Example of transferring data from CSV file to Excel worksheet

Updated 6/17/2020

Examples Enterprise Build A2019

In this example, you build a bot to update the product inventory in an Excel worksheet with new product names from a CSV file. Use actions from the CSV/TXT, Excel advanced, IF/ELSE, and Loop packages.

Prerequisites

Before you start building the bot, create the following data sets on your desktop in the specified file formats:

Data set 1: ProductInventory.xlsx

Item number	Name	Count	Category	Unit price	Taxable
A0001	Milk	15	Grocery	3	N
A0002	Eggs	6	Grocery	4	N
A0003	Flower	3	Garden	10	Y
A0004	Table	1	Home	50	Y
A0005	Towel	4	Home	10	Y
A0006	Dog Food	16	Pet	22	N
A0007	Paint	43	Home	12	Y

Data set 2: NewProductNames.csv

Item number	Name
A0005	Hand Towel
A0002	Chicken Eggs

Item number	Name
A0003	Sunflower
A0004	Coffee Table
A0006	Dog Food - Small Dogs
A0007	Paint - Dark Blue
A0001	2% Milk

Procedure

1. Open a new bot:
   a. From the Automation Anywhere Enterprise web interface, select **Bots** > (and then)**My bots**.
   b. Click **Create TaskBot**.
   c. Enter a bot name.
   d. Accept the default folder location **\Bots**.
   To change where your bot is stored, click **Choose** and follow the prompts.
   e. Click **Create and Edit**.
2. Open the `NewProductNames.csv` file that you just created.
   a. Double-click or drag the **CSV/TXT** > (and then)**Open** action.
   b. In the **Session name** field, enter `session 1` .
   c. Provide the file path to `NewProductNames.csv`.
   d. Select the **Contains header** option.
   e. Click **Apply**.
3. Open the `ProductInventory.xlsx` file that you just created.
   a. Double-click or drag the **Excel advanced** > (and then)**Open** action.
   b. In the **Session name** field, enter `session 1` .
   c. Provide the file path to `ProductInventory.xlsx`.
   d. Choose to open the file in **Read-write** mode.

   e. Select the **Contains header** option.
   f. Click **Apply**.

4. Use the **Go to cell** action to indicate the first cell in which to update the product names.
   a. Double-click or drag the **Excel advanced** > (and then)**Go to cell** action.
   b. In the **Session name** field, enter `session 1` .
   c. Select the **Specific cell** option and enter `B2` .
   d. Click **Apply**.
5. Use a **Loop** action to retrieve the cell values in each row from `ProductInventory.xlsx`.
   a. Double-click or drag the **Loop** action.
   b. Select the **Excel Advanced** > (and then)**For each row in worksheet** iterator.
   c. In the **Session name** field, enter `session 1` .
   d. In the **Loop through** field, select **All rows**.
   e. In the **Assign current value to this variable** field, create a *rInventory* variable.
   f. Click **Apply**.
6. Use a **Loop** action to retrieve the cell values in each row from `NewProductNames.csv`.
   a. Drag the **Loop** action into the **For each row in worksheet Loop** container.
   b. Select the **For each row in CSV/TXT** iterator.
   c. In the **Session name** field, enter `session 1` .
   d. In the **Assign current value to this variable** field, create a *rNewProduct* variable.
   e. Click **Apply**.
7. Use an **If** action to compare the item number from `ProductInventory.xlsx` to the item number from `NewProductNames.csv` to ensure they are the same before moving on to the next action.
   a. Double-click or drag the **If** action into the **For each row in csv/txt Loop** container.
   b. Select the **String** condition.
   c. In the **Source value** field, input *rInventory[0]*.
   d. Select the **Equals to (=)** operator.
   e. In the **Target value** field, input *rNewProduct[0]*.
   f. Click **Apply**.
8. Use the **Set cell** and **Go to cell** actions to update the product name and move to the cell below.

  a. Double-click or drag the **Excel Advanced** > (and then)**Set cell** action.

  b. In the **Session name** field, enter `session 1` .

  c. Select the **Active cell** option.

  d. In the **Cell value** field, input *rNewProduct{Name}*

  e. Click **Apply**.

  f. Double-click or drag the **Excel Advanced** > (and then)**Go to cell** action.

  g. In the **Session name** field, enter `session 1` .

  h. From the **Active cell** drop-down list, select **One cell below**.

  i. Click **Apply**.

9. Insert an alternative to the **If** action: if the item numbers are not the same, the bot continues to the next row in `NewProductNames.csv`.

  a. Drag the **If** > (and then)**Else** action.

  b. Drag the **Loop** > (and then)**Continue** action.

10. Close the files.

  a. Double-click or drag the **Excel advanced** > (and then)**Close** action.

  b. In the **Session name** field, enter `session 1` .

  c. Select the **Save changes** option.

  d. Click **Apply**.

  e. Double-click or drag the **CSV/TXT** > (and then)**Close** action.

  f. In the **Session name** field, enter `session 1` .

  g. Click **Apply**.

11. Click **Save**.

12. Run the bot.

The bot updates the `ProductInventory.xlsx` file to look like the following table:

Data set 3: ProductInventory.xlsx

Item number	Name	Count	Category	Unit price	Taxable
A0001	2% Milk	15	Grocery	3	N
A0002	Chicken Eggs	6	Grocery	4	N

Item number	Name	Count	Category	Unit price	Taxable
A0003	Sunflower	3	Garden	10	Y
A0004	Coffee Table	1	Home	50	Y
A0005	Hand Towel	4	Home	10	Y
A0006	Dog Food - Small Dogs	16	Pet	22	N
A0007	Paint - Dark Blue	43	Home	12	Y

Send Feedback

Automation Anywhere.com
Contact Us
Privacy
Terms
Trademark
Products
Customers
Careers