



Merge Excel Spreadsheets Example

Readme

Version 1.0
06/02/2020

Table of Contents

1. Introduction.....	3
1.1 Overview.....	3
1.2 Use Cases	3
2. Requirements & Prerequisites.....	4
2.1 System Requirements.....	4
2.2 Prerequisites	4
3. Getting Started.....	5
3.1 Quick Start	5
3.1.1 Setup	5
3.1.2 Configuration	6
4. Support & FAQs	10
4.1 Support	10
4.2 FAQs.....	10
Appendix A: Record of Changes	11
Appendix B: References	12

1. Introduction

This document contains all essential information for the user to make full use of this A2019 ***Merge Excel Spreadsheets Example***. It includes a description of the functions and capabilities and step-by-step procedures for setup & configuration of the ***Merge Excel Spreadsheets Example***.

1.1 Overview

This bot is a simple starter example of using the Excel Advanced package in Automation Anywhere Enterprise A2019. It reads two spreadsheets and determines where they have common values in column A, such as a product number. When there is a match, it reads columns G, H, and I from the first spreadsheet (source) and copies them over to the same columns in the second (target). The bot loops through every row of both spreadsheets to ensure all updates are completed.

Use this bot as a starting point to get familiar with the Excel Advanced package, and change the number of times the bot loops to update whichever columns you choose.

1.2 Use cases

The key use cases include:

- Open and read one or more Excel Spreadsheets
- Open and write to an Excel Spreadsheet
- Insert new data into an existing spreadsheet
- Copy data from one Excel Spreadsheet to another and save the changes
- Repeat an action more than once (loop)

2. Requirements & Prerequisites

2.1 System Requirements

[Enterprise A2019 \(Cloud deployed\) and Community Edition device requirements.](#)

Review the machine hardware specifications, operating system versions, and browser types supported by Automation Anywhere Enterprise for creating and running bots and command packages as an Enterprise A2019 (Cloud deployed) or Community Edition user on your local machine.

2.2 Prerequisites

Microsoft Excel required on the bot runner device.

A2019 build 4088 or higher required.

Excel Advanced package version 2.0.0-20200422-000103 or higher.

3. Getting Started

3.1 Quick Start

3.1.1 Setup

Two Excel spreadsheets required as input, source and target. Data in source spreadsheet will be copied to target in columns G, H, and I.

Examples:

Spreadsheet 1 (source):

	A	B	C	D	E	F	G	H	I
1	Item number	Name	Count	Category	Unit price	Taxable	Backorder	Perishable	Locale
2	A0001	Milk	15	Grocery	3	N	Y	Y	US
3	A0005	Towel	4	Home	10	Y	N	N	US
4	A0006	Dog Food	16	Pet	22	N	N	N	US
5	A0007	Paint	43	Home	12	Y	N	N	IN
6									

Spreadsheet 2 (target):

	A	B	C	D	E	F	G	H	I
1	Item number	Name	Brand	Popular	Unit price	TTS			
2	A0001	Milk	Wholesome Foods	Y	3	1			
3	A0002	Eggs	Dairy Love	Y	4	1			
4	A0003	Flower	Spring Me Up!	N	10	3			
5	A0004	Table	Woods Are Us	N	50	10			
6	A0005	Towel	Claire Dane	N	10	6			
7	A0006	Dog Food	Purina	Y	22	3			
8	A0007	Paint	Color the World	N	12	2			

Example output in Spreadsheet 2 (columns G, H and I have been populated):

A	B	C	D	E	F	G	H	I
Item number	Name	Brand	Popular	Unit price	TTS	Backorder	Perishable	Locale
A0001	Milk	Wholesome Foods	Y	3	1	Y	Y	US
A0002	Eggs	Dairy Love	Y	4	1			
A0003	Flower	Spring Me Up!	N	10	3			
A0004	Table	Woods Are Us	N	50	10			
A0005	Towel	Claire Dane	N	10	6	N	N	US
A0006	Dog Food	Purity	Y	22	3	N	N	US
A0007	Paint	Color the World	N	12	2	N	N	IN

3.1.2 Configuration and Use

Import the A2019 Excel Advanced Package in the Control Room under Bots > Packages.

This chart shows Excel Advanced Package commands used by the bot:

Excel Advanced Package commands used by the bot			
Command	Function	Parameter	Value
Open	Opens an Excel Workbook for input/output	Session Name	Identifies an Excel file by Session
		File Name	Physical location of Excel files
		Open In	Access mode Read-Only or Read-Write Read-Write mode for Spreadsheet 2
Go to cell	Moves the cursor to a specific cell in a Microsoft Excel spreadsheet or a CSV file.	Session Name	Identifies an Excel file by Session
		Specific Cell	

			Excel spreadsheet Cell location, such as A1
Get single cell	Retrieves the values from a single cell in a Microsoft Excel spreadsheet or a CSV file and stores them in a string variable.	Session Name Cell Option Store cell contents to	Identifies an Excel file by Session Active cell or Specific Cell String variable to populate with cell contents
Find	Finds a string in a Microsoft Excel spreadsheet or a CSV file.	Session Name From Till Find Search options Assign cell addresses to variable	Identifies an Excel file by Session Start of search range End of search range Value to search for Loop through search by row or column List variable to populate with cell address where search criteria found
Set cell	Sets a value in the Active cell or Specific cell in a Microsoft Excel spreadsheet or a CSV file. You can also use this action to set a formula.	Session Name Cell Option Cell Value	Identifies an Excel file by Session Active cell or Specific Cell Variable value to copy to cell
Close	Closes the current workbook and provides an option to Save changes when closing the file.	Session Name Save changes when closing file	Identifies an Excel file by Session When selected, file changes write to disk

Input Variables				
<i>Variable Name</i>	<i>Type</i>	<i>Mandatory</i>	<i>Purpose</i>	<i>Example Input</i>
<i>vFoundCellNumber</i>	<i>String</i>	Yes	Cell location in target spreadsheet where key value matches source spreadsheet value	A39
<i>vFoundListSizeNumber</i>	<i>Number</i>	Yes	Count of how many cells match the key looked up in target spreadsheet	1
<i>vHeaderText</i>	<i>String</i>	Yes	Stores column header text from source spreadsheet to update target spreadsheet	Backorder
<i>vNewRowRow</i>	<i>Record</i>	Yes	Store current row from source spreadsheet in a single record while looping	Single row from a spreadsheet
<i>vOldRowRow</i>	<i>List of Strings</i>	Yes	List of cell addresses in target spreadsheet where key value found	A14
<i>vSourceCell</i>	<i>String</i>	Yes	Cell <i>value</i> (not address) to look up in target spreadsheet)	A00339
<i>vTargetCell</i>	<i>String</i>	Yes	1) Check this value against vSourceCell to confirm they match	A00339

Output Variables				
<i>Variable Name</i>	<i>Type</i>	<i>Mandatory</i>	<i>Purpose</i>	<i>Example Output</i>
<i>vHeaderText</i>	<i>String</i>	Yes	Set cell on target spreadsheet header	Backorder
<i>vTargetCell</i>	<i>String</i>	Yes	Update target spreadsheet in corresponding cell where key row found	A00339

Edit lines 5 and 9 of the bot to indicate the file locations of your source and target spreadsheets. See section 3, above, for detailed example. Then you can run the bot.

4. Support & FAQs

4.1 Support

Free bots are not officially supported. You can get access to Community Support through the following channels:

- You can get access to Community Support, connecting with other Automation Anywhere customers and developers on [APeople](#) – the [Bot Building Forum](#), the [Bot Store Support Forum](#), or the [Developers Everywhere Group](#).
- Automation Anywhere also provides a [Product Documentation portal](#) which can be accessed for more information about our products and guidance on [Enterprise A2019](#).

4.2 FAQs

For questions relating to Enterprise A2019: See the [Enterprise A2019 FAQs](#).

Appendix A: Record of Changes

No.	Version Number	Date of Change	Author	Notes
1	1.0	06/05/2020	Gina Baronian-Moore	Initial Version

Appendix B: References

No.	Topic	Reference Link
1	Overview of Enterprise A2019	Click here
2	Guidance: Building basic A2019 bots	Click here
3	Guidance: Building A2019 action packages	Click here
4	APeople Community Forum	Click here
5	Automation Anywhere University	Click here