Build your first IBM RPA bot

Hands-on Lab

Version 1.3 (for on-prem deployment)

Public

Authors:

Jukka Juselius - jukka.juselius@fi.ibm.com

Paul Pacholski – pacholsk@ca.ibm.com

Changes for BP Virtual Workshop by Jukka Juselius

© Copyright International Business Machines Corporation 2020

© Copyright International Business Machines Corporation 2020

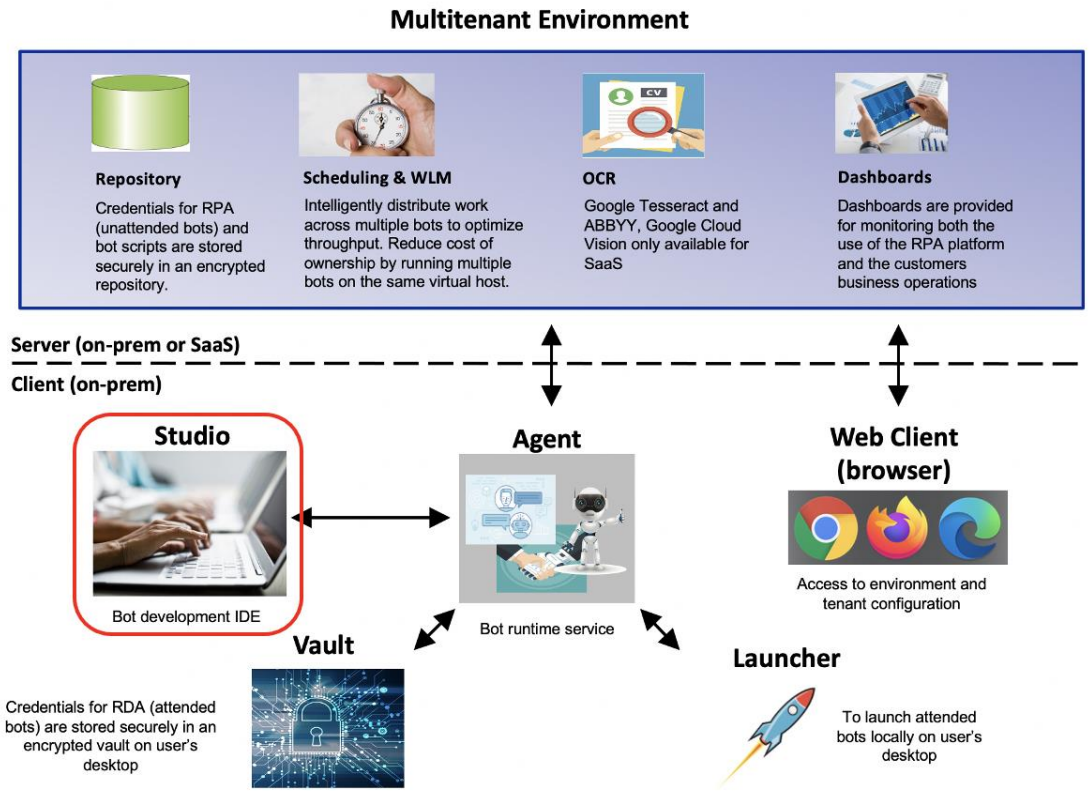
Table of Contents

1. Introduction 4
   - Use case 4
   - Prerequisites 4
2. Lab Instructions 5
   - Start IBM RPA Studio 5
   - Create a new automation 5
   - Create your bot script - Login to JK Automation website 6
   - Test your automation and continue implementation 13
   - Read CSV 14
   - Store row data in variables 16
   - Set Sales Leads input fields 19
   - Define Subroutines 23
   - Add loop to iterate through the data table 25
   - Logout from JK automation website 28
3. Work with Completed Automation 31
   - Open completed lab 31
   - Check input csv file location 31
   - Run the automation 31

© Copyright International Business Machines Corporation 2020

1. Introduction

This hands-on lab you will learn how to use IBM RPA Studio to create a simple RPA automation.



Use case

In this exercise you will build a robot to automate processing of sales leads that arrive in a CSV/Excel format. Each row of the file represents a separate sales lead. The sales leads need to be manually entered (copy/paste) by an analyst into the online opportunity system of record (JK Automation Sales Leads). This task is error prone and the analysts repeatedly ask if this can be automated, but it has never been a company priority. Until now 😉

Prerequisites

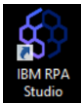
Nothing. Your environment is already configured, and everything is set. Ready to start?

© Copyright International Business Machines Corporation 2020

2. Lab Instructions

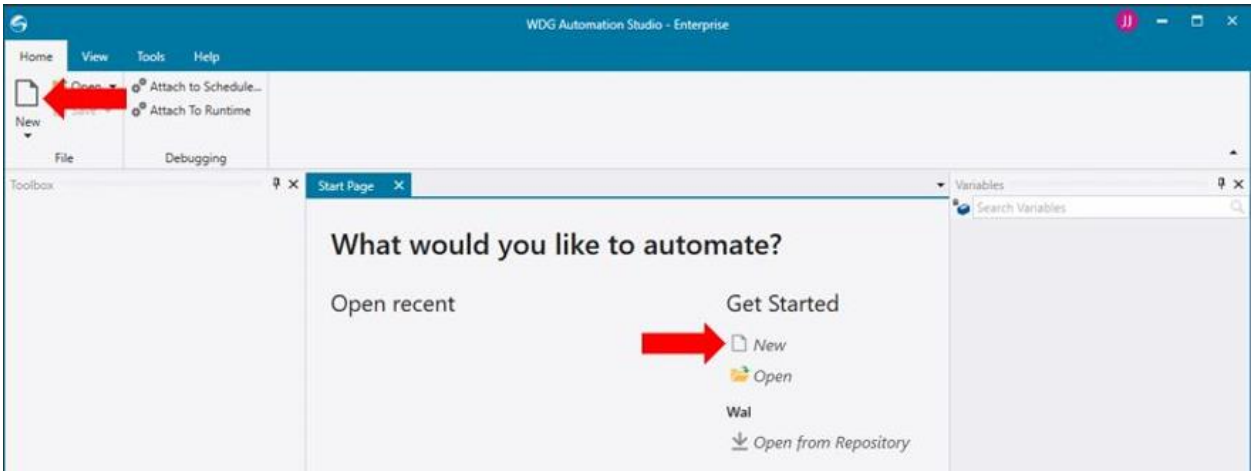
Start IBM RPA Studio

_1. Double-click the `IBM RPA Studio shortcut` in your desktop.



Create a new automation

_1. Create a new WAL file by clicking the `New` icon from the top toolbar or `New` under "Get Started" section.

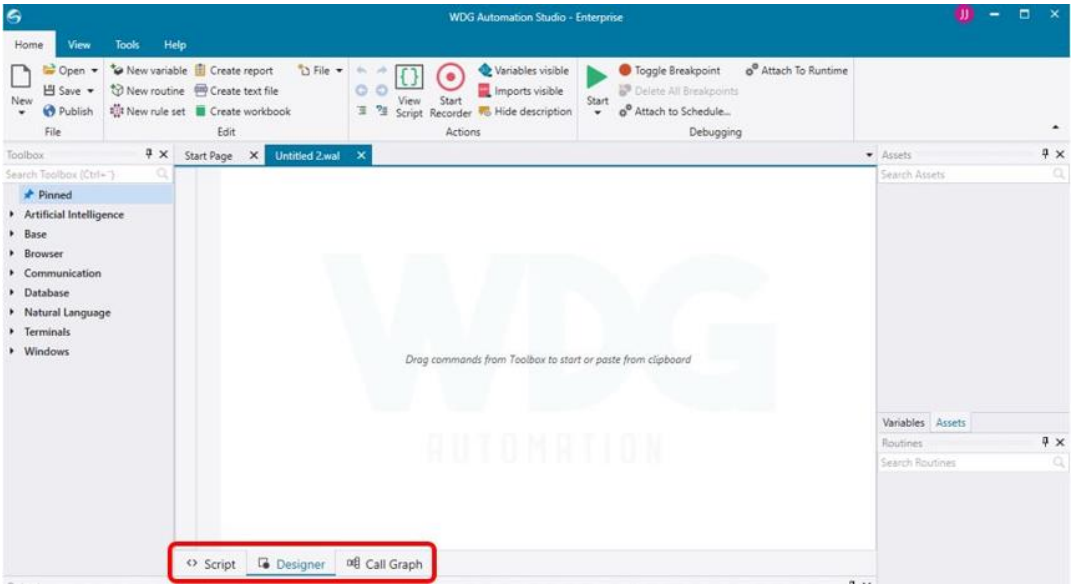


_2. Select `WAL file` and click `Open`.

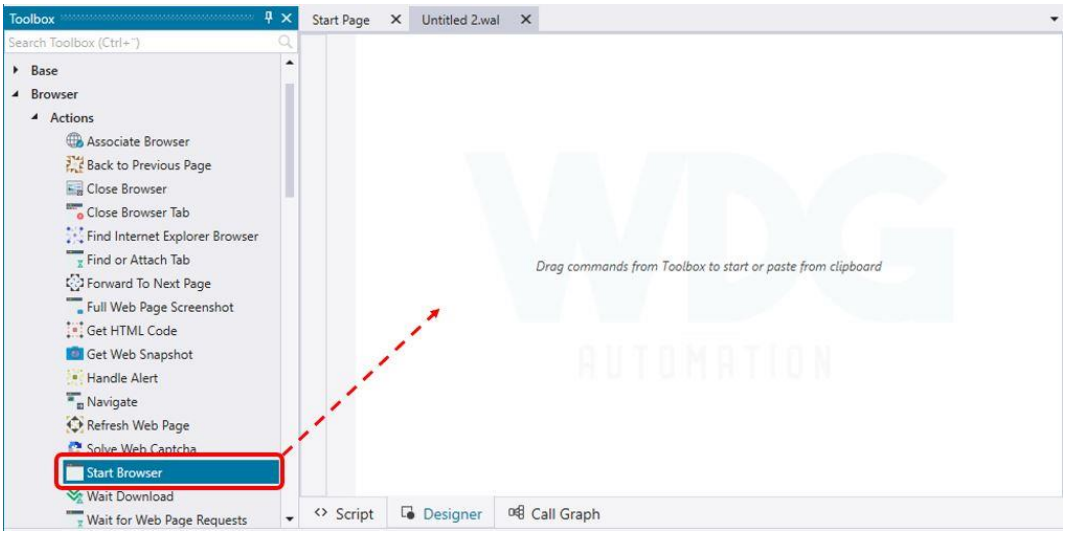
© Copyright International Business Machines Corporation 2020

Create your bot script - Login to JK Automation website

_1. You should now have your Studio opened with an empty WAL file in your Designer view. Note that you can change your view between *Script*, *Designer* and *Call Graph*.

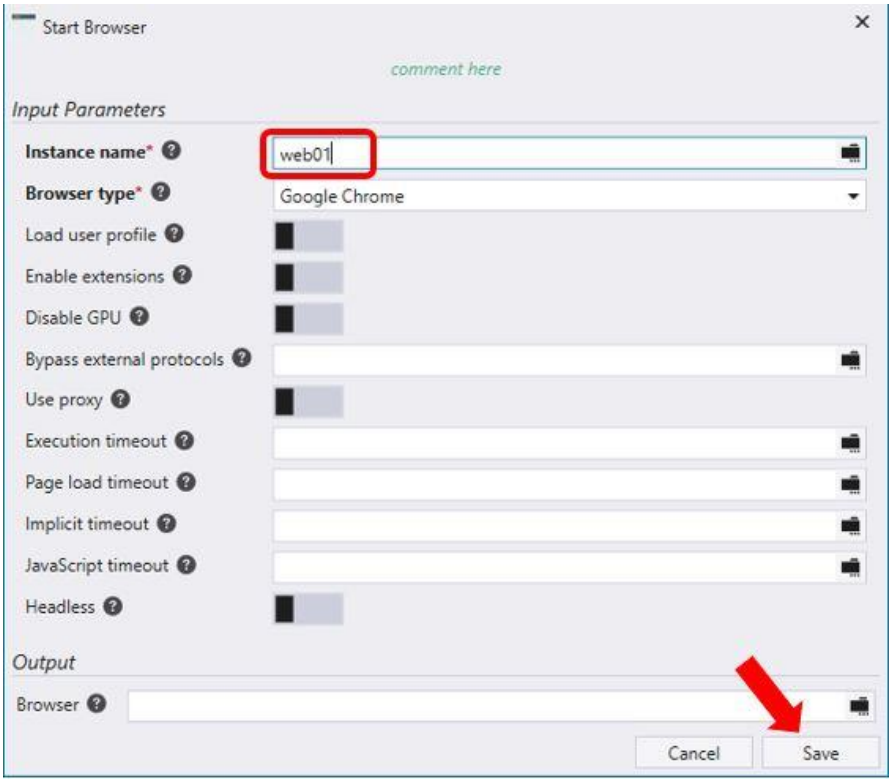


_2. Add `Start Browser` command from your Toolbox to your Designer view. You can find the command under ***Browser*** > ***Actions***. Drag and drop it to your canvas.



© Copyright International Business Machines Corporation 2020

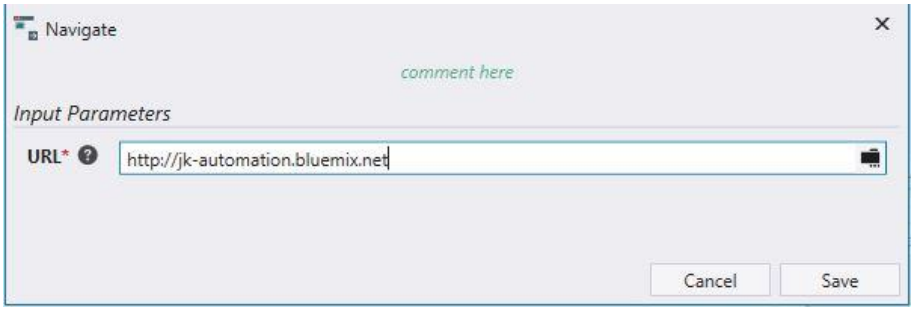
_3. The command configuration opens. You need to set name to your browser instance (for example `web01`). Keep `Google Chrome` selected as your browser type, if you have Chrome installed. You can also change the browser to one you want to use. Click `Save` button.



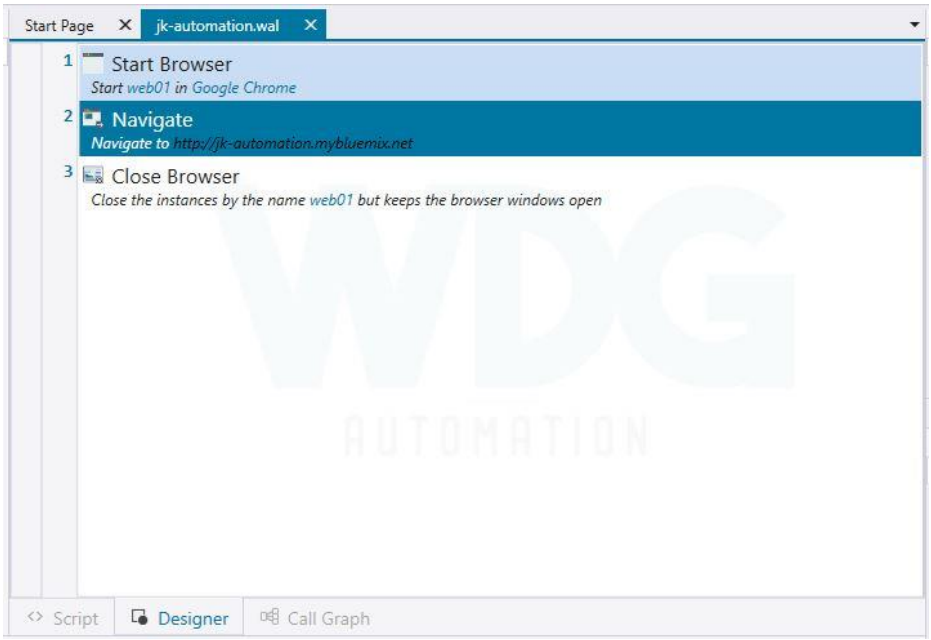
_4. `Close Browser` configuration window pops up automatically. Make sure to set the same instance name value that you used for `Start Browser`. Also set `Keep browser open` as enabled. This helps you during the implementation so that browser is not closed when you test your automation and you can continue using it for defining next actions for your automation. Click `Save` button to close the configuration window.

© Copyright International Business Machines Corporation 2020

_5. Next command we want is `Navigate`. You can find it also under *Browser* > *Actions*. Drag and drop it below the `Start Browser` command to your Designer view. Set the URL to **http://jk-automation.mybluemix.net** and click the `Save` button.



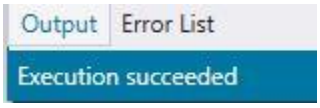
_6. **Save** your work by hitting `Ctrl+S` or the `Save icon` from the top toolbar. Navigate to *Desktop → IBM RPA Lab Resources → Lab1 – Build your first IBM RPA bot* and save your automation WAL file using name **jk-automation**. Your automation should now look as follows.



_7. **Run** your automation to test that it works and JK Automation website is opened. Click the green `Run` icon form the top toolbar. Alternatively, you can hit `F5`.

© Copyright International Business Machines Corporation 2020

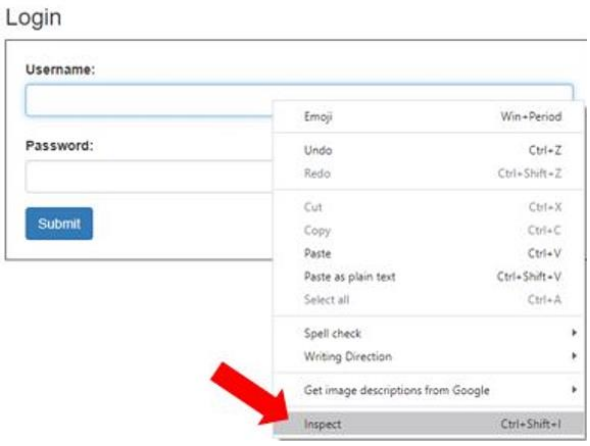
_8. You should see new Chrome Browser window opened and JK Automation login page opened. **Note!** For the first time this might take a couple of seconds. When you go back to your Studio window, you should see `Execution succeeded` text at the bottom left-hand side corner. ***Make sure to leave the browser window open!***



_9. Nice! Your first run with IBM RPA 👍Let's keep on going! Next we will need to automate the login to JK Automation website.

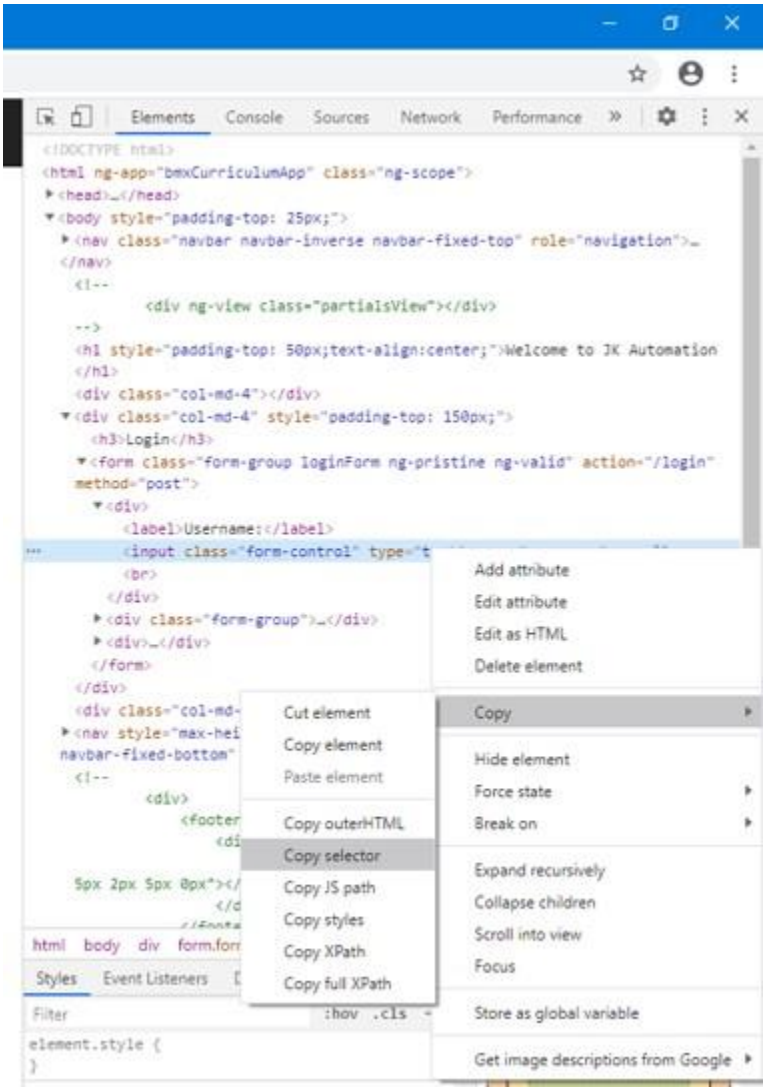
IBM RPA has a web recorder plugin for Chrome, but for this lab we're doing things manually to learn how we can select web elements just using basic tools. IBM RPA supports several selectors for fields: id, name, css, xpath, id + name.

_10. Open JK Automation browser window that you should have open. **Right-click** the `Username` field and from the opened menu select `Inspect`.



© Copyright International Business Machines Corporation 2020

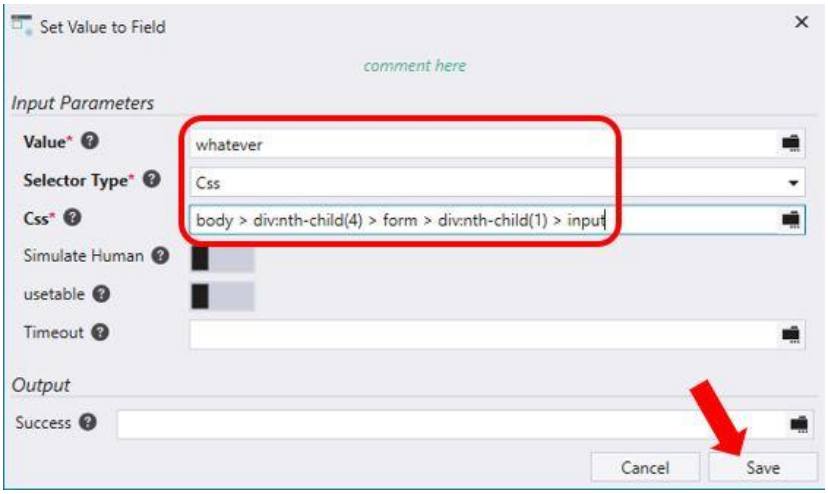
_11. This open your browsers element inspector with the Username field element selected. **Right-click** the selected element, select `Copy` > `Copy selector`. This will copy the CSS Selector for the element to your clipboard.



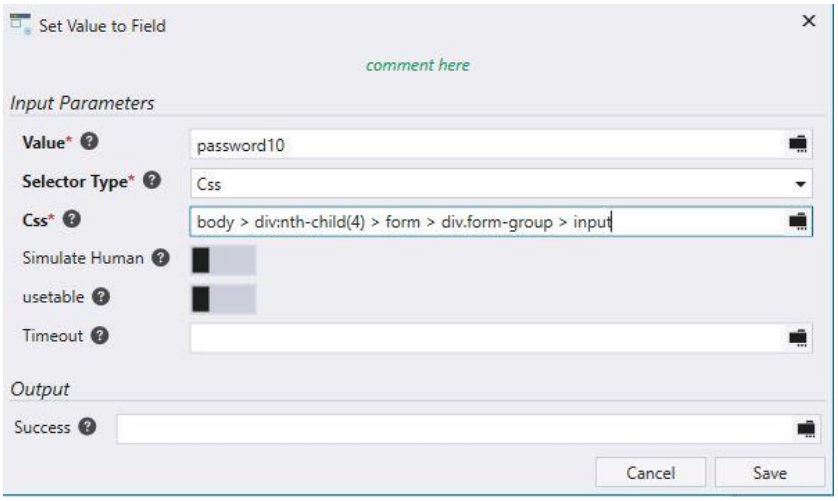
© Copyright International Business Machines Corporation 2020

_12. Go back to your IBM RPA Studio and add `Set Value to Field` (Browser > Fields > Set Value to Field) command to your automation just under `Navigation` command.

_13. When the command configuration window opens set *Value* to "**whatever**" (username you use does not matter), *Selector Type* to **Css** and *Css* to **the value from your clipboard** (the selector you copied from element inspector). Finally click `Save`.



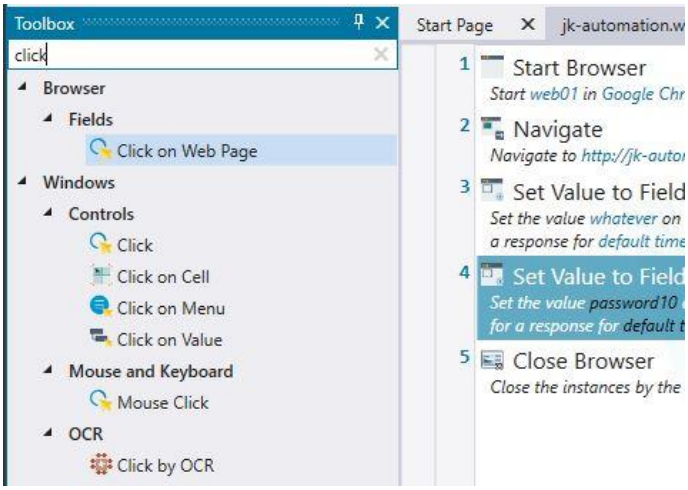
_14. Similarly add new `Set Value to Field` command below the previous for the `Password` field (you need to copy the selector for it as well). Set *Value* to **password10**.



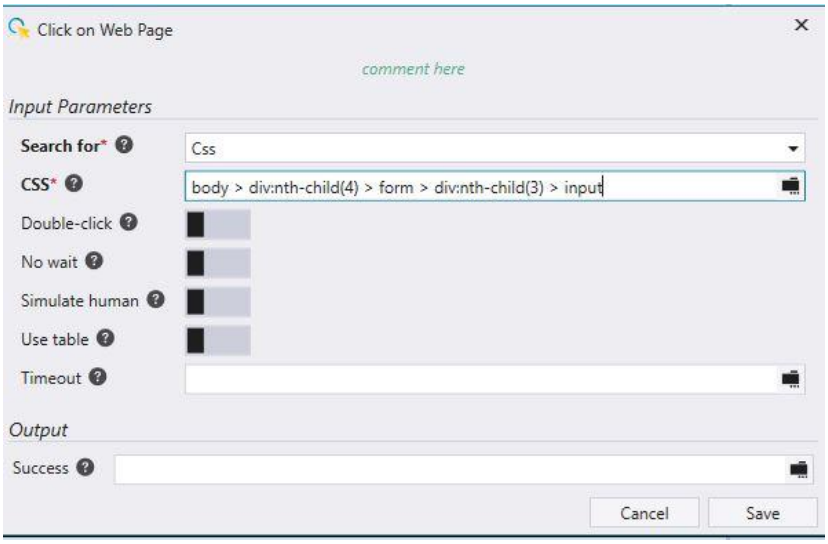
Now, let's finish the login sequence by adding a command to click the `Submit` button in the JK Automation login page.

© Copyright International Business Machines Corporation 2020

_15. We'll do that by adding command `Click on Web Page` after the Set Value to Field commands. ***Note*** that you can use toolbox search field to find commands. To display all the commands related to clicking, type **click** to the search field.



_16. Get the selector for the Submit button from your browsers element inspector as you earlier did for the Username and Password fields.



© Copyright International Business Machines Corporation 2020

Your automation should look like as follows. Nice, good job!

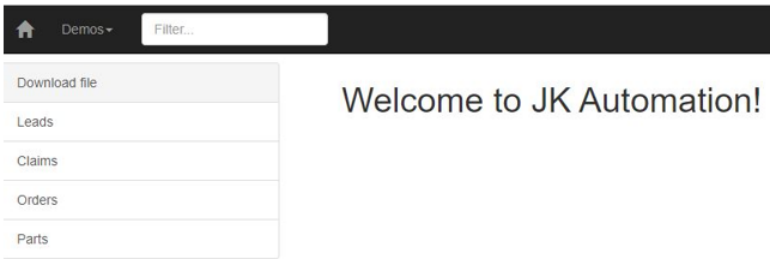
Test your automation and continue implementation

_1. Save your work.

_2. Close your browser with JK Automation login page

_3. **Run** your automation to test that it works and JK Automation website is opened. Click the green `Run` icon form the top toolbar. Alternatively, you can hit `F5`.

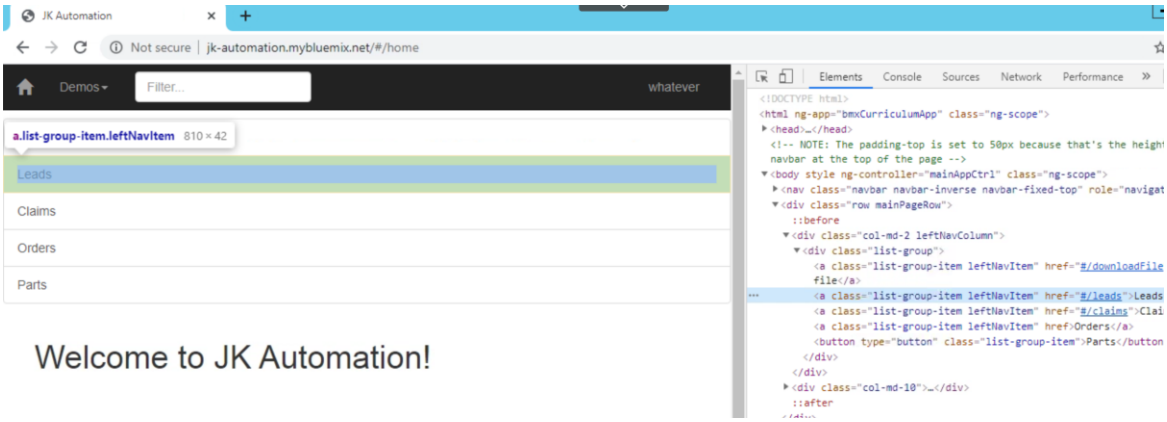
You should see your bot executing, opening the JK Automation web page and logging in with the information that you used to develop the login sequence. When your bot is finished, you should see the JK Automation welcome page.

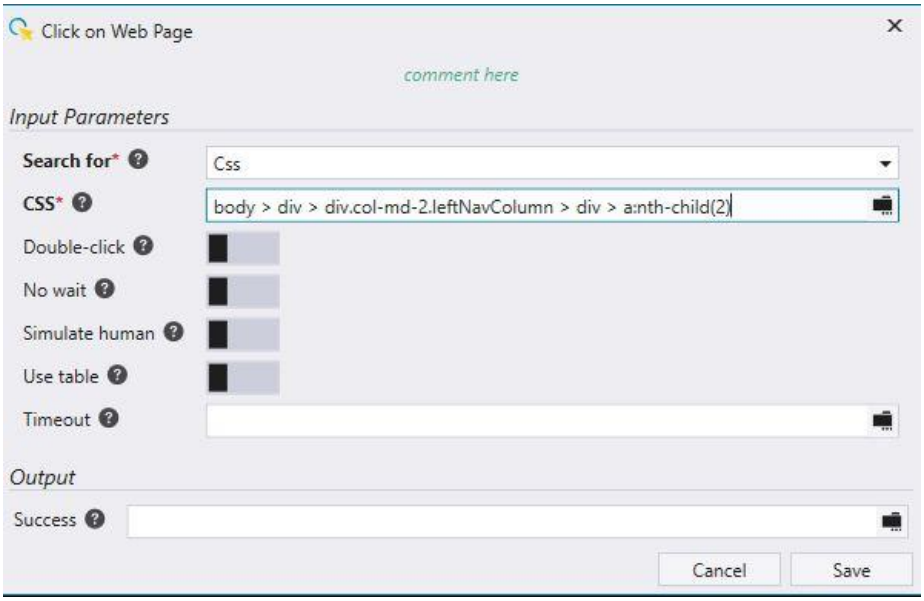


© Copyright International Business Machines Corporation 2020

Read CSV

_1. Since we want to handle *Leads*, add a `Click on Web Page` command to your automation to click the `Leads` link in the left-hand side menu. Yet again, use your browsers element inspector to get the needed selector for the Leads link.

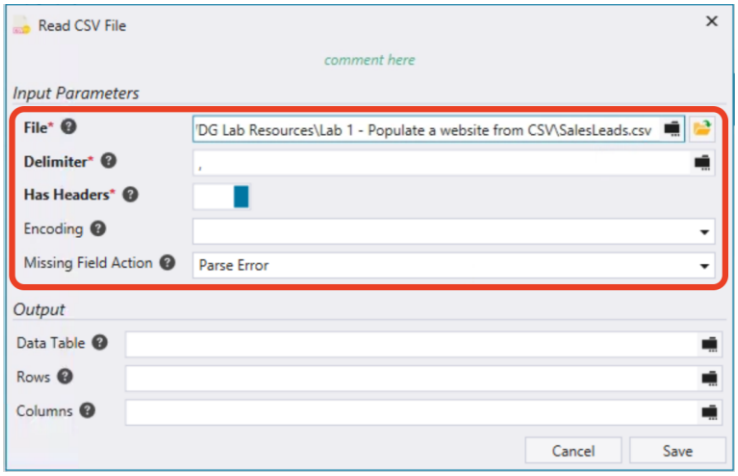




© Copyright International Business Machines Corporation 2020

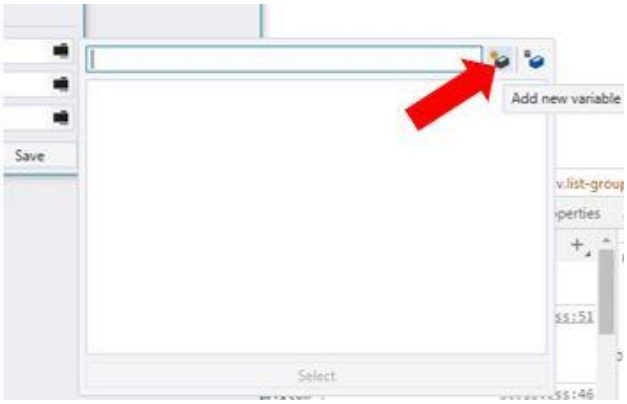
_2. Use toolbox search to find `Read CSV File` command and add it after your last Click on Web Page command.

_3. When the command configuration window opens, use the folder browse icon to select the file **SalesLeads.csv** from folder *Desktop → IBM RPA Lab Resources → Lab 1 – Build your first IBM RPA bot.* Leave all the other selections as they are in default.



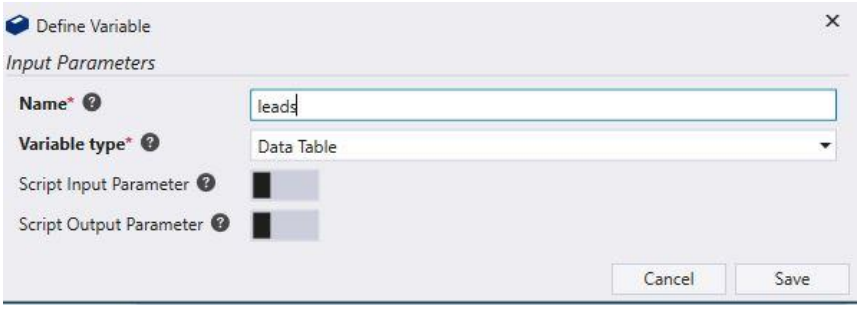
There are three outputs for the command: *Data Table* (holds the data), *Rows* (number of rows in the data table) and *Columns* (number of columns in the data table). We want to store these to variables.

_4. First, **Click** the folder icon besides the Data Table field to open variable list (as shown in picture above). Next, click `Add new variable` icon and the `Define Variable` window opens.

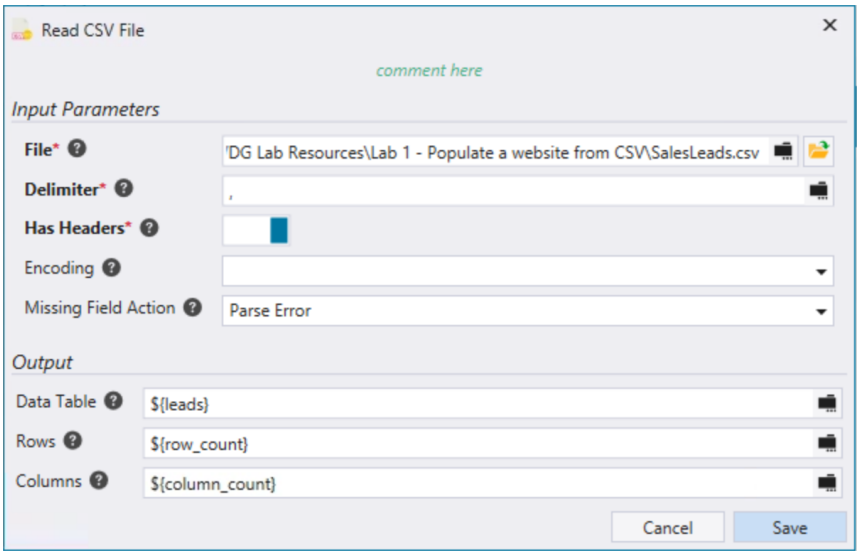


© Copyright International Business Machines Corporation 2020

_5. Name the variable as **leads**. Notice that the variable type is automatically set to match the output type (Data Table). Click `Save`.



_6. Repeat the previous steps to create variables also for other outputs *Rows* and *Columns*. Name them **row_count** and **column_count**, respectively. Your `Read CSV File` command configuration should now look as follows. Click `Save`.



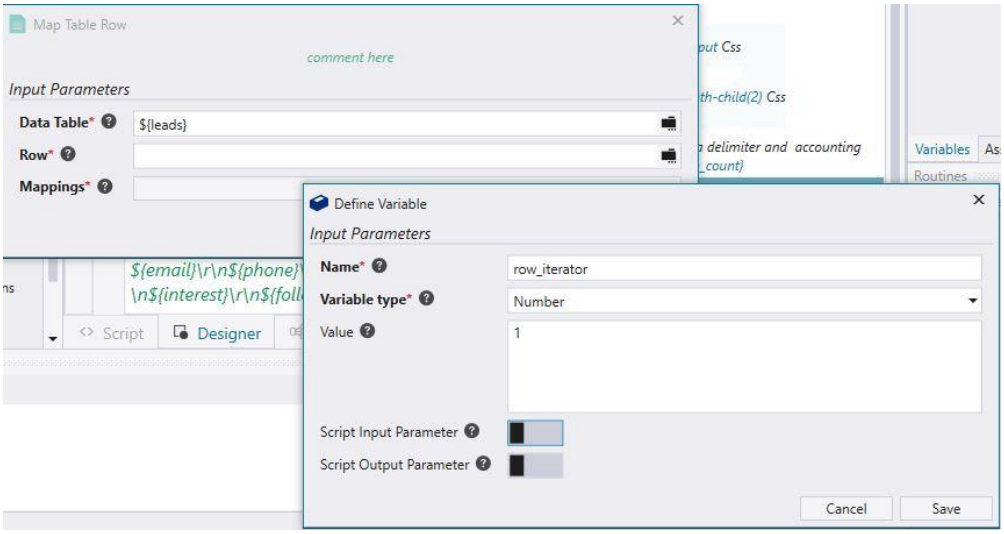
Store row data in variables

Now, we obviously want to iterate through all the rows within the data table and insert the data to JK Automation Sales Leads page. Let's focus first on getting a lead read from the data table.

_1. Use your toolbox search to find `Map Table Row` command and then drag and drop it under the `Read CSV File` command in your Designer view.

© Copyright International Business Machines Corporation 2020

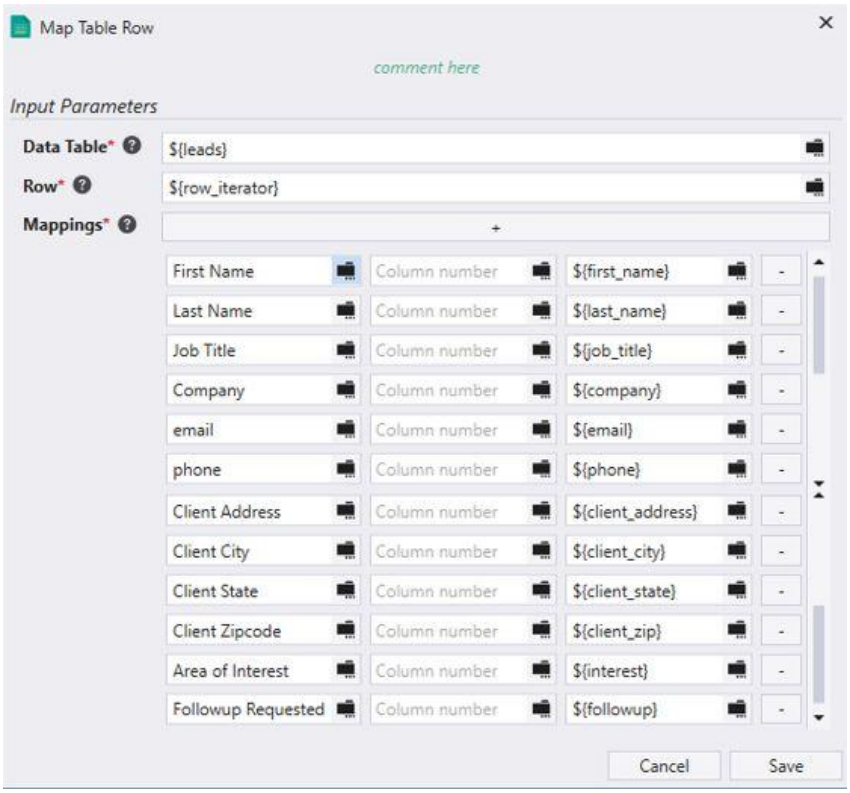
_2. When the configuration window opens, select variable **leads** to `Data Table` and then create a new variable for Row called row_iterator (since we want a variable to help us iterate through the data table) and set the default value for it to **1**.



Next we need to create mappings for the data table row. If you open the CSV file that we're using for the lab with Excel / Notepad, you can see that there is a header row (with column names) followed by 12 data rows and also 12 different columns: **First Name**, **Last Name**, **Job Tile**, **Company**, **email**, **phone**, **Client Address**, **Client City**, **Client State**, **Client Zipcode**, **Area of Interest** and **Followup Requested**.

© Copyright International Business Machines Corporation 2020

_3. Click `plus sign (+)` within your `Map Table Row` configuration window to add all the needed mappings for your row. On the left side, you need to type in the `Column Name` from the CSV file (watch out for typos) to identify the data for the mapping, on the right side, you need to first create a new variable to hold the data within the bot. **Map and name the variables as follows**.



TIP:

*To save time, using the **script view** in your Studio you can cut and paste the following line:*

***mapTableRow** --dataTable ${leads} --row ${row_iterator} --mappings "name=First Name=${first_name},name=Last Name=${last_name},name=Job Title=${job_title},name=Company=${company},name=email=${email},name=phone=${phone},name=Client Address=${client_address},name=Client City=${client_city},name=Client Zipcode=${client_zipcode},name=Area of Interest=${interest},name=Followup Requested=${followup},name=Client State=${client_state}"*

If do this, you also need to copy the new variable definitions to the start of your script:

defVar --name first_name --type String
defVar --name last_name --type String
defVar --name job_title --type String
defVar --name company --type String
defVar --name email --type String
defVar --name phone --type String

© Copyright International Business Machines Corporation 2020

defVar --name client_address --type String
defVar --name client_city --type String
defVar --name client_zipcode --type String
defVar --name interest --type String
defVar --name followup --type String
defVar --name client_state --type String

You can find both of these from the lab folder as text files that you use for copying and pasting.

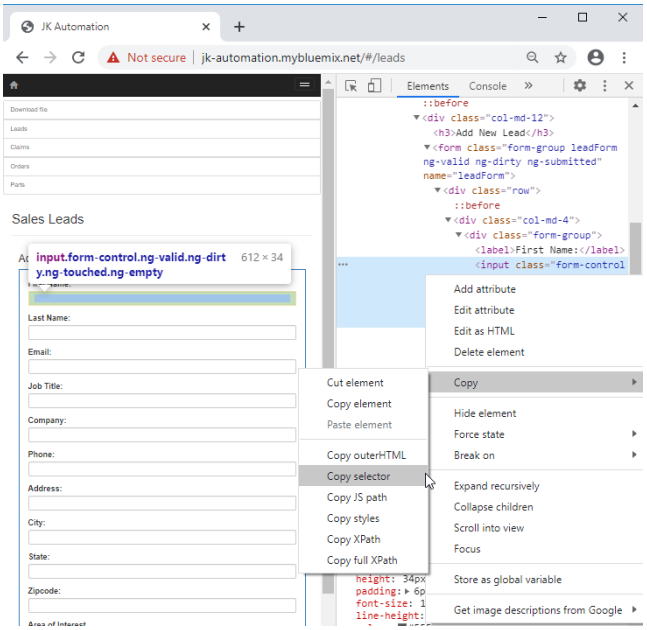
_4. Click `Save` to save and close the configuration. Now we need to insert the data to JK Automation Sales Leads page that you should still have open in your browser. Click **Leads** from the left-hand side navigation menu, if not already on Sales Leads page.

If you have closed JK Automation web page, you can always run your current automation to get the JK Automation Sales Leads page opened and continue from there. For each run, a new browser will be opened. While Studio is able to deal with these multiple open browser windows without issues, it helps to close all but one from time to time to avoid confusing the developer.

Set Sales Leads input fields

You will use a series of `Set Value to Field` commands to insert data that we just extracted to JK Automation Sales Leads page and to its matching fields. Let's go through the first mapping together.

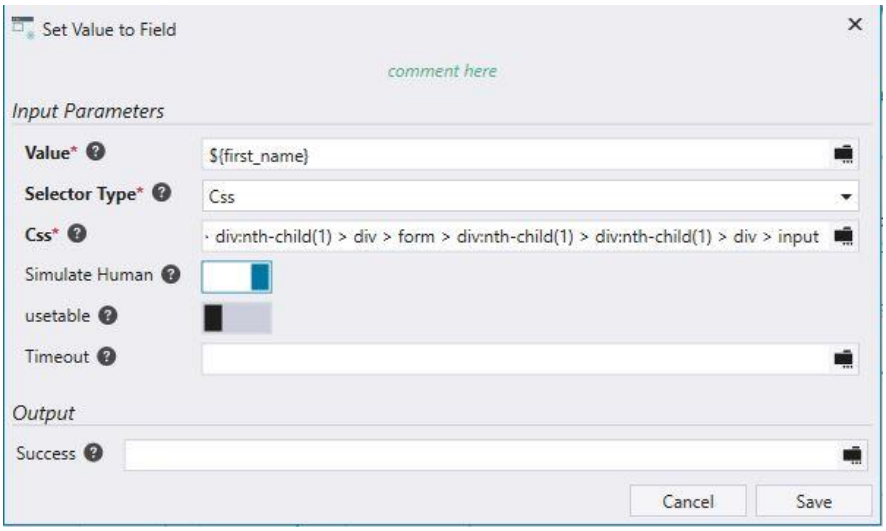
_1. Right-click the `First Name` input field in the web page, select `Inspect` > `Copy` > `Copy Selector`.



© Copyright International Business Machines Corporation 2020

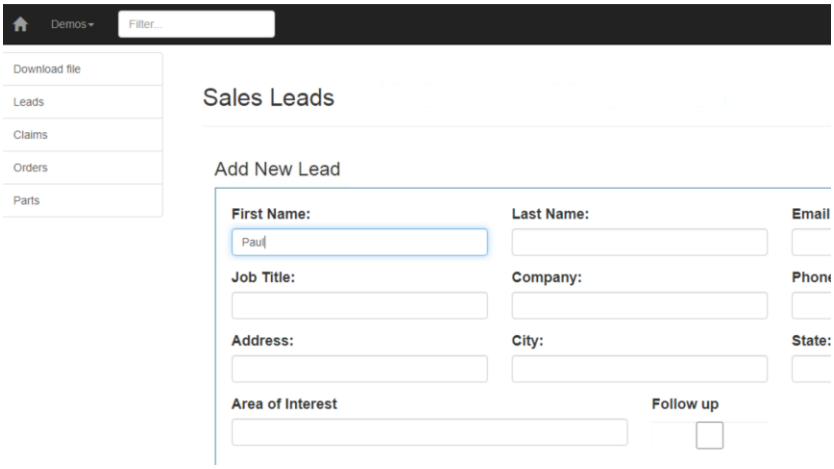
_2. Add `Set Value to Field` command to your automation under `Map Table Row` command.

_3. For `Selector Type` select **Css,** for `Css` copy the CSS Selector from (as shown in Step 1), for `Value` use the extracted variable **first_name**. Enable `Simulate Human` option. Click `Save`.



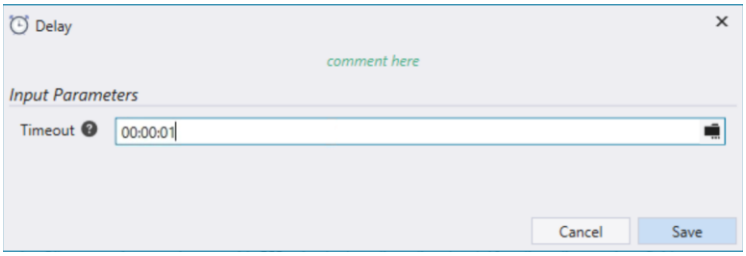
© Copyright International Business Machines Corporation 2020

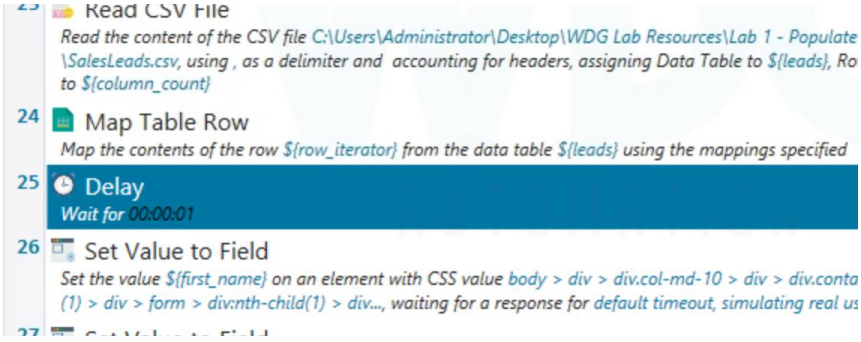
_4. Close the browser with JK Automation site and run your automation to test it. When the script finishes, you should see the Sales Lead page with an entry to First Name input field from your first sales lead.



NOTE! If you do not see First Name field filled in, the automation seems to be too "quick" for the web site, this is quite normal. For demo, we can fix this by adding a small delay before starting to fill in the input fields. Normally you would need to wait for certain elements to appear and be ready. We'll do this in easy way now 😉

Drag and drop **Delay** command *after the Map Table Row* command and *before the first Set Value to Field* command. Set time out to **00:00:01** for one second.





Close the browser with JK Automation and **run** your script again to verify that also the first name field gets populated.

© Copyright International Business Machines Corporation 2020

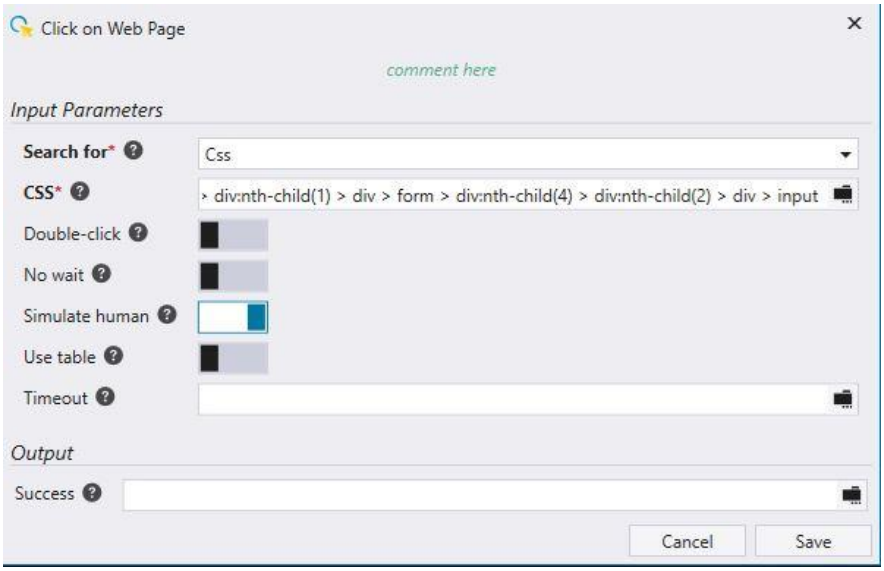
_5. Now, set all the remaining 11 input fields (Area of Interest being the last one) as you did with the First Name field. Remember to enable Simulate Human option.

Note, you can use copy-paste in your Designer view to easily copy commands.

_6. Add `If` command (Under *Base* > *Flow Control*) to your automation under the last `Set Value to Field` command. Configure it using the **followup** variable as follows.

_7. Add `Click on Web Page` command between *If* and *End If*.

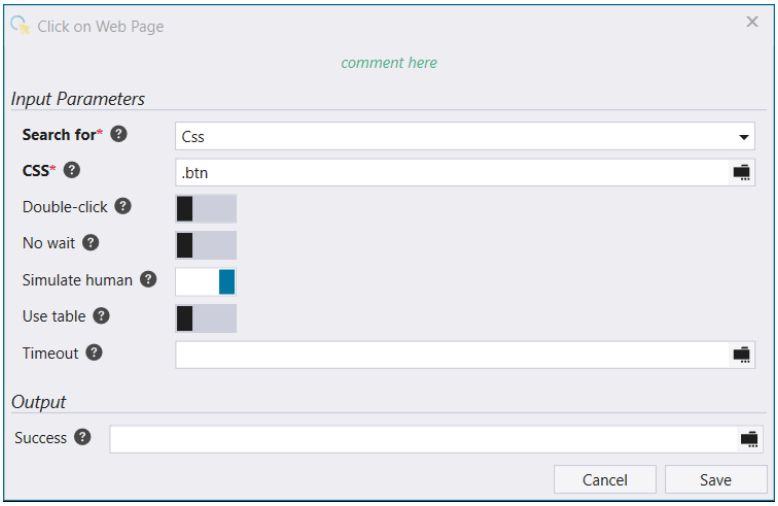
_8. For `Selector Type` select **Css,** for `Css` copy the CSS Selector for the ***Follow up*** **check box** in the Sales Leads web page. Enable `Simulate Human` option. Click `Save`.



_9. Add `Click on Web Page` command under *End If* to click the **Submit** button in the Sales Leads web page.

© Copyright International Business Machines Corporation 2020

_10. For `Selector Type` select **Css,** for `Css` copy the CSS Selector for the ***Submit*** button web. Enable `Simulate Human` option. Click `Save`.



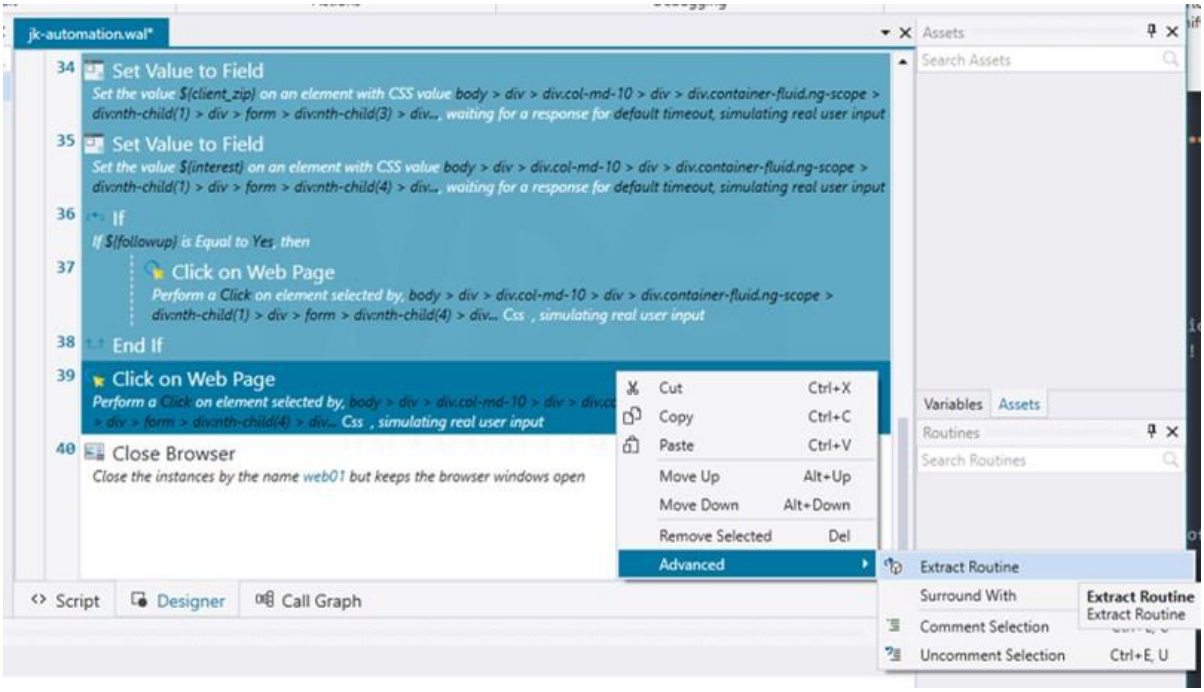
_11. **Save** your work, **close your browser** showing the JK Automation web site and **run** your automation by hitting the `Run` icon in the top toolbar. You should see your automation executing and entering the first sales lead (Paul Pacholski) to the JK Automation Sales Leads page. Nice!

Define Subroutines

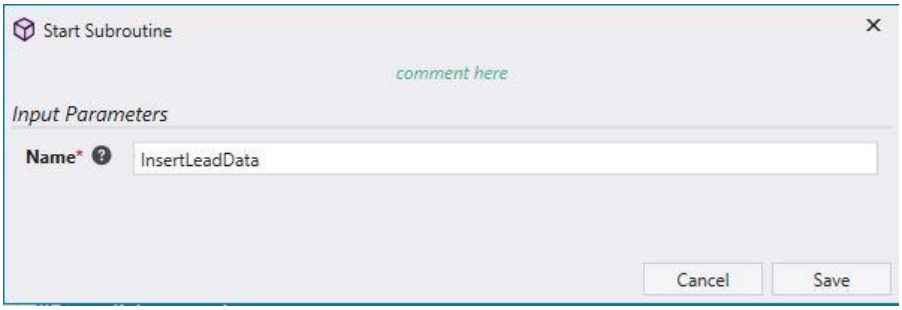
IBM RPA Studio allows you to define and group parts of your automation as sub-routines. This can be useful to group logic parts of your automation script and make your automation more easily understandable.

© Copyright International Business Machines Corporation 2020

_1. In your WGD Studio Designer view, select (*Shift+Click*) **all commands** between *Map Table Row* and *Close Browser* (**NOTE!** Do **NOT** select *Map Table Row* and *Close Browser*!), right-click one of the selected commands and select **Advanced** > **Extract Routine**.

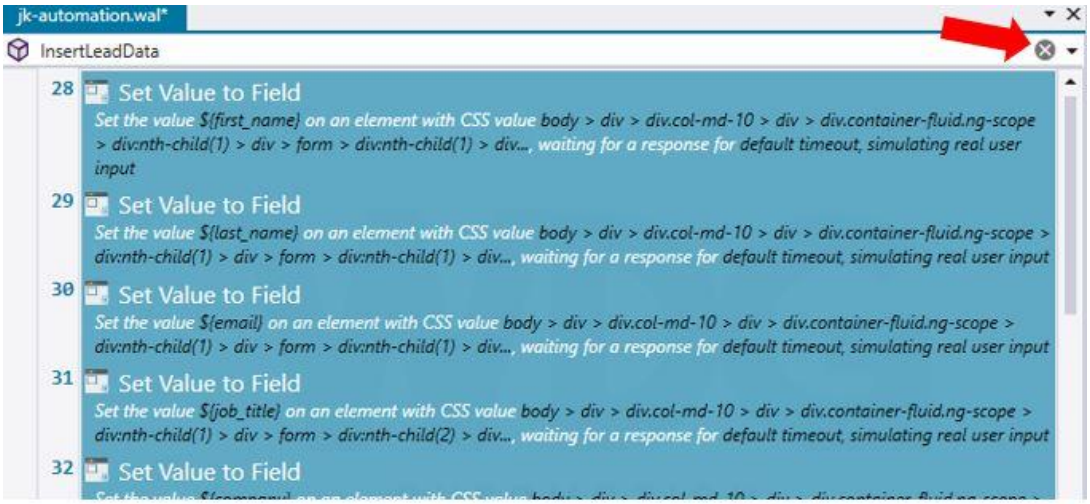


_2. Name your sub-routine as **InsertLeadData**. Click `Save`.

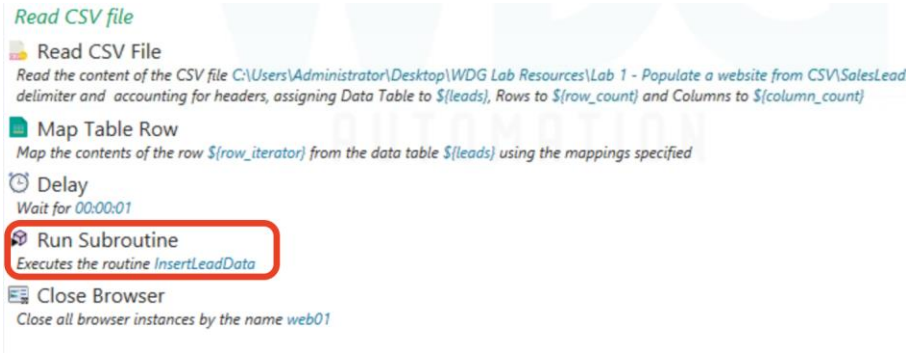


© Copyright International Business Machines Corporation 2020

_3. Studio will show the extracted sub-routine. You can close it and return to "main" automation by clicking the *close icon* after the name of the sub-routine.



_4. You can now see the extracted sub-routine in your "main" automation script.



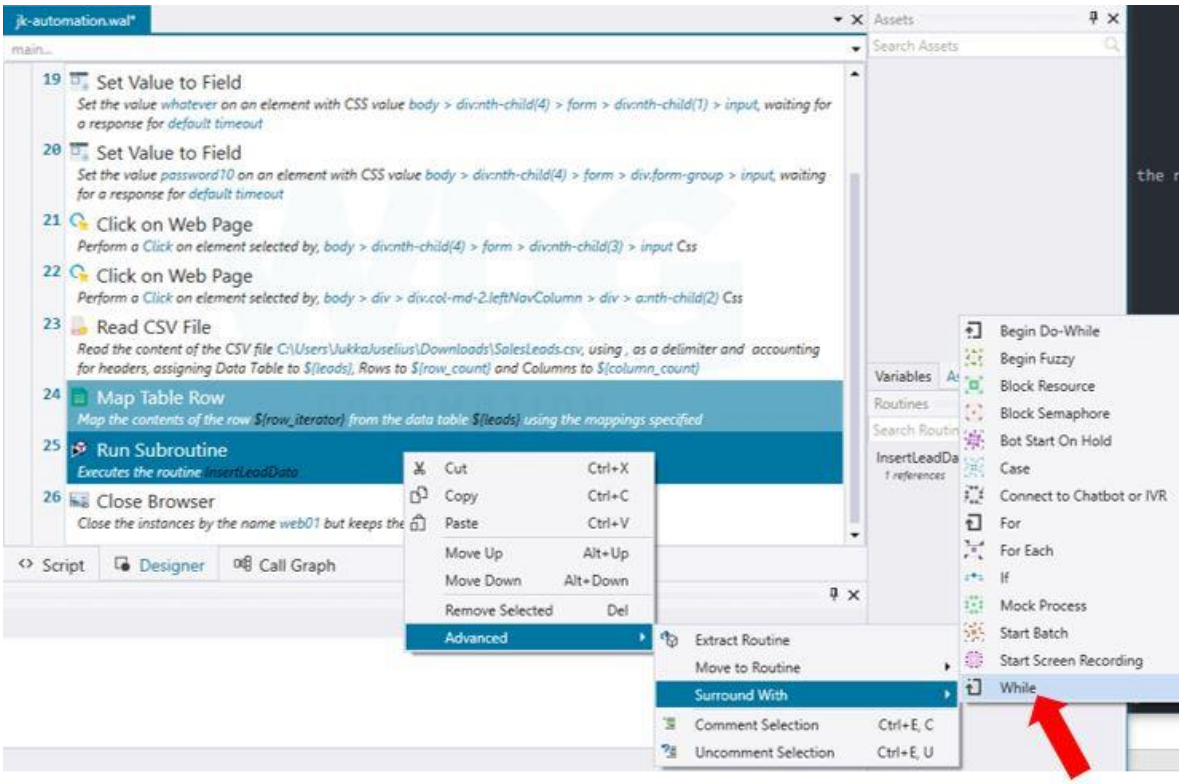
Note, variables in IBM RPA Studio are global. This means that you do not need to map variables between your main automation script and its sub-routines.

Add loop to iterate through the data table

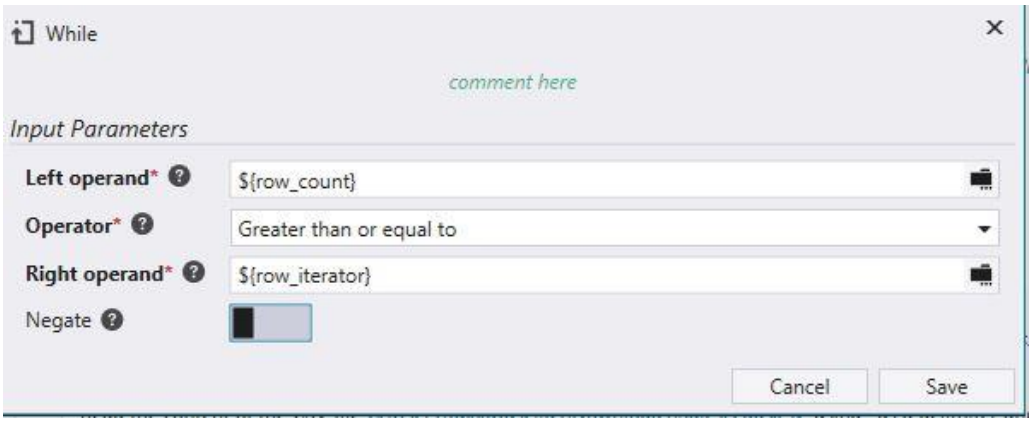
We have our automation almost ready. One of thing we still need to do, is to add a looping structure to iterate through all the sales leads within the data table that we read from the CSV file. IBM RPA studio offers several structures to do this. We will use a `While` loop and use the row_count variable to go through all the records.

© Copyright International Business Machines Corporation 2020

_1. Select the `Map Table Row` and the `Run Subroutine` commands, **right-click** one of the selected commands, and select ***Advanced*** > ***Surround With*** > ***While***.

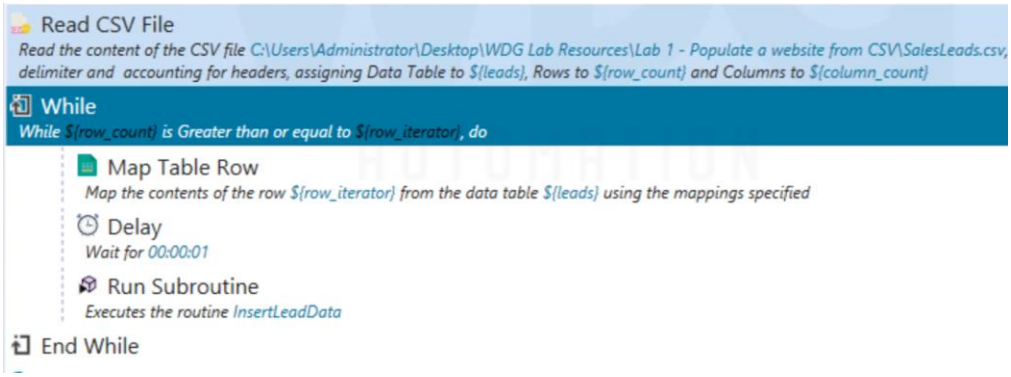


_2. Configure the While command using the **row_count** and **row_iterator** variables as follows. Click `Save`.



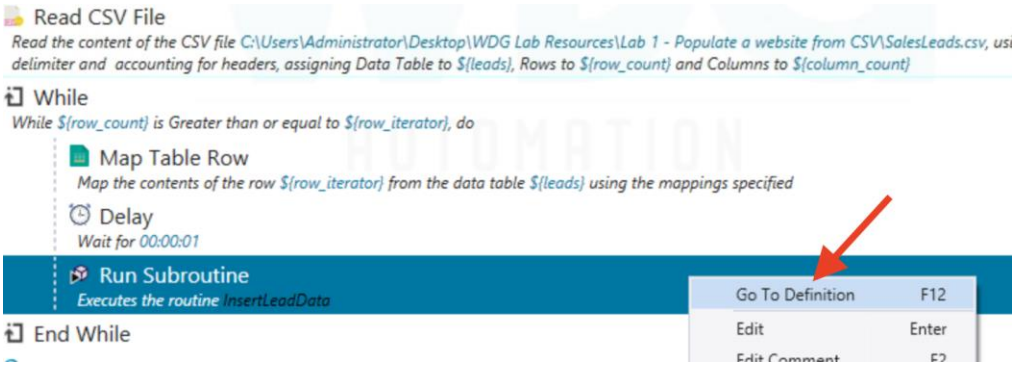
© Copyright International Business Machines Corporation 2020

You will now see the While loop in your main automation script.



Good job! But something is missing... Right! We need still need to make sure that we increase our **row_iterator** value for each iteration.

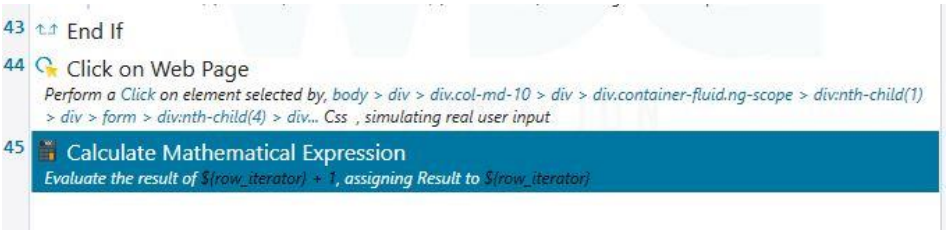
_3. **Right-click** the `Run Subroutine` inside the While loop and select **Go To Definition**.



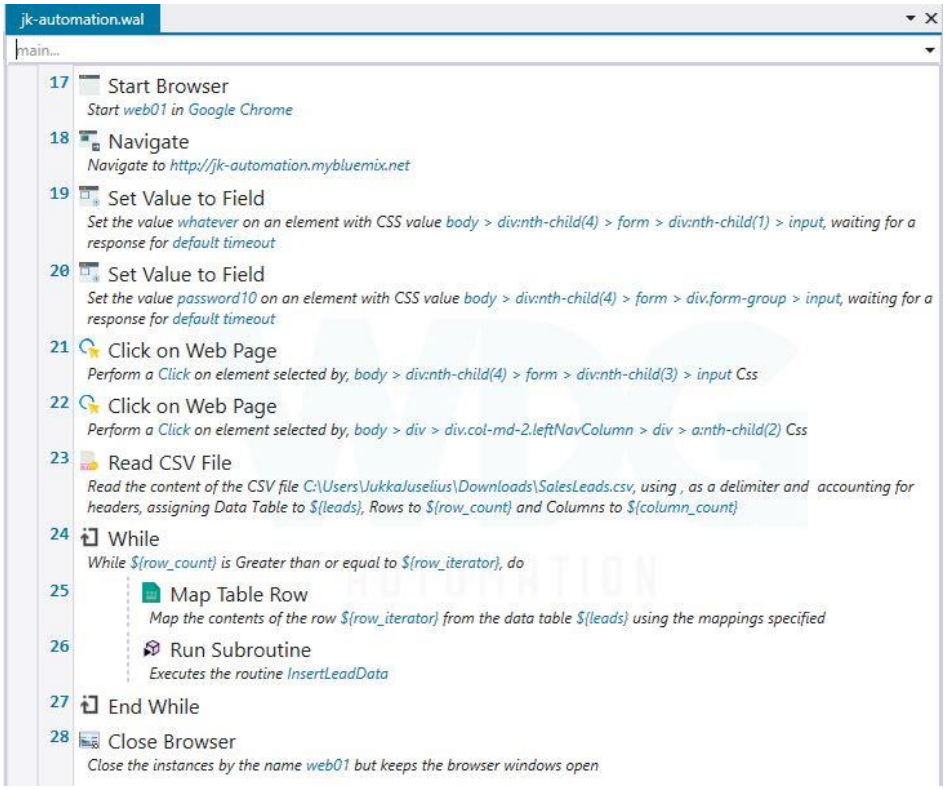
_4. Add `Calculate Mathematical Expression` command (Base > Numeric) as the last command of the sub-routine. Configure it as follows. We want to add +1 at the end of each iteration. Don't forget to set the output result to ${row_iterator}!

© Copyright International Business Machines Corporation 2020

You should now have the command added to your sub-routine.



_5. Close the sub-routine to go back to main automation script. **Save** your work.



_6. Close your current browser window showing the JK Automation web site and **Run** your automation to test that it works and JK Automation website is opened. Click the green `Run` icon form the top toolbar. Alternatively, you can hit `F5`.

You should see the automation / bot executing and adding all the sales leads to the JK Automation web site. When the bot finished running you should now see all 12 rows from the excel spreadsheet in the Sales Leads table, first one being *Paul* and the last one *Stuart*.

Logout from JK automation website

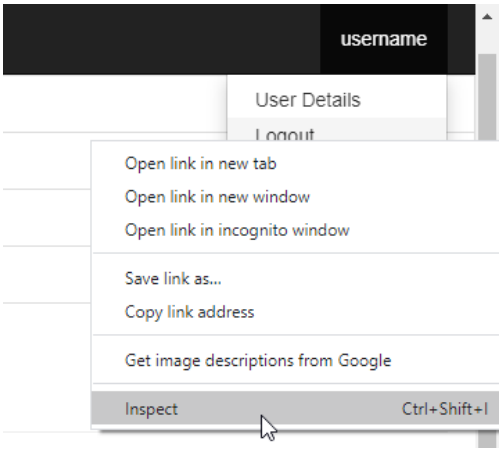
Recall that the `Close Browser` command was configured to left browser open for testing.

© Copyright International Business Machines Corporation 2020

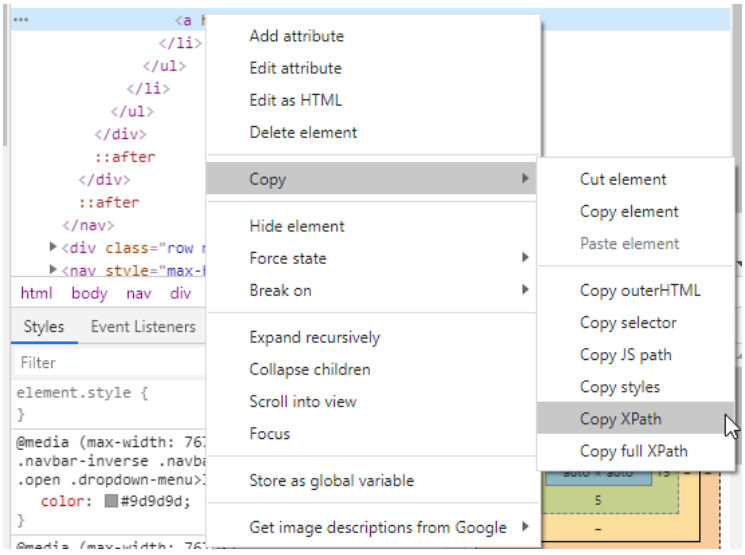
To complete our automation. Let's implement logout from the JK Automation site by adding one `Click on Web Page` command at the end of your automation, just before the `Close Browser` command.

_1. C**lick your username** on the right-hand side upper corner of the web page.

_2. Select **the logout link** from the menu, right click and then select **Inspect**.



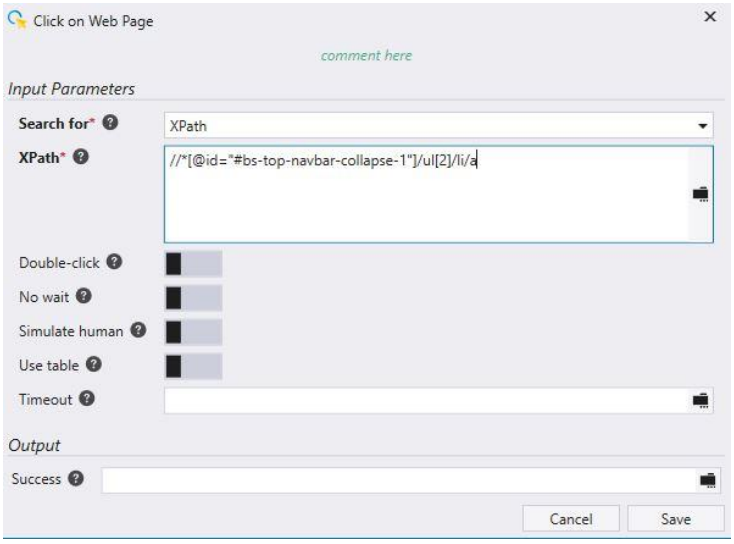
_3. Select **Copy > Copy Xpath**.



© Copyright International Business Machines Corporation 2020

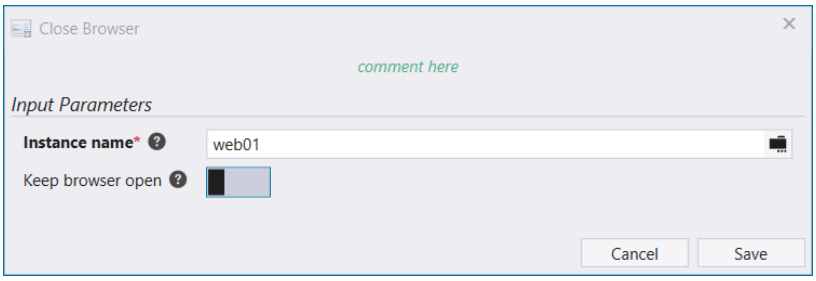
_4. Add **Click on Web Page** command after *End While* command.

_5. Set *Search for* to **XPath** and *Xpath* to **the value from your clipboard** (the XPath you copied from element inspector). Finally click `Save`.



The last three commands in the Designer should now look similar to this:

_6. Double click `Close Browser command` at the end of your automation and change the `Keep browser open` **off** and click `Save`.



_7. Test your automation again. You should now see the browse close after the automation completes.

Well done! Congrats finishing the exercise! Hope you learned some of the basics on how to use IBM RPA Studio. Keep on automating 😊

THIS COMPLETES THIS HANDS-ON LAB

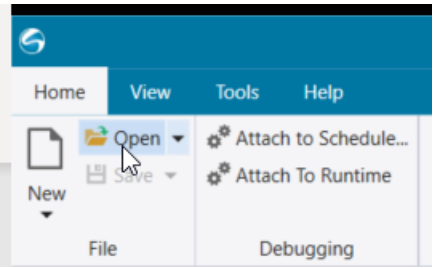
© Copyright International Business Machines Corporation 2020

3. Work with Completed Automation

If you just want to run the automation without authoring it or if you just want to look at a completed solution, follow these instructions below.

Open completed lab

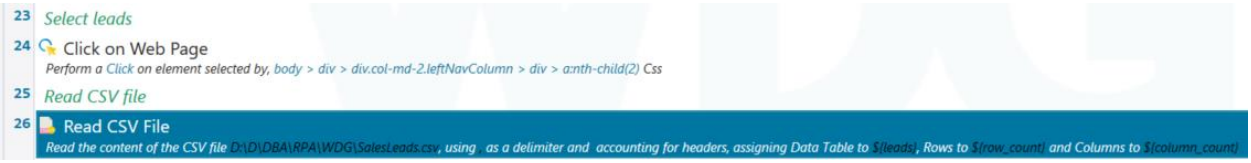
_1. In IBM RPA Studio, Select `Open`.



_2. Navigate to **sales-lead-automation-completed.wal** in folder *Desktop → IBM RPA Lab Resources → Lab 1 – Build your first IBM RPA bot* and click `Open`.

Check input csv file location

_1. Double click on the `Read CSV File` command



_2. Make sure that the location of the file if set to *Desktop → IBM RPA Lab Resources → Lab 1 – Build your first IBM RPA bot →* **SalesLeads.csv**.

Run the automation

_1. Click the green `Run` icon form the top toolbar. Alternatively, you can hit `F5`.

© Copyright International Business Machines Corporation 2020