



# HarePoint Workflow Manager Extensions for SharePoint on premise

## Administrator's Guide



Product version 1.2

4/4/2023

( This Page Intentionally Left Blank )

## Table of Contents

Table of Contents .....	3
I. Introduction .....	13
Description .....	13
II. Product Architecture.....	13
How the product works .....	13
III. Installation .....	14
Prerequisites .....	14
The distributive package .....	14
Installing solution.....	15
Checking configuration .....	18
Trial mode and licensing .....	18
IV. Update .....	19
Before updating .....	19
Getting the distributive package.....	19
Updating solution.....	20
V. Uninstallation.....	23
General uninstall information.....	23
Deactivate features.....	23
Remove trust.....	24
Retract solution.....	25
Remove IIS site.....	26
VI. Activating or deactivating custom actions on a site .....	27
General information .....	27
Managing features .....	27
VII. Security console .....	28
General security information.....	28
Generating a token .....	29
Using tokens.....	30
VIII. Licensing .....	31

Trial mode .....	31
Licensing model .....	31
Entering license keys.....	32
IX. Creating a workflow in SharePoint Designer .....	32
Connecting SharePoint Designer .....	32
Creating a workflow .....	33
X. List of workflow actions .....	35
1. Active Directory activities .....	35
1.1 Add Member to Active Directory Group.....	35
1.2 Change Active Directory Account Settings.....	36
1.3 Change Active Directory Group Settings.....	37
1.4 Create Group in Active Directory .....	37
1.5 Create Mail Contact in Active Directory .....	38
1.6 Create User in Active Directory.....	39
1.7 Delete Group from Active Directory .....	39
1.8 Delete User from Active Directory.....	40
1.9 Find User in Active Directory .....	40
1.10 Find User in Active Directory by Query.....	41
1.11 Get Active Directory Groups Where User is Member.....	41
1.12 Get E-Mails of Active Directory Group Members .....	42
1.13 Get User's Attribute from Active Directory .....	42
1.14 Get User's Manager from Active Directory.....	43
1.15 Get User's Multivalued Attribute from Active Directory .....	43
1.16 Move User into Organizational Unit .....	44
1.17 Remove Member from Active Directory Group.....	44
1.18 Set Active Directory Options.....	45
1.19 Set User's Attribute in Active Directory .....	45
1.20 Set User's Multivalued Attribute in Active Directory.....	46
1.21 Set User's Password in Active Directory .....	46
1.22 Set User's Photo in Active Directory .....	47
2. Administration activities .....	47

2.1 Activate Feature by GUID.....	47
2.2 Add User to SharePoint Group.....	47
2.3 Add User to SharePoint Site.....	48
2.4 Alternate CSS and Site Logo .....	48
2.5 Change SharePoint Group Owner .....	48
2.6 Change SharePoint Group Settings .....	49
2.7 Change SharePoint List Settings.....	50
2.8 Create SharePoint Group .....	51
2.9 Create SharePoint Site .....	51
2.10 Create Site Collection.....	52
2.11 Deactivate Feature By GUID .....	52
2.12 Delete SharePoint Group .....	53
2.13 Delete SharePoint Site .....	53
2.14 Delete Site Collection.....	53
2.15 Delete Site Collection from Recycle Bin.....	53
2.16 Deploy Theme To Web.....	54
2.17 Get SharePoint User Membership Status .....	54
2.18 Get Site User MultiValued Property .....	55
2.19 Get Site User SingleValue Property.....	55
2.20 Inject Custom CSS .....	56
2.21 Remove CSS and Site Logo .....	56
2.22 Remove Custom Action.....	56
2.23 Remove User from SharePoint Group .....	56
2.24 Remove User from SharePoint Site.....	57
2.25 Rename Site at Url .....	57
2.26 Set Master Page .....	57
2.27 Set SharePoint Admin Center URL .....	58
2.28 Set SharePoint Site Navigation Options.....	58
2.29 Set Site Theme Based On Name .....	59
3. Cloud Print activities .....	59
3.1 Delete Print Job.....	59

3.2 Get Print Job Status.....	59
3.3 Get Printer Status.....	60
3.4 Set Printing Options .....	60
3.5 Submit Print Job .....	61
4. Conversion activities .....	61
4.1 Convert Document.....	61
5. Date/Time activities .....	63
5.1 Convert Date from Local Time to UTC .....	63
5.2 Convert Date from UTC to Local Time .....	63
5.3 Convert Date to Text.....	64
5.4 Evaluate Due Date.....	65
5.5 Get Date Property .....	65
5.6 Get Interval Between Dates.....	66
5.7 Get Working Time Between Dates.....	67
5.8 Set DateTime Constant .....	67
6. Development activities .....	68
6.1 Convert JSON to XML .....	68
6.2 Execute PowerShell Script.....	69
6.3 Execute SQL Command .....	70
6.4 Execute SQL Command into List .....	71
6.5 Query XML.....	72
6.6 Transform XML.....	74
6.7 Update XML.....	76
7. Dictionary activities.....	78
7.1 Add an Item to a Dictionary .....	78
7.2 Remove Duplicate Values from Dictionary .....	79
7.3 Remove Item from Dictionary.....	79
8. Document Library activities .....	80
8.1 Check In Item at URL .....	80
8.2 Check Out Item at URL .....	80
8.3 Copy Document to List Item Attachment .....	80

8.4 Copy Folder Structure .....	81
8.5 Copy or Move Document .....	81
8.6 Copy or Move Folder.....	82
8.7 Create Document from Text .....	82
8.8 Create Folder in the Document Library at URL .....	83
8.9 Create Link to a Document .....	83
8.10 Delete Folder in the Document Library at URL .....	84
8.11 Discard Check Out Item at URL .....	84
8.12 Discard Publish Item .....	84
8.13 Get Document Lock Status.....	84
8.14 Get Entry From Folder Property Bag.....	85
8.15 Get Text Document Content .....	85
8.16 Publish Item .....	85
8.17 Rename Document at URL.....	86
8.18 Upload Net File to Document Library at URL.....	86
9. E-Mail activities .....	87
9.1 Add Custom Headers to E-mail .....	87
9.2 Get E-mail Property.....	87
9.3 Receive All E-Mails to List from IMAP Server.....	88
9.4 Receive All E-mails to List from POP3 Server .....	89
9.5 Receive E-Mail from IMAP server .....	90
9.6 Receive E-Mail from POP3 Server .....	91
9.7 Save E-Mail Attachments to List Item .....	93
9.8 Send E-Mail with Document as Attachment.....	93
9.9 Send E-Mail with Http Files Attachments .....	94
9.10 Send E-Mail with List Item Attachments.....	94
9.11 Send E-Mail with SharePoint Attachments.....	95
9.12 Send E-Mail with Word Document as a Body .....	96
9.13 Send E-Mail without Attachments .....	97
9.14 Set E-mail Delivery and Processing Options .....	98
9.15 Set SMTP Server Options .....	99

10. Exchange activities .....	100
10.1 Cancel Exchange Calendar Item.....	100
10.2 Create Exchange Calendar Item.....	101
10.3 Delete Exchange Calendar Item.....	101
10.4 Update Exchange Calendar Item.....	102
10.5 Update Exchange Calendar Item Advanced Properties .....	102
10.6 Update Exchange Calendar Item Attendees .....	103
11. Images activities.....	104
11.1 Convert Image.....	104
11.2 Flip Image.....	104
11.3 Get EXIF Property from Image .....	105
11.4 Get Image Size.....	106
11.5 Get Image Type .....	106
11.6 Get XMP Property .....	109
11.7 Resize Image .....	109
11.8 Rotate Image.....	110
11.9 Set EXIF Property on Image.....	111
11.10 Set Image Transparent Color .....	111
11.11 Set XMP Property.....	112
12. Lists activities .....	112
12.1 Add Content Type to List.....	112
12.2 Add Existing Column to Content Type .....	112
12.3 Approve Item at URL.....	113
12.4 Copy List Item at URL .....	113
12.5 Copy List Item Attachments to Document Library.....	113
12.6 Copy List Item Attachments to List Item.....	114
12.7 Create Content Type .....	114
12.8 Create List .....	115
12.9 Create List Item at URL.....	115
12.10 Create Site Column .....	116
12.11 Delete Entire SharePoint List at Url .....	117



12.12 Delete Item at URL .....	117
12.13 Find List Item by CAML.....	117
12.14 Get Field Value at URL .....	118
12.15 Get List Item Attachments .....	118
12.16 Get List Item URL.....	118
12.17 Get List URL .....	119
12.18 Loop Through and Update List Items.....	120
12.19 Move List Item to Subfolder .....	122
12.20 Reject Item at URL.....	122
12.21 Remove All Attachments.....	122
12.22 Remove Content Type from List .....	123
12.23 Set Content Type Document Template.....	123
12.24 Set Default Content Type on List .....	123
12.25 Update List Item at URL .....	124
13. Network activities .....	125
13.1 Create Directory on FTP Server.....	125
13.2 Delete Directory from FTP Server .....	125
13.3 Delete File from FTP Server .....	126
13.4 Download Document from FTP Server .....	126
13.5 Download Documents from FTP Folder .....	127
13.6 Fetch RSS Feed to Html.....	127
13.7 Fetch RSS Item .....	128
13.8 Send HTTP GET Request.....	128
13.9 Send HTTP OAuth Request.....	129
13.10 Send HTTP POST Request.....	131
13.11 Send XMPP Instant Message.....	132
13.12 Set FTP Server Options.....	132
13.13 Set OAuth Authentication .....	133
13.14 Shorten Url .....	134
13.15 Upload Document to FTP Server.....	134
13.16 Upload Documents to FTP Folder .....	135

13.17 Upload List Item Attachments to FTP Server .....	135
14. Office activities.....	136
14.1 Accept Revision Changes in Word Document.....	136
14.2 Add Digital Signature .....	136
14.3 Add New Row into Word Document .....	137
14.4 Add Watermark to Word Document .....	137
14.5 Find Text in Word Document.....	138
14.6 Get Excel Cell Value.....	138
14.7 Get Office Document Property .....	138
14.8 Get Word Document Page Count .....	139
14.9 Read Word Document .....	140
14.10 Read Word Document To Text.....	140
14.11 Replace External Hyperlink in Word Document .....	140
14.12 Replace Text in Word Document .....	141
14.13 Set Excel Cell Value .....	142
14.14 Set Office Document Property.....	142
14.15 Update Multiple Fields in Word Document.....	143
14.16 Update Picture in Word Document .....	144
14.17 Update Word Document.....	145
15. Pdf activities.....	147
15.1 Add Image Watermark to PDF .....	147
15.2 Add PDF Letterhead .....	148
15.3 Add Text Watermark to PDF .....	148
15.4 Merge PDF Documents .....	149
15.5 Sign PDF Document.....	149
15.6 Split PDF Documents.....	150
16. Permissions activities.....	151
16.1 Change Permissions Inheritance at URL .....	151
16.2 Delete All Permissions at URL .....	151
16.3 Delete All User Permissions at URL.....	152
16.4 Delete Permissions at Url.....	152

16.5 Grant Permissions at Url .....	152
17. Social activities .....	153
17.1 Create Group in Yammer .....	153
17.2 Create MailChimp Campaign .....	153
17.3 Create User in Yammer .....	154
17.4 Delete MailChimp Campaign .....	155
17.5 Delete User from Yammer .....	155
17.6 Follow user/site/tag/document in SharePoint .....	155
17.7 Get User's Property from Yammer.....	156
17.8 Invite User to Yammer .....	156
17.9 Publish Message to Social Media via dlvr.it .....	156
17.10 Publish Message to Yammer .....	157
17.11 Replicate MailChimp Campaign .....	157
17.12 Schedule MailChimp Campaign .....	158
17.13 Send MailChimp Campaign .....	159
17.14 Send Twitter Message.....	159
17.15 Set User's Property in Yammer .....	161
17.16 Stop following user/site/tag/document in SharePoint.....	161
17.17 Subscribe to MailChimp List.....	162
17.18 Unsubscribe from MailChimp List.....	162
17.19 Update Group in Yammer .....	163
18. Strings activities .....	163
18.1 Capture All Occurrences of Text .....	163
18.2 Capture Text with Regular Expression .....	164
18.3 Convert Base64 to Text .....	165
18.4 Convert HTML To Text .....	166
18.5 Convert Number To Text.....	166
18.6 Convert Text Case .....	167
18.7 Convert Text to Base64 .....	167
18.8 Decode Web Text.....	168
18.9 Encode Web Text .....	169

18.10 Evaluate Mathematical Expression .....	170
18.11 Extract Substring .....	171
18.12 Get Random Number .....	171
18.13 Get Random String .....	172
18.14 Get String Length .....	172
18.15 Join Text .....	172
18.16 Log to History List (HarePoint) .....	173
18.17 Replace Text with Regular Expression .....	173
18.18 Set Regular Expression Options .....	174
18.19 Split Text.....	175
18.20 Trim Text .....	175
19. Taxonomy activities .....	176
19.1 Add Language To Term Store.....	176
19.2 Create Taxonomy Group .....	176
19.3 Create Taxonomy Term.....	177
19.4 Create Taxonomy Term Set.....	177
19.5 Delete Taxonomy Group .....	178
19.6 Delete Taxonomy Term.....	178
19.7 Delete Taxonomy Term Set.....	178
19.8 Update Taxonomy Group.....	179
19.9 Update Taxonomy Term .....	179
19.10 Update Taxonomy Term Set .....	180
20. Workflow activities .....	180
20.1 Get Workflow Status.....	180
20.2 Set Authentication .....	181
20.3 Start Workflow .....	182
20.4 Terminate Workflow .....	183
Appendix A. Technical support .....	184

## I. Introduction

### Description

HarePoint Workflow Manager Extensions for SharePoint on premise is a software product that vastly expands the capabilities of SharePoint Designer to create the “2013 workflow platform” type workflows for SharePoint and in the most of cases allows avoiding custom development and coding to achieve almost any scenario.

The product offers a wide range of ready-to-use workflow actions with a fundamentally new functionality to workflows, such as: managing document libraries and lists, reading and writing Microsoft Office documents, document conversion to PDF and other formats, image conversion, processing and resizing, SharePoint administration, executing SQL commands, operating through networking protocols, such as HTTP, SMTP, POP3, FTP, RSS, XMPP and Twitter, improved activities for sending and receiving e-mails, finally a large set of actions that facilitate work with dictionaries, strings and regular expressions, date and time.

The product is fully supported by all Microsoft workflow development tools and there is no need to adapt the existing workflows, or migrate them to a new development environment.

The **key features** of HarePoint Workflow Extensions are:

- ✓ easy, quick and trouble-free deployment and update
- ✓ complete integration with SharePoint Designer which is a freeware and widely used tool, thus minimum training costs
- ✓ over 240 custom actions in addition to the built-in actions in SharePoint Designer
- ✓ full support in SharePoint Designer
- ✓ the custom actions are also supported by [HarePoint Workflow Designer](#)

## II. Product Architecture

### How the product works

The “2013 workflow platform” type workflows run on a separate engine, which is called **Workflow Manager**. Since this engine is technically no longer a part of SharePoint (unlike the “2010 workflow platform”), it provides a set of advantages, such as improved performance, reliability, scalability, and control of the workflows.

HarePoint Workflow Manager Extensions product installs the following set of components:

- **SharePoint application** that contains workflow actions definitions used by SharePoint Designer or HarePoint Workflow Designer
- **IIS site** that hosts the engine for HarePoint Workflow Manager Extensions
- Small **SQL database** to keep the workflow data and cache

- **Certificates** used to establish **Trust relationship** to exchange data securely between Workflow Manager and HarePoint Workflow Manager Extensions engines

User creates a workflow using **SharePoint Designer** or **HarePoint Workflow Designer**, where the additional actions from HarePoint Workflow Manager Extensions are available in 19 new categories.

When a workflow is executed by a Workflow Manager engine, once it reaches the custom action from HarePoint, the data are sent to the HarePoint workflow engine, then a response with the action results is returned to the Workflow Manager and the workflow continues.

The process of exchanging data with the HarePoint workflow engine is completely seamless and remains unnoticed for SharePoint users and administrators, thus retaining the experience of working with native SharePoint workflows.

## III. Installation

### Prerequisites

**Note:** You need to have a **farm administrator** permissions on your SharePoint farm to install the product.

- **Windows Server 2008** or newer
- **SharePoint 2013/2016/2019/SE**
- **IIS 7.5** or newer
- **.NET Framework 4.6.1** and above is required
- **Web Deploy 3.6** and above is required
- The **64-bit** edition of **SQL Server 2008 R2** or newer

### The distributive package

The latest version of the product can be downloaded using the following direct link:

<https://www.harepoint.com/DownloadProducts/HarePointWorkflowExtensions2016.zip>

Unpack the zip file to a new folder on the server where the installation will be initiated.

The distributive package contains:

- **Resources** folder – contains all the necessary resource files for the deployment. Specifically, this is a default location for **certificate** files (.cer and .pfx) that will be created upon installation, and may later be used for installing product updates.
- **Install.ps1** – the main PowerShell script file used to conduct the installation or update procedures.

- **Install.txt** – the file with the most detailed and up-to-date instructions on how to use the current version of the Install.ps1 script. **Please read carefully before proceeding with the installation or update!**
- **Install.Parameters.config** – the configuration file that is created automatically when the Install.ps1 script is running. It stores the parameters used previously to run installation or update again with the same parameters.
- **Install.log** – the log file that will be created automatically when the Install.ps1 script is running. This is the full log of the installation process, with all the parameters used and all the script outputs. This file can be used for further technical reference.
- **ReadMe.txt** - the file containing the version number of the given release.

## Installing solution

The product is installed using the fully automated PowerShell script.

Open the **PowerShell Console** and navigate to the folder where you have unzipped the distributive package (see previous paragraph).

Run the installation script using the following command:

**\.Install.ps1**

**Note:** The script saves parameters from the previous run, so if you have already executed the script earlier, it will first prompt if the same parameters should be used, or if you would like to specify all the parameters from scratch:

```
Attached Parameters

Certificate path : Resources\HWE2013Cert.pfx
Site name       : Hwe2013
Http port      : 8187
Sql Server     : SP2016-PREV\SHAREPOINT
Sql Database   : Hwe2013StateStorage
Installation mode : 1

Run a script with these parameters? [Y/n]
```

Enter “n” to start from scratch (or “y” to proceed with the indicated settings).

Choose the **installation mode**:

```
Specify installation mode:
1. Full
2. Workflow actions IIS web site
3. SharePoint package with action definitions and management pages
Enter mode (default is 1):
```

1. **Full** – in this mode, all the components will be installed, including both the IIS site hosting the HarePoint workflow engine and the SharePoint app with action definitions and management pages will be deployed (effectively, it combines the modes 2 and 3 below).
2. **Workflow actions IIS web site** – in this mode, only the IIS site hosting the HarePoint workflow engine will be installed
3. **SharePoint package with action definitions and management pages** - in this mode, only the SharePoint WSP package will be deployed

Depending on your deployment scenario and farm topology, you may be installing these components either to a single server (mode 1, the simplest case), or to different servers (modes 2 and 3).

In this example we will proceed with the default **Full** mode – enter “1”:

```
Enter mode (default is 1):
1
Please enter relative path to certificate (default is Resources\HWE2013Cert.pfx):
```

The script prompts for the path where the **certificate file** (.pfx) is located.

Since this is a new install, the file does not yet exist, so simply click enter.

Provide the **password** for the newly created certificate:

```
Please enter relative path to certificate (default is Resources\HWE2013Cert.pfx):
Certificate file C:\Users\AdminSP2016\Downloads\HWE_1.0.21328_j\does not exist, creating new one
Please enter password for the new certificate: _
```

The new certificate file (.pfx) will be created.

**Note:** be sure to remember the certificate password – it may be needed afterwards during the product update or reinstall.

At the next step, specify the **name for the IIS site** that will host the HarePoint workflow engine or keep it default *Hwe2013*:

```
Please enter site name (default is Hwe2013):
```

The next step is to create or connect to a SQL database that will be used to store the workflow statuses and cache. Specify the **SQL server name**, including the **instance** name, if necessary:

```
Site path: C:\inetpub\hwe2013
Application pool: HWE2013AppPool
Please enter SQL server name (it can be SERVER, SERVER\INSTANCE or SERVER\INSTANCE,PORT)
```

Then specify the **SQL database name**, or keep it default *Hwe2013StateStorage*:

```
Please enter database name (default is Hwe2013StateStorage):
```



After successful connection to the database, the script will proceed to **deploying the IIS site**.

Specify the port number to configure the site binding, or keep it default 8187:

```
Deploying web site ...
Modifying web.config ...
HarePoint Workflow Extensions IIS Web successfully installed
Please enter a port number for a site (default is 8187):
_
```

The installation process will continue; this may take some time.

The script will be updating you about the status:

```
Setting up trust and registering app principal...
Connecting to Central Administration web...
Exporting .cer certificate...
Creating new HarePoint Workflow Extensions 2013 STS cert trusted root authority
HarePoint Workflow Extensions 2013 STS cert trusted root authority successfully created
Creating new HarePoint Workflow Extensions 2013 STS trusted security token issuer
HarePoint Workflow Extensions 2013 STS trusted security token issuer successfully created
Registering app principal
App principal has been successfully registered
Waiting to finish solution timer job
Finished the solution timer job
Waiting to finish solution timer job
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Finished the solution timer job
Removing solution from the farm
Waiting to finish solution timer job
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Finished the solution timer job
Successfully deployed the package
```

The final step is to **validate the installation and activate the product**.

Enter the **SharePoint site URL** where the custom workflow actions from HarePoint Workflow Manager Extensions will be used:

```
Do you want to validate installation and activate HarePoint Workflow Extensions on specified web site? [Y/n]
```

**Note:** if you need to have the custom actions from HarePoint Workflow Manager Extensions available on other SharePoint sites as well, please follow the instructions of the [VI. Activating or deactivating custom actions on a site](#) chapter, after completing the installation.

The product installation is now complete.

You can close the PowerShell console window.

The installation process log has been saved to **Install.log** file in the same folder where the script is located. Keep this file for further technical reference.

## Checking configuration

After the installation, check the **URL** of the HarePoint Workflow Manager Extensions IIS site in the product settings.

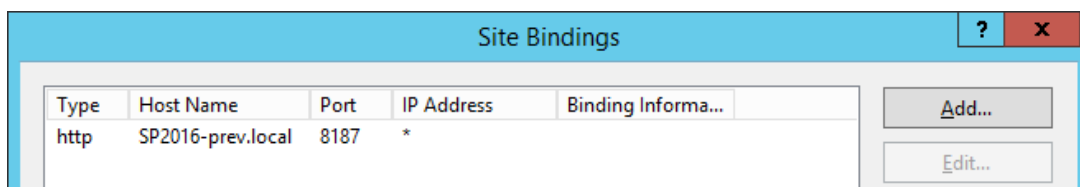
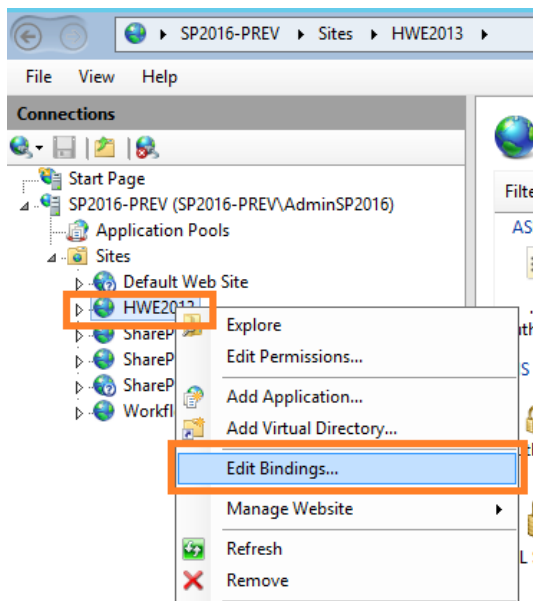
Open **Central Administration – HarePoint Workflow Extensions 2013/2016**:

HarePoint Workflow Extensions IIS web site URL

Specify the URL of the IIS website where the HarePoint Workflow Extensions web application is deployed

http://sp2016-prev.local:8187/

The URL and port number need to correspond to the settings of the **HWE2013** (default name) site in IIS Manager, site bindings:



## Trial mode and licensing

If the product is installed for the first time on this farm, the **30-days** trial mode will be activated.

For details on licensing and instructions to **enter the license keys**, refer to the [VIII. Licensing](#) section.

## IV. Update

### Before updating

We constantly work on HarePoint Workflow Manager Extensions, introducing new workflow actions, and extending the capabilities of the existing ones.

We recommend having an up-to-date version of HarePoint Workflow Manager Extensions installed.

### Current version

You can review the currently installed version in **Central Administration – HarePoint Workflow Extensions 2013/2016**:



### Requirements

You need to have a **farm administrator** permissions on your SharePoint farm to update the product.

### Getting the distributive package

The latest version of the product can be downloaded using the following direct link:

<https://www.harepoint.com/DownloadProducts/HarePointWorkflowExtensions2016.zip>

Unpack the zip file to a new folder on the server where the installation will be initiated.

The distributive package contains:

- **Resources** folder – contains all the necessary resource files for the deployment.  
**Note:** this is a default location for **certificate** files (.cer and .pfx) that are created upon installation, so it is convenient to **copy them from the previously used distributive package to the new one** before starting an update.
- **Install.ps1** – the main PowerShell script file used to conduct the installation or update procedures.
- **Install.txt** – the file with the most detailed and up-to-date instructions on how to use the current version of the Install.ps1 script. **Please read carefully before proceeding with the**

## installation or update!

- **Install.Parameters.config** – the configuration file that is created automatically when the Install.ps1 script is running. It stores the parameters used previously to run installation or update again with the same parameters.

**Note:** It may be convenient to **copy this file from the previously used distributive package to the new one** before starting an update, to use the same parameters

- **Install.log** – the log file that is created automatically when the Install.ps1 script is running. This is the full log of the update process, with all the parameters used and all the script outputs. This file can be used for further technical reference.
- **ReadMe.txt** - the file containing the version number of the given release.

## Updating solution

The product is updated using the fully automated PowerShell script.

Open the **PowerShell Console** and navigate to the folder where you have unzipped the distributive package (see previous paragraph).

Run the installation script using the following command:

### \.Install.ps1

**Note:** The script saves parameters from the previous run in **Install.Parameters.config** file, so if you have already executed the script earlier, it will first prompt if the same parameters should be used, or if you would like to specify all the parameters from scratch:

```
Attached Parameters

Certificate path : Resources\HWE2013Cert.pfx
Site name       : Hwe2013
Http port       : 8187
Sql Server      : SP2016-PREV\SHAREPOINT
Sql Database    : Hwe2013StateStorage
Installation mode : 1

Run a script with these parameters? [Y/n]_
```

Enter “n” to start from scratch (or “y” to proceed with the indicated parameters).

Choose the **update mode**:

```
Specify installation mode:
1. Full
2. Workflow actions IIS web site
3. SharePoint package with action definitions and management pages
Enter mode (default is 1):
```

1. **Full** – in this mode, all the components will be update, including both the IIS site hosting the HarePoint workflow engine and the SharePoint app with action definitions and management pages will be deployed (effectively, it combines the modes 2 and 3 below).
2. **Workflow actions IIS web site** – in this mode, only the IIS site hosting the HarePoint workflow engine will be updated
3. **SharePoint package with action definitions and management pages** - in this mode, only the SharePoint WSP package will be updated

Depending on your deployment scenario and farm topology, these components can be deployed either to a single server (mode 1, the simplest case), or to different servers (modes 2 and 3). The update would need to be installed accordingly.

There can be recommendations from HarePoint Technical Support if it's sufficient to update just one of the components in this release, so you could be running the partial update (mode 2 or mode 3) rather than the full update (mode 1).

In this example we will proceed with the default **Full** mode – enter “1”:

```
Enter mode (default is 1):
1
Please enter relative path to certificate (default is Resources\HWE2013Cert.pfx):
```

The script prompts for the path where the **certificate file** (.pfx) is located.

If you have copied the .pfx file from the previous distributive package, then you can specify the path to it. Alternatively, if you don't have it, simply continue and the new .pfx file will be created.

Provide the **password** for the certificate:

```
Please enter relative path to certificate (default is Resources\HWE2013Cert.pfx):
Certificate file C:\Users\AdminSP2016\Downloads\HwME_1.0.21328_j\does not exist, creating new one
Please enter password for the new certificate: _
```

**Note:** be sure to remember the certificate password – it may be needed afterwards during the product update or reinstall.

At the next step, specify the **name for the IIS site** that hosts the HarePoint workflow engine or use the default name *Hwe2013*:

```
Please enter site name (default is Hwe2013):
```

The next step is to connect to a SQL database that stores the workflow statuses and cache.

Specify the **SQL server name**, including the **instance** name, if necessary:

```
Site path: C:\inetpub\hwe2013
Application pool: HWE2013AppPool
Please enter SQL server name (it can be SERVER, SERVER\INSTANCE or SERVER\INSTANCE,PORT)
```

Then specify the **SQL database name**, or use the default name *Hwe2013StateStorage*:

```
Please enter database name (default is Hwe2013StateStorage):
```

After successful connection to the database, the script will proceed to **redeploying the IIS site**.

Specify the port number to configure the site binding, or use the default port 8187:

```
Deploying web site ...
Modifying web.config ...
HarePoint Workflow Extensions IIS Web successfully installed
Please enter a port number for a site (default is 8187):
```

The installation process will continue; this may take some time.

The script will be updating you about the status:

```
Setting up trust and registering app principal...
Connecting to Central Administration web...
Exporting .cer certificate...
Removing existing HarePoint Workflow Extensions 2013 STS cert trusted root authority
HarePoint Workflow Extensions 2013 STS cert trusted root authority successfully removed
Creating new HarePoint Workflow Extensions 2013 STS cert trusted root authority
HarePoint Workflow Extensions 2013 STS cert trusted root authority successfully created
Removing existing HarePoint Workflow Extensions 2013 STS trusted security token issuer
HarePoint Workflow Extensions 2013 STS trusted security token issuer successfully removed
Creating new HarePoint Workflow Extensions 2013 STS trusted security token issuer
HarePoint Workflow Extensions 2013 STS trusted security token issuer successfully created
Registering app principal
App principal has been successfully registered
Waiting to finish solution timer job
Finished the solution timer job
Waiting to finish solution timer job
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Finished the solution timer job
Removing solution from the farm
Waiting to finish solution timer job
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Please wait... either a retraction or deployment is happening
Finished the solution timer job
Successfully deployed the package
```

The final step is to **validate the installation and activate the product**.

Enter the **SharePoint site URL** where the custom workflow actions from HarePoint Workflow Manager

Extensions are used:

```
Do you want to validate installation and activate HarePoint Workflow Extensions on specified web site? [Y/n]
```

The installer will then display a warning that you are about to recycle the “**HarePoint Workflow Extensions 2013 Actions**” site level feature – confirm this with entering “y”. Recycling is needed to apply changes.

**Note:** if you’re using workflows with HarePoint Workflow Manager Extensions actions on other sites, be sure to recycle the site feature manually on those sites to apply changes. For more details, refer to [Managing features](#) chapter.

The product update is now complete.

You can close the PowerShell console window. The update process log has been saved to **Install.log** file in the same folder where the script is located.

## V. Uninstallation

### General uninstall information

Uninstalling HarePoint Workflow Extensions **will not remove or affect any of the existing workflows** in any way.

The workflows containing HarePoint actions will not be functional after the uninstallation.

If you reinstall HarePoint Workflow Manager Extensions afterwards, the functionality of the workflows containing HarePoint actions **will be restored**, without the need to republish the workflows.

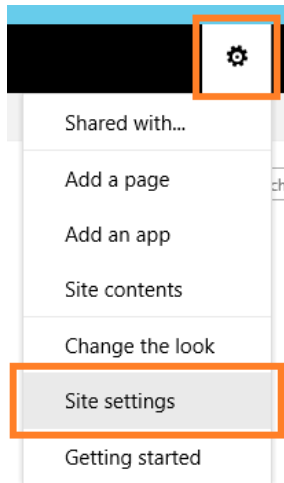
To **uninstall** HarePoint Workflow Manager Extensions completely, follow the steps, as described in the following paragraphs.

### Deactivate features

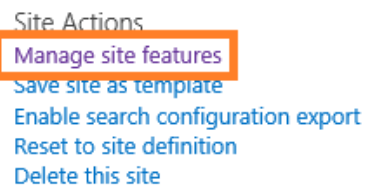
The availability of custom actions from HarePoint Workflow Manager Extensions is managed individually per each site by the site feature:

- **HarePoint Workflow Extensions 2013 Actions**

This feature needs to be deactivated **on each site** where it was activated.  
Open the SharePoint site, navigate to **Site Settings**:



Click **Manage site features** under **Site Actions**:



Scroll down to find the **HarePoint Workflow Extensions 2013 Actions** feature:



Deactivate this feature.

## Remove trust

Navigate to **Central Administration – Security – General Security – Manage trust**:



# Security



## Users

[Manage the farm administrators group](#) | [Approve or reject distribution groups](#) | [Specify web application user policy](#)



## General Security

[Configure managed accounts](#) | [Configure service accounts](#) | [Configure password change settings](#) | [Specify authentication providers](#) | [Manage trust](#) | [Manage antivirus settings](#) | [Define blocked file types](#) | [Manage web part security](#) | [Configure self-service site creation](#)



## Information policy

[Configure information rights management](#) | [Configure Information Management Policy](#)

Click on **HarePoint Workflow Extensions 2013 STS cert**, then click **Delete** button on a ribbon:

The screenshot shows the 'TRUST RELATIONSHIPS' tab on the 'Security' ribbon. The 'Delete' button is highlighted with an orange box. Below the ribbon, a table lists various certificates. The 'HarePoint Workflow Extensions 2013 STS cert' is highlighted with an orange box.

Central Administration	Name
Application Management	00000005-0000-0000-c000-000000000000_7bc3fa5c-f1e9-48ea-ba53-48144ae1950c
System Settings	00000005-0000-0000-c000-000000000000_8d86d38e-d991-4501-9b6d-267e21d5d0e8
Monitoring	00000005-0000-0000-c000-000000000000_981bbd6b-396a-48b4-858a-944eb18f4c7f
Backup and Restore	00000005-0000-0000-c000-000000000000_a895c156-023d-4b93-8339-b212b219ca63
Security	00000005-0000-0000-c000-000000000000_ec1ef6df-017c-47f0-a82e-15c9f36138f6
Upgrade and Migration	00000005-0000-0000-c000-000000000000_fcf2b5ad-9812-4547-ae8f-1fd4aa1adda5
General Application Settings	local
	Workflow Manager Farm

Alternatively, you can use the following PowerShell command:

**Remove-SPTtrustedRootAuthority -Identity "HarePoint Workflow Extensions 2013 STS cert"**

## Retract solution

Navigate to **Central Administration – System settings – Farm Management – Manage farm solutions:**

# System Settings



## Servers

[Manage servers in this farm](#) | [Manage services in this farm](#) |  
[Manage services on server](#) | [Convert server role in this farm](#)



## E-Mail and Text Messages (SMS)

[Configure outgoing e-mail settings](#) | [Configure incoming e-mail settings](#) |  
[Configure mobile account](#)



## Farm Management

[Configure alternate access mappings](#) | [Manage farm features](#) |  
[Manage farm solutions](#) | [Manage user solutions](#) |  
[Configure privacy options](#) | [Configure cross-firewall access zone](#)

Click on **hwe2013.wsp**.

In the Solution Properties, click **Retract** solution:

## Solution Properties

[Retract Solution](#) | [Back to Solutions](#)

Name:	hwe2013.wsp
Type:	Core Solution
Contains Web Application Resource:	No
Contains Global Assembly:	Yes
Contains Code Access Security Policy:	No
Deployment Server Type:	Front-end Web server
Deployment Status:	Deployed
Deployed To:	Globally deployed.
Last Operation Result:	The solution was successfully deployed.
Last Operation Details:	SP2016-PREV : The solution was successfully deployed.

After the solution is retracted, you can click **Remove solution**.

Alternatively, you can use the two PowerShell commands:

```
Uninstall-SPsolution -identity "hwe2013.wsp"
```

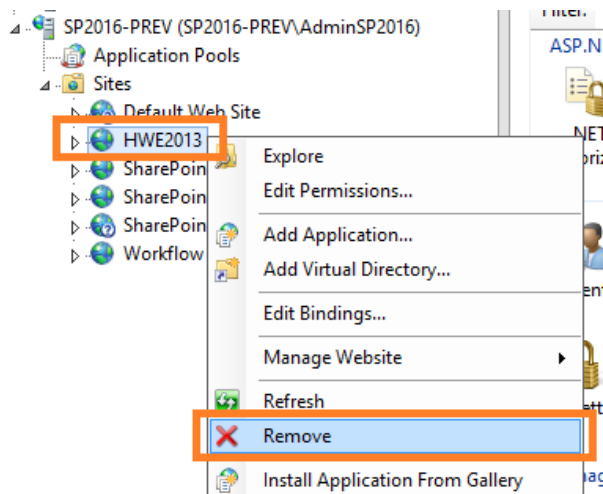
```
Remove-SPsolution -identity "hwe2013.wsp"
```

## Remove IIS site

Open **IIS Manager**, expand **Sites**.

Locate the **HWE2013** site, right-click it and select **Remove**:

**Note:** “HWE2013” is a default name of the site; a different site name could have been assigned while installing the product. For more details on the installation process, refer to **Install.log** file from the [distributive package](#) that was used for installation.



The product is now completely removed from your SharePoint farm.

## VI. Activating or deactivating custom actions on a site

### General information

The availability of custom actions from HarePoint Workflow Manager Extensions is managed individually per each site by the site feature:

- **HarePoint Workflow Extensions 2013 Actions**

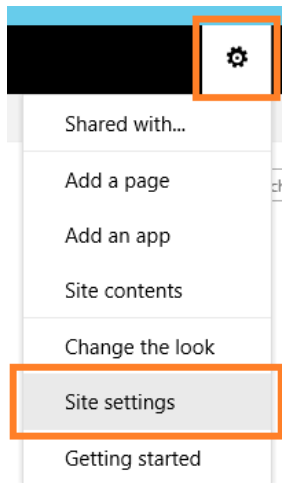
This feature is **automatically activated** on the site that was specified during the last step of [installation](#) process.

However, if you need to have HarePoint Workflow Manager Extensions actions available on other sites as well, the site feature there needs to be activated manually, see next section.

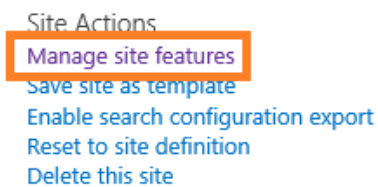
### Managing features

Follows these steps to make HarePoint Workflow Manager Extensions actions available on a specific site:

Open the SharePoint site, navigate to **Site Settings**:



Click **Manage site features** under **Site Actions**:



Scroll down to find the HarePoint Workflow Extensions 2013 Actions feature:



Activate the **HarePoint Workflow Extensions 2013 Actions** feature.

Restart SharePoint Designer if necessary. The custom actions from HarePoint Workflow Manager Extensions will now become available for workflows on this site.

If you need to **disable** the custom actions on this site, deactivate the **HarePoint Workflow Extensions 2013 Actions** feature:



## VII. Security console

### General security information

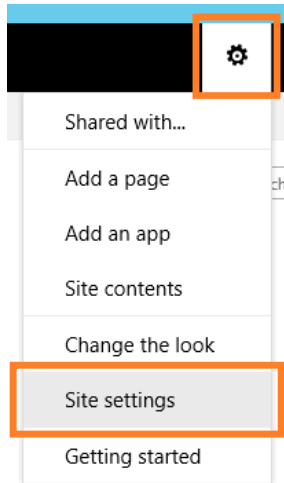
With HarePoint Workflow Manager Extensions actions it is possible to interact with various services that require authentication (SharePoint sites, FTP, SQL, PowerShell, SMTP, and so on).

For improved security, HarePoint Workflow Manager Extensions do not allow transferring credentials in clear text; the main approach is based on using **tokens**.

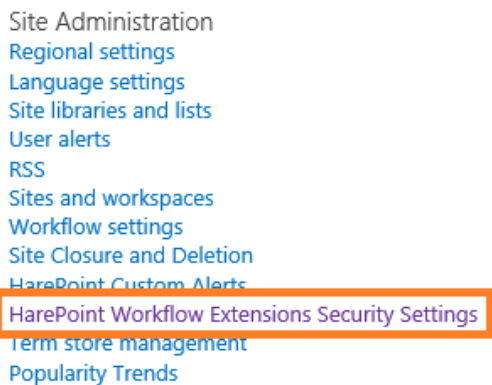
Tokens (to access a service) can be generated on the **Security console**, and then used in **Set Authentication** action in a workflow to pass it to the corresponding workflow action, as described below.

## Generating a token

Open the SharePoint site, navigate to **Site Settings**:



Under **Site Administration**, click **HarePoint Workflow Extensions Security Settings**:



The **Security console** page will open:

## HarePoint Workflow Extensions - Security



Access Token Generation  
[SharePoint](#) | [Login&Password](#) | [PrintNode](#)

In this Security console, you can generate the following tokens:

- **SharePoint** – to access the current SharePoint site:  
SharePoint Access Token

**Success**  
eyJUeXBlljoyLCJJc0VuY3J5cHRIZCI6ZmFsc2UsIkRhdGEiOiJleUpWYzJWeVNXUWlPaUpzJWpneE5UWTVOVEUxTFRjek1UZ3pNakEzTLnpZNUxUVXdNQ0lzSWtsZWVmxSR0YyWWwvRmFdeE16QlVNVGc2TkRZNk1EUXVOVFf3T1RBME5Gb2lMQ0pWY21raU9pSm9kSFJ3T2k4dlIteGhMbU52YINKOSJ9

- **Login&Password** – to access any service using login and password (including SharePoint sites too):  
Login&Password Token

Login:

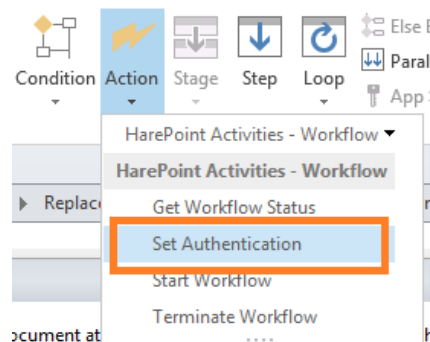
Password:

- **PrintNode** – to interact with Google Cloud Print service:  
PrintNode Token

ApiKey:

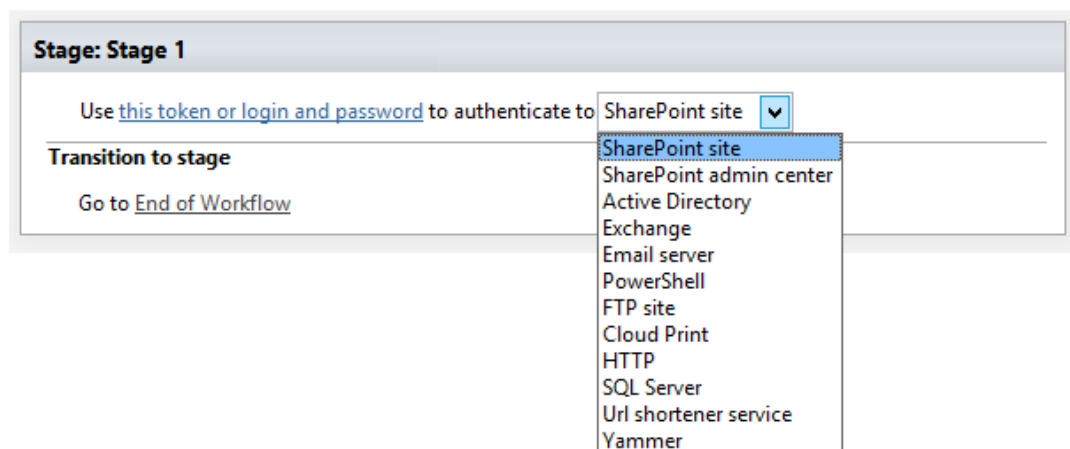
## Using tokens

To deliver the tokens obtained on the Security console to the workflow actions using them, a special workflow action is used – **Set Authentication** (located under **Workflows** category):



This action should be placed above all other actions requesting authentication.

In the action, select the **service** to authenticate to, and then **paste the token** to the first parameter of the action:



**Note:** to simplify access to the current SharePoint site to interact with its lists and libraries, HarePoint Workflow Manager Extensions implement a special mechanism called **Autologon**. With the help of this mechanism, in most cases using the Set Authentication action + token for the current SharePoint site is not needed in a workflow.

However, it may still be required in some cases, so if the workflow fails and history log indicates “Authentication is required”, simply generate a token and add the Set Authentication action to the workflow.

## VIII. Licensing

### Trial mode

If the product is installed for the first time on this farm, the **30-days** trial mode will be activated.

The trial mode is fully functional, and HarePoint Workflow Manager Extensions actions can be used without any limitations.

### Licensing model

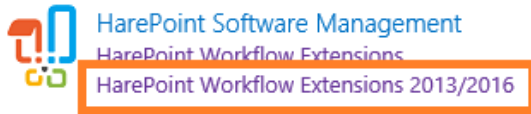
The product is licensed based on the number of **Web Front-End\*** servers in your farm. Each server requires a separate license key that needs to be entered on a licensing page.

\*The server is identified as Web Front-End server if it has “SharePoint Foundation Web Application” service running (Central Administration – Manage servers in this farm). In some cases, this service may be running also on Application servers – please contact [HarePoint Support](#) for further guidance.

The number of workflows using HarePoint Workflow Manager Extensions is not limited.

## Entering license keys

The licensing page for HarePoint Workflow Manager Extensions is located in **Central Administration** under **HarePoint Software Management**:



Click the **HarePoint Workflow Extensions 2013/2016** link:

## HarePoint Workflow Extensions 2013/2016

A screenshot of the HarePoint Workflow Extensions 2013/2016 licensing page. The page has a light gray background. On the left, there is a section titled 'Licenses' with the subtitle 'Specify license keys for HarePoint Workflow Extensions'. On the right, there is a section titled 'Version: 1.0.21328' with the subtitle '1 server was found where the product is installed.' Below this, there is a table with one row containing the text 'CT34' followed by a blurred license key and 'PP'. The table is enclosed in a thin gray border.

Note the **number of servers** detected, each requiring a separate license key (see [previous paragraph](#) for more details).

Enter the corresponding number of license keys, delimited by a semicolon (;) then click **Save**.

**Note:** if there are less license keys entered than number of servers detected, the product will switch either to **trial mode** or to **trial mode expired**. With the trial mode expired, the product is not functional.

For any questions regarding the licensing or entering license keys, please contact [HarePoint Technical Support](#).

## IX. Creating a workflow in SharePoint Designer

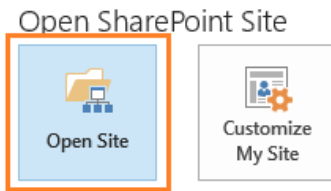
### Connecting SharePoint Designer

As soon as the product is installed, you can start creating workflows that will use HarePoint Workflow Manager Extensions actions.

This can be done using the standard tool – **SharePoint Designer**, or a freeware graphical designer from HarePoint – [HarePoint Workflow Designer](#).

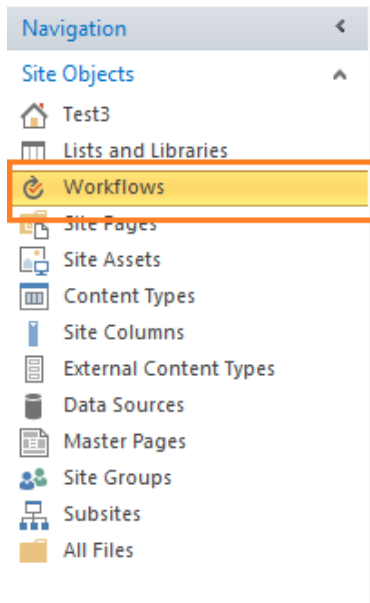
Launch **SharePoint Designer** and click **Open Site**:



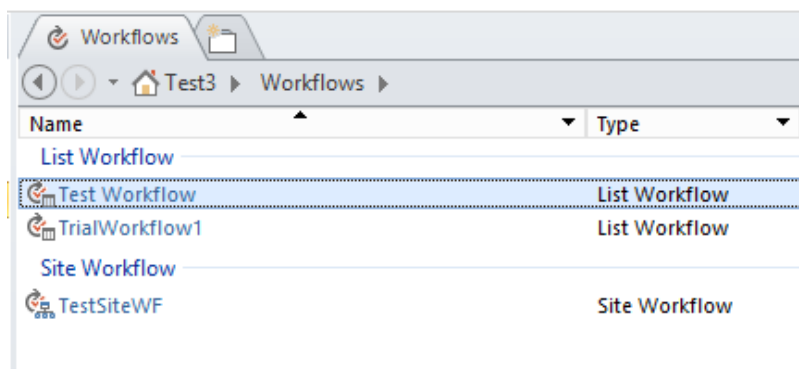


Enter the URL of the SharePoint site.

Click on **Workflows** in the **Navigation** pane:



Your existing workflows (if any) will be displayed in the right pane:



## Creating a workflow

Decide what workflow type you need, and click the respective button on the ribbon:



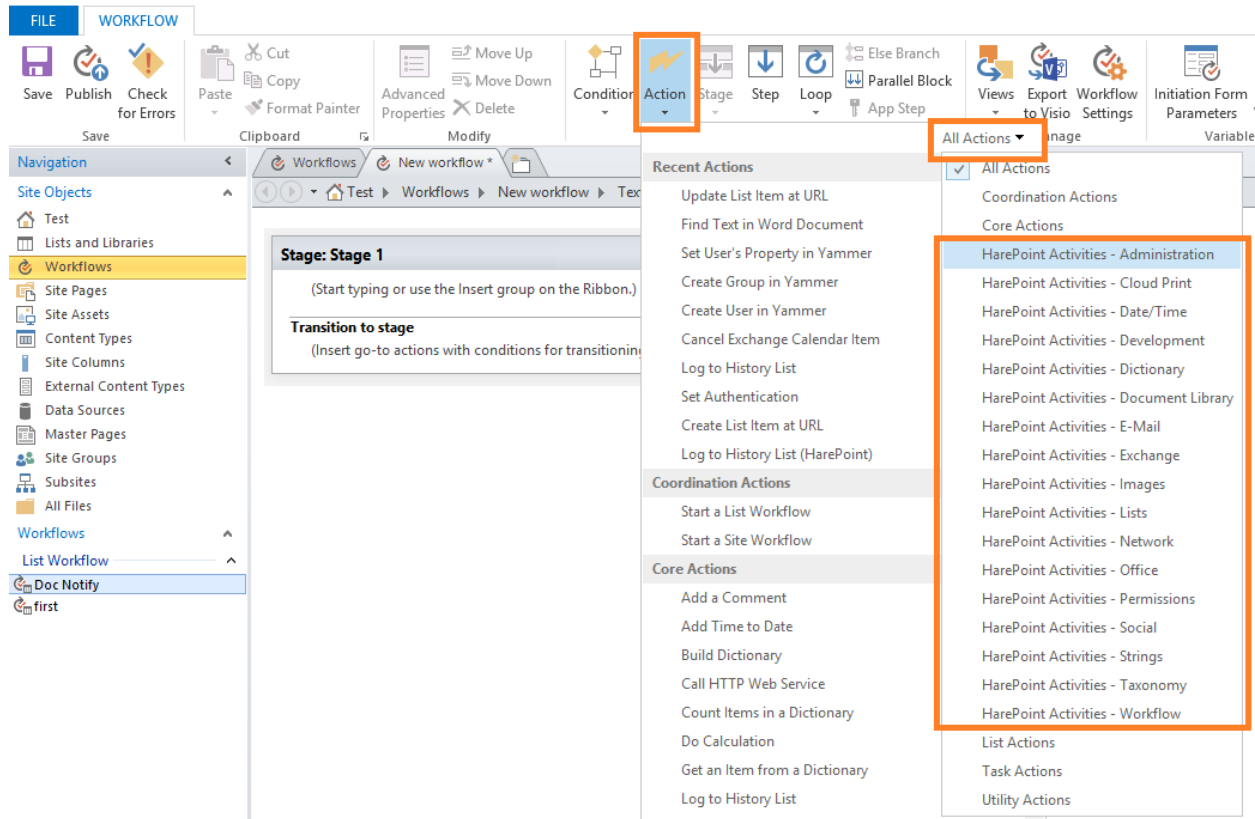
For **List workflow**, point to a **list** or **document library** this workflow should refer to.

Enter the **name** for the new workflow. Make sure the **Platform Type** is set to **SharePoint 2013 Workflow**  
**Workflow:**

A screenshot of a dialog box titled 'Create List Workflow - Documents'. The dialog has a light blue header bar with a question mark icon and a close button (X). Inside the dialog, there is a section titled 'Add a new workflow to your list' with a circular arrow icon. Below this, there is a section titled 'Enter a name and description for your new workflow'. It contains two text input fields: 'Name:' with the text 'New workflow' and 'Description:' with the text 'let's use HarePoint Workflow Extensions!'. Below these fields is a section titled 'Choose the platform to build your workflow on'. It contains a dropdown menu labeled 'Platform Type:' with the selected value 'SharePoint 2013 Workflow'. The dropdown menu is highlighted with an orange border. At the bottom right of the dialog are 'OK' and 'Cancel' buttons.

New workflow will be created and opened for editing.

Go to **Actions** drop-down and click on **All actions** to show all available workflow categories. You will see **19 categories** from HarePoint Workflow Extensions:



Refer to the **detailed description of each workflow action and all its parameters** in the next chapter [List of workflow actions](#).

## X. List of workflow actions

### 1. Active Directory activities

#### 1.1 Add Member to Active Directory Group

This workflow action is used to add one or several users or groups to specified Active Directory security or distribution group.

Mail contacts also can be added.

**Note:** In order to specify AD credentials, add the *Set Authentication* activity to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Add current user to members of specified group in Active Directory.

#### Parameters:

**current user** - Logon name of the user or group: DOMAIN\user or user@domain.com. Leave the field empty to add the current user.

To add **multiple users**, provide their logon names semicolon-delimited:  
mapilab\test1;mapilab\test2;mapilab\test3

To add **mail contacts**, provide the e-mail address of a contact.

**specified** - Logon name of the group: DOMAIN\group or group@domain.com.

## 1.2 Change Active Directory Account Settings

This workflow action is used to enable or disable a user account or to change its settings.

**See Also:** *Create User in Active Directory*

**Note:** In order to specify AD credentials, add the *Set Authentication* action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Change logon name account settings: enable, change password at next logon: keep existing, password never expires: keep existing, account expires: keep existing

#### Parameters:

**logon name** - Logon name of the user: DOMAIN\user or user@domain.com.

**enable/disable** - *Enable or disable* user account.

**change password** - User must change password at next logon: *yes/no* or *keep existing* settings

**password never expires** - Password never expires: *yes/no* or *keep existing* settings.

**account expires: yes** - Select if the account expires: *yes/no* or *keep existing* settings. If Yes is selected, please specify the expiry date in the next parameter.

**on this date** - specify the expiry date of the account, if this account expires (previous parameter set to Yes).

## 1.3 Change Active Directory Group Settings

This workflow action is used to change the Active Directory group settings.

**Note:** In order to specify AD credentials, add the *Set Authentication* action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Change specified group settings, group type: keep existing, description: keep existing, email: keep existing

### Parameters:

**specified group** - specify an AD group name that needs to be changed.

**group type: keep existing** - (optional) if a group type needs to be changed, select the new group type from a drop-down list.

The available options are:

- *universal security group*
- *global security group*
- *domain local security group*
- *universal distribution group*
- *global distribution group*
- *domain local distribution group*

**description: keep existing** - (optional) if a group description needs to be changed, specify the new group description.

**email: keep existing** - (optional) if a group email address needs to be changed, specify the new group email address.

## 1.4 Create Group in Active Directory

This workflow action will create a new group in the specified Active Directory container.

**See also:** "Find Group in Active Directory" action can be used prior to this action to check if the AD group already exists.

**Note:** In order to specify AD credentials, add the Set Authentication action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Create new group in this container with specified group name.

**Parameters:**

**this** - Distinguished name of the Active Directory container: OU=Test,DC=domain,DC=com. Use Active Directory Explorer to get correct path: <http://technet.microsoft.com/en-us/sysinternals/bb963907>

**specified** - Logon name of the group: DOMAIN\group or group@domain.com.

## 1.5 Create Mail Contact in Active Directory

This workflow action creates a new mail contact in Active Directory with specified parameters.

Create new contact in this container with specified name, this display name, this alias, this external e-mail, mail enable: yes.

**Parameters:**

**this container** - Specify the distinguished name or canonical name of the container where the contact will be created.

**with specified name** - Specify the name for the contact object in AD

**this display name** - Specify the name that will be displayed to the user.

**this alias** - (for mail-enabled contacts only) The internal e-mail address of the contact.

**this external e-mail** - External e-mail address (outside of organization).

**mail enable: yes** - Enable (or disable) receiving e-mails to this contact via Microsoft Outlook and Microsoft Exchange.

## 1.6 Create User in Active Directory

This workflow action creates a new user account in a specified Active Directory container. The newly created account will be enabled; you can disable the account and change its settings with *Change Active Directory Account Settings* action.

**Note:** In order to specify AD credentials, add the *Set Authentication* action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Create new user in this container with specified logon name, this first name, this last name, this password. Optional: use this as user principal name.

### Parameters:

**this container** - Distinguished name of the Active Directory container, like OU=Test,DC=domain,DC=com.

**specified** - User's logon name. Domain name in the user's logon name is ignored (correct syntax is DOMAIN\username or username@domain).

**Important note:** Logon name is technically limited to **20 characters**. If you need to have a longer name, specify here only the first 20 characters of user logon name (or just any random characters) and use the optional parameter "user principal name" to specify a longer name, see below.

**this first name** - User's first name.

**this last name** - User's last name.

**this password** - User's password. Password may be blank. Use *Change Active Directory Account Settings* to ask the user to change the password at first login.

**use this as user principal name** - (optional) Specify user principal name that can be used instead of user logon name.

**Note:** user principal name is technically limited to 64 characters.

## 1.7 Delete Group from Active Directory

This workflow action will delete a specified security or distribution group from Active Directory.

**Note:** In order to specify AD credentials, add the *Set Authentication* activity to the workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Delete group with this group name from Active Directory.

**Parameters:**

**this** - Logon name of the group: DOMAIN\group or group@domain.com.

## 1.8 Delete User from Active Directory

This workflow action will delete a specified user from Active Directory.

**Note:** In order to specify AD credentials, add the *Set Authentication* action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Delete user with this logon name from Active Directory.

**Parameters:**

**this** - Logon name of the user: DOMAIN\user or user@domain.com.

## 1.9 Find User in Active Directory

This workflow action is used to find a SharePoint user in Active Directory and store the user's logon name in the output variable. The action returns an empty string if the user is not found in Active Directory.

**Note:** In order to specify AD credentials, add the *Set Authentication* action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Find current user in Active Directory and store logon name in Variable: variable.

**Parameters:**

**current user** - Name of SharePoint user.

**Variable: variable** - Output string variable with a logon name like DOMAIN\username. Or empty string if user is not found in Active Directory.



## 1.10 Find User in Active Directory by Query

This workflow action is used to find a user or group in Active Directory by a specified query. You can use two type of queries.

First is **simple query** - the format in the style of ab INI file:

```
<AD_attribute_name1>=<value1>  
<AD_attribute_name2>=<value2>
```

For example:

```
displayName=John Smith
```

Second, is **LDAP format**. For example:

```
(&(objectCategory=user)(objectClass=user)(displayName=John Smith)(department=Marketing))
```

Find user in Active Directory by simple query using this query and store logon name in Variable: variable.

### Parameters:

**simple query** - One of the following query types:

- *simple query* (for .INI-style query)
- *LDAP filter* (for LDAP query)

**this query** - String with the query.

**Variable: variable** - Output string variable with a logon name like DOMAIN\username. Empty string if user is not found in Active Directory.

## 1.11 Get Active Directory Groups Where User is Member

This workflow action will return the list of groups where specified user is a member.

**Note:** In order to specify AD credentials, add the *Set Authentication* activity to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Get list of the Active Directory groups where user current user is member. Store list of these groups in Variable: GroupsDictionary, success of the searching in Variable: UserFound.

### Parameters:

**current user** - The name of Active Directory user for reading names of his groups (like DOMAIN\Account). Leave the field empty to use current user.

**Variable: GroupsArray** - Output dictionary variable with the list of groups.

**Variable: UserFound** - Output boolean variable, *true* indicates that user was found in Active Directory.

## 1.12 Get E-Mails of Active Directory Group Members

This workflow action returns the email addresses of an Active Directory group's members to a dictionary variable.

**Note:** In order to specify AD credentials, add the *Set Authentication* activity to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Get user e-mails from specified group. Expand nested groups: No. Store result in Variable: EmailsArray.

### Parameters:

**specified group** - Name of the group from which we want to read member e-mails (like: DOMAIN\group).

**expand nested groups** - Include or not members of nested groups.

**Variable: EmailsArray** - Output array variable with the list of the e-mails of Active Directory group members.

## 1.13 Get User's Attribute from Active Directory

This workflow action will return the value of a user's specified Active Directory user attribute.

**Note:** In order to specify AD credentials, add the *Set Authentication* activity to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

**See Also:** *Get User's Multivalued Attribute from Active Directory*.

**Note:** to get attribute of any other AD object, refer to "Get Object's Attribute from Active Directory" action.

Get this attribute of current user from Active Directory. Store attribute value in Variable: variable1, success of searching in Variable: variable2

#### Parameters:

**this attribute** - Name of the attribute.

**current user** - Name of the user from which we want to read the attribute value (like DOMAIN\Account). Leave the field empty to use the current user.

**Variable: variable1** - Output variable for the obtained attribute value.

**Variable: variable2** - Output boolean variable, *true* indicates that the user (not the attribute!) was found in Active Directory.

### 1.14 Get User's Manager from Active Directory

This workflow action is used to get a manager's display name for a specified Active Directory user or a group.

**Note:** In order to specify AD credentials, add the Set Authentication action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Get manager of current user from Active Directory. Store manager's display name in Variable: variable1, success of the searching in Variable: variable2.

#### Parameters:

**current user** - The name of Active Directory user or a group whose manager's name is to be obtained (like DOMAIN\Account). Leave the field empty to use current user.

**Variable: variable1** - Output variable for the obtained manager display name.

**Variable: variable2** - Output boolean variable, *true* indicates that the user or group (not the manager) was found in Active Directory.

### 1.15 Get User's Multivalued Attribute from Active Directory

This workflow action is used to get a multi-valued attribute of a specified Active Directory user.

**Note:** This action returns a dictionary variable.

**See Also:** *Get User's Attribute from Active Directory.*

Get multi-valued attribute this attribute of current user from Active Directory. Store attribute value in Variable: ArrayVariable, success of the searching in Variable: variable.

**Parameters:**

**this attribute** - Name of attribute.

**current user** - User logon name (like DOMAIN\Account) or current user by default.

**Variable: ArrayVariable** - Output dictionary variable with attribute values.

**Variable: variable** - Output boolean variable, *true* indicates that user was found in Active Directory.

## 1.16 Move User into Organizational Unit

This workflow action is used to move a specified AD user into a specified AD organizational unit or container.

**Note:** In order to specify AD credentials, add the Set Authentication action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Move Active Directory user with this logon name into specified organizational unit.

**Parameters:**

**with this logon name** - Logon name of the Active Directory user (e.g. DOMAIN\username).

**specified** - Distinguished name or canonical name of the Organizational Unit or Active Directory container (e.g. OU=Test,DC=company,DC=local or company.local/Users).

## 1.17 Remove Member from Active Directory Group

This workflow action will remove a user or group from an Active Directory security or distribution group.

Mail contacts can also be removed.

Deleting multiple users is supported.

**Note:** In order to specify AD credentials, add the *Set Active Directory Authentication (impersonate)* activity to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

Remove current user from members of specified group in Active Directory.

**Parameters:**

**current user** - Logon name of the user or group: DOMAIN\user or user@domain.com. Leave the field empty to use the current user.

To delete **multiple** users, specify them delimiting by a semicolon (;).

Example: DOMAIN\user1; DOMAIN\user2; DOMAIN\user3

**specified** - Logon name of the group: DOMAIN\group or group@domain.com.

To remove a **mail contact**, specify its e-mail address.

**Examples:**

Remove COMPANY\jsmith from finance@company.com group in Active Directory

## 1.18 Set Active Directory Options

This workflow action is used to specify where the users should be searched in.

Set Active Directory Options: Search in the current forest with specified DNS name .

**Parameters:**

**current forest** - Specify the search scope to find the necessary user. Possible values:

- *current*
- *current and trusted*
- *custom* - in this case, additionally specify the forest DNS name

**with specified DNS name** - specify the DNS name of the AD forest.

## 1.19 Set User's Attribute in Active Directory

This workflow action is used to set an Active Directory attribute of a specified user.

**Note:** In order to specify AD credentials, add the *Set Authentication* action to the same workflow step first. Otherwise, AD will be accessed as the user who started the workflow or as the workflow author in the case of an impersonation step.

**See Also:** *Set User's Multivalued Attribute in Active Directory.*

Set this attribute of current user in Active Directory to this value.

**Parameters:**

**this attribute** - specify the name of AD attribute to be set.

**current user** - specify the user whose attribute will be changed

**this value** - specify the new value the AD attribute.

## 1.20 Set User's Multivalued Attribute in Active Directory

This workflow action is used to set a multi-valued Active Directory attribute for a specified user.

**Note:** This action uses a dictionary variable.

**Tip:** This action can be used to set AD attribute to empty (null). To do this, create an empty dictionary, then set the AD attribute to this array.

**See Also:** *Set User's Attribute in Active Directory.*

Set multi-valued attribute this attribute of current user in Active Directory to this value.

**Parameters:**

**this attribute** - Name of attribute.

**current user** - User logon name (like DOMAIN\Account) or current user by default.

**this value** - Dictionary variable with values.

## 1.21 Set User's Password in Active Directory

This workflow action is used to set the password for a specified Active Directory user.

Set logon name account password in Active Directory to this password.

**Parameters:**

**logon name** - Logon name of the user: DOMAIN\user or user@domain.com.

**this password** - String with new password.

## 1.22 Set User's Photo in Active Directory

This workflow action is used to set or remove a photo in the Active Directory user's profile ('thumbnailPicture' attribute).

JPG image format is required.

**Note:** The size of the photo is limited to 100KB.

Set current user's photo in Active Directory to photo at this url.

### Parameters:

**current user** - specify the user whose photo will be changed

**photo at this url** - specify the URL of the photo (stored in SharePoint document library) that will be uploaded to the user's AD profile

## 2. Administration activities

### 2.1 Activate Feature by GUID

This workflow action is used to activate a site or site collection feature by its GUID (forcibly, if necessary).

Activate feature with this GUID for the Site at this URL . Force: Yes.

### Parameters:

**this GUID** - the GUID of the feature to be activated.

**Site** - select whether the feature belongs to a *Site* or *Site Collection*.

**this URL** - the site or site collection URL.

**Yes** - select if the feature should be installed in Force mode. (*Yes/No*)

### 2.2 Add User to SharePoint Group

This action adds one or more users to one or more SharePoint groups.

Add specified users to specified group at this site.

#### Parameters:

**specified** users - One or more logon names of users separated by semicolon.

**specified** group - Group name, as displayed under People and Groups \ All Groups.

This parameter also accepts **arrays of groups**. In this case, user(s) will be added to all groups from the array.

**this** - Full URL of a site.

### 2.3 Add User to SharePoint Site

This workflow action is used to add local or Active Directory users to a SharePoint site.

Add specified users to this site

#### Parameters:

**specified users** - One or more logon user names separated by semicolons (;). For example, DOMAIN\account or MACHINE\account.

**this site** - Full URL of the site.

### 2.4 Alternate CSS and Site Logo

This workflow action is used to set alternative CSS and logo to a site.

Set alternate CSS and site logo to this site. CSS file this, logo file this.

#### Parameters:

**this site** - site URL

**CSS file this** - specify the CSS file to be applied

**logo file this** - specify the logo file to be applied

### 2.5 Change SharePoint Group Owner

This workflow action is used to change the owner of a SharePoint group.



Change specified group owner at current site url to current user account

**Parameters:**

**specified group** - specify the SharePoint group name

**current site url** - (optional) specify the address of the site where the group should be searched at.

By default, the current site is used.

**current user** - (optional) specify the name of the account or the group that should be set as the group owner.

By default, current user will be set as the group owner.

## 2.6 Change SharePoint Group Settings

This workflow action is used to change the settings of a SharePoint group.

Change specified group settings at current site url, about me: keep existing, view membership: keep existing, edit membership: keep existing, allow membership requests: keep existing, auto-accept requests: keep existing.

**Parameters:**

**specified group** - type in the name of SharePoint group to change.

**current site url** - keep it default to point to a current site, or specify the URL of a different site where the SharePoint group belongs to.

**about me: keep existing** - (optional) specify the new text for group description.

**view membership: keep existing** - (optional) specify who can view members of this group:

- *keep existing*
- *group members*
- *everyone*

**edit membership: keep existing** - (optional) specify who can edit group membership:

- *keep existing*
- *group owner*

- *group members*

**allow membership requests:** keep existing - (optional) specify if membership requests are allowed:

- *keep existing*
- *yes* - in this case additional parameter to this email appears
- *no*

**to this e-mail** - (only if previous parameter is set to *yes*) specify e-mail address for membership requests

**auto-accept requests:** keep existing - (optional) specify if membership requests should be automatically accepted:

- *keep existing*
- *yes*
- *no*

## 2.7 Change SharePoint List Settings

This workflow action is used to change the settings of SharePoint list.

Change list settings at this list url , require approval: keep existing , require checkout: keep existing , create versions: keep existing , enable content types: keep existing , experience: keep existing

### Parameters:

**this list url** - Specify the full path to the list or document library, which settings need to be changed.

**require approval** - Specify whether new items or changes to existing items should remain in a draft state until they have been approved.

*Keep existing, yes, no*

**require checkout** - Specify whether users must check out documents before making changes in this document library.

*Keep existing, yes, no*

**create versions** - Specify whether a version is created each time you edit a file in this document library.

*Keep existing, major, major and minor*

**enable content types** - Specify whether to allow the management of content types on this document library.

*Keep existing, yes, no*

**experience** - specify which user experience should be used for this list or library.

*Keep existing, auto, new experience, classic experience*

## 2.8 Create SharePoint Group

This workflow action creates a SharePoint group with specified parameters.

Create specified group with this description at site url and make current user the owner, can view: Group Members , can edit: Group Owner .

### Parameters:

**specified group** - enter the group name.

**this description** - the description for this group.

**site url** - the full site URL where the group should be created.

**current user** - (optional) specify the name of account or a group that will be set as this group owner.

By default, the current user will be set as the owner.

**can view: Group Members** - specify an account or group which can view the members of this group.

**can edit: Group Owner** - specify an account or a group which can edit group membership.

## 2.9 Create SharePoint Site

This action allows you to create a new site anywhere within the SharePoint environment.

Create new site with specified title at this URL, using this template and this description, use unique permissions No.

### Parameters:

**specified** - Title of the new site.

**this url** - Full URL for the new site.

**this** template - Template for the new site. It might be the **name** of a site template, e.g. Team Site, Blank Site, Blog and etc., or it might be the **ID** of a site template, e.g. STS#0, STS#1 and so on.

**this** description - New site description.

**No** - By default, the new site inherits permissions from its parent Site or the Site Collection. Unique permissions can be created. If you select No, a single set of user permissions will be shared by both sites, and user permissions on the new site will be editable only by an administrator of the parent site.

## 2.10 Create Site Collection

This action workflow action creates a new site collection.

Create new site collection with specified title at this url , using this template and this owner. Site collection storage quota 1000 and resource quota 1000 .

### Parameters:

**specified** title - Title of new site collection.

**this url** - URL of new site collection.

**this** template - Template for the new site. It might be **name** of site template, e.g. Team Site, Blank Site, Blog and etc., or it might be the **ID** of a site template, e.g. STS#0, STS#1, and so on.

**this owner** - specify the account to be owner of this site collection.

storage quota 1000 - (in Megabytes) specify the storage quota for the new site collection.

resource quota 1000 - (in Megabytes) specify the resource quota for the new site collection.

## 2.11 Deactivate Feature By GUID

This workflow action is used to deactivate Site or Site Collection features based on their GUID. Deactivation can be forced, if necessary.

Deactivate feature with this GUID for the Site at this URL . Force: Yes .

### Parameters:

**this GUID** - specify the GUID of the feature to be deactivated.

**Site** - select the scope of the feature: *Site* or *Site Collection*.

**this URL** - specify the URL of the Site or Site Collection where a feature needs to be deactivated.

**Yes** - choose whether deactivation should be forced: *Yes, No*.

## 2.12 Delete SharePoint Group

This workflow action is used to delete a SharePoint group from a specified site.

Delete specified group from this site url.

### Parameters:

**specified** - Name of group.

**site url** - Full URL of a site.

## 2.13 Delete SharePoint Site

This action deletes a site anywhere within the SharePoint environment.

Delete site at this url.

### Parameters:

**this url** - Full URL of site to delete.

## 2.14 Delete Site Collection

This workflow action deletes a site collection.

Delete site collection at this url.

### Parameters:

**this url** - Full URL to site collection.

## 2.15 Delete Site Collection from Recycle Bin

This workflow action deletes a Site Collection permanently by removing it from the Recycle Bin.

Delete site collection at this url.

#### Parameters:

**this url** - Full URL of the site collection.

## 2.16 Deploy Theme To Web

This workflow action is used to deploy a custom theme to a SharePoint Site.

Deploy theme with this name to this site. Theme file path , font file path , image file path , master page name .

#### Parameters:

**this name** - enter the name of the theme.

**this site** - specify the URL of the site where the theme will be deployed.

**theme file path** - specify the path of the theme file.

**font file path** - specify the path of the font file.

**image file path** - specify the path of the image file.

**master page name** - specify the name of the master page.

## 2.17 Get SharePoint User Membership Status

This workflow action is used to check if a specified user is a member of a SharePoint group.

If this user is member of this SharePoint group, set Variable: isMember to yes

#### Parameters:

**this user** - specify the user name as DOMAIN\username

**this SharePoint group** - specify the name of SharePoint group

**Variable: isMember** - specify the output variable

## 2.18 Get Site User MultiValued Property

This workflow action is used to get a specified multi-value property from a user's profile on the given site. This action returns an empty string if the property is not found.

A list of all available properties is located in Central Administration \ Manage User Profile Properties.

Get (Custom) specified property for this user and store result in Variable: userProperty

### Parameters:

**(Custom)** - Select the property from the drop-down list, or select *Custom* and specify it in the next parameter

**specified** - Property name (not the display name of the property!).

**this user** - User whose property should be received.

**Variable: userProperty** - Output string variable.

## 2.19 Get Site User SingleValue Property

This workflow action is used to get a specified single-value property from a user's profile on the site. This action returns an empty string if the property is not found.

You can get the list of all available properties in Central Administration \ Manage User Profile Properties.

Get (Custom) specified property for this user and store result in Variable: userProperty

### Parameters:

**(Custom)** - Select the property from the drop-down list, or select *Custom* and specify it in the next parameter

**specified** - Property name (not the display name of the property!).

**this user** - User whose property should be received.

**Variable: userProperty** - Output string variable.

## 2.20 Inject Custom CSS

This workflow action is used to inject custom CSS in a site.

Inject custom CSS to this site. CSS file this and this action name.

### Parameters:

**this site** - specify the URL of the site where CSS needs to be injected.

**CSS file this** - specify the URL of the CSS file.

**this action name** - specify the action name.

## 2.21 Remove CSS and Site Logo

This workflow action is used to remove CSS and the logo from the specific site.

Remove CSS and site logo from this site.

### Parameters:

**this site** - specify the URL of the site.

## 2.22 Remove Custom Action

This workflow action is used to remove a custom action from the Site Actions menu on a specified site.

Remove custom action with this name at this site.

### Parameters:

**this name** - specify the name of the custom action.

**this site** - specify the URL of the site.

## 2.23 Remove User from SharePoint Group

This workflow action is used to remove a user from a SharePoint group.

Remove specified user from specified group at this site.

### Parameters:



**specified user** - The login name of the user to be removed.

**specified group** - Group name, as it displayed in People and Groups \ All Groups.

**this** - Full URL of site.

## 2.24 Remove User from SharePoint Site

This workflow action is used to remove user from SharePoint site and from all site groups.

Remove specified user from this site.

### Parameters:

**specified** - the login name of the user.

**this** - Full URL of a site.

## 2.25 Rename Site at Url

This workflow action is used to set a new title for a site specified by URL anywhere within the SharePoint environment.

Rename site at this url to this title.

### Parameters:

**this url** - Full URL of site.

**this title** - New title.

## 2.26 Set Master Page

This workflow action sets the master page for the current or any other site.

Set this master page as default for the current site.

### Parameters:

**this masterpage** - specify the name of the masterpage or its URL relative to the server.

### Examples:

- *v4.master*
- *default.master*
- */\_catalogs/masterpage/v4.master*

**the current site** - (Optional) specify the absolute URL of the site to set where the masterpage is being set. Use *current site* to set the masterpage for the current site.

## 2.27 Set SharePoint Admin Center URL

This workflow action is used to specify the URL of SharePoint Admin Center, to be referred to by the next actions in the workflow.

Use this url as SharePoint admin center url

### Parameters:

**this url** - specify the full URL of the SharePoint Admin Center

## 2.28 Set SharePoint Site Navigation Options

This workflow action sets the navigation options for a SharePoint site (such as Quick Launch, Top Link Bar).

Set options for site at this url, show on parent's Quick Launch: don't change , show on parent's top link bar: don't change , inherit parent's top link bar: don't change

### Parameters:

**site at this url** - specify the URL of the site.

**show on parent's Quick Launch: don't change**- specify whether you need to display or hide this site from the Quick Launch of the parent site.

**Available values:** *yes, no, don't change* (keep current setting)

**show on parent's top link bar: don't change** - specify whether you need to display or hide this site from the Top Link Bar of the parent site.

**Available values:** *yes, no, don't change* (keep current setting)

**inherit parent's top link bar: don't change** - choose to display the Top Link Bar from the parent site or display a Top Link Bar for this particular site.

**Available values:** *yes, no, don't change* (keep current setting)

## 2.29 Set Site Theme Based On Name

This workflow action allows you to set the site theme based on a theme name.

Set this site theme based on this name.

### Parameters:

**this site** - specify the URL of the site.

**this name** - specify the name of the theme to be set.

## 3. Cloud Print activities

### 3.1 Delete Print Job

This workflow action deletes a print job with a specified ID.

Delete print job with this id.

### Parameters:

**this id** - specify the ID of the print job that needs to be deleted.

### 3.2 Get Print Job Status

This workflow action is used to get the status of a specified print job.

Get status of print job with this id and store status in jobStatus.

### Parameters:

**this id** - specify the ID of the print job to get its status.

The print job ID can be obtained from the *Submit Job* action.

**jobStatus** - specify the variable to store the print job status.

### 3.3 Get Printer Status

This workflow action is used to get the status of a cloud printer by its name or ID.

Get status of printer name or id of printer and store status in printerStatus .

#### Parameters:

**printer name or id of printer** - specify either the cloud printer name or cloud printer ID to get its status.

**printerStatus** - specify the variable to store the cloud printer status.

### 3.4 Set Printing Options

This workflow action is used to set printing options for a cloud print job.

Set number of copies: 1 , page range: all , Portrait orientation, Do not fit to page , 2-sided printing: No and print in Color .

#### Parameters:

**number of copies: 1** - specify the number of document copies to be printed.

**page range: all** - specify the page range to be printed. By default, all pages will be printed.

**Portrait orientation** - specify the orientation of the document:

- *Portrait*
- *Landscape*

**Do not fit to page** - specify the document scaling:

- *Do not fit to page*
- *Fit to page*
- *Grow to page*
- *Shrink to page*
- *Fill page*

**2-sided printing: No** - specify whether the document needs should be printed on both sides of the paper:

- *No*
- *Long edge*
- *Short edge*

**print in Color** - select the color mode for printing:

- *Color*
- *Monochrome*
- *Auto*

### 3.5 Submit Print Job

This workflow action submits a cloud print job.

Print document from this URL with default content type on the specified printer . Store job's id to jobId .

**Parameters:**

**this URL** - specify the URL of the document to be printed.

**default content type** - (optional) specify the content type of the document to be printed.

**specified printer** - specify the printer name or printer ID.

**jobId** - specify the variable to store the print job ID.

This Job ID is used to get the status of this print job - **see also** Get Job Status action.

## 4. Conversion activities

### 4.1 Convert Document

This workflow action converts the specified document to a different format and stores the output file in a document library.

This action processes files only of supported formats; files of other formats will be skipped. The list of supported input formats:

- *Open XML Document (.docx, .docm, .dotx, .dotm)*
- *Word 97-2003 Document (.doc, .dot)*
- *Rich Text Format (.rtf)*
- *Web Page (.htm, .html, .mht, .mhtml)*
- *Word 2003 XML Document (.xml)*

The list of supported output formats:

- *Word Document (docx)*
- *Word 97-2003 Document (doc)*
- *PDF*
- *PDF/A*
- *RTF*
- *XPS*

**Note:** This action requires "Word Automation Services" running on the server. To check this service, open Central Administration, go to System Settings and open the "Manage services on server" page. Unfortunately, this service is not available on SharePoint Foundation.

Convert document at this URL to PDF format and save to the folder at specified URL and with the same name. Overwrite existing document

#### Parameters:

**this URL** - specify the full URL of the input document

**PDF** - specify the output format:

- *Word Document (docx)*
- *Word 97-2003 Document (doc)*
- *PDF*
- *PDF/A*
- *RTF*
- *XPS*

**folder at specified URL** - specify the URL of a document library, or a folder within a document library to store the output file.

**the same name** - choose if the output file should have the same name as the original file, or specify a name directly.

**Overwrite** - choose if a document should be overwritten, if a file with such name already exists in the target folder.

## 5. Date/Time activities

### 5.1 Convert Date from Local Time to UTC

This workflow action converts a local date/time variable to a Coordinated Universal Time (UTC) date/time. The UTC is equal to the local time minus the UTC offset taken from the site's Regional Settings. The conversion also takes into account the daylight-saving time rule.

Convert specified date from Local Time to UTC and store result in Variable: convertedToUTCDate

#### Parameters:

**specified date** - Input date or Date/Time variable.

**Variable: convertedToUTCDate** - Output Date/Time variable.

### 5.2 Convert Date from UTC to Local Time

This workflow action is used to convert Coordinated Universal Time (UTC) to local time. The local time is equal to the UTC time plus the UTC offset taken from the site's Regional Settings. The conversion also takes into account the daylight-saving time rule.

Convert specified date from UTC to Local Time and store result in Variable: convertedToLocalTimeDate

#### Parameters:

**specified date** - Input UTC date or date/time variable

**Variable: convertedToLocalTimeDate** - Output local time/date variable

## 5.3 Convert Date to Text

This workflow action converts a date/time variable to a string using a specified format.

A detailed description of format string syntax for each type is contained in following articles:

- Standard DateTime Format Strings - [http://msdn.microsoft.com/en-us/library/aa720651\(v=VS.71\).aspx](http://msdn.microsoft.com/en-us/library/aa720651(v=VS.71).aspx)
- Custom DateTime Format Strings - [http://msdn.microsoft.com/en-us/library/aa719856\(v=VS.71\).aspx](http://msdn.microsoft.com/en-us/library/aa719856(v=VS.71).aspx)
- DateTime Formatting Sample - [http://msdn.microsoft.com/en-us/library/aa326720\(v=VS.71\).aspx](http://msdn.microsoft.com/en-us/library/aa326720(v=VS.71).aspx)

In the case of a wrongly formatted string, the date will be converted with the default format and a warning message will be logged in the workflow history log.

Convert specified date to text with specified format and store result in Variable: textDate

### Parameters:

**specified date** - Input date/time variable.

**specified format** - Format string.

**Variable: textDate** - Output string variable.

### Examples:

Set DateTime constant Now and store result in Variable: d

then Convert Variable: d to text with MMMM dd yyyy HH:mm:ss.ffffff and store result in Variable: strDate

then Get Variable: d day property and store result in Variable: day

then Get Variable: d day of week property and store result in Variable: dow

then Get Variable: d day of year property and store result in Variable: doy

then Get Variable: d hour property and store result in Variable: h

then Get Variable: d millisecond property and store result in Variable: ms

then Get Variable: d minute property and store result in Variable: min

then Get Variable: d month property and store result in Variable: m



then Get Variable: d second property and store result in Variable: sec

then Get Variable: d year property and store result in Variable: y

then Log [%Variable: strDate%] : [%Variable: day%], [%Variable: dow%], [%Variable: doy%], [%Variable: h%], [%Variable: ms%], [%Variable: min%], [%Variable: m%], [%Variable: sec%], [%Variable: y%] to the workflow history list

Result in workflow history file:

April 25 2011 14:16:26.1626432 : 25, 2, 115, 14, 162, 16, 4, 26, 2011

## 5.4 Evaluate Due Date

This workflow action calculates a due date by adding a specified amount of time to a starting date while excluding non-working hours according to the site's Regional Settings.

Evaluate due date from this time for this many hours, 5 minutes and store result in Variable: dueDate

### Parameters:

**this time** - Starting date/time.

**this many** hours - Duration in hours.

**5** minutes - Duration in minutes.

**Variable: dueDate** - Output date/time variable.

## 5.5 Get Date Property

This workflow action is used to get one of the numeric values (like year, or day of the week) of a date/time variable.

See examples section for details.

Get this date day property and store result in Variable: dateProperty.

### Parameters:

**this date** - Input date/time variable.

**day** - What property of the date to store, available values are: day, day of week, day of year, hour, millisecond, minute, month, second, and year.

See examples section for details.

**Variable: dateProperty** - Output integer variable.

#### Examples:

*Set DateTime constant Now and store result in Variable: d*

*then Convert Variable: d to text with MMMM dd yyyy HH:MM:ss.ffffff and store result in Variable: strDate*

*then Get Variable: d day property and store result in Variable: day*

*then Get Variable: d day of week property and store result in Variable: dow*

*then Get Variable: d day of year property and store result in Variable: doy*

*then Get Variable: d hour property and store result in Variable: h*

*then Get Variable: d millisecond property and store result in Variable: ms*

*then Get Variable: d minute property and store result in Variable: min*

*then Get Variable: d month property and store result in Variable: m*

*then Get Variable: d second property and store result in Variable: sec*

*then Get Variable: d year property and store result in Variable: y*

*then Log [%Variable: strDate%] : [%Variable: day%], [%Variable: dow%], [%Variable: doy%], [%Variable: h%], [%Variable: ms%], [%Variable: min%], [%Variable: m%], [%Variable: sec%], [%Variable: y%] to the workflow history list*

Result in workflow history list:

April 25 2011 14:04:26.1626432 : 25, 2, 115, 14, 162, 16, 4, 26, 2011

## 5.6 Get Interval Between Dates

This workflow action calculates the interval between two specified dates in specified units and saves the result to a variable.

Get minutes between this date and this date, store interval in Variable: interval

#### Parameters:

get **minutes** - select the unit for the time interval. Possible values: *minutes, hours, days, weeks, months, years*.

between **this date** - the starting date

and **this date** - the ending date

**Variable: interval** - the Number variable where the result will be stored

## 5.7 Get Working Time Between Dates

This workflow action is used to calculate the working days or working hours between the two dates.

The working days and hours are taken from **Regional Settings** of a SharePoint site.

Get working days between this date and this date, store interval in Variable: interval

### Parameters:

**days** - choose to calculate *days* or *hours*.

In *days* mode, the time is not taken into account; both start and end days are included.

In *hours* mode, the time is taken into account.

**Example:** "Monday 00:00 till Friday 00:00" will return 32 working hours. In order to get 40 working hours, specify "Monday 00:00 till Friday 23:59".

**this date** - specify the start date

**this date** - specify the end date

**Variable: interval** - specify a variable to store the output

## 5.8 Set DateTime Constant

This workflow action is used to initialize date/time variables. See parameters and the example section for details.

Set DateTime constant **Now** and store result in **Variable: dateConstant**

### Parameters:

**Now** - One of the following date/time values: *Now, UtcNow, Today, Yesterday, Tomorrow, Beginning of this Week, End of this Week, Beginning of next Week, End of next Week, Beginning of this Month, End of this Month, Beginning of next Month, End of next Month, Beginning of this Year, End of this Year, Beginning of next Year, End of next Year*. See examples section for details.

**Variable: dateConstant** - Output date/time variable.

#### Examples:

Set DateTime constant Now and store result in Variable: vNow

then Set DateTime constant Today and store result in Variable: vToday

then Set DateTime constant End of this Week and store result in Variable: vEndOfWeek

then Set DateTime constant Beginning of next Year and store result in Variable: vBeginningNextYear

then Log [%Variable: vNow%] \* [%Variable: vToday%] \* [%Variable: vEndOfWeek%] \* [%Variable: vBeginningNextYear%] to the workflow history list

Result in workflow history list:

4/25/2011 2:56:12 PM \* 4/25/2011 12:00:00 AM \* 4/30/2011 12:00:00 AM \* 1/1/2012 12:00:00 AM

## 6. Development activities

### 6.1 Convert JSON to XML

This workflow action is used to convert a JSON string to an XML document.

Convert this JSON to XML and store result in Variable: XMLOutput

#### Parameters:

**this JSON** - specify the JSON string to be converted.

**Note:** The JSON string should contain the root element.

#### Example:

```
{ "table": { "item":  
[{"Column1": "Value1", "Column2": "12.03.2022", "Column3": "User1"},
```

```
{"Column1":"Value2","Column2":"10.02.2022","Column3":"User2"}]
```

**Variable: XMLOutput** - specify the string variable to store the XML output

## 6.2 Execute PowerShell Script

This workflow action is used to execute a custom PowerShell script within a workflow.

You can also run **.ps1** files - see *Examples* section for more details.

In the code, the object named `$Context` is available. With this object, you can easily get access to workflow context, the output Result variable or to any other workflow variable, activity execution context, or the workflow history list (using `Write-Host`).

Additionally, two variables with user credentials are available in the code: `$credential` - NetworkCredential object, `$pscredential` - PSCredential object.

See *Examples* section below.

**Note:** use **"Set Authentication"** action to specify the account that will be used to run the script.

Execute this PowerShell script and store result in Variable: results.

### Parameters:

**this PowerShell script** - PowerShell script, see example below.

**Variable: results** - Output variable. This variable is accessible in the code as `$Context.Result`.

### Examples:

#### #1

*Set encrypted passwords with sfDdv1yb+3... string*

*Execute `$x = $Context.GetWorkflowVariable("v...` and store result in Variable: results. Run as encrypted password account.*

The complete code for Execute PowerShell Script action is the following:

```
$x = $Context.GetWorkflowVariable("v1")
$Context.LogComment("test comment")
Write-Host "test"
$Context.Result = "result string"
```

```
$Context.SetWorkflowVariable("v1", "sample_value")
Write-Host $Context.WorkflowContext.Web.Title
Get-ExecutionPolicy -Scope Process
Get-ExecutionPolicy -Scope CurrentUser
Get-ExecutionPolicy -Scope LocalMachine
Get-ExecutionPolicy
```

## #2

Run a **.ps1** file using this workflow action:

```
C:\test\test.ps1 $Context
```

In this case, the **\$Context** object will be available in a PowerShell script.

To use **\$Context** within this **.ps1** file, add this line to the beginning of this file:

```
$context = $args[0]
```

You can then refer to workflow variables in **.ps1** file as follows:

```
$context.GetWorkflowVariable("v1")
```

## #3

If you need to transfer **multiple parameters** to a **.ps1** script, use **space** as delimiter:

```
C:\test\test.ps1 $Variable1 $Variable2
```

To use them in the **.ps1** file, add this line to the beginning of this file:

```
$variable1 = $args[0]
```

```
$variable2 = $args[1]
```

Note: If the parameter itself contains a space, enclose in in **single quotes**:

```
C:\test\test.ps1 $Variable1 'two words'
```

## 6.3 Execute SQL Command

This workflow action is used to execute an SQL command on Microsoft SQL Server or Azure SQL Database.

For an SQL language reference, see your Microsoft SQL Server documentation.

**See Also:** *Log To History List* (HarePoint) to write long strings to the history list.

Execute this SQL command using specified connection string and store result in Variable: outputDictionary, number of records in Variable: recordsAffected

#### Parameters:

**this SQL command** - String, expression or variable with valid SQL command.

**specified connection string** - Database connection string. See your DMBS documentation for connection string format.

**Variable: outputDictionary** - Output dictionary variable for retrieved records. If only one record was retrieved by the SELECT command (value of *recordsAffected* is 1) then this dictionary contains the record. If more than one record was retrieved by the SELECT command (value of *recordsAffected* is 2 or greater) then this dictionary contains values of the first column of recordset.

**Variable: recordsAffected** - Output integer variable with the number of records affected by command execution.

#### Examples:

Execute SELECT TOP 1 \* FROM Table1 using  
Server=tcp:myazuredb1.database.windows.net,1433;Database=mydb1;User  
ID=user@myazuredb1;Password=qwerty;Trusted Connection=False;Encrypt=True;Connection  
Timeout=30; and store result in Variable: outputDictionary, number of records in Variable: recordsAffected

Log [%Variable: outputDictionary%] [%Variable: recordsAffected%] to the workflow history list

## 6.4 Execute SQL Command into List

This workflow action is used to execute an SQL command on Microsoft SQL Server or Azure SQL Database and store the results in a specified SharePoint list. Each row from the results represents a new list item. You can specify alternate column mapping in parameters.

For example, you have a list with two columns named *Title* and *Due Date*. And you have the following SQL query: *SELECT Name, Date from Table*. Then you can set column mapping to the following:

*Name=Title*

*Date=Due Date*

**Note:** for SharePoint **lookup fields**, the data should be provided in the data source as '*ID;#Value*'

Execute this SQL command using specified connection string and add results into list at this url using this column mapping, store number of records in Variable: recordsAffected

**Parameters:**

**this SQL command** - String, expression or variable with valid SQL command.

**specified connection string** - Database connection string. See your DMBS documentation for connection string format.

**this url** - Full URL of the list where items should be added.

**this column mapping** - (Optional) Multiline string with alternate column mapping, each on new line, in the following format:

*DB\_Column\_Name1=SP\_Column\_Name1*

*DB\_Column\_Name2=SP\_Column\_Name2*

**Variable: recordsAffected** - Output integer variable with the number of records affected by command execution.

## 6.5 Query XML

This workflow action is used to query an XML document using XPath query language.

**See Also:**

- XPath Language Reference: <http://www.w3.org/TR/xpath/>
- XPath Online Tutorial: <http://www.w3schools.com/xpath/default.asp>
- Online XPath Query Tool: <http://www.bit-101.com/xpath/>

Query this XML using XPath query this query and store resulting text in Variable: queryXml

**Parameters:**

**this XML** - Input string, expression or variable with XML document.

**this query** - XPath query string. See examples section for details.



**text** - One of three available options: *text*, *inner XML* or *outer XML*.

**Variable: queryXml** - Output string variable.

#### Examples:

Send HTTP request to <http://www.ecb.int/stats/eurofxref/eurofxref-daily.xml> with this login and this password. Use GET method with this headers. Store response in Variable: Xml and status code in Variable: Code

then Query Variable: Xml using XPath query `//Cube[@currency='USD']/@rate` and store resulting text in Variable: EuroUsd

then Log European Central Bank EUR to USD conversion rate is: [%Variable: EuroUsd%] to the workflow history list

This sample takes XML document with the currency conversion rates from the European Central Bank, and gets the actual EUR/USD conversion rate.

Result in the workflow history list:

*The request was sent successfully: http://www.ecb.int/stats/eurofxref/eurofxref-daily.xml*

*European Central Bank EUR to USD conversion rate is: 1.4453*

Example of XML document of European Central Bank:

```
<gesmes:Envelope xmlns:gesmes="http://www.gesmes.org/xml/2002-08-01"
xmlns="http://www.ecb.int/vocabulary/2002-08-01/eurofxref">
```

```
<gesmes:subject>Reference rates</gesmes:subject>
```

```
<gesmes:Sender>
```

```
<gesmes:name>European Central Bank</gesmes:name>
```

```
</gesmes:Sender>
```

```
<Cube>
```

```
<Cube time="2011-06-30">
```

```
<Cube currency="USD" rate="1.4453"/>
```

```
<Cube currency="JPY" rate="116.25"/>
```

```
<Cube currency="BGN" rate="1.9558"/>
```

```

<Cube currency="CZK" rate="24.345"/>

<Cube currency="DKK" rate="7.4587"/>

<Cube currency="GBP" rate="0.90255"/>

</Cube>

</Cube>

</gesmes:Envelope>

```

## 6.6 Transform XML

This workflow action is used for XML transformations using XSLT templates.

### See Also:

- XSL Transformations Version 1.0: <http://www.w3.org/TR/xslt>
- Online XSLT Tutorial: <http://www.w3schools.com/xsl/default.asp>
- Online XSLT Design Tool: <http://www.w3schools.com/xsl/tryxslt.asp>  
xmlfile=catalog&xsltfile=catalog

Transform this XML using XSLT this transform and store result in Variable: transformedXML

### Parameters:

**this XML** - Input string, expression or variable with XML document.

**this transform** - Input string, expression or variable with XSLT template.

**Variable: transformedXML** - Output string variable.

### Examples:

Send HTTP request to <http://www.ecb.int/stats/eurofxref/eurofxref-daily.xml> with this login and this password. Use GET method with this headers. Store response in Variable: Xml and status code in Variable: Code

then Transform Variable: Xml using XSLT `<xsl:stylesheet version="1.0" xmlns:xsl=...` and store result in Variable: Html

then Create document from Variable:

Html in <https://testsite/Shared%20Documents/Rates.Html>. Overwrite existing document.

This sample takes an XML document with currency conversion rates from the European Central Bank and generates an HTML document with currency rates in a document library.

For an example of an XML document, please see the examples section of *Query XML* workflow action overview.

Full text of XSLT template for this sample:

```
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:gesmes="http://www.gesmes.org/xml/2002-08-01"
xmlns:cubes="http://www.ecb.int/vocabulary/2002-08-01/eurofxref">
<xsl:template match="/">
  <html>
  <body>
    <h2><xsl:value-of select="//gesmes:name"/>: <xsl:value-of select="//cubes:Cube/@time" /></h2>
    <table border="1">
      <tr bgcolor="#9acd32">
        <th>Currency</th>
        <th>Rate</th>
      </tr>
      <xsl:for-each select="//cubes:Cube/cubes:Cube/cubes:Cube">
        <tr>
          <td><xsl:value-of select="@currency"/></td>
          <td><xsl:value-of select="@rate"/></td>
        </tr>
      </xsl:for-each>
    </table>
  </body>
</html>
```

```
</table>

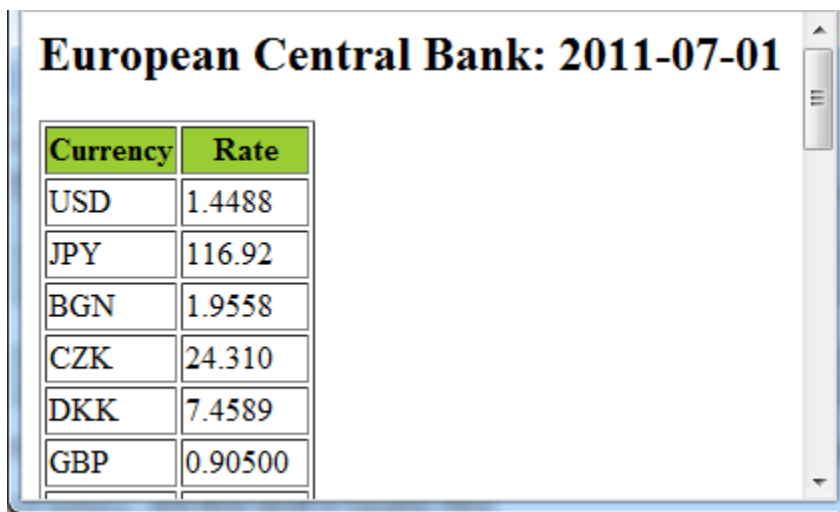
</body>

</html>

</xsl:template>

</xsl:stylesheet>
```

Example of output HTML document:



Currency	Rate
USD	1.4488
JPY	116.92
BGN	1.9558
CZK	24.310
DKK	7.4589
GBP	0.90500

## 6.7 Update XML

This workflow action updates a node value or appends a new child node to an XML document using XPath query.

### See Also:

- XPath Language Reference: <http://www.w3.org/TR/xpath/>
- XPath Online Tutorial: <http://www.w3schools.com/xpath/default.asp>
- Online XPath Query Tool: <http://www.bit-101.com/xpath/>

Update this XML, update node value using XPath this query with this value, store result in Variable: updatedXml

### Parameters:

**this XML** - Input text, expression or variable with XML document.

**update node value** - Two options are available: *update node value* or *append child node*.

**this query** - Input text, expression or variable with XPath query.

**this value** - New node value or new child node XML.

**Variable: updatedXml** - Output string variable for updated XML document.

### Examples:

#### 1. Update node sample

Send HTTP request to <http://www.ecb.int/stats/eurofxref/eurofxref-daily.xml> with this login and this password. Use GET method with this headers. Store response in Variable: Xml and status code in Variable: Code

then Query [%Variable: Xml%] using XPath query //Cube[@currency='USD']/@rate and store resulting text in Variable: EuroUsd

then Set Variable: EuroUsdNumber to Variable: EuroUsd

then Calculate Variable: EuroUsdNumber multiply by 1,1 (Output to Variable: EuroUsdCorrected)

then Update [%Variable: Xml%], update node value using XPath //Cube[@currency='USD']/@rate with [%Variable: EuroUsdCorrected%], store result in Variable: NewXml

then Transform [%Variable: NewXml%] using XSLT <xsl:stylesheet version="1.0" xmlns:xsl=... and store result in Variable: Html

then Create document from Variable:

Html in <http://gorlach7/Shared%20Documents/Rates.Html>. Overwrite existing document.

This sample takes an XML document with currency conversion rates from the European Central Bank, increases the EUR/USD conversion rate by 10%, and generates an HTML document with currency rates in a document library.

Note that the type of EuroUsdNumber variable is Number. For the full text of the XSLT template and an example of output HTML, see the examples section of the *Transform XML* activity overview. For an example of source XML document, see the examples section of the *Query XML* activity overview.

#### 2. Append child node sample

Send HTTP request to <http://www.ecb.int/stats/eurofxref/eurofxref-daily.xml> with this login and this password. Use GET method with this headers. Store response in Variable: Xml and status code in Variable: Code

then Update [%Variable: Xml%], append child node value using XPath //Cube/Cube[@currency='USD'] with <Cube currency="EUR" rate="1.0000"/>, store result in Variable: NewXml

then Transform [%Variable: NewXml%] using XSLT <xsl:stylesheet version="1.0" xmlns:xsl=... and store result in Variable: Html

then Create document from Variable: Html in http://qorlach7/Shared%20Documents/Rates.Html. Overwrite existing document.

This sample takes an XML document with currency conversion rates from the European Central Bank, adds a new node with the EUR/EUR rate 1.0000, and generates an HTML document with currency rates in a document library.

For the full text of the XSLT template and an example of output HTML, see the examples section of the *Transform XML* activity overview. For an example of source XML document, see the examples section of the *Query XML* activity overview.

## 7. Dictionary activities

### 7.1 Add an Item to a Dictionary

This workflow action is used to add a new item to a dictionary variable.
Changes can be made to an existing dictionary variable, or a new dictionary variable can be created.
<b>Note:</b> an item is always added to the <b>end</b> of the dictionary variable. In case you need to insert an item to a <b>specific index</b> , use <i>Insert Dictionary Item at Index</i> .
<b>See Also:</b> Understanding Dictionary actions in SharePoint Designer 2013 ( <a href="http://msdn.microsoft.com/en-us/library/office/jj554504(v=office.15).aspx">http://msdn.microsoft.com/en-us/library/office/jj554504(v=office.15).aspx</a> )

Add item as String with this name to the dictionary variable. If item with the same name exists, display error . Save result to: Variable: outputDictionary.

#### Parameters:

**item** - Text of variable of any supported type.

**String** - Type of the new dictionary item. The following types are supported: *Date/Time, Integer, Number, String, Boolean, Dictionary, Guid*.

**this** name - Name of the new dictionary item.

**dictionary variable** - the source dictionary variable.

**display error** - choose what the workflow should do if such item already exists in a dictionary: stop and display an error, or replace the value.

**Variable: outputDictionary** - the target dictionary variable.

The source dictionary variable can be specified here as well; in this case, all changes will be made to it.

## 7.2 Remove Duplicate Values from Dictionary

This workflow action removes duplicated values from the dictionary variable. Changes can be made to an existing dictionary variable, or a new dictionary variable can be created.

Remove duplicate values from dictionary variable. Result at Variable: outputDictionary.

### Parameters:

**dictionary variable** - the source dictionary variable

**Variable: outputDictionary** - the target dictionary variable.

The source dictionary variable can be specified here as well; in that case all changes will be made to it.

## 7.3 Remove Item from Dictionary

This workflow action removes an item with a specific name or value from a dictionary variable.

Remove item with specified name as String from the dictionary variable. Result at: Variable: outputDictionary.

### Parameters:

**specified** - specify the name or value of the item, depending on the next parameter.

**name** - Selector with two values: *name* or *value*.

**String** - One of the supported types: *Date/Time, Integer, Number, String, Boolean, Integer, Guid*.

Always use *String* type when you are looking for the item by name.

If you are searching by value, the type of this value should correspond to the type of the dictionary item.

**dictionary variable** - the source dictionary variable.

**Variable: outputDictionary** - the target dictionary variable. The source dictionary variable can be specified here as well; in that case all changes will be made to it.

## 8. Document Library activities

### 8.1 Check In Item at URL

This workflow action is used to check in a specified item with a comment.

Do minor check in of list item at specified url with specified comment

#### Parameters:

**minor check in** - Check in type, one of the following options: *minor check in*, *major check in*, or *overwrite check in*.

**specified url** - Full list item URL including site name.

**specified comment** - Comment text.

### 8.2 Check Out Item at URL

This workflow action is used to check out list item specified by URL.

Check out item at specified url

#### Parameters:

**specified url** - Full URL of list item including site name.

### 8.3 Copy Document to List Item Attachment

This workflow action copies a specified document to a list item as an attachment.

Copy document at this url to list item at this url. Overwrite existing items.



#### Parameters:

Copy document at **this url** - the URL of the document to be copied

to list item at **this url** - the URL of the target list item

**Overwrite** - select to *overwrite* or *do not overwrite* existing items

### 8.4 Copy Folder Structure

This workflow action recreates a folder structure in a target folder, identical to the structure of the source folder. No content is copied, only empty folders are created.

Copy folder structure of any content type from a folder at this url to the folder at this url.

#### Parameters:

**any content type** - specify the content types that will be taken into account

**a folder at this url** - specify the full URL of the source folder

**to the folder at this url** - specify the full URL of the target folder

### 8.5 Copy or Move Document

This workflow action can copy, move or rename documents in SharePoint document libraries.

It can also **copy** a document to **multiple folders**, if they are specified in a **dictionary variable**.

Copy document at this url to the folder at specified url and with the same name. Overwrite existing document. Stop workflow on error.

#### Parameters:

**Copy** - Copy or move document.

**this url** - full URL of document

**specified url** - URL of the target folder, may be the same as source folder.

Dictionaries of URLs are also supported for **copying** a document to all folders mentioned in a dictionary.

**the same** - (optional) the new file name and extension for the document.

**Overwrite** - If the folder already contains file with the same name, then you can apply one of two options. *Overwrite* - replace file by new one. *Do not overwrite* - skip this file.

**Note:** This action adds new file version when overwriting

**Stop workflow** - choose to *stop workflow* on error, or to *skip optional columns*.

## 8.6 Copy or Move Folder

This workflow action copies or moves the files from the specified folder to another folder. Can optionally include files in subfolders or choose only specific content types.

Copy all documents of this content type from a folder at this url to the folder at this url. Overwrite existing documents. Include subfolders. Stop workflow on error.

### Parameters:

**Copy/Move** - select to *copy* or *move* files.

**all** - choose *all* documents or *the folder and all* documents

**this content type** - (optional) specify documents of which content types should be affected

**folder at this url** - specify the URL of the source folder

**to the folder at this url** - specify the URL of the target folder

**Overwrite/Do not overwrite** - specify if existing files with the same names in the target folder should be overwritten, or whether the workflow should stop with an error.

**Include/Do not include subfolders** - specify if the subfolders in the source folder should also be copied/moved.

**Stop workflow** - choose to *stop workflow* on error, to *skip optional columns*, or to *skip document*.

## 8.7 Create Document from Text

This workflow action adds a new text document or replaces an existing document with a new file created from a string variable.

Create document from this text in file URL. Overwrite existing document.

### Parameters:

**this text** - Document body.

**file URL** - Full URL for the new document, including site name, file name and extension.

**Overwrite** - If the folder already contains file with the same name, then you can apply one of these three options:

- *Overwrite* - replace file in library with specified file.
- *Do not overwrite* - generate an error.
- *Append unique identifier*- add string like \_20e0ad06-4261-4349-8a37-894fa7574f42 to the end of file name.

## 8.8 Create Folder in the Document Library at URL

This workflow action is used to create a subfolder in a document library or in a folder of a document library.

Create specified folder at folder URL.

**Parameters:**

**specified** - Folder name.

**URL** - Full URL with site name of document library or folder in document library.

## 8.9 Create Link to a Document

This workflow action creates a link in a specified library to a document from another library

Create link to document with this url at this library url

**Parameters:**

**this url** - the url of the document for which the link will be created

**this library url** - URL of the library where the link will be created

## 8.10 Delete Folder in the Document Library at URL

This workflow action is used to delete a subfolder in a document library or in a folder of a document library.

Delete specified folder at folder URL

### Parameters:

**specified** - Subfolder name to delete.

**URL** - Full URL with site name of document library or name of the folder in the document library.

## 8.11 Discard Check Out Item at URL

This workflow action is used to remove the check out of a specified list item.

Discard Check Out item at specified url

### Parameters:

**specified url** - Full list item URL including site name.

## 8.12 Discard Publish Item

This workflow action is used to discard list item publishing.

Discard publish list item at this url with specified comment

### Parameters:

**this url** - Full URL of target list item.

**specified comment** - Comment text.

## 8.13 Get Document Lock Status

This workflow action is used to retrieve the lock status of the document (is locked for edit or available)

Get lock status of the document at this URL and store in variable Variable: isLocked

### Parameters:

**document at this URL** - specify the full URL of the document.

**Variable: isLocked** - specify the variable to store the output value.

## 8.14 Get Entry From Folder Property Bag

This workflow action is used to get a specified entry value from a property bag of the parent folder of a specified list item.

For the list of property bag entries, refer to: [http://msdn.microsoft.com/en-us/library/microsoft.sharepoint.spfolder.properties\(v=office.12\).aspx](http://msdn.microsoft.com/en-us/library/microsoft.sharepoint.spfolder.properties(v=office.12).aspx)

This action errors out if the property is not found.

Get specified entry value from list item at this URL parent folder property bag and store in Variable: propertyValue

### Parameters:

**specified** - Name of the entry in property bag.

**Example:** vti\_timelastmodified.

**this url** - Full URL of list item.

**Variable: variable** - Output variable.

## 8.15 Get Text Document Content

This workflow action is used to get the content of the text document and save it to variable.

Get content of the document at this URL and store in Variable: content.

### Parameters:

**this URL** - the full URL of the text document.

**Variable: content** - the variable to store the document content

## 8.16 Publish Item

This workflow action is used to submit a list item for content approval with a specified comment.

Publish list item at this url with specified comment.

**Parameters:**

**this URL** - Full URL of target list item.

**specified** - Comment text.

## 8.17 Rename Document at URL

This workflow action is used to rename a list item specified by URL.

Rename document at specified URL to this file name

**Parameters:**

**specified URL** - Full URL of list item including site name.

**this file name** - New file name without extension.

## 8.18 Upload Net File to Document Library at URL

This workflow action is used to upload a file to a document library or its subfolder.

Upload file from this URL to the folder at this URL . Overwrite existing document.

**Parameters:**

**from this URL** - the full URL of the file to be uploaded

**at this URL** - specify the full URL of the document library or its subfolder where the file should be uploaded

**Overwrite** - If the document library or subfolder already contains a file with the same name, then you can apply one of three options:

- *Overwrite* - replace file in library with the specified file.
- *Do not overwrite* - generate an error.
- *Append unique identifier*- add string like \_20e0ad06-4261-4349-8a37-894fa7574f42 to the end of the new file's name.

## 9. E-Mail activities

### 9.1 Add Custom Headers to E-mail

This workflow action adds the specified e-mail headers to the e-mails sent by the HarePoint *Send E-Mail ...* actions.

**Note:** This action must precede *Send E-Mail ...* actions in a workflow - see **Examples** section.

Add these custom headers to outgoing e-mail messages

#### Parameters:

**these custom headers** - specify the headers to be added. Multiple headers can be specified, delimited by a new line.

#### Header examples:

- *Return-Receipt-To: sender@example.com* - request a receipt
- *Importance: high* - set importance level as High
- *Sensitivity: company-confidential* - set the e-mail sensitivity as confidential

For more details, please refer to <https://tools.ietf.org/html/rfc4021>

#### Examples:

##### Requesting a receipt for an outgoing e-mail

*Add Return-Receipt-To: test@harepoint.com custom headers to outgoing e-mail messages.*

*then Send e-mail to John Smith with Normal importance, blind carbon copy to this address, use test@harepoint.com as sender*

**Note:** e-mail addresses in both actions need to be the same!

### 9.2 Get E-mail Property

This workflow action is used to retrieve message properties from the mime body.

Get HTML body from this mime body and store in Variable: emailProperty.

#### Parameters:

**HTML body** - One of the available message properties. The list of available properties includes: HTML body, TEXT body, sender name, sender e-mail, CC recipients, reply to, date sent, importance, and mail headers.

**this mime** - Mime body of the received e-mail message.

**Variable: emailProperty** - Output string value of the specified property.

#### Examples:

Receive e-mail from pop-mail.outlook.com:995, secure connection type SSL, login testuser@outlook.com and password Test1Pwd. Store e-mail sender in Variable: outputSender, recipient in Variable: outputRecipients, subject in Variable: outputSubject, and MIME body in Variable outputBody.

then Get mail headers from Variable: outputBody and store in Variable: msgHeader

then Copy 250 characters from start of Variable: msgHeader ( Output to Variable: result )

then Log Variable: result to the workflow history list

### 9.3 Receive All E-Mails to List from IMAP Server

This workflow action is used to download all messages from a specified IMAP mailbox from the default folder (usually INBOX) to a prepared SharePoint list. The messages are deleted from the mailbox after adding them to the SharePoint list.

The target SharePoint list should contain the following fields that will be filled with values from the downloaded messages:

Column	Data Type	E-mail header
Title	Single line of text	Subject
Description	Multiple lines of text / enhanced rich text	Html Body
FromName	Single line of text	Sender display name
FromAddress	Single line of text	Sender e-mail address
FromUser	Person or Group	The program tries to find SharePoint user with the message sender e-mail address.
Date	Date and Time	Date Sent
Importance	Single line of text	Message importance: Low, Normal, High
TextBody	Multiple lines of text	Text body of the message
ToString	Single line of text	"To" field with names and e-mail addresses
CCString	Single line of text	"CC" field with names and e-mail addresses
ReplyToString	Single line of text	"Reply-To" header
MailHeaders	Multiple lines of text	All message headers



**Note:** To provide login and password, use **Set Authentication** action.

Later versions support **OAuth 2.0** to receive emails from office365.com. In this case, additional steps are needed, please follow [this guide](#) to obtain Client ID, Tenant ID, and client secret.

Specify login as **your\_email@domain.com@ClientID@TenantID**, password as **client secret**.

Download all e-mails from [this IMAP server](#), secure connection type SSL. Store them with attachments into the list at [this url](#). Store inline images in folder at [this url](#).

**Parameters:**

**all emails** - choose to download *all* emails, or the *specified number* of emails.

**this IMAP server** - IMAP server name or IP-address. By default, port 143 is used, but you can specify an alternative port with a colon, e.g. mail.company.com:1143.

**type SSL** - Secure connection type. Available values: *None*, *SSL*. Note, that an IMAP server with SSL usually listens for incoming connections on port 993.

**with** - Specify whether to attach message attachments to the list item or not.

**this url** - Full URL of the list.

**this url** - (Optional) Full URL of the document library where inline attachments will be stored.

## 9.4 Receive All E-mails to List from POP3 Server

This workflow action is used to download all messages from a specified POP3 mailbox to a prepared SharePoint list. The messages are deleted from the mailbox after adding them to the SharePoint list.

The target SharePoint list should contain the following fields that will be filled with values from the downloaded messages:

Column	Data Type	E-mail header
Title	Single line of text	Subject
Description	Multiple lines of text / enhanced rich text	Html Body
FromName	Single line of text	Sender display name
FromAddress	Single line of text	Sender e-mail address
FromUser	Person or Group	The program tries to find SharePoint user with the message sender e-mail address.
Date	Date and Time	Date Sent
Importance	Single line of text	Message importance: Low, Normal, High
TextBody	Multiple lines of text	Text body of the message
ToString	Single line of text	"To" field with names and e-mail addresses
CCString	Single line of text	"CC" field with names and e-mail addresses
ReplyToString	Single line of text	"Reply-To" header
MailHeaders	Multiple lines of text	All message headers

**Note:** To provide login and password, use **Set Authentication** action.

Later versions support **OAuth 2.0** to receive emails from office365.com. In this case, additional steps are needed, please follow [this guide](#) to obtain Client ID, Tenant ID, and client secret.

Specify login as **your\_email@domain.com@ClientID@TenantID**, password as **client secret**.

Download all e-mails from this server, secure connection type SSL. Store them with attachments into the list at this URL. Store inline images in folder at this URL.

#### Parameters:

**all emails** - specify to download *all* emails, or only *specific number* of emails.

**this server** - POP3 server name or IP-address. By default, port 110 is used, but you can specify an alternative port with a colon, like mail.company.com:1110.

type **None** - Secure connection type. Available values: None, SSL. Note that a POP3 server with SSL usually listens for incoming connections on port 995.

**with** - Specifies whether to attach message attachments to list item or not.

**this url** - Full URL of the list.

**this url** - (Optional) Full URL of the document library where inline attachments will be stored.

## 9.5 Receive E-Mail from IMAP server

This workflow action is used to receive e-mail messages from an IMAP mail server. This action gets a list of messages in the mailbox, receives the first unread message and deletes it from mailbox.

This action generates an error when authentication fails or when the server is unavailable, and returns empty values when the mailbox is empty if there are no unread messages.

**Note:** To provide login and password, use **Set Authentication** action.

Later versions support **OAuth 2.0** to receive emails from office365.com. In this case, additional steps are needed, please follow [this guide](#) to obtain Client ID, Tenant ID, and client secret.

Specify login as **your\_email@domain.com@ClientID@TenantID**, password as **client secret**.

Receive e-mail from [this IMAP server](#), secure connection type None. Store e-mail sender in Variable:outputSender, recipient in Variable:outputRecipients, subject in Variable:outputSubject and HTML body in Variable:outputBody.

#### Parameters:

**this server** - IMAP server name or IP-address.

By default, port 143 is used, but you can specify an alternative port with a colon, like mail.company.com:1430

type **None** - Secure connection type. Available values: *None*, *SSL*. Note that an IMAP server with SSL usually listens for incoming connections on port 993.

**Variable: outputSender** - Output string with sender's address of received e-mail message.

**Variable: outputRecipients** - Output string with recipients' addresses of received e-mail message.

**Variable: outputSubject** - Output string with subject of received e-mail message.

**HTML** - Select to return message body as *HTML*, *text* or *MIME* (including encoded attachments and other content).

**Variable: outputBody** - Body of received e-mail message. The type of body is defined by the previous parameter. It can be one of the following: *HTML*, *TEXT* or *MIME*.

#### Examples:

Receive email from [imap.gmail.com:993](#), secure connection type SSL. Store e-mail sender in Variable: sender, recipient in Variable: recipient, subject in Variable: subject and HTML body in Variable: body.

## 9.6 Receive E-Mail from POP3 Server

This workflow action is used to receive e-mail messages from a POP3 mail server. The action gets a list of messages in a mailbox, retrieves the first one, and deletes the received message from the mailbox.

This action generates an error when authentication fails or the server is unavailable, and returns an empty value when the mailbox is empty.

**Note:** To provide login and password, use **Set Authentication** action.

Later versions support **OAuth 2.0** to receive emails from office365.com. In this case, additional steps are needed, please follow [this guide](#) to obtain Client ID, Tenant ID, and client secret.

Specify login as **your\_email@domain.com@ClientID@TenantID**, password as **client secret**.

Receive e-mail from [this server](#), secure connection type [None](#). Store e-mail sender in [Variable:outputSender](#), recipient in [Variable:outputRecipients](#), subject in [Variable:outputSubject](#) and [HTML](#) body in [Variable:outputBody](#).

#### Parameters:

**[this server](#)** - POP3 server name or IP-address.

By default, port 110 is used, but you can specify an alternative port with colon, like mail.company.com:1110.

type **[None](#)** - Secure connection type. Available values: *None*, *SSL*.

Note, that a POP3 server with SSL usually listens for incoming connections on port 995.

**Variable: [outputSender](#)** - Output string with the sender's address from the received e-mail message.

**Variable: [outputRecipients](#)** - Output string with the recipients' addresses from the received e-mail message.

**Variable: [outputSubject](#)** - Output string with subject of the received e-mail message.

**[HTML](#)** - Select to return the message body as *HTML*, *text* or *MIME* (including encoded attachments and other content).

**Variable: [outputBody](#)** - The body of the received e-mail message. The type of body is defined by the previous parameter. It can be one of the following: *HTML*, *TEXT* or *MIME*.

#### Examples:

Receive email from [pop.gmail.com:995](#), secure connection type [SSL](#), login [harepoint@gmail.com](#) and password [pwd123](#). Store e-mail sender in [Variable: sender](#), recipient in [Variable: recipient](#), subject in [Variable: subject](#) and [HTML](#) body in [Variable: body](#).

## 9.7 Save E-Mail Attachments to List Item

This workflow action is used to retrieve attachments from an e-mail message and attach them to a specified list item.

**See Also:** *Receive E-Mail from POP3 Server* and *Receive E-Mail from IMAP Server*

Save all attachments from this mime body to list item at this url. Store names of saved attachments to Variable: attachmentNames

### Parameters:

all - File masks, separated by semicolon. Leave this field empty to save all attachments.

this mime - String with the message mime body.

this url - Full URL of list item.

**Variable: attachmentNames** - Array of the saved attachment names.

## 9.8 Send E-Mail with Document as Attachment

This workflow action is used to send an e-mail with a document from a document library as an attachment.

**Note:** Before using this action, you should define the SMTP connection settings with *Set SMTP Server Options*.

**See Also:** *Set SMTP Server Options*.

Send e-mail to this address with Normal importance, blind carbon copy to this address, display this address as sender and include document this document as attachment in the e-mail.

### Parameters:

e-mail to this address - Click here to specify message recipients, carbon copy recipients, message subject and body.

SharePoint groups are supported as recipients.

**Normal** - E-mail importance. Available options: *Normal, Low, High*.

blind carbon copy to this address - (Optional) Blind carbon copy recipients.

**this address** as sender - Sender address.

**this document** - Item from document library. For example, *Current Item* for the document library workflow.

## 9.9 Send E-Mail with Http Files Attachments

This workflow action is used to send an e-mail with one or more attachments from the Internet, available with HTTP protocol.

**Note:** Before using this action, you should define SMTP connection settings with *Set SMTP Server Options* action.

**See Also:** "Set SMTP Server Options" action.

Send e-mail to this address with Normal importance, blind carbon copy to this address, use this address as sender and include files from these URLs as attachments.

### Parameters:

e-mail to **this address** - Click here to specify message recipients, carbon copy recipients, message subject and body. You may ignore SharePoint Designer warning "If a recipient is not a valid SharePoint user, he or she will not receive workflow emails".

SharePoint groups are supported as recipients.

**Normal** - E-mail importance. Available options: *Normal, Low, High*.

blind carbon copy to **this address** - (optional) Dictionary variable with e-mails of blind carbon copy recipients.

**this address** as sender - Sender address to be displayed in *From* field of thee-mail. It may be overridden by your SMTP server.

Format: "User Name" <mailbox@server.com>

**this URLs** - Multi-line string with one or more URL of file to attach. Each file name must be on a separate line.

## 9.10 Send E-Mail with List Item Attachments

This workflow action is used to send an e-mail with attachments from a list item. For example, you can send a Task or Calendar item's attachments with this action.

**Note:** Before using this action, you should define SMTP connection settings with *Set SMTP Server Options*.

**See Also:** *Set SMTP Server Options*

Send e-mail to [this address](#) with **Normal** importance, blind carbon copy to [this address](#), use [this address](#) as sender and include all files attached to [this item](#) to attachments of e-mail.

**Parameters:**

e-mail to [this address](#) - Click here to specify message recipients, carbon copy recipients, message subject and body.

SharePoint groups are supported as recipients.

**Normal** - E-mail importance. Available options: *Normal, Low, High*.

blind carbon copy to [this address](#) - (Optional) Blind carbon copy recipients.

[this address](#) as sender - Sender address.

[this item](#) - Attachments of this list item (like Calendar or Task) will be attached to e-mail.

## 9.11 Send E-Mail with SharePoint Attachments

This workflow action is used to send an e-mail with SharePoint files as attachments.

**Note:** Before using this action, you should define SMTP connection settings with *Set SMTP Server Options*.

**See Also:** *Set SMTP Server Options*

Send e-mail to [this address](#) with **Normal** importance, blind carbon copy to [this address](#), use [this address](#) as sender and include all files at [this URLs](#) to attachments of e-mail.

**Parameters:**

e-mail to [this address](#) - Click here to specify message recipients, carbon copy recipients, message subject and body.

SharePoint groups are supported as recipients.

**Normal** - E-mail importance. Available options: *Normal, Low, High*.

blind carbon copy to [this address](#) - (Optional) Blind carbon copy recipients.

**this address as sender** - specify the email address that will be displayed as sender address.

files at **this URLs** - a multi-line string of full URLs of the files to be attached. Can contain only one URL, if necessary.

## 9.12 Send E-Mail with Word Document as a Body

This workflow action is used to send an e-mail with a Word document as a body. It is useful for sending invoices, surveys, and any other documents with complex formatting.

Send e-mail to this address, CC to this address, BCC to this address with this subject and Word body at this url, use this address as sender and attach files at these urls

### Parameters:

**e-mail to this address** - specify the main recipients of an e-mail (TO field)

**CC to this address** - (optional) specify carbon copy recipients (CC field)

**BCC to this address** - (optional) specify blind carbon copy recipients (BCC field)

**this subject** - specify the subject for an e-mail

**Word body at this url** - specify Encoded Absolute URL of a Word document that will be used as a body of an e-mail.

To get a proper URL, you can use the following:

- **Current Item: Encoded Absolute URL** lookup
- **Find list item at URL** action
- **Get List Item Url** action with *Native form* in the first parameter

**this address as sender** - (optional) specify an alternative e-mail address that will be displayed as Sent From in the e-mail.

**attach files at these urls** - specify an **array** of URLs of files that need to be attached.

**Note:** in order to get an array of URLs, you can use the following actions:

- **Find List Item at URL**
- **Get List Item Attachments**
- **Create Array** (*String* type)



**Example** of creating an array:

Create String array with *http://company.local/Documents/Sample.pdf* items and store in Variable: Array

## 9.13 Send E-Mail without Attachments

This workflow action sends an e-mail without any attachments.

**Note:** Before using this action, you should define SMTP connection settings using *Set SMTP Server Options*.

**See Also:** *Set SMTP Server Options*

Send e-mail to this address with Normal importance, blind carbon copy to this address, use this address as sender.

### Parameters:

e-mail to **this address** - Click here to specify message recipients, carbon copy recipients, and message subject and body. You may ignore the SharePoint Designer warning "If a recipient is not a valid SharePoint user, he or she will not receive workflow emails".  
SharePoint groups are supported as recipients.

**Normal** - E-mail importance. Available options: *Normal, Low, High*.

blind carbon copy to **this address** - (optional) Dictionary variable with e-mail addresses of blind carbon copy recipients.

**this address** as sender - Sender address to be displayed in *From* field of the e-mail. It may be overridden by your SMTP server.

Format: "User Name" <mailbox@server.com>

### Examples:

Use SMTP server *smtp-mail.outlook.com:587*, secure connection type *SSL*,  
login *testuser1@outlook.com* and password *test1pwd*. Do not convert message body to plain text.

then Send e-mail to *testuser3@gmail.com* with Normal importance, blind carbon copy to this address,  
use *testuser1@outlook.com* as sender.

## 9.14 Set E-mail Delivery and Processing Options

This workflow action is used to send personalized e-mails.

This action needs to be placed prior to "Send E-Mail..." actions.

**See also:** "Set SMTP Server Options" action.

Send regular e-mails. Expand macros in e-mail body and subject

### Parameters:

**Regular** - choose to send regular or personal e-mails:

- *regular*
- *personal*

**Expand** - choose to expand or not expand macros in subject and body of an e-mail:

- *Expand*
- *Do not expand*

Is set to "Expand", the following Recipient's macros will be available in subject and body of an e-mail:

- %Name% Recipient's display name. Example: John Smith.
- %FirstName% Recipient's first name. Example: John.
- %LastName% Recipient's last name. Example: Smith.
- %Department% Recipient's department.
- %AccountName% Recipient's account name. Example: HAREPOINT\jsmith.
- %JobTitle% Recipient's job title.
- %Manager% Recipient's manager account name.
- %MobilePhone% Recipient's Mobile phone number.
- %WorkPhone% Recipient's Work phone number.
- %WorkEmail% Recipient's Work E-mail address.
- %Office% Recipient's office.

**Note:** Recipient's macros are only available in personal e-mails.

The similar macros are available for the Sender:

- %SenderName% Sender's display name. Example: John Smith.
- %SenderFirstName% Sender's first name. Example: John.
- %SenderLastName% Sender's last name. Example: Smith.
- %SenderDepartment% Sender's department.
- %SenderAccountName% Sender's account name. Example HAREPOINT\jsmith.
- :%SenderJobTitle% Sender's job title.
- %SenderManager% Sender's manager account name.
- %SenderMobilePhone% Sender's mobile phone number.
- %SenderWorkPhone% Sender's work phone number.
- %SenderWorkEmail% Sender's work E-mail address.
- %SenderOffice% Sender's office.

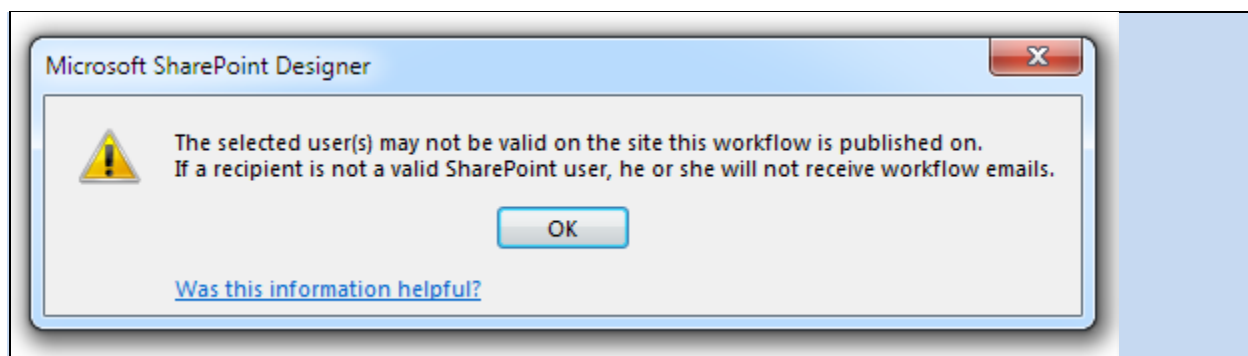
**Note:** Sender's macros are available both in regular and in personal e-mails.

**Note:** macros only expand for **SharePoint users**! If the corresponding SharePoint has not been found, the workflow action will display an error. An e-mail will **not** be sent with macros that cannot be expanded.

## 9.15 Set SMTP Server Options

This workflow action is used to define SMTP server settings for the HarePoint actions *Send E-Mail with HTTP Files Attachments* and *Send E-Mail without attachments*.

You can ignore the following warning in SharePoint Designer:



Use SMTP server this server, secure connection: Yes. Do not convert message body to plain text.

**Parameters:**

**this server** - SMTP server name or IP-address.

By default, port 25 is used, but you can specify an alternative port with a colon, like mail.company.com:125.

**Yes** - Specify if secure connection should be used. Available values: *Yes*, *No*.

**Do not convert** - Specifies whether to convert message body to plain text or not.

**Examples:**

Use SMTP server smtp-mail.outlook.com:587, secure connection Yes. Do not convert message body to plain text.

then Send e-mail to testuser3@gmail.com with Normal importance, blind carbon copy to this address, use testuser1@outlook.com as sender.

## 10. Exchange activities

### 10.1 Cancel Exchange Calendar Item

This workflow action cancels an Exchange calendar item with a specified cancellation text.

Cancel calendar item with this ID and this cancellation message text.

**Parameters:**

**this ID** - specify the item ID to be cancelled.

**this cancellation message** - type the cancellation message to be displayed.

## 10.2 Create Exchange Calendar Item

This workflow action is used to create a new calendar item in Exchange.
Note that this action only creates an event, but doesn't add attendees to it - this can be done by <b>Update Exchange Calendar Item Attendees</b> action.
The advanced parameters of an event (such as reminder interval, importance level, etc.) can be changed, if necessary, by <b>Update Exchange Calendar Item Advanced Properties</b> action.
<b>Important note:</b> <b>Set Authentication</b> action needs to be used prior to this action to authenticate against the Exchange services.
See also: <b>Send Meeting Request</b> action is equivalent to this action, but additionally sends meeting requests to attendees.

Create calendar item with this subject, this location, this message, this start date, this end date, is all day event: no . Store calendar Id to Variable: calendarId

### Parameters:

<b>with <u>this</u> subject</b> - specify the subject of an event
<b><u>this</u> location</b> - specify the location of the event
<b><u>this</u> message</b> - specify the event message
<b><u>this</u> start date</b> - specify the start date of the event
<b><u>this</u> end date</b> - specify the end date of the event
<b>is all day event - <u>no</u></b> - choose <i>yes</i> or <i>no</i>
<b>Store calendar ID in <u>Variable: calendarID</u></b> - specify a variable to store the ID of an item.
This is a global ID and it is the same for all event attendees. It is used in other actions from "Exchange" category.

## 10.3 Delete Exchange Calendar Item

This workflow action deletes an Exchange calendar item. Several deletion methods are available. See description of the second parameter for more details.

Delete calendar item with this ID using move to deleted items delete mode.

**Parameters:**

**this ID** - specify the item ID to be deleted.

**move to deleted items** - select the deletion method:

- *move to deleted items*
- *hard delete*
- *soft delete*

## 10.4 Update Exchange Calendar Item

This workflow action updates an Exchange calendar item with new basic information.

**See also:** *Update Exchange Calendar Item Advanced Properties*

Update calendar item with this ID. Set fields: this subject, this location and this message body. Set this date as start and this date as end (this is all day event: No ).

**Parameters:**

**this ID** - specify the item ID to be cancelled

**this subject** - specify the new subject

**this location** - specify the new location

**this message body** - specify the new message body

**this date as start** - set the new start date

**this date as end** - set the new end date

**all day event: No** - select if this is an all day event: *yes, no*.

## 10.5 Update Exchange Calendar Item Advanced Properties

This workflow action allows updating advanced properties of an Exchange calendar item.

**See also:** *Update Exchange Calendar Item* to update the basic properties of an Exchange calendar item.

Update calendar item with calendarId ID. Set fields: reminder 15 minutes, Normal importance, response requested Yes , allow new time proposal Yes , this categories list.

**Parameters:**

**calendarId** - specify the ID of the Exchange calendar item.

**reminder 15 minutes** - set the new value for the reminder for this event, in minutes.

**Normal importance** - set the new importance level:

- *Normal*
- *High*
- *Low*

**response requested Yes** - select if response is requested: *yes, no*.

**allow new time proposal Yes** - select if a new time proposal is allowed: *yes, no*.

**this categories list** - specify the list of categories from a variable.

## 10.6 Update Exchange Calendar Item Attendees

This workflow action allows you to update the list of required and/or optional attendees for the specified Exchange calendar item.

**See also:** *Update Exchange Calendar Item* to update the basic properties of the Exchange calendar item, and *Update Exchange Calendar Item Advanced Properties* to update the advanced properties of the Exchange calendar item.

Update calendar item with calendarId ID. Set required attendees to value of this variable, optional attendees to value of this variable.

**Parameters:**

**calendarId** - specify the ID of the Exchange calendar item.

**Set required attendees to value of this variable** - provide the variable with the list of required attendees for this event.

**optional attendees to value of this variable** - provide the variable with the list of optional attendees for this event.

## 11. Images activities

### 11.1 Convert Image

This workflow action converts an image to one of the supported output formats: Bitmap Picture (bmp), GIF, JPG, PNG, TGA, TIFF.

The output file will have the same name and the extension according your selection in the second parameter, and will be placed in the same document library or folder where the source file is stored.

**Note:** See the overview of *Get Image Type* for the list of supported file formats.

Convert image at specified URL to JPG format and store URL to converted image in Variable: convertedImageUrl. Overwrite existing image.

#### Parameters:

**specified URL** - Full URL of the image in Document Library.

**JPG** - One of the supported output formats: Bitmap Picture (bmp), GIF, JPG, PNG, TGA, TIFF.

**Variable: convertedImageUrl** - Output string variable with the URL of a new image.

**Overwrite** - Choose to overwrite the file or not, in case the library already contains a file with the same name as the output file.

#### Examples:

See the example in the *Get Image Type* activity description.

### 11.2 Flip Image

This workflow action is used to flip an image in one of the supported formats horizontally or vertically.

**Note:** See the *Get Image Type* overview for the list of supported file formats.

Flip image at specified URL horizontally

#### Parameters:

**specified URL** - URL of the image in Document Library.

**horizontally** - How to flip the image: horizontally or vertically.



## 11.3 Get EXIF Property from Image

This workflow action is used to get an EXIF property from an image in a supported format. With this action you can get the following properties:

• Title
• Camera Manufacturer
• Camera Model
• Exposure Time
• F number
• Date and Time
• Focal Length
• Orientation
• Exposure program
• ISO speed rating
• Flash
• Exposure mode
• Artist
• White balance
• Scene capture type
• Exposure bias
• Aperture
• Maximum lens aperture
• North or South Latitude
• East or West Longitude
• Scene type
• Software

**See also:** Exchangeable image file format at Wikipedia ([http://en.wikipedia.org/wiki/Exchangeable\\_image\\_file\\_format](http://en.wikipedia.org/wiki/Exchangeable_image_file_format) ).

**Note:** See the *Get Image Type* activity overview for the list of supported file formats.

Get EXIF property Title from image at this URL and store in Variable: exifProp

**Parameters:**

**Title** - Name of the EXIF property (see list above).

**this URL** - URL of the picture in Document Library.

**Variable: exifProp** - Output string variable. Empty string if the property is not found.

## 11.4 Get Image Size

This workflow action is used to get an image size in pixels. In case of an unknown file format, this action will terminate the workflow with an error.

**Note:** see the list of supported formats in the description of *Get Image Type*.

Get size of the image at specified URL and store width in Variable: imageWidth and height in Variable: imageHeight

**Parameters:**

**specified URL** - Full URL of the image.

**Variable: imageWidth** - Output integer variable for image width in pixels.

**Variable: imageHeight** - Output integer variable for image height in pixels.

## 11.5 Get Image Type

This workflow action analyzes a file extension and returns an image type if the file type is supported by the Product image processing engine:

- BMP - Windows or OS/2 Bitmap File (\*.BMP)
- ICO - Windows Icon (\*.ICO)
- JPEG - Independent JPEG Group (\*.JPG, \*.JIF, \*.JPEG, \*.JPE)

• JNG - JPEG Network Graphics (*.JNG)
• KOALA - Commodore 64 Koala format (*.KOA)
• LBM - Amiga IFF (*.IFF, *.LBM)
• IFF - Amiga IFF (*.IFF, *.LBM)
• MNG - Multiple Network Graphics (*.MNG)
• PBM - Portable Bitmap (ASCII) (*.PBM)
• PBMRAW - Portable Bitmap (BINARY) (*.PBM)
• PCD - Kodak PhotoCD (*.PCD)
• PCX - Zsoft Paintbrush PCX bitmap format (*.PCX)
• PGM - Portable Graymap (ASCII) (*.PGM)
• PGMRAW - Portable Graymap (BINARY) (*.PGM)
• PNG - Portable Network Graphics (*.PNG)
• PPM - Portable Pixelmap (ASCII) (*.PPM)
• PPMRAW - Portable Pixelmap (BINARY) (*.PPM)
• RAS - Sun Rasterfile (*.RAS)
• TARGA - truevision Targa files (*.TGA, *.TARGA)
• TIFF - Tagged Image File Format (*.TIF, *.TIFF)
• WBMP - Wireless Bitmap (*.WBMP)
• PSD - Adobe Photoshop (*.PSD)
• CUT - Dr. Halo (*.CUT)
• XBM - X11 Bitmap Format (*.XBM)
• XPM - X11 Pixmap Format (*.XPM)
• DDS - DirectDraw Surface (*.DDS)
• GIF - Graphics Interchange Format (*.GIF)
• HDR - High Dynamic Range (*.HDR)

• FAXG3 - Raw Fax format CCITT G3 (*.G3)
• SGI - Silicon Graphics SGI image format (*.SGI)
• EXR - OpenEXR format (*.EXR)
• J2K - JPEG-2000 format (*.J2K, *.J2C)
• JP2 - JPEG-2000 format (*.JP2)
• PFM - Portable FloatMap (*.PFM)
• PICT - Macintosh PICT (*.PICT)
• RAW - RAW camera image (*.*)

For example, for JPG, JIF, JPEG and JPE files, the returned image type will be JPEG. Product can also load RAW files from digital photo cameras (CRW/CR2, NEF, RAF, DNG, MOS, KDC, DCR, etc.), virtually all RAW formats are supported).

For unsupported formats the returned image type will be UNKNOWN.

Get type of the image in specified URL and store result in Variable: imageType and Variable: isImageCompressed

#### Parameters:

**specified URL** - Full URL of the picture.

**Variable: imageType** - Output string with the type of image ('UNKNOWN' for unsupported types).

**Variable: isImageCompressed** - Output boolean variable. True indicates that the returned format uses image compression.

#### Examples:

*Get type of the image in Current Item:Encoded Absolute URL and store result in Variable: imageType and Variable: isImageCompressed*

*If Variable: isImageCompressed equals No*

*If Variable: imageType not equals UNKNOWN*

*Convert image at Current Item:Encoded Absolute URL to JPG format and store url to converted image in Variable: convertedImageUrl. Overwrite existing image.*

*then Delete item at [%Current Item:Encoded Absolute URL%] list item url*

Else

Log Type: [%Variable: imageType%] to the workflow history list

This workflow automatically converts uncompressed images to JPG format.

## 11.6 Get XMP Property

This workflow action gets the specified XMP property from the specified document and saves it to a variable.

Get specified XMP metadata property from document at this url and store result in Variable: variable1

### Parameters:

**specified XMP metadata** - Select the required XMP metadata property from the drop-down list.

**document at this url** - Specify the full URL of the document.

For the current document, use **Current Item:Encoded Absolute URL** lookup.

**Variable: variable1** - Specify the variable name to store the result.

### Examples:

Get Dublin Core - Subject XMP metadata property from Current Item: Encoded Absolute URL and store in Variable:XMPSubject.

## 11.7 Resize Image

This workflow action resizes an image. With this action you can easily avoid huge images from digital cameras in SharePoint picture libraries.

**Note:** See 'Get Image Type' activity overview for the list of supported file formats.

Resize image at specified URL. Set width: 200, height: 100 (keep aspect ratio: yes) and append ' resize' to filename. Do not process image if it is smaller than target.

### Parameters:

**specified URL** - Full URL of the image in Document Library.

**width 200** - Maximum width of converted image, in pixels.

**height 100** - Maximum height of converted image, in pixels.

**keep aspect ratio** - Yes or no. In the first case, the output picture will have one dimension exactly as specified by the width/height parameters, but the other dimension may be smaller than what is specified by width/height.

**append 'resize' to filename** - Two options are available:

- *append 'resize' to filename* - saves the output to a new file with postfix **\_resize** in the name
- *overwrite existing file* - overwrites the source file.

**Do not process** - Process, do not process, or copy without resizing source file if it has smaller dimensions than those specified in the height and width parameters.

#### Examples:

Get type of the image in Current Item:Encoded Absolute URL and store result in Variable: imageType and Variable: isImageCompressed

If Variable: imageType equals JPEG

Resize image at Current Item:Encoded Absolute URL set width: 800 , height: 600 (keep aspect ratio: yes ) and overwrite existing file . Do not process image if it is smaller than target

## 11.8 Rotate Image

This workflow action is used to rotate an image in a supported format 90, 180 or 270 degrees clockwise or counter-clockwise.

**Note:** See the *Get Image Type* overview for the list of supported file formats.

Rotate image at specified URL 90 degrees Clockwise.

#### Parameters:

**specified URL** - URL of the image in Document Library.

**90** - How many degrees to rotate the image: *90, 180, 270*.

**Clockwise** - How to rotate the image: *clockwise or counter-clockwise*.

## 11.9 Set EXIF Property on Image

This workflow action sets the specified EXIF property for the specified image.

Set EXIF property Title on image at this url to variable value

### Parameters:

**Title** - the required EXIF property to be set can be selected from the drop-down list.

Available values: Title, Subject, Artist, Copyright, Date and Time, Software, GPS Latitude, GPS Longitude, GPS Altitude

**this url** - URL of the image

**variable** - value to be set for the EXIF property

### Examples:

Set EXIF property Title on image at Current Item:Encoded Absolute URL to Trafalgar Square value

## 11.10 Set Image Transparent Color

This workflow action is used to set the transparent color for an image in one of the supported formats. Images with a transparency can be saved in PNG or GIF formats only.

The transparent color is a property of the image, transparency is not a property of the color. Once you have set a color to be the transparent color, any regions of the image in that color that were drawn previously will be transparent.

**Note:** See the *Get Image Type* overview for the list of supported file formats.

Set #FF00FF color of the image at specified URL transparent, convert to PNG.

### Parameters:

**#FF00FF** - Color to mark as transparent. You can use hexadecimal RGB values (like #RRGGBB) or the name of a system-defined color from this list: [http://msdn.microsoft.com/en-us/library/system.drawing.knowncolor\(v5.90\).aspx](http://msdn.microsoft.com/en-us/library/system.drawing.knowncolor(v5.90).aspx)

**specified URL** - URL of the image in Document Library.

**PNG** - Output format for the image with transparency: *PNG* or *GIF*.

## 11.11 Set XMP Property

This workflow action sets the specified XMP metadata property for a document to specified value.

Set specified XMP metadata property in document at this url to this value

### Parameters:

**specified XMP metadata** - Choose XMP metadata property from the drop-down list.

**document at this url** - Specify the full URL of the document.

For the current document, use **Current Item: Encoded Absolute URL** lookup.

**this value** - Specify the value to be set.

In case if property field contains multiple properties, they should be delimited by semicolon (;)

### Example:

Set Dublin Core - Subject XMP metadata property in url to construction; house; building.

### Examples:

Set Dublin Core - Description XMP metadata property in Current Item: Encoded Absolute URL to Some description here.

## 12. Lists activities

### 12.1 Add Content Type to List

This workflow action adds a specified content type to a list.

Add this content type to list at this list url.

### Parameters:

**this** - Name of the content type.

**this list url** - Full URL of the target list.

### 12.2 Add Existing Column to Content Type

This workflow action adds an existing column to a content type.



Add this existing column to specified content type.

**Parameters:**

**this existing column** - specify the name of the existing column to be added

**specified content type** - specify the content type name to which the column will be added

## 12.3 Approve Item at URL

This workflow action is used to approve a list item specified by URL.

Approve this list item url item with specified comment.

**Parameters:**

**this list item url** - Full URL of the list item.

**specified** - Comment text.

## 12.4 Copy List Item at URL

This workflow action is used to copy a list item from one list to another.

**Note:** This action can copy multiple list items if a dictionary of URLs is provided as the input parameter.

Copy list item at this URL to the list at this list URL. Store output item URL in Variable: url

**Parameters:**

**this url** - Full URL of the source list item. Dictionaries of the URLs are also supported; in this case all items for this dictionary will be copied.

**this list url** - Full URL of the target list.

**Variable: url** - Output string variable for the URL of copied list item.

## 12.5 Copy List Item Attachments to Document Library

This workflow action is used to copy attachments from a specified list item to a document library specified by URL.

Copy attachments of list item at this url to document library at specified url. Overwrite if items exist.

**Parameters:**

**this url** - Full URL of list item.

**specified url** - Full URL of target document library.

**Overwrite** - If the document library already contains a file with the same name, you can choose to *Overwrite* it, skip this file (*Do not overwrite*), or to *Append unique ID* to the file name.

## 12.6 Copy List Item Attachments to List Item

This workflow action is used to copy attachments of one list item to attachments of another list item.

Copy attachments of list item at this url to list item at specified url. Overwrite existing items

**Parameters:**

**this url** - Full URL of source list item.

**specified url** - Full URL of target list item.

**Overwrite** - If target item already contains an attachment with the same name, then you can choose one of two options:

- *Overwrite* - replace existing file with new one
- *Do not overwrite* - skip this file.

## 12.7 Create Content Type

This workflow action creates a new content type.

Create a new content type with this title, this description based on this content type , put into this sorting group .

**Parameters:**

**this title** - title of the new content type

**this description** - description of the new content type

**based on this content type** - name of the parent content type

**this sorting group** - specify the sorting group to put the new content type in.

## 12.8 Create List

Creates a new SharePoint list using the specified template.

Create list with this title and this description based on this template. Store list URL to Variable: listUrl.

### Parameters:

**this title** - Title of new list.

**this description** - Description of the new list.

**this template** - Name of the list of the document library template.

**Variable: listUrl** - This variable will contain the full URL of the created list.

## 12.9 Create List Item at URL

This workflow action creates a new list item and sets the values of the list item's fields. Values are specified as a multi-line string of one of two available formats.

First, INI file format. For example:

Position=Team Leader

First Name=John

Last Name=Smith

Age=23

Second, XML format, like:

<item>

<field title="Position">Team Leader</Field>

<field title="First Name">John</Field>

<field title="Last Name">Smith</Field>

<field title="Age">23</Field>

</item>

**See Also:** *Update List Item at Url*

Create list item with specified values at this list URL. Store new list item URL in Variable: itemURL.

**Parameters:**

**specified values** - Values for list item fields in INI or XML format.

For **Hyperlink** type fields, it is also possible to specify a description using *Url#description*:

**Example:**

*Title=new task*

*Url=http://www.harepoint.com*

*Url#description=HarePoint website*

**Managed metadata** columns are supported, and it is possible to specify term label names from the Terms Set. Multiple labels should be delimited by a semicolon (;)

**Example:**

*FieldName=Label1;Label2;Label3*

**this list URL** - Full URL of list.

**Variable: itemURL** - Output string variable for the URL of the created list item.

## 12.10 Create Site Column

This workflow action creates a new site column on a site based on a CAML query.

Create a new site column on the site at this URL based on this CAML

**Parameters:**

**this CAML** - CAML query.

For details on CAML syntax, please refer to [http://msdn.microsoft.com/en-us/library/office/ms437580\(v=office.15\).aspx](http://msdn.microsoft.com/en-us/library/office/ms437580(v=office.15).aspx)

## 12.11 Delete Entire SharePoint List at Url

This workflow action is used to delete the entire SharePoint list with all the list items completely.

**Important note:** The indicated SharePoint list will be deleted permanently and cannot be restored afterwards! If you need to remove just a single item in a list, use another action "Delete List Item at URL".

Delete the SharePoint list at this url

### Parameters:

this url - specify full URL of a SharePoint to be deleted.

## 12.12 Delete Item at URL

This workflow action is used to delete a list item specified by URL.

Delete item at specified list item URL

### Parameters:

specified - Full URL of list item.

## 12.13 Find List Item by CAML

This workflow action is used to find list items using a CAML query.

**See Also:** *Find List Item at Url*

Find item in the list at this URL by this CAML query. Store top 1000 result urls in Variable: foundItems.

### Parameters:

this URL - Full URL of list.

this CAML query - String with the CAML query.

top 1000 - Specify number of results to be stored.

**Variable: foundItems** - Output dictionary variable with URLs of the matching list items.

## 12.14 Get Field Value at URL

This workflow action is used to get a field value from a list item specified by URL.

Get value from this field of list item at this url and store result in Variable: fieldValue

### Parameters:

**this field** - Field name.

**this url** - Full URL of list item.

**Variable: fieldValue** - Output variable for the field value.

## 12.15 Get List Item Attachments

This workflow action is used to get names, file sizes or URLs of list item attachments.

Get Names of attachments of list item at this url and store to Variable: attachments

### Parameters:

**Names** - One of three options: *Names*, *URLs* or *File Sizes*

**this** - Full URL of list item.

**Variable: attachments** - Output string array with names or URLs, or a number array with file sizes.

## 12.16 Get List Item URL

This workflow action is used to get the URL of a specified list item, or the URL of the edit form or view form of a specified list item.

Get Display Form url for list item this list item (current version) and store result in Variable: itemUrl.

### Parameters:

**Display Form** - One of three options:

- *Display Form*
- *Edit Form*

- *Native Form*

In case of a document library, *Native Form* is the URL of file.

**this list item** - List item, e.g. Current Item.

**current** version - (optional) specify the version of the list item.

If versioning is enabled for the list, you can get the data from the previous versions of the item.

Two formats can be used:

- -1, -2, -3, ... - to specify previous versions relative to the current version
- 1.0, 2.0, 3.0 - to specify the required version number directly

**Variable: itemUrl**- Output string variable.

## 12.17 Get List URL

This workflow action is used to get one of seven possible URL types for a list on the current SharePoint site:

- Display Form
- Edit Form
- New Form
- Native Form
- Default Form

**Note:** Most workflow actions work with the Native Form of URL.

Get Display Form URL of this list and store to Variable: listUrl

**Parameters:**

**Display Form** - Type of URL to return.

**this** - List on current site.

**Variable: listUrl** - Output string variable.

## 12.18 Loop Through and Update List Items

This workflow action is used to update multiple items in a list with the same data (set certain column(s) of those items to a certain value(s)). The necessary items are selected by a CAML query.

Loop through items by this CAML query in list at this URL and update specified values.

### Parameters:

**this CAML query** - specify a CAML query to select items that need to be updated.

**Example:** all list items where the values of **Title** column begin with **test1**

```
<Where>
<BeginsWith>
<FieldRef Name="Title" />
<Value Type="Text">test1</Value>
</BeginsWith>
</Where>
```

**list at this URL** - specify URL of a list where items need to be updated.

**specified values** - specify new values for the columns that need to be changed for the selected items, in INI or XML format:

First one is the format in style of **INI** file:

Position=Team Leader

First Name=John

Last Name=Smith

Age=23

Second one is an **XML** format:

```
<item>
```

```
<field title="Position">Team Leader</field>
```

```
<field title="First Name">John</field>
```

```
<field title="Last Name">Smith</field>
```

```
<field title="Age">23</field>
```



</item>

#### Various field types are supported:

a) For **Lookup** fields, specify only the ID of the lookup item:

**Example:**

*Country=27*

**Multiselect Lookup fields** are supported, values can be either comma (,) or semicolon (;) delimited.

**See also:** The IDs of the items can be obtained by using two actions: **Find List Item at URL** to find necessary item and **Get Field Value** to get its ID.

b) For **Hyperlink** type fields it is also possible to update description using *Url#description*:

**Example:**

*Title=new task*

*Url=http://www.harepoint.com*

*Url#description=HarePoint website*

c) **Managed metadata** columns are supported, it is possible to specify term label names from Terms Set. Multiple labels should be delimited by a semicolon (;)

**Example:**

*FieldName=Label1;Label2;Label3*

d) **Choice** and **Checkboxes** types fields are supported. Specify the values delimited by a semicolon (;)

**Example:**

*Title=New Order*

*Order=Pizza;Pasta;Black Tea*

e) You can use a **special (alternative) format** to get values of workflow variables. To use it, the first line in this parameter should be **UseCustomVars** - this keyword enables this functionality. In the next lines you can use any workflow variable by its name: %VariableName%. This also supports arrays - you can get array item values as %ArrayName[0]%, %ArrayName[1]%, and so on.

**Example (for INI format):**

*UseCustomVars*

*Title=%ResultArray[0]%*

*Order=%ResultArray[1]%*

*Size=%ResultArray[2]%*

*Ingredients=%ResultArray[3]%*

In this example the list item fields will get the values of array variable *ResultArray*.

## 12.19 Move List Item to Subfolder

This workflow action is used to move a list item to a subfolder in the same list. Version history and attachments for the list item are retained.

Move list item at this url to this subfolder. Store output item url in Variable: outputUrl.

**See also:** to copy or move list item(s) to a different list, use "Copy List item at URL" and "Delete Item at URL" actions.

### Parameters:

**item at this url** - specify the URL of a list item to move.

For current item, use **Current Item: Encoded Absolute URL** lookup.

**this subfolder** - type in the name of a subfolder in the same list as the list item.

**Variable: outputUrl** - specify the string variable to store the URL of the list item in the new location.

## 12.20 Reject Item at URL

This workflow action is used to reject an item in a specified list.

Reject list item at this url item with specified comment.

### Parameters:

**this** - Full URL of list item.

**specified** - Comment text.

## 12.21 Remove All Attachments

This workflow action is used to remove attachments by name or by file mask from a list item specified by URL.

Remove all attachments from list item at this URL. Store number of removed attachments to Variable: removedCount

**Parameters:**

**all** - Multi-line string with one attachment file name or file mask per line. Leave blank to remove all attachments.

**this URL** - Full URL of list item (e.g. Current Item: Encoded Absolute URL).

**Variable: removedCount** - Output integer variable to store number of removed attachments.

## 12.22 Remove Content Type from List

This workflow action removes the specified content type from a list.

Remove this content type from list at this list URL.

**Parameters:**

**this** - Name of the content type.

**this list url** - Full URL of the target list.

## 12.23 Set Content Type Document Template

This workflow action sets a document template for a content type.

Set document template url for this content type to this url, copy document to the resource folder.

**Parameters:**

**this content type** - specify the content type name

**to this url** - specify the URL of the document that will be used as a template.

**copy** - select to copy/do not copy the document template to the resource folder.

## 12.24 Set Default Content Type on List

This workflow action sets a specified content type as the default one for the specified list. If the list does not contain the specified content type, it will be added automatically.

Set this content type as default content type on the list at this list URL.

**Parameters:**

**this content type** - Name of the content type.

**this list URL** - Full URL of the target list.

## 12.25 Update List Item at URL

This workflow action is used to set values of list item fields. It is possible to update **multiple** list items in a single action by providing a dictionary of URLs as an input parameter.

Values are specified as a multi-line string in one of two formats.

First, INI file format. For example:

Position=Team Leader

First Name=John

Last Name=Smith

Age=23

Second, XML format:

<item>

<field title="Position">Team Leader</field>

<field title="First Name">John</field>

<field title="Last Name">Smith</field>

<field title="Age">23</field>

</item>

**See Also:** *Create List Item*

Update list item with specified values at this list item URL.

**Parameters:**

**specified values** - Values for list item fields in INI or XML format.

For **Hyperlink** type fields it is also possible to update the description using *Url#description*:

**Example:**

*Title=new task*

*Url=http://www.harepoint.com*

*Url#description=HarePoint website*

**Multiselect Lookup fields** are supported, values can be either comma (,) or semicolon (;) delimited.

**Managed metadata** columns are supported, and it is possible to specify term label names from the Terms Set. Multiple labels should be delimited by a semicolon (;)

**Example:**

*FieldName=Label1;Label2;Label3*

**this list item URL** - Full URL of list item. **Dictionaries** of full URLs are also supported, in this case **multiple** items will be updated.

## 13. Network activities

### 13.1 Create Directory on FTP Server

This workflow action creates a directory on an FTP server.

To specify the server name, login and password, use *Set Authentication*.

Create specified directory in this folder on FTP server.

**Parameters:**

**specified** - Directory name on FTP server.

**this** - Folder name to create.

**Examples:**

See example in the *Set FTP Server Credentials* action description.

### 13.2 Delete Directory from FTP Server

This workflow action deletes a specified directory on the FTP server.

To specify the server name, login and password, use *Set Authentication*.

Delete this directory from FTP server. Do not delete directory if directory is not empty.

**Parameters:**

**this** - Directory name to delete.

**Do not delete directory** - Specifies whether to *delete child items* or *not to delete directory* at all.

### 13.3 Delete File from FTP Server

This workflow action deletes a specified file from the FTP server.

To specify the server name, login and password, use *Set Authentication*.

Delete this file from this directory on FTP server.

**Parameters:**

**this** file - File name to delete.

**this** directory - Directory name on the FTP server.

### 13.4 Download Document from FTP Server

This workflow action is used to download a document from FTP server to a SharePoint document library.

**Note:** "**Set Authentication**" action needs to be used before this workflow action to specify the parameters of the FTP server.

Download this document from FTP server into document library at this url folder with the same name. Overwrite if the file already exists.

**Parameters:**

**this file** - specify the server-relative URL of the file on FTP server. (The FTP server URL is specified in "Set FTP Server Options" action earlier).

**document library at this url** - Encoded absolute URL of the document library, or a folder in it, where the downloaded document should be saved.

**the same name** - (optional) specify the new document name to be used. By default, the original name will be used.

**Do not overwrite** - specify if the file should be overwritten if it already exists: *Do not overwrite, overwrite.*

## 13.5 Download Documents from FTP Folder

This workflow action is used to download multiple files from an FTP folder.

**Note:** use "Set Authentication" action to authenticate against the FTP server.

**See also:** "Download Document from FTP Server" action

Recursively download all files from FTP folder to this folder url. Overwrite existing documents.

### Parameters:

**Recursively** - choose whether to download the files within the subfolders (*recursively*) or not (*non-recursively*)

**all files** - specify to download *all* files (default), or only certain file types - use the standard *file mask*, e.g. \*.txt to download only the text documents.

**FTP folder** - specify the folder of an FTP server

**this folder url** - specify the URL of the local document library folder to receive the files.

**Overwrite** - choose to *overwrite/do not overwrite* if the files with these names already exist in the target location.

## 13.6 Fetch RSS Feed to Html

This workflow action is used to fetch all items from an RSS feed and print using a specified template.

Fetch RSS from this URL print by `<p>[DATE]: <a href="[URL]">[TITLE]</a></p><p>[DESCRIPTION]</p>` template and store in Variable: outputHtml

### Parameters:

**this URL** - URL of RSS feed.

**template** - Printing template for RSS item. [DATE], [URL], [TITLE] and [DESCRIPTION] in the template will be replaced by the corresponding RSS item fields.

**outputHtml** - Output string variable, will contain concatenated string with all feed items.

## 13.7 Fetch RSS Item

This workflow action is used to fetch the top item from an RSS feed.

**Note:** this workflow action can return long strings, use the built-in SharePoint workflow actions like *Extract Substring from Start of String* to prepare long strings for the *Log to History List* action.

Fetch RSS from this URL and store item number 1's title in Variable: title, link in Variable: link, description Variable: description and publication date in Variable: date.

### Parameters:

**this URL** - RSS feed address.

item number 1's - Number of items to fetch.

**Variable: title** - Output string variable for RSS item title.

**Variable: link** - Output string variable for RSS item link.

**Variable: description** - Output string variable for RSS item text.

**Variable: date** - Output date variable for RSS item publication date.

## 13.8 Send HTTP GET Request

This workflow action is used to send an HTTP GET request.

**See Also:** *Encode Web Text* and *Send HTTP POST Request*

Send HTTP request to this URL. Use GET method with this headers. Store response in Variable: response and status code in Variable: statusCode

### Parameters:

**this URL** - Target URL with parameters.

**this headers** - (Optional) Additional request headers each on new line. For example: *Accept-Language: fr*.

**Variable: response** - Output string variable to receive the remote server response, usually contains the requested web page body.

**Variable: statusCode** - Output integer variable for HTTP status code (example: 200).



## Examples:

Encode [%Current Item:Approver Comments%] using URL encoder and store in Variable: param.

then Send HTTP request to http://mysite/ q=[%Variable: param%] . Use GET method with this headers.  
Store response in Variable: pageBody and status code in Variable: code

## 13.9 Send HTTP OAuth Request

This workflow action sends an HTTP OAuth request. Various HTTP methods are available.

**Note:** *Set OAuth Authentication* needs to be used prior to this action.

See also: *Set OAuth Authentication*

Send HTTP OAuth request to this url using GET method with these headers and this request. Store response in variable.

### Parameters:

this url - this is the API endpoint, the URL of the web-resource to send the request to.

Please refer to "Developers API" documentation for details.

using GET method - select the HTTP method. Possible values:

- GET
- POST
- PUT
- DELETE

Please refer to "Developers API" documentation to determine which method should be used.

these headers - (Optional) specify the additional headers for this request, if necessary.

this request - (Optional) specify the request body, if necessary.

Variable - specify the variable where the server response will be saved.

## Examples:

### How to work with xero.com: Creating a new contact

(see also **Set OAuth Authentication** action)

**Step 1.** Obtain the base64-encoded private key. It can be generated using the command line utility *openssl*:

```
openssl gensrsa -out privatekey.pem 1024
```

more details here: <http://developer.xero.com/documentation/advanced-docs/public-private-keypair/>

**Step 2.** Obtain consumer key and consumer secret here: <https://app.xero.com/Application/List>

**Step 3.** Set up a workflow (this is the example to **create a new Contact**):

- Set xero.com OAuth authentication. Use LLWHHD7WS505 key, PRNCI5KNCHU76NSP secret, sign request with RSA-SHA1 MIICWwIBIjchPkmJKwcOL..., request token at this url, get an access token at this url
- then Send HTTP OAuth request to <https://api.xero.com/api.xro/2.0/cont...> using PUT method with these headers and <Contact> <Name>ABCDEFGG</Name> <F... request. Store response in Variable:variable
- then Log Variable:variable to the workflow history list

Comments for workflow action 1:

- The consumer key and consumer secret from **Step 2** are used in this action in the respective parameters
- The base64-encoded private key from **Step 1** is used in the additional parameter after RSA-SHA1

Comments for workflow action 2:

According to the xero.com API documentation, the parameters are filled as follows:

- request URL is <https://api.xero.com/api.xro/2.0/Contacts>
- HTTP method that is used is PUT
- request body is:

```
<Contact>
<Name>ABCDEFGG</Name>
<FirstName>John</FirstName>
<LastName>Smith</LastName>
<EmailAddress>john.smith@gmail.com</EmailAddress>
<Addresses>
<Address>
<AddressType>POBOX</AddressType>
<AddressLine1>P O Box 123</AddressLine1>
```

```

    <City>Wellington</City>
    <PostalCode>6011</PostalCode>
  </Address>
</Addresses>
<BankAccountDetails>01-0123-0123456-00</BankAccountDetails>
<TaxNumber>12-345-678</TaxNumber>
<AccountsReceivableTaxType>OUTPUT</AccountsReceivableTaxType>
<AccountsPayableTaxType>INPUT</AccountsPayableTaxType>
<DefaultCurrency>NZD</DefaultCurrency>
</Contact>

```

### 13.10 Send HTTP POST Request

This workflow action is used to send an HTTP POST request.

**See Also:** *Encode Web Text* and *Send HTTP GET Request*

Send HTTP request to this URL. Use POST method with this headers and this request. Store response in Variable: response and status code in Variable: statusCode

#### Parameters:

**this URL** - Target URL.

**this headers** - (Optional) Additional request headers each on new line. For example: *Accept-Language: fr.*

**this request** - (Optional) The body of the request. For example: *name=john&age=23.*

**Variable: response** - Output string variable to receive the remote server response, usually contains the requested web page body.

**Variable: statusCode** - Output integer variable for HTTP status code (example: 200).

#### Examples:

Encode [%Current Item:Approver Comments%] using URL encoder and store in Variable: param.

then Send HTTP request to http://mysite/script.aspx. Use POST method with this headers and query=[%Variable: param%] request. Store response in Variable: pageBody and status code in Variable: code

## 13.11 Send XMPP Instant Message

This workflow action is used to send instant messages to XMPP-compatible messaging clients like Google Talk.

A list of compatible clients is available here: <http://xmpp.org/xmpp-software/clients/>

Send XMPP instant message to this JID by using JID: this JID, password: this password and optional server: this server.

### Parameters:

message - Message text

to this JID - Receiver's JID

using JID: this JID - Sender's JID

this password - Sender's password

this server - (Optional) Network host to log on

### Examples:

Send XMPP message *Hello!* to myfriend@gmail.com by using JID: myaccount@gmail.com, password: mypassword and optional server: talk.google.com.

## 13.12 Set FTP Server Options

This workflow action is used to set FTP server name and protocol for other FTP-related activities like "Upload Document to FTP Server" or "Create Directory on FTP Server".

This activity must be placed above all other FTP-related activities in the step.

Later versions also support SFTP (SSH) servers.

**Note:** The credentials to the FTP server are provided using "Set Authentication" action.

Set FTP server options: use specified FTP server, specified protocol.

### Parameters:

specified FTP server - specify the URL of the FTP server

specified protocol - specify the protocol for the connection. Available options:

- *FTP (no SSL)* - simple FTP connection without encryption. Default port is usually 21.
- *FTPS (FTP over SSL)* - FTP connection using SSL. Default port is usually 990.
- *SFTP (SSH)* - Transferring files over SSH. Default port is usually 22.

### 13.13 Set OAuth Authentication

This workflow action is used to set OAuth authentication.

**See also:** *Send HTTP OAuth Request* action

Set general OAuth authentication. Use this consumer key, this consumer secret, sign request with default, request token at this url, get access token at this url

#### Parameters:

**general** - select the site to authenticate against. Possible values:

- xero.com
- general

**this consumer key** - specify the consumer key used for authentication. You can obtain it from the respective web site, usually in "Developers API" section.

**this consumer secret** - specify the consumer secret used for authentication. You can obtain it from the respective web site, usually in "Developers API" section.

**sign request with default** - select the method to sign the requests to the server. Possible values:

- default
- plain text
- hmac-sha1
- rsa-sha1 (**required for xero.com**)

Please refer to "Developers API" section for more details.

When **rsa-sha1** is selected, an **additional parameter** would need to be filled:

**RSA key** - (only when rsa-sha1 is selected in previous parameter) paste the base64-encoded private key from **.pem** file here. (The instructions on how to obtain .pem file are available in the description on the respective site).

**request token at this url** - (Optional) specify the URL to obtain the request token. It is used during the authentication process, e.g. for digital signature.

**access token at this url** - (Optional) specify the URL to exchange a request token for an access token. It is used to access API functionality.

#### Examples:

Please refer to *Send HTTP OAuth Request* for an example on how to use these actions for **xero.com**.

### 13.14 Shorten Url

This workflow action generates a short URL using online providers for short URLs.

Shorten this url with goo.gl and store result in Variable:shortenUrl

#### Parameters:

**this url** - specify URL to be shorted

**with goo.gl** - select the provider

Currently two providers are available:

- *Goo.gl* - does not require registration, so login and password are not required
- *Bit.ly* - requires registration on Bitly.com, login and password should be specified in action parameters.

**store result in Variable:variable** - the shortened URL is stored in this variable.

### 13.15 Upload Document to FTP Server

This workflow action uploads a list item on an FTP server.

To specify the server name, login and password, use *Set Authentication*.

Upload specified document to FTP server into this folder. If file already exists do nothing.

#### Parameters:

**specified** - List item to upload.

**this** - FTP directory name.

**do nothing** - Action taken if the file already exists. Four options are available:

- *do nothing*
- *treat as an error* - generate error and stop workflow
- *upload with unique name* - add unique identifier like {11fba226-59d6-4f31-b9c4-e43ddc67e048} to the end of file name
- *overwrite* the existing file

#### Examples:

See example in *Set FTP Server Credentials*.

### 13.16 Upload Documents to FTP Folder

This workflow action is used to upload multiple documents from a specific folder of a document library to a folder on an FTP server.

**Note:** Use "Set Authentication" action to authenticate to the FTP server.

**See also:** "Upload Document to FTP Server" action

Recursively upload all documents from this folder url to FTP folder. Overwrite existing files.

#### Parameters:

**Recursively** - choose to upload files from subfolders (*recursively*), or not (*non-recursively*).

**all documents** - specify to upload *all* documents, or only documents of certain types - use *standard file mask*, e.g. \*.txt to upload only the text files.

**Overwrite** - choose to *overwrite*/*do not overwrite* if the files with these names already exist in the FTP folder.

### 13.17 Upload List Item Attachments to FTP Server

This workflow action is used to upload list item attachments to an FTP server directory.

To specify the server name, login and password, use *Set Authentication*.

Upload specified list item attachments to FTP server into this folder. If the file already exists do nothing.

#### Parameters:

**specified** - List item whose attachments will be uploaded, e.g. Current Item.

**this** - Directory name on FTP server.

**do nothing** - Action taken if the file already exists. Four options are available:

- *do nothing*
- *treat as an error* - generate error and stop workflow
- *upload with unique name* - add unique identifier like {11fba226-59d6-4f31-b9c4-e43ddc67e048} to the end of file name
- *overwrite* the existing file

## 14. Office activities

### 14.1 Accept Revision Changes in Word Document

This workflow action accepts all revision changes in the specified Word document.

**Note:** Only DOCX documents are supported.

Accept all revision changes in Word document at this URL

#### Parameters:

**this URL** - the URL of the Word document, e.g. Current Item: Encoded Absolute URL

### 14.2 Add Digital Signature

This workflow action is used to add a digital signature to the specified document. PFX certificate file and password are needed for this procedure.

Add digital signature to the document at this url using certificate at this url protected with this password

#### Parameters:

**document at this url** - specify the full URL to a document that needs to be signed.

To point to a current document (that workflow has triggered for), you can use Current Item: Encoded Absolute URL lookup.



**certificate at this url** - specify the full URL to a .pfx certificate file.

**this password** - specify the password for the PFX certificate provided

### 14.3 Add New Row into Word Document

This workflow action adds a copy of the last row in the Repeating Section Content Control. After this has been added, it can be modified using *Update Word Document*.

To create a Repeating Section Content Control, highlight the necessary row in the Word Document, and on the ribbon on the Developer Tab click 'Add Repeating Section Content Control' button.

**See Also:** An article with a detailed example can be found here:

<http://www.harepoint.com/Products/HarePointWorkflowExtensions/Example-Repeating-Content-Workflow.aspx>

Add new row into this repeating section content control of this .docx document

#### Parameters:

**this repeating section** - the name of the repeating content control

**this .docx** document - the reference to the document, e.g. Current Item

### 14.4 Add Watermark to Word Document

This workflow action is used to add watermarks to Word documents.

**Note:** Only **DOCX** format (OpenXML) of Word documents is supported.

**See also:** "Add watermark to PDF document" and "Set PDF Watermark Options" actions to add watermarks to PDF documents.

Add image at this url as a watermark to the Word document at this url

#### Parameters:

**image at this url** - specify the full URL of an image to be used as a watermark

**Word document at this url** - specify the full URL of a Word document to add a watermark to.

## 14.5 Find Text in Word Document

This workflow action is used to check for the presence of a specified substring in a Microsoft Word document.

**Note:** only DOCX documents are supported.

Find this text in this document, match case: no. Store result in Variable: isFind

### Parameters:

**this text** - Input string to find in Word document.

**this** - Reference to the document, e.g. Current Item.

**no** - Is the search case-sensitive *Yes* or *no*.

**Variable: variable** - Output boolean variable.

## 14.6 Get Excel Cell Value

This workflow action is used to get cell value at Microsoft Excel document on a SharePoint server.

**Note:** Only xlsx format documents are supported.

Get value of this cell of specified sheet in this document and store in Variable: cellValue

### Parameters:

**this** - Cell address, e.g. A1.

**specified** - Name of the sheet, e.g. Sheet1.

**this** - the reference to an XLSX document, e.g. Current Item.

**Variable: CellValue** - Output string variable.

## 14.7 Get Office Document Property

This workflow action is used to get the property of a document on a SharePoint server. With this action, you can get the following properties:

- Title

• Author
• Keywords
• Subject
• Comments
• Last Modified By
• Category
• Manager
• Company

**Note:** only Open Office XML (Microsoft Office 2007-2013) formats are supported: DOCX, XLSX, PPTX.

**See Also:** Open Office XML at Wikipedia ( [http://en.wikipedia.org/wiki/Office\\_Open\\_XML](http://en.wikipedia.org/wiki/Office_Open_XML) ).

Get this property of this office document and store in Variable: docProperty

#### Parameters:

**this property** - Office document property name (see description above).

**this** - Reference to document, e.g. Current Item.

**Variable: docProperty** - Output string variable.

## 14.8 Get Word Document Page Count

This workflow action is used to get the page count of a word document.

Only .docx (Word 2007 and later) version is supported.

**Note:** the .docx file only stores the static information about the page count, which is updated when the document is rendered by MS Word application. That is, this count **may be incorrect** if the document was closed before rendering was complete, or if specific views (such as "Web document" view) has been used.

Get the page count of document at this url and save to Variable: variable

#### Parameters:

**this url** - specify the full URL of the document to be processed

**save to Variable: variable** - specify the output variable, to store the page count.

## 14.9 Read Word Document

This workflow action is used to get the text of a named content control of a Microsoft Word document. See *Update Word Document* for details of how to add named content control to a Microsoft Word document.

**Note:** only DOCX documents are supported.

Read this content control of this document and store its value in Variable: docText

### Parameters:

**this content control** - Content control name, see overview section for more details.

**this** - Reference to a document, e.g. Current Item.

**Variable: docText** - Output string variable.

## 14.10 Read Word Document To Text

This workflow action is used to get the text of a Microsoft Word document and save it to a string variable.

**Note:** only DOCX documents are supported.

Read this document and store its text in Variable: docText

### Parameters:

**this** - Reference to a document, e.g. Current Item.

**Variable: docText** - Output string variable.

## 14.11 Replace External Hyperlink in Word Document

This workflow action is used to replace the URLs and/or text labels in external hyperlinks in Word document.

**Note:** only DOCX (OpenXML) format documents are supported.

Replace all occurrences of this hyperlink with this hyperlink in document at this url, apply changes to: both URL and text, match case: no, increment version number: yes.

**Parameters:**

**all occurrences** - choose which of the matching hyperlinks should be replaced:

- *all occurrences*
- *first occurrence*
- *last occurrence*

**this hyperlink** - specify the text string to search in hyperlink URL and/or text label

**this hyperlink** - specify the replacement text string.

**Note:** When replacing URLs, make sure that resulting string after replacement will be a valid URL.

**document at this url** - specify the full URL of the document that should be changed.

**both URL and text** - choose if

- *only text labels*
- *only URLs*
- *both*

should be replaced

**match case: no** - choose if both search and replacement should be case sensitive, or not.

**increment version number: yes** - choose to create a new version of a document, or keep changes within the current version.

## 14.12 Replace Text in Word Document

This workflow action is used to replace text in Microsoft Word document. This action can be used for generation of bills and invoices, filling forms and addresses on envelopes within a workflow.

**Note:** only DOCX documents are supported.

**See Also:** 'Update Word Document' action to update content control fields, 'Replace external hyperlinks in Word document' action to replace hyperlink URLs and/or text labels.

Replace all occurrences of this text in this document with this text, match case: no .

**Parameters:**

**all occurrences** - With this action it is possible to replace *all occurrences, first occurrence or last occurrence* of specified pattern.

**this text** - Text to replace.

**this** - Reference to a document, e.g. Current Item.

**this text** - New string.

**no** - Case sensitive: *yes* or *no*.

### 14.13 Set Excel Cell Value

This workflow action is used to set cell value at Microsoft Excel document on a SharePoint server.

**Note:** Only xlsx format documents are supported

Set value of this cell of specified sheet in this document to this value

**Parameters:**

**this** - Cell address, e.g. A1.

**specified** - Name of the sheet, e.g. Sheet1.

**this** - Reference to an XLSX document, e.g. Current Item.

**this value** - String, Integer, Number, Date/Time or Boolean variable with a new cell value.

### 14.14 Set Office Document Property

This workflow action is used to set the value of document property of a document on a SharePoint server. With this action, you can set the value for the following properties:

- Title
- Author

• Keywords
• Subject
• Comments
• Last Modified By
• Category
• Manager
• Company

**Note:** MS Office 'Open XML' (Microsoft Office 2007-2013) formats are supported only: DOCX, XLSX, PPTX.

**See Also:** MS Office 'Open XML' at Wikipedia ( [http://en.wikipedia.org/wiki/Office\\_Open\\_XML](http://en.wikipedia.org/wiki/Office_Open_XML) ).

Set this property of this office document and to this value

**Parameters:**

**this property** - Office document property name (see description above).

**this** - Reference to a document, e.g. Current Item.

**this value** - String variable with the new value.

## 14.15 Update Multiple Fields in Word Document

This workflow action is used to update multiple content control fields in Word document in a single run, which is more convenient and efficient than using "Update Word Document" action several times in a workflow.

**See also:** "Update Word document" action to update a single content control field.

**Note:** only DOCX (OpenXML) Word files are supported.

Set values of the following content controls to the specified values (all occurrences) for document at this url, text format: plain text.

**Parameters:**

**the specified values** - specify the content control field names and their new values. Two formats are supported:

#### INI-style:

```
control name1=value1  
control name2=value2  
...
```

#### XML-format:

```
<document>  
<field title="control name1">value1</field>  
<field title="control name2">value2</field>  
<field title="control name3">value3</field>  
...  
</document>
```

**Note:** New line breaks are only possible with Rich Text content control fields, **<br/>** tag and **HTML** format selected - see below.

**all occurrences** - specify which content control fields should be updated if the document has several content control fields with the same name:

- *all occurrences*
- *last occurrence (repeating content control)*

**at this url** - specify the full URL of a document to be processed.

**text format** - select the format for **all** content control fields that will be updated. Two options available:

- *plain text*
- *HTML* - supports tags for text formatting, including **<br/>** tag for new line break

**Note:** If you need to update content controls of different formats, either group them by the same formats and update them in two "Update Multiple Fields In Word Document" actions, or use "Update Word Document" action for each content control individually.

## 14.16 Update Picture in Word Document

This workflow action sets the picture located at a specified URL to the picture content control of the specified document.

**See Also:** *Update Word Document*



Set this picture content control of this document to picture at this url

**Parameters:**

**this picture content control** - the picture content control to be set

**this** document - the URL of the document to be modified

**this url** - the direct URL to the picture within this SharePoint farm

**Examples:**

Set logo of Invoices document to picture at <https://spsite/Shared%20Documents/newpic.jpg>

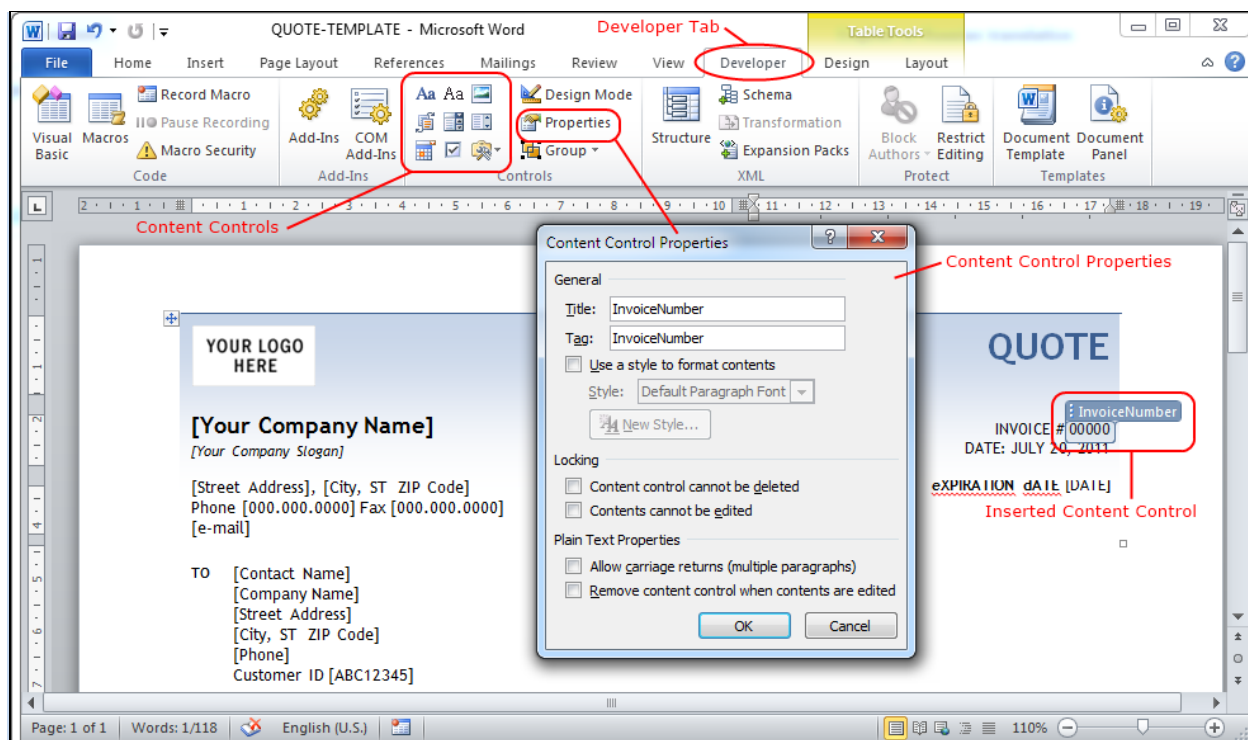
## 14.17 Update Word Document

This workflow action updates the text of a named content control in a Microsoft Word document. This action can be used for generation of bills and invoices, filling forms, and addresses on envelopes within a workflow.

To place a content control, you should first enable **Developer** Tab in Microsoft Word:

1. Click the Microsoft Office Button , and then click Word Options.
2. Click Popular.
3. Select the Show Developer tab in the Ribbon check box, and then click OK.

On the Developer Tab, select the content control on the Ribbon, insert it into the document and assign it a name using the Properties button:



**See Also:** 'Replace Text in Word Document' action; 'Update Multiple Fields In Word Document' action to update multiple content control fields in one run; 'Replace external hyperlinks in Word document' action to replace URLs and/or text labels.

Set value of this content control (all occurrences) of this document to this value, text format: plain text

#### Parameters:

**this content control** - Content control name, see overview section for details.

**all occurrences** - specify which content controls with the specified name should be updated, if the document has several content controls with the same name:

- *all occurrences*
- *last occurrence (repeating content control)*

**this** document - Reference to document, e.g. Current Item.

**this value** - String with the new text for content control.

**plain text** - specify the format of the text. Available values: *plain text*, *HTML*.

#### Examples:

### Example 1: text content controls

*Set value of InvoiceNumber (all occurrences) of Current Item .docx document to 12345, text format:HTML*

This workflow action replaces the content of "InvoiceNumber" content control with 12345.

### Example 2: checkbox content controls

*Set value of Checkbox1 (all occurrences) of Current Item .docx document to true, text format: plain text*

This workflow action sets the "Checkbox1" content control as checked.

## 15. Pdf activities

### 15.1 Add Image Watermark to PDF

This workflow action is used to add an image watermark to a PDF document.

**See also:** "Add Text Watermark to PDF" action.

Add image watermark at this url to pdf at this url, place in background layer at middle center, rotate 0 degrees and zoom 100 %

#### Parameters:

**watermark at this url** - specify the full URL of an image that will be added as a watermark.

It is recommended to upload an image to a document library in SharePoint first, and then point to it there.

**pdf at this url** - specify the full URL of a PDF document

**background layer** - specify the layer of the watermark: *background* or *foreground*

**middle center** - specify the watermark alignment:

- *top left*
- *top center*
- *top right*
- *middle left*

- *middle center*
- *middle right*
- *bottom left*
- *bottom center*
- *bottom right*
- *random*

**rotate 0 degrees** - specify if watermark image needs to be rotated.

**zoom 100%** - specify how the watermark image needs to be scaled

## 15.2 Add PDF Letterhead

This workflow action is used to add a PDF letterhead to an existing PDF document.

Add pdf letterhead at [this url](#) to the document at [this url](#)

**Parameters:**

**letterhead at [this url](#)** - specify the full URL of the letterhead.

**document at [this url](#)** - specify the full URL of the PDF document,

## 15.3 Add Text Watermark to PDF

This workflow action is used to add a text watermark to an existing PDF document.

**See also:** "Add Image Watermark to PDF" action.

Add [this text](#) watermark to pdf at [this url](#), place in [background](#) layer, at [middle center](#), use [this](#) font, [this](#) size, [black](#) color, [100%](#) opacity

**Parameters:**

**[this text](#) watermark** - specify the text for the watermark.

**pdf at [this url](#)** - specify the full URL of a PDF document.

**[background](#) layer** - choose the layer for the watermark: *background* or *foreground*.

**middle center** - specify the watermark alignment:

- *top left*
- *top center*
- *top right*
- *middle left*
- *middle center*
- *middle right*
- *bottom left*
- *bottom center*
- *bottom right*
- *random*

**this font** - specify the font name for the watermark.

**this size** - specify the font size.

**black color** - specify the font color

**100% opacity** - specify the watermark text opacity

## 15.4 Merge PDF Documents

This workflow action is used to merge several PDF documents into one.

Merge pdf files at these urls and save merged pdf to this document url

**Parameters:**

**these urls** - specify the URLs of PDF documents to be merged.

**this document url** - specify the URL of an output document.

## 15.5 Sign PDF Document

This workflow action adds the specified picture (usually a handwritten signature) to a PDF document.

Sign the document at [this url](#) using image at [this url](#) and put it on page 1 at [top left](#) corner, indent left: 1 in., top: 1 in

**Parameters:**

<b>document at <a href="#">this url</a></b> - specify the full URL of a PDF document to add a picture.
<b>image at <a href="#">this url</a></b> - specify the full URL of a picture to be added.
Supported picture formats: gif, png, jpg, bmp, tiff, wmf, emf, ico.
<b>page <u>1</u></b> - specify the page number in PDF document to add the picture to.
You can use '-1' to point to the last page in a document.
<b><a href="#">top left corner</a></b> - specify where the picture should be located on the page:
<i>top left, top center, top right, middle left, middle center, middle right, bottom left, bottom center, bottom right</i>
<b>left: <u>1</u> in</b> - specify the picture indent from the left side, in inches
<b>top: <u>1</u> in</b> - specify the picture indent from the top, in inches

## 15.6 Split PDF Documents

This workflow action is used to split a PDF document into several PDF documents.

Split pdf document at [this url](#) by [this number of pages](#) and save files in the folder at [this url](#), [overwrite](#) existing files

**Parameters:**

<b>document at <a href="#">this url</a></b> -specify the full URL of an original PDF document.
<b><a href="#">this</a></b> - specify the value for the next parameter.
<b><a href="#">number of pages</a></b> - specify how the PDF document should be split - by <i>number of pages</i> or by <i>bookmark level</i> .
The corresponding value (e.g. number of pages) needs to be specified in the previous parameter.
<b>folder at <a href="#">this url</a></b> - specify the full URL of a target document library/folder to store the output files.
<b><a href="#">overwrite</a></b> - choose to <i>overwrite</i> or <i>do not overwrite</i> if the documents with such names already exist in the target document library

## 16. Permissions activities

### 16.1 Change Permissions Inheritance at URL

This workflow action starts or stops inherited permissions for a SharePoint site, list, list folder or list item specified by URL.

The action can **process multiple items** if a dictionary of URLs is provided.

Start inheriting permissions at this url

#### Parameters:

**Start/stop** - Action to perform concerning inheritance.

**this url** - Full URL of a SharePoint site, list, list folder or list item.

**Dictionaries of URLs** are also supported. In this case all items from the dictionary will be processed.

#### Examples:

Set Variable: item to [%Workflow Context:Current Item URL%].

then Stop inheriting permissions at Variable: item.

### 16.2 Delete All Permissions at URL

This workflow action is used to delete all permissions from a SharePoint site, list, list folder or list item specified by URL.

Permissions for **multiple items** can be deleted with this action if a dictionary of URLs is provided.

Delete all permissions at this url

#### Parameters:

**this url** - Full URL of a SharePoint site, list, list folder or list item.

**Dictionaries of URLs** are also supported. In this case all items from the dictionary will be processed.

## 16.3 Delete All User Permissions at URL

This workflow action is used to delete all permissions for specified user from a SharePoint site, list, list folder or list item specified by URL.

Permissions for **multiple items** can be deleted with this action if a dictionary of URLs is provided.

Delete all permissions at this url for this user.

### Parameters:

**this url** - Full URL of a SharePoint site, list, list folder or list item.

**Dictionaries of URLs** are also supported. In this case all items from the dictionary will be processed.

**this user** - specify the user to delete the permissions for.

## 16.4 Delete Permissions at Url

This workflow action is used to remove site users permissions from SharePoint site, list, list folder or list item specified by URL.

Delete specified permissions to person at this url.

### Parameters:

**specified permissions** - specify permissions to delete

**person** - specify a person to delete permissions for

**this url** - specify the URL of a site, list, list folder, etc.

## 16.5 Grant Permissions at Url

This workflow action is used to grant permissions for site users to a SharePoint site, list, list folder or list item specified by URL.

Grant specified permissions to person at this url.

### Parameters:

**specified permissions** - specify permissions to be granted to a person.

**person** - specify the person to grant the permissions for.



**this url** - specify the URL of a site, list, list folder, etc

## 17. Social activities

### 17.1 Create Group in Yammer

This workflow action creates a new group in Yammer with a specified title and description. The group can be public or private.

Create new public group in Yammer with this title, this description, use this access code

#### Parameters:

**public group** - choose if the new group should be *public* or *private*.

**this description** - specify description for the group

use **this access code** - specify the access code:

There are two ways to receive it:

- Using **Social Authorize Tool** from the **Client Components** of HarePoint Workflow Extensions (use Installer to install Client Components)
- Using the following link: <http://auth.mapilab.com/Yammer/RequestToken>

### 17.2 Create MailChimp Campaign

This workflow action is used to create a new MailChimp campaign.

**Note:** this action only creates a campaign, but does not launch it.

**See also:** *Send MailChimp campaign*

Create a new MailChimp campaign from html with this subject and this body to recipient name and recipient emails list. Use this api key, store campaign id in Variable: variable.

#### Parameters:

**html/plain text** - select whether e-mail body should be in HTML or Plain Text.

**this subject** - the e-mail subject, and also the campaign name on MailChimp.

**this body** - e-mail body

**recipient name** - (Optional) the recipient name for "To:" field.

**Macros** are supported:

- *E-mail address \*|EMAIL|\**
- *First name \*|FNAME|\**
- *Last name \*|LNAME|\**
- *Unsubscribe link \*|UNSUB|\**
- *Forward to a Friend Link \*|FORWARD|\**
- *Update Profile Link \*|UPDATE\_PROFILE|\**
- *MonkeyRewards Affiliate Link \*|REWARDS|\**

**recipient emails list** - the name of the recipient e-mails list from MailChimp. This list contains the e-mails of the recipients for this campaign.

**this api key** - the authorization key. See <http://kb.mailchimp.com/accounts/management/about-api-keys>

**store campaign id in Variable:variable** - specify the variable where the ID for this campaign will be stored.

## 17.3 Create User in Yammer

This workflow action creates a new user in the Yammer network with specified parameters.

**Note:** Use "Set Authentication" action to specify access credentials

Create user in Yammer with this e-mail, this full name, this job title, this department

**Parameters:**

**this e-mail** - specify the e-mail address of the new user

**this full name** - specify the full name of the new user

**this job title** - specify the job title of the new user

**this department** - specify the department of the new user

## 17.4 Delete MailChimp Campaign

This workflow action is used to delete a [MailChimp](#) campaign permanently.

**Important note:** the deleted campaign cannot be restored!

Delete [this](#) MailChimp campaign permanently. Use [this](#) api key.

### Parameters:

**[this](#) MailChimp campaign** - specify either the *ID* or *name* of the campaign to be deleted.

**Note:** If the name is specified and there are multiple campaigns with this name, the campaign most recently created will be removed.

**[this](#) api key** - the authorization key. See <http://kb.mailchimp.com/accounts/management/about-api-keys>

## 17.5 Delete User from Yammer

This workflow action deletes a specified user from Yammer. The e-mail address is used to identify the user.

Delete user with [this](#) e-mail from Yammer

### Parameters:

**user with [this](#) e-mail** - specify the e-mail address of the user

## 17.6 Follow user/site/tag/document in SharePoint

This workflow action allows following of a certain user/site/tag/document in SharePoint.

Follow [this](#) [user](#).

### Parameters:

**[this](#)** - specify the name of the user, URL of the site, tag name, or document name to follow.

**[user](#)** - the following values are available:

- *User*
- *Document*

- *Site*
- *Tag*

## 17.7 Get User's Property from Yammer

This workflow action gets the value of a specified property from the profile of the specified user in Yammer.

Get specified property of Yammer user with this e-mail address, store result in Variable: userProperty

### Parameters:

**specified property** - specify the profile property to get. Available properties are:

*About Me, Birthday, Department, Expertise, External Profiles, Full Name, Job Title, IM Provider, IM Username, Interests, Kids' Names, Location, Mobile Phone, Significant Other, Work Phone, Employer, Position Held, Job Description, Employment Start Year, Employment End Year, School, Degree/Diploma, School Description, Start Year, End Year.*

**user with this e-mail address** - specify the user's e-mail address

**store result in Variable: userProperty** - specify the variable to store the output

## 17.8 Invite User to Yammer

This workflow action initiates sending an invitation e-mail from the Yammer network to a specified e-mail address

Send Yammer invitation e-mail to this e-mail address

### Parameters:

**this e-mail address** - specify the recipient's e-mail address

## 17.9 Publish Message to Social Media via dlvr.it

This workflow action allows posting to Facebook, Twitter, Google+ and other social networks using dlvr.it.

**Note:** This action requires registration on dlvr.it.

A free subscription allows posting to 3 networks.

Publish message with this text, this title and attach image at this url using this dlvr.it API key and this route

**Parameters:**

**this text** - message text

**this title** - (optional) message title

**attach image at this url** - (optional) path to an image to be attached to a post

**his API key** - user's secret key to access dlvr.it API

The API key can be obtained here: <https://app.dlvr.it/settings/account>

**this route** - route name for dlvr.it that will be used to post message.

The route is set up in dlvr.it (<https://app.dlvr.it/deliveries>) right after you log into it. Add RSS there, then add those social networks that you wish to repost to.

Specify the route name in this parameter.

## 17.10 Publish Message to Yammer

This workflow action posts a message to any group of Yammer corporate network; a file can be attached from a specified document library.

Publish this message to All Company Yammer group, attach file at this url.

**Parameters:**

**this message** - a text of the message to be posted

to **All Company** Yammer group - select the Yammer group where this message should be posted

attach **file at this url** - specify the URL of the file from a Document Library in SharePoint

## 17.11 Replicate MailChimp Campaign

This workflow action is used to replicate (create a copy of) an existing MailChimp campaign.

Replicate this MailChimp campaign. Use this api key. Store replicated campaign id in Variable: variable.

**Parameters:**

**this MailChimp campaign** - specify either the *ID* or *name* of the campaign to be replicated.

**Note:** If the name is specified and there are multiple campaigns with this name, the campaign most recently created will be replicated.

**this api key** - the authorization key. See <http://kb.mailchimp.com/accounts/management/about-api-keys>

**store campaign id in Variable:variable** - specify the variable where the ID for the new campaign will be stored.

## 17.12 Schedule MailChimp Campaign

This workflow action is used to schedule a MailChimp campaign to be sent on a certain date at a certain time.

**Note:** A campaign can also be sent immediately, see the *Send MailChimp Campaign* action.

Schedule this MailChimp campaign for this time. Use this api key.

**Parameters:**

**this MailChimp campaign** - specify either the *ID* or *name* of the campaign to be scheduled.

**Note:** If the name is specified and there are multiple campaigns with this name, the campaign most recently created will be scheduled.

**this time** - specify the time in GMT (Greenwich Mean Time) time zone, in 24h format as follows:

yyyy-mm-dd hh:mm:ss

**Example:**

2013-12-30 20:30:00

**this api key** - the authorization key. See <http://kb.mailchimp.com/accounts/management/about-api-keys>

## 17.13 Send MailChimp Campaign

This workflow action is used to initiate sending of a [MailChimp](#) campaign immediately.

**Note:** A campaign sending can also be scheduled - see the *Schedule MailChimp Campaign* action.

Send [this](#) MailChimp campaign immediately. Use [this](#) api key.

### Parameters:

**[this MailChimp campaign](#)** - specify either the *ID* or *name* of the campaign to be sent.

**Note:** If the name is specified and there are multiple campaigns with this name, the most recently created campaign will be sent.

**[this api key](#)** - the authorization key. See <http://kb.mailchimp.com/accounts/management/about-api-keys>

## 17.14 Send Twitter Message

This workflow action sends a message to twitter. To set up this action, you must create an authorization token with the "Twitter Authorization Tool" (available for free at HarePoint.Com).

Run the tool and click the "Request Token" button. After that, a Twitter web-page will be opened:

Enter your credentials and press the Allow button on the web page to get the PIN number:



**You've successfully granted access to HarePoint Workflow Activities !**

Simply return to HarePoint Workflow Activities and enter the following PIN to complete the process.

**3591757**

Enter the PIN number to the "Twitter Authorization Tool" and get an authorization token:

HarePoint Workflow Activities

Authorize token:  
Mjc2OTI3Mjl4LVZlVDJpcXBScUQ2dw1UQ0t3QkVsMWhjQ1cxdkhkdG81R21B  
cINOUVYkzMzg4MkZBR1DNUQwLTQ0RUQ1QUVEMC0yOURDNDgyNTdFOUQ  
xdEZuRWhPwFVHwUpQbEJuU1gzTjhYc3I5SDBkT1BLdZlzeUwzWk5B

Request Token

Copy to Clipboard

Press the "Copy to Clipboard" button and paste the authorization token to the second parameter of this action.

Send message with specified twitter authorize token.

**Parameters:**



**message** - Message text. Maximum length is 140 characters, longer messages will be cut off automatically.

**specified** - Twitter authorization token generated with the Twitter Authorization Tool.

## 17.15 Set User's Property in Yammer

This workflow action sets a specified property in the profile of a specified user in Yammer. The E-mail address is used to identify the user.

Set specified property of Yammer user with this e-mail address to this value

### Parameters:

**specified property** - select the property to set. Available properties are: *About Me, Birthday, Department, Expertise, External Profiles, Full Name, Job Title, IM Provider, IM Username, Interests, Kids' Names, Location, Mobile Phone, Significant Other, Work Phone, Employer, Position Held, Job Description, Employment Start Year, Employment End Year, School, Degree/Diploma, School Description, Start Year, End Year.*

**user with this e-mail address** - specify the user's e-mail address

**this value** - specify the new value for this property

## 17.16 Stop following user/site/tag/document in SharePoint

With this workflow action you can stop following a user/site/tag/document in SharePoint.

Stop following this User.

### Parameters:

**this** - specify the name of the user, URL of the site, tag name, or document name to stop following.

**User** - following values are available:

- *User*
- *Document*
- *Site*
- *Tag*

## 17.17 Subscribe to MailChimp List

This workflow action adds a new subscriber to a MailChimp list.

Subscribe this e-mail address to this MailChimp list. Use this api key.

### Parameters:

**this e-mail address** - specify the e-mail address of a new subscriber.

**this MailChimp list** - specify the name of the MailChimp list where the new subscriber needs to be added.

**this api key** - the authorization key. See <http://kb.mailchimp.com/accounts/management/about-api-keys>

## 17.18 Unsubscribe from MailChimp List

This workflow action is used to unsubscribe people from your MailChimp list.

Unsubscribe this e-mail address from this MailChimp list. Use this api key, delete completely: no, send goodbye: yes, notify: yes .

### Parameters:

**this e-mail address** - specify e-mail address of a person who needs to be unsubscribed.

**this MailChimp list** - specify the name of the MailChimp list from which the subscriber should be removed.

**this api key** - the authorization key. See <http://kb.mailchimp.com/accounts/management/about-api-keys>

**delete completely: no/yes** - select whether the subscriber should be removed completely from the list, or only marked as unsubscribed.

**send goodbye: yes/no** - specify whether an e-mail should be sent to a person, informing that he or she is now unsubscribed.

**notify: yes/no** - specify whether an e-mail should be sent to the administrator of the MailChimp list informing that a user has been unsubscribed.

## 17.19 Update Group in Yammer

This workflow action modifies the parameters of an existing group in Yammer.

Update this group in Yammer, set this title, this description, make it public

### Parameters:

**this group** - specify the group to be updated

**this title** - specify the new title for the group

**this description** - specify the new description for the group

**make it public** - choose to make this group public or private

## 18. Strings activities

### 18.1 Capture All Occurrences of Text

This workflow action is used to store all matches of a regular expression in a specified text to output to an array variable.

Regular expressions provide a powerful, flexible, and efficient method for processing text. To learn more about regular expressions, see: <http://msdn.microsoft.com/en-us/library/hs600312.aspx>

For a regular expression language reference, see: <http://msdn.microsoft.com/en-us/library/az24scfc.aspx>

**Note:** To set options for this activity, like case sensitivity, use *Set Regular Expression Options*.

Capture regular expression in this text and store all occurrences in Variable: matchResults

### Parameters:

**regular expression** - Regular expression pattern. For example: `"b(\w+ )s1\b"`.

**this text** - Input text, expression, item field or variable.

**Variable: matchResults** - Output dictionary variable. Each item contains one match of the regular expression.

### Examples:

*Set regular expression options: ignore case - Yes, is pattern culture invariant - Yes, use multiline mode - No, use single line mode - No, search right to left - No. Store options to Variable: REOptions*

Capture \b[A-Z0-9.\_%]+@[A-Z0-9.-]+\.[A-Z]{2,4}\b in xxx kkk@my.com owkk@ @kkk s@i12.com ddd and store all occurrences in Variable: allOccurrences

then Join Variable: allOccurrences at index 0 next -1 items using ; and store in Variable: emails

then Log Variable: emails to the workflow history list

Result in workflow history list:

kkk@my.com; s@i12.com

## 18.2 Capture Text with Regular Expression

This workflow action is used to store the first match of a regular expression in a specified text to output a text variable.

Regular expressions provide a powerful, flexible, and efficient method for processing text. To learn more about regular expressions, see: <http://msdn.microsoft.com/en-us/library/hs600312.aspx>

For a regular expression language reference, see: <http://msdn.microsoft.com/en-us/library/az24scfc.aspx>

**Note:** To set the options for this activity, like case sensitivity, use *Set Regular Expression Options*.

Capture regular expression in this text and store in Variable: firstMatchResult

### Parameters:

**regular expression** - Regular expression pattern. For example, '\b(\w+ )s\1\b'.

**this text** - Input text, expression, item field or variable.

**Variable: firstMatchResult** - Output string variable with the text of the first match of the regular expression.

### Examples:

Set regular expression options: ignore case - No, is pattern culture invariant - Yes, use multiline mode - No, use singleline mode - No, search right to left - No.

then Capture .est in IT IS TEST and store in Variable: r1

then Set regular expression options: ignore case - Yes, is pattern culture invariant - Yes, use multiline mode - No, use singleline mode - No, search right to left - No.

then Capture .est in IT IS TEST and store in Variable: r2

then Log <[%Variable: r1%]>, <[%Variable: r2%]> to the workflow history list

Result in workflow history list:

<>, <TEST>

## 18.3 Convert Base64 to Text

This workflow activity will decode Base64 string, interpret the resulting data as a string in one of five available encodings, and copy to the output system string variable.

Available encodings:

- default - 8-bit encoding of the current operating system's ANSI code page.
- ASCII - 7-bit ASCII encoding.
- Unicode - also known as UTF16, represents each character as a sequence of one or two 16-bit integers.
- UTF8 - uses one byte for any ASCII characters that have the same code values in both UTF-8 and ASCII encoding, and up to four bytes for other characters.
- UTF32 - uses exactly 32 bits for each character.

Because all default encodings and ANSI encoding lose data (see Examples section), your workflow might need to use UTF8 instead.

Convert specified Base64 text to text using default encoding and store result in Variable: convertedText.

### Parameters:

**specified Base64 text** - Input text, expression, item field or variable.

**default** - One of five available encodings.

**Variable: convertedText** - Output string variable.

### Examples:

Let's encode German word 'Grüß' with different encodings with 'Convert Text to Base64' activity on the machine with Russian ANSI code page and decode the results back with the same encodings with *Convert Base64 to Text*.

- *default*: encoded string 'R3J1Pw==', decoded string 'Gru '

- *ASCII*: encoded string 'R3I/Pw==', decoded string 'Gr '
- *Unicode*: encoded string 'RwByAPwA3wA=', decoded string 'Grüß'
- *UTF8*: encoded string 'R3LDvMOF', decoded string 'Grüß'
- *UTF32*: encoded string 'RwAAAHIAAAD8AAAA3wAAAA==', decoded string 'Grüß'

As you can see, all three Unicode encodings save the German national characters unchanged. With 7-bit ASCII encoding, all national German characters were lost. In the case of default encoding, the input string was initially mapped to a Russian ANSI code page, where the German 'ü' was mapped to Latin 'u' and the last 'ß' was lost, because it has no suitable mapping to the Russian code page.

## 18.4 Convert HTML To Text

This workflow action is used to strip HTML tags from a specified string and to return the result to an output variable.

Convert this HTML code to plain text and store result in Variable: convertedText.

### Parameters:

**this** - Text, expression, item field or variable to convert.

**Variable: convertedText** - Output string variable.

## 18.5 Convert Number To Text

This workflow action is used to convert an integer number into text using a specified language. Supported languages: English, French, German, Russian.

Convert this number to text in English and store result in Variable: convertedText.

### Parameters:

**this** number - Input integer variable to convert.

**English** - Select the language to be used.

**Variable: convertedText** - Output string variable.

### Examples:

Convert Variable: totalPrice number to text in English and store result in Variable: totalPriceText.

then Convert Price: [%Variable: totalPriceText%] to title case and store in Variable: stringPrice

## 18.6 Convert Text Case

Converts specified text to UPPER CASE, lower case, Title Case or Sentence case.

**See Also:** *Convert Number To Text*

Convert this text to uppercase and store in Variable: convertedText.

### Parameters:

**this text** - Text, expression, item field or variable to convert.

**uppercase** - One of the following conversion types: *uppercase, lowercase, title case, sentence case*.

**Variable: convertedText** - Output string variable.

### Examples:

Convert Variable: totalPrice number to text in English and store result in Variable: totalPriceText.

then Convert Price: [%Variable: totalPriceText%] to title case and store in Variable: stringPrice

## 18.7 Convert Text to Base64

This workflow activity will encode a specified text using one of five available encodings and convert the encoded text to Base64.

Available encodings:

- default - 8-bit encoding of the current operating system's ANSI code page.
- ASCII - 7-bit ASCII encoding.
- Unicode - also known as UTF16, represents each character as a sequence of one or two 16-bit integers.
- UTF8 - uses one byte for any ASCII characters, which have the same code values in both UTF-8 and ASCII encoding, and up to four bytes for other characters.
- UTF32 - uses exactly 32 bits for each character.

Because all Default encodings and ANSI encoding lose data (see Examples section), your workflow might need to use UTF8 instead.

Convert specified text to Base64 using default encoding and store result in Variable: convertedText.

#### Parameters:

**specified text** - Text, expression, item field or variable to encode.

**default encoding** - Select one of five available encodings.

**Variable: convertedText** - Output string variable.

#### Examples:

Let's encode German word 'Grüß' with different encodings with the *Convert Text to Base64* activity on a machine with a Russian ANSI code page and decode the results back with the same encodings with *Convert Base64 to Text*.

- *default*: encoded string 'R3J1Pw==', decoded string 'Gru '
- *ASCII*: encoded string 'R3I/Pw==', decoded string 'Gr '
- *Unicode*: encoded string 'RwByAPwA3wA=', decoded string 'Grüß'
- *UTF8*: encoded string 'R3LDvMOF', decoded string 'Grüß'
- *UTF32*: encoded string 'RwAAAHIAAAD8AAAA3wAAAA==', decoded string 'Grüß'

As you can see, all three Unicode encodings save the German national characters unchanged. With 7-bit ASCII encoding, all national German characters were lost. In the case of default encoding, the input string was initially mapped to Russian ANSI code page, where the German 'ü' was mapped to a Latin 'u' and the last 'ß' was lost, because it has no suitable mapping to the Russian code page.

## 18.8 Decode Web Text

This workflow action decodes text with one of two available decoding methods and returns the result to an output variable.

Available decode methods:

1. *url decoder* - Converts a URL-encoded string into a decoded string. If characters such as blanks and punctuation are passed in an HTTP stream, they might be misinterpreted at the receiving end. URL encoding converts characters that are not allowed in a URL into character-entity equivalents; URL decoding reverses the encoding. For example, when embedded in a block of



text to be transmitted in a URL, the characters < and > are encoded as %3c and %3e; this method decodes it back.

2. *html decoder* - Converts a string that has been HTML-encoded for HTTP transmission into a decoded string. If characters such as blanks and punctuation are passed in an HTTP stream, they might be misinterpreted at the receiving end. HTML encoding converts characters that are not allowed in HTML into character-entity equivalents; HTML decoding reverses the encoding. For example, when embedded in a block of text, the characters < and > are encoded as &lt; and &gt; for HTTP transmission, this method decodes it back.

**See Also:** *Encode Web Text*

Decode specified text using URL decoder and store in Variable: decodedText.

**Parameters:**

**specified text** - Text, expression, item field or variable to decode.

**URL decoder** - One of two decoding methods: *url decoder* or *html decoder*.

**Variable: decodedText** - Output string variable.

## 18.9 Encode Web Text

Processes text with one of four available encoding methods and returns result to output variable.

Available encoding methods:

1. *URL encoder* - Can be used to encode the entire URL, including query-string values. If characters such as blanks and punctuation are passed in an HTTP stream, they might be misinterpreted at the receiving end. URL encoding converts characters that are not allowed in a URL into character-entity equivalents. For example, when the characters < and > are embedded in a block of text to be transmitted in a URL, they are encoded as %3c and %3e. Note, that this method converts each space character to a plus character (+).
2. *URL path encoder* - The same as above, but converts each space character into the string "%20", which represents a space in hexadecimal notation. Use this method when you encode the path portion of a URL in order to guarantee a consistent decoded URL, regardless of which platform or browser performs the decoding.
3. *html encoder* - Converts a string to an HTML-encoded string. If characters such as blanks and punctuation are passed in an HTTP stream, they might be misinterpreted at the receiving end. HTML encoding converts characters that are not allowed in HTML into character-entity

equivalents. For example, when embedded in a block of text, the characters < and > are encoded as &lt; and &gt; for HTTP transmission.

4. *html attribute encoder* - Minimally converts a string to an HTML-encoded string. It converts only quotation marks ("), ampersands (&), and left angle brackets (<) to equivalent character entities. It is considerably faster than the *html encoder*. The result string should be used only for double-quoted (") attributes, security issues might arise when using a result string with single-quoted (') attributes.

Encode specified text using URL encoder and store in Variable: encodedText.

**Parameters:**

**specified text** - Text, expression, item field or variable to encode.

**URL encoder** - One of four available encoding methods: *url encoder*, *url path encoder*, *html encoder*, and *html attribute encoder*.

**Variable: encodedText** - Output string variable.

**Examples:**

Encode [%Current Item:Approver Comments%] using URL encoder and store in Variable: param.

then Send HTTP request to http://mysite/ q=[%Variable: param%] with this login and this password.

Use GET method with this headers. Store response in Variable: pageBody and status code in Variable: code

## 18.10 Evaluate Mathematical Expression

This workflow action is used to evaluate the specified mathematical expression.

Evaluate this mathematical expression and store result in Variable: exprResult

**Parameters:**

**this** - specify the mathematical expression.

**Supported operators:** + - \* /

**Note:** there are more supported operators and functions, for more details, please refer to <https://ncalc.codeplex.com/>

**Variable: exprResult** - specify the variable to keep the result.

## 18.11 Extract Substring

This workflow action extracts a substring from a string specified by a starting index and substring length. A substring length of -1 indicates that the substring extends to the end of the string.

Extract substring from this text, starting at index 0 with -1 length and store in Variable: Substring

### Parameters:

**this text** - the string expression that a substring will be taken from

**starting at index 0** - specify the starting index. 0 stands for the beginning of the string expression.

**-1 length** - specify the length of a substring to be extracted. -1 indicates the end of the string expression.

**Variable: Substring** - a variable to store the result.

## 18.12 Get Random Number

This workflow action is used to get a random number within the specified range.

**Note:** The range of the returned values includes the lower limit value but does not include the higher limit value.

**See Also:** *Get Random String*

Get random number within the range from 0 to 100 and store in Variable: randomNumber.

### Parameters:

**0** - The lower limit of the range. May be negative.

**100** - The higher limit of the range. May be negative.

**Variable: randomNumber** - Output integer variable.

### 18.13 Get Random String

This workflow action is used to get a random string which contains characters from a specified set.

By default, the following set is used: abcdefghijklmnopqrstuvwxyz0123456789

**See Also:** *Get Random Number*

Get random string 16 characters long containing alphanumeric symbols and store in Variable:randomString.

#### Parameters:

16 - Output string length.

alphanumeric - The set of characters for output string.

Variable:randomString - Output string variable.

### 18.14 Get String Length

Returns the length of a string or string expression, in terms of the number of characters

Get length of this string and store result to Variable: stringLength

#### Parameters:

this - Input string variable.

stringLength - Output integer variable.

### 18.15 Join Text

This workflow action is used to join strings from an array to one string using a specified delimiter.

**See Also:** *Split Text* and *Capture All Occurrences of Text*

Joins string dictionary at index 0 next -1 items using specified delimiter and store in Variable: joinedText

#### Parameters:

string dictionary - Input dictionary.

index 0 - Zero-based index of the first dictionary item to join.

**next -1** - Zero-based index of the last dictionary item to join. -1 means 'to the end of the dictionary'.

**specified delimiter** - String placed between two dictionary items.

**Variable: joinedText** - Output string variable.

#### Examples:

*Set regular expression options: ignore case - Yes, is pattern culture invariant - Yes, use multiline mode - No, use single line mode - No, search right to left - No. Store options to Variable: REOptions*

*Capture \b[A-Z0-9. %-]+@[A-Z0-9.-]+\.[A-Z]{2,4}\b in xxx ipi@my.com owkk@ @ipi s@i12.com ddd and store all occurrences in Variable: allOccurrences, use options from Variable: REOptions*

*then Join Variable: allOccurrences at index 0 next -1 items using ; and store in Variable: emails*

*then Log Variable: emails to the workflow history list*

Result in workflow history list:

ipi@my.com; s@i12.com

## 18.16 Log to History List (HarePoint)

This action is similar to the *Log to History List* core action. But the core action has a 255 characters limitation and a workflow may be suspended if the string is longer. The HarePoint action divides long strings into sections of 255 characters or less and writes them to the Workflow History without risk of workflow suspension.

Log message to the workflow history list.

#### Parameters:

**message** - Message text for Workflow History.

## 18.17 Replace Text with Regular Expression

This workflow action is used to replace all matches of regular expression in the specified text with a specified string.

Regular expressions provide a powerful, flexible, and efficient method for processing text. To learn more about regular expressions, see: <http://msdn.microsoft.com/en-us/library/hs600312.aspx>

For a regular expression language reference, see: <http://msdn.microsoft.com/en-us/library/az24scfc.aspx>

**Note:** To set options for this activity, like case sensitivity, use *Set Regular Expression Options*.

Find regular expression in this text, replace with this text and store in Variable: ReplaceResult

**Parameters:**

**regular expression** - Regular expression pattern. For example, '\b(\w+ )\s1\b'.

**this text** - Input text, expression, item field or variable.

**replace with this text** - String to replace all matches.

**Variable: ReplaceResult** - Output string variable.

## 18.18 Set Regular Expression Options

This workflow action sets the options for regular expression actions like *Capture Text with Regular Expression*, *Capture All Occurrences of Text* or *Replace Text with Regular Expression*.

Once added to a workflow, this action will set regular expressions options for all the following related actions.

If necessary, you can add more *Set Regular Expressions Options* actions to set new options for subsequent actions.

Set regular expression options: ignore case - Yes, is pattern culture invariant - Yes, use multiline mode - No, use singleline mode - No, search right to left - No

**Parameters:**

**ignore case** - Ignore case in regular expression and in the input string.

**pattern is culture invariant** - Specifies that cultural differences in language are ignored. Culture is used to obtain culture-specific information, such as casing rules and the alphabetical order of individual characters.

**multiline mode** - Multiline mode. Changes the meaning of ^ and \$ so they match at the beginning and end, respectively, of any line, and not just the beginning and end of the entire string.

**singleline mode** - Specifies single-line mode. Changes the meaning of the dot (.) so it matches every character (instead of every character except \n).

**search right to left** - Specifies that the search will be from right to left instead of from left to right.

**Examples:**

*Set regular expression options: ignore case - No, is pattern culture invariant - Yes, use multiline mode - No, use singleline mode - No, search right to left - No.*

*then Capture .est in IT IS TEST and store in Variable: r1*

*then Set regular expression options: ignore case - Yes, is pattern culture invariant - Yes, use multiline mode - No, use singleline mode - No, search right to left - No.*

*then Capture .est in IT IS TEST and store in Variable: r2*

*then Log <[%Variable: r1%]>, <[%Variable: r2%]> to the workflow history list*

Result in workflow history list:

<>, <TEST>

## 18.19 Split Text

This workflow action is used to split text and store results to a strings dictionary. You can specify string delimiters (one per line) or character delimiters (each character in specified string is delimiter).

**See Also:** *Join Text*

Splits specified text using specified delimiters (string, one per line) and store in Variable: stringsDictionary

**Parameters:**

**specified text** - Text, expression, item field or variable.

**specified delimiters** - String or multi-line string with delimiters.

**string, one per line** - Type of delimiters. You can choose string delimiters (one per line) or character delimiters (each character in specified string is delimiter).

**Variable: stringsDictionary** - Output strings dictionary.

## 18.20 Trim Text

Removes white-space characters from the beginning and/or end of a specified text.

The advantage (if compared to the out-of-the-box SharePoint "Trim String" action) is that it's possible to remove either only leading or only trailing white-space characters.

Removes *leading and trailing* white-space characters from this text and store in Variable: trimmedText

**Parameters:**

**leading and trailing** - One of three options: *leading*, *trailing*, *leading and trailing*.

**this text** - Text, expression, item field or variable.

**Variable: trimmedText** - Output variable.

## 19. Taxonomy activities

### 19.1 Add Language To Term Store

This workflow action is used to add a language to a specified term store.

Add English language to Default term store.

**Parameters:**

**English language** - select the language to be added from the drop-down list.

**Default term store** - type the name of the term store where the new language needs to be added.

### 19.2 Create Taxonomy Group

This workflow action is used to create a new taxonomy group in a specified term store.

Create taxonomy group with this name in Default term store. Store guid to Variable: groupGuid.

**Parameters:**

**this name** - type the name for the new taxonomy group.

**Default term store** - type the name of the term store where the new group needs to be created.

**Variable: groupGuid** - select a variable where the GUID of the new group will be stored.



## 19.3 Create Taxonomy Term

This workflow action is used to create a new taxonomy term in a specified term store and a specified term set.

Create taxonomy term with this name in Default term store, this group and this term set. Use English language. Store guid to Variable: termGuid.

### Parameters:

**this name** - type the name for the new taxonomy term.

**Default term store** - type the term store name.

**this group** - type the name of the group.

**this term set** - type the name of the term set.

**English language** - select the language for the new taxonomy term. If necessary, more languages can be added to term store using *Add Language To Term Store*.

**Variable: termGuid** - select the variable to store the GUID of a new term.

## 19.4 Create Taxonomy Term Set

This workflow action creates a new taxonomy term set in a specified term store.

Create taxonomy term set with this name in Default term store and group with this name. Use English language. Store guid to Variable: termSetGuid.

### Parameters:

**this name** - type the name for the new taxonomy term.

**Default term store** - type the term store name.

**this group** - type the name of the group.

**English language** - select the language for the new taxonomy term. If necessary, more languages can be added to the term store using *Add Language To Term Store*.

**Variable: termSetGuid** - select a variable to store the GUID of a new term set.

## 19.5 Delete Taxonomy Group

This workflow action deletes a specified taxonomy group.

Delete taxonomy group with this name in Default term store.

### Parameters:

**this name** - type the name of the taxonomy group to be deleted.

**Default term store** - type the name of the term store where a new group needs to be deleted.

## 19.6 Delete Taxonomy Term

This workflow action is used to delete a specified taxonomy term from a specified term set, term group and term store, for a specified language.

Delete taxonomy term with this name in Default term store, this group and this term set. Use English language.

### Parameters:

**this name** - type the name of the term to be deleted.

**Default term store** - type the name of the term store.

**this group** - type the name of the group

**this term set** - type the name of a term set.

**English language** - select the language for the taxonomy term.

## 19.7 Delete Taxonomy Term Set

This workflow action is used to delete a specified term set from a specified taxonomy term store and group.

Delete taxonomy term set with this name in Default term store and group with this name. Use English language.

### Parameters:

**this name** - enter the name of the term set to be deleted.

**Default term store** - the name of the term set.

**this name** - the name of the group.

**English language** - select the language.

## 19.8 Update Taxonomy Group

This workflow action is used to update the name and the description of a taxonomy group.

Update taxonomy group with this name in Default term store. Set this as new name and this as description.

### Parameters:

**this name** - enter the name of the taxonomy group to be updated.

**Default term store** - the name of the term store.

**this as new name** - the new name for the taxonomy group.

**this as description** - the new description for the taxonomy group.

## 19.9 Update Taxonomy Term

This workflow action is used to update the parameters of a taxonomy term.

Update taxonomy term with this name in Default term store, this group and this term set. Set this as new term name, this as description, this as default label and Yes term is available for tagging. Use English language.

### Parameters:

**this name** - enter the name of a term to be updated.

**Default term store** - the name of the term store where the term is located.

**this group** - the name of the group where the term is located

**this term set** - enter the name of the term set where this term belongs.

**this as new term name** - enter the new name for this term.

**this as description** - enter the new description for this term.

**this as a default label** - the new default label for this term.

**Yes** - is the term available for tagging, *yes* or *no*.

## 19.10 Update Taxonomy Term Set

This workflow action is used to update the parameters of a taxonomy term set.

Update taxonomy term set with this name in Default term store and group with this name. Set this as new name, this as description, this as contact and Yes term set is open for term creation. Use English language.

### Parameters:

**this name** - enter the name of the term set to be updated.

**Default term store** - the name of the term store.

**group with this name** - the name of the group.

**this as new name** - the new name for this term set.

**this as description** - enter the new description for this term set.

**this as contact** - the new contact for this term set.

**Yes** - specify whether the term is open for new term creation, *yes* or *no*.

**English language** - select the language for the new taxonomy term.

## 20. Workflow activities

### 20.1 Get Workflow Status

This workflow action is used to get the status of a specific workflow.

**Note:** Only 2013 workflows are supported!

Get status of workflow with this id started on current site, store status in Variable: status and last status change time in Variable: lastModified

### Parameters:

**this id** - specify the Instance ID of the workflow whose status is to be determined. You can obtain it from the *Start Workflow* action.

**current site** - specify the full URL of the site where the workflow is located. Leave this field blank (unchanged) for the current site.

**Variable: status** - the variable to store the status of the workflow

**Variable: lastModified** - the variable to store the last status change time.

## 20.2 Set Authentication

This is a universal workflow action that is used to authenticate to various services:

- SharePoint site
- SharePoint admin center
- Active Directory
- SMTP server
- Google Cloud Print
- PowerShell
- FTP site
- Exchange
- HTTP
- SQL
- URL shortener service
- Yammer

Use this token or login and password to authenticate to this

### Parameters:

**this token or login and password** - specify either of the following:

- token

- encrypted token
- encrypted credentials (login and password)

These can be generated in the **Security** section of the Management Console (in SharePoint Online) or **Site Settings - Site Administration - HarePoint Workflow Extensions Security Settings** (SharePoint on premise).

**Note:** For better security, it is not possible to specify login and password in plain text.

**this** - select the service where authentication is performed.

Available values are mentioned above.

## 20.3 Start Workflow

This workflow action starts a SharePoint 2013 or SharePoint 2010 list or site workflow. Unlike the out-of-the-box actions which start workflows, this action lets you transfer workflow initiation parameters to the workflow by using a corresponding dictionary variable.

Start this SharePoint 2013 workflow on the this url with WorkflowParameters parameters and store instance ID in Variable: instanceID

### Parameters:

**this** - type the workflow name (e.g. as seen in SharePoint Designer).

**SharePoint 2013** - select whether the workflow is a SharePoint 2013 or SharePoint 2010 workflow.

**this url** - specify the full URL of the site, if it is a site workflow, or the full URL of a list item if it is a list workflow.

**WorkflowParameters** - (optional) specify the workflow initiation parameters. Use the out-of-the-box "Build Dictionary" action to create a variable (dictionary type) that contains the necessary workflows parameters.

**Variable: instanceID** - a variable to store the workflow instance ID. This instance ID is used in other actions from "Workflow" category.

## 20.4 Terminate Workflow

This workflow action is used to terminate SharePoint 2013 or cancel SharePoint 2010 workflows (Microsoft uses different terminology for the same process for 2013 and 2010 workflows).

Terminate SharePoint 2013 workflow with this instance ID from site current site.

### Parameters:

**Terminate SharePoint 2013** - select if the workflow you need to stop is 2010 or 2013 workflow type.

**with this instance ID** - specify the instance ID for the workflow to be terminated. It can be obtained from the *Start Workflow* action.

**current site** - specify the full URL of the site where the workflow is running. Leave this field blank (unchanged) for the current site.

## Appendix A. Technical support

In case if you have any technical questions, please contact our Support:

- Create a **ticket** in our **Help Desk** on <https://www.harepoint.com/Support>  
**OR**
- Send us an **e-mail** to [support@harepoint.com](mailto:support@harepoint.com)

If you have questions regarding the **licensing**, feel free to contact our Sales Department directly via [sales@harepoint.com](mailto:sales@harepoint.com)