PUBLIC

RPA As Easy As 1-2-3: Low-Code Design for Microsoft Office and SAP Software

INT165

Exercise 1: Starting with Excel

Xiaohui XUE / SAP
Moumita Bera / SAP

THE BEST RUN SAP

TABLE OF CONTENTS

INTRODUCTION

This workshop aims to introduce the new features available in SAP Intelligent RPA 2.0 and how they can be used to simplify the business integration with Microsoft office and with SAP tools.

In this workflow, you will create process on Excel.

[Note: We are expecting that you have created a trial account.]

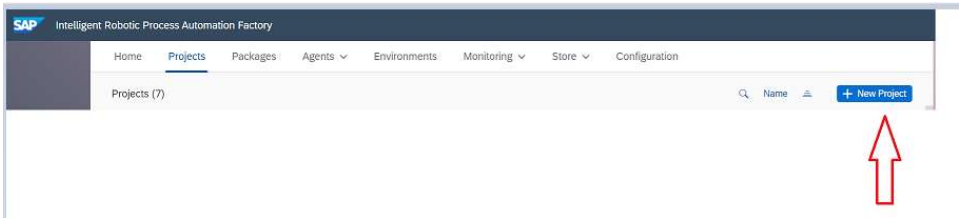
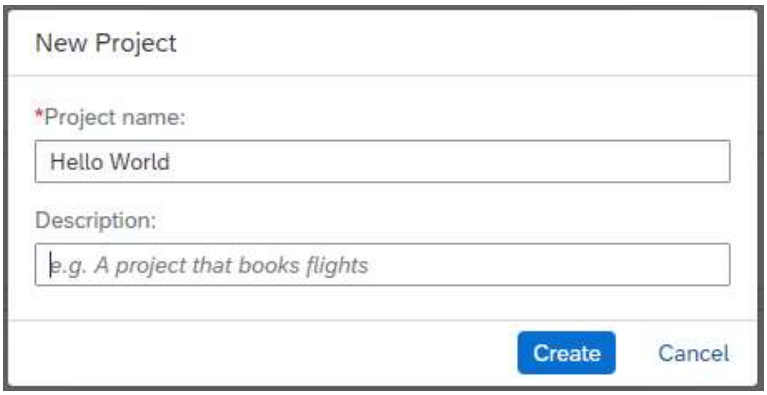
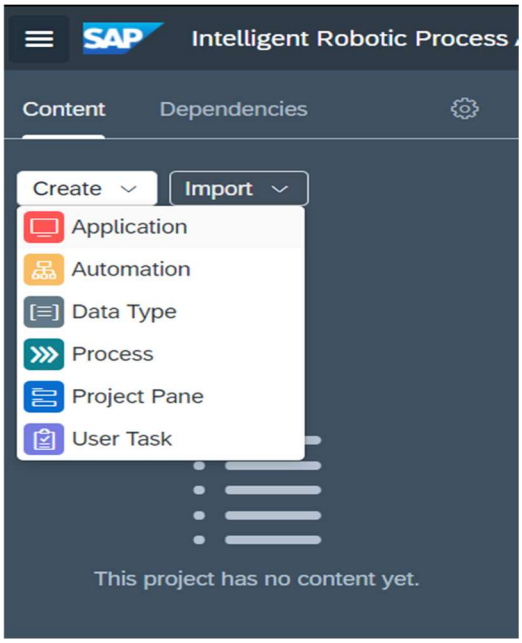
EXERCICE 1: CREATE BOT ON EXCEL

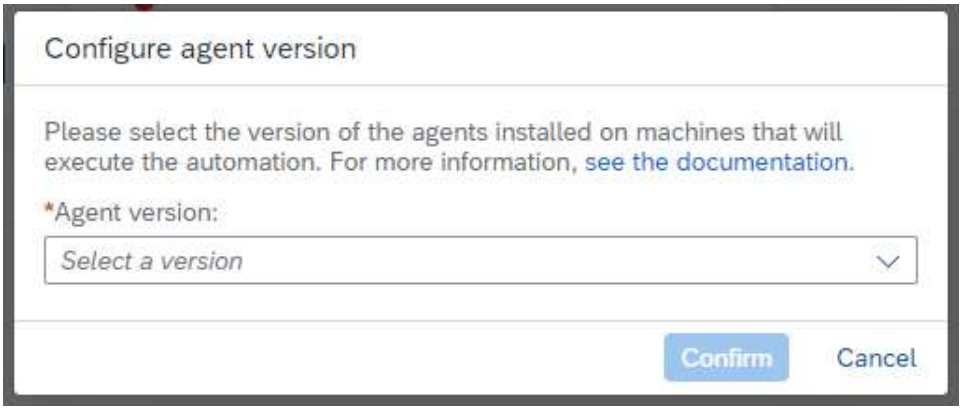
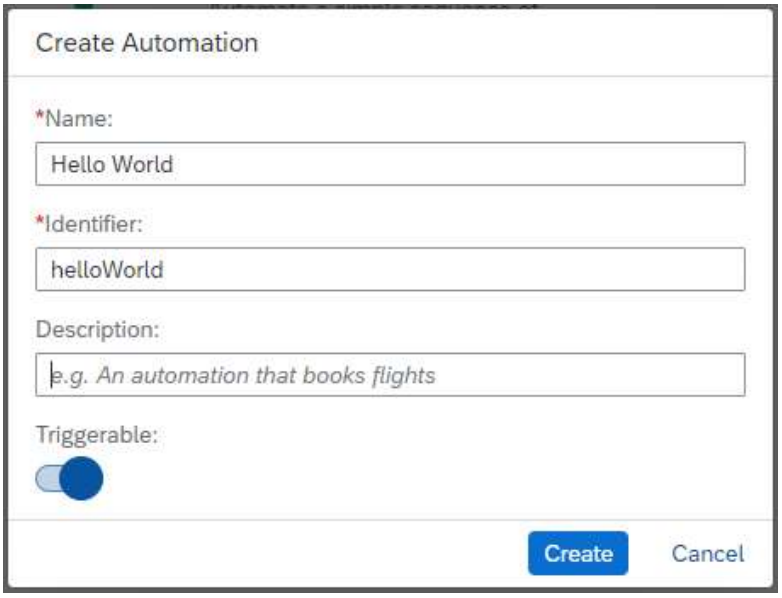
Estimated time: 30 minutes

The aim of this exercise is to create a simple automation using the Excel library. You will see how to create a simple automation which:

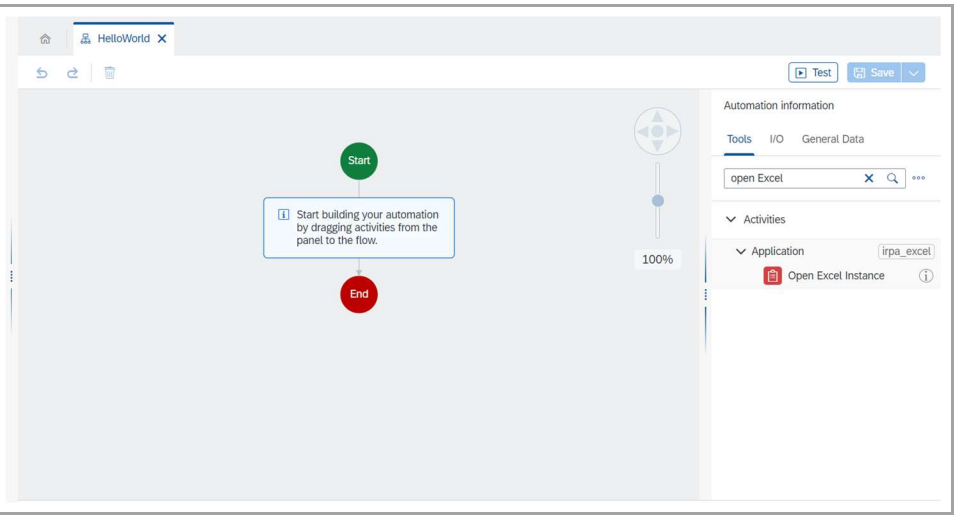
- ... creates an Excel file
- … inserts text in a cell of the created Excel file

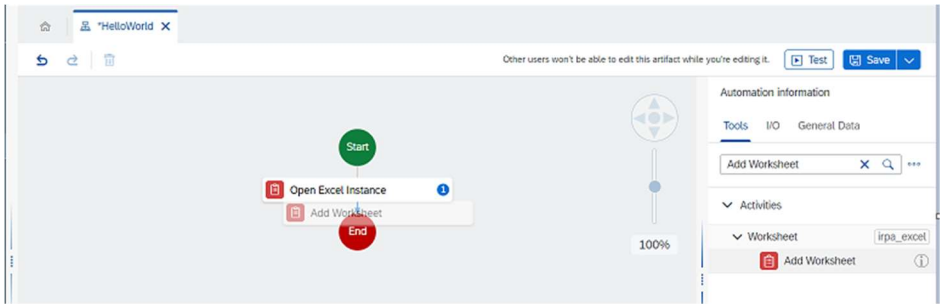
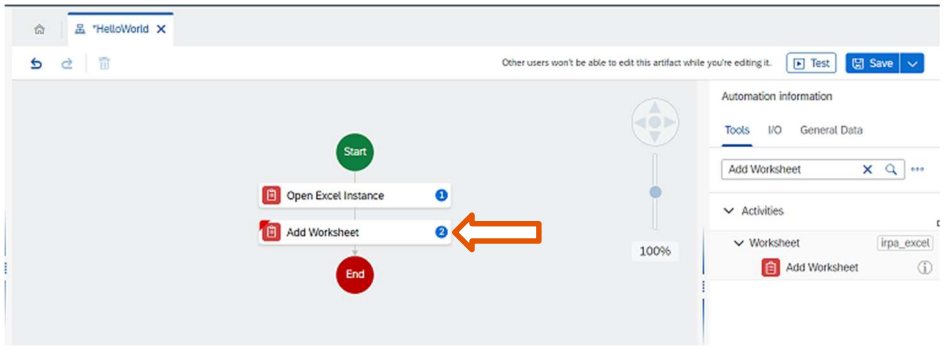
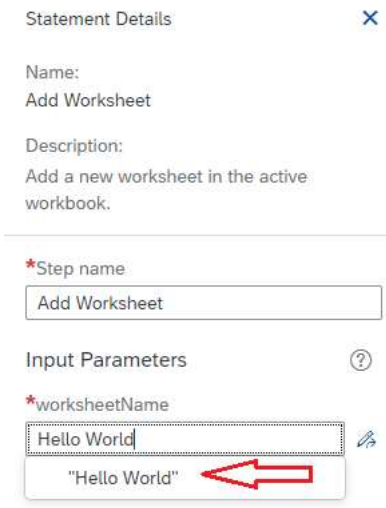
Creation of the automation

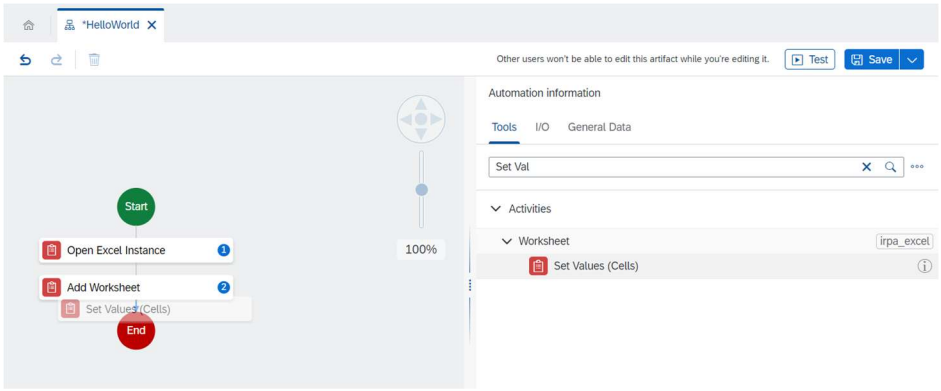
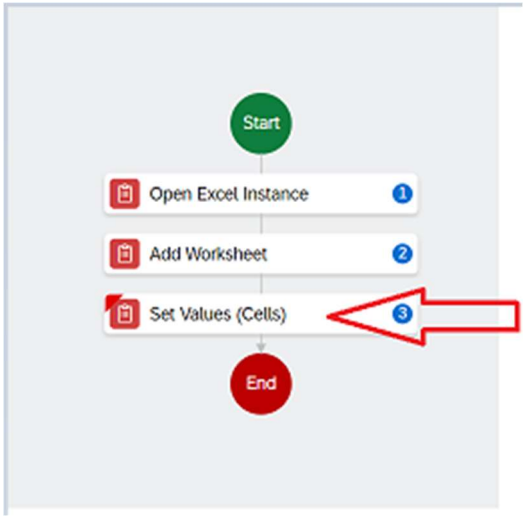
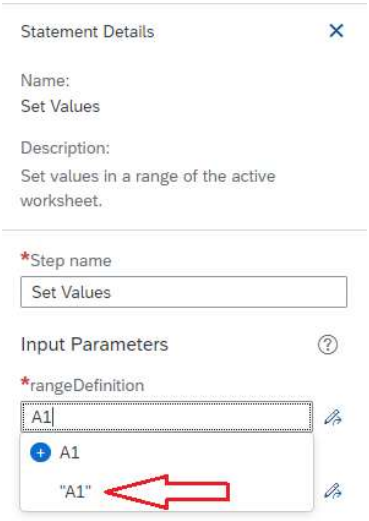
Explanation	Screenshot
Go to **Projects** Tab. Click on **New Project**	
A new popup appears. **Choose a name** for the project (here HelloWorld), then **click** on the Create button.	
Click on the Create button to expand the dropdown menu, then **select** Automation.	

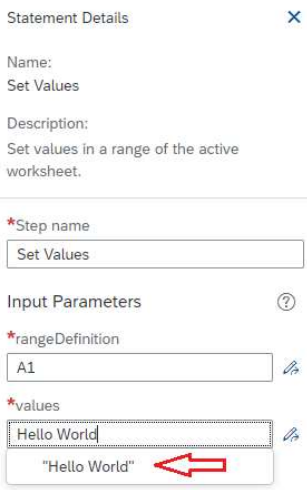
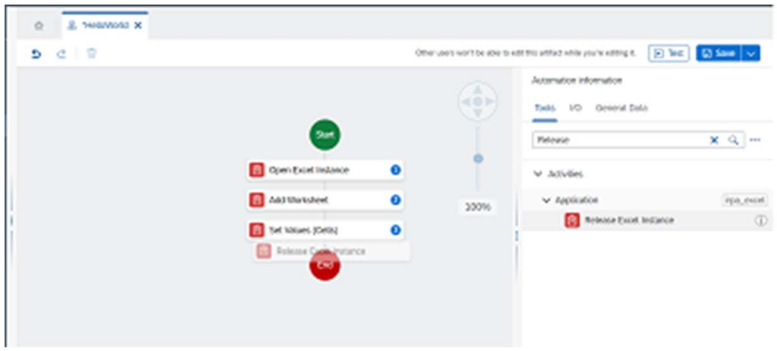
In the pop-up dialog, select the **agent version** for this project. In this exercise, the version can be the one marked with the label **Local**. **Click** on **Confirm**.	
A new popup appears. **Choose a name** for the automation (here HelloWorld), then **click** on the Create button.	

Adding Excel Activities

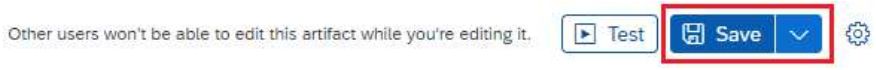
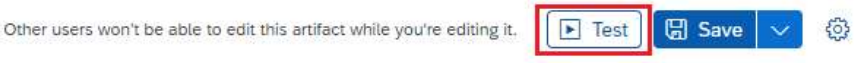
Search for "open excel" in the search bar of the right-hand side panel. **Drag & drop** the Open Excel Instance activity from the Tools panel into the workflow, between the green and red circle.	

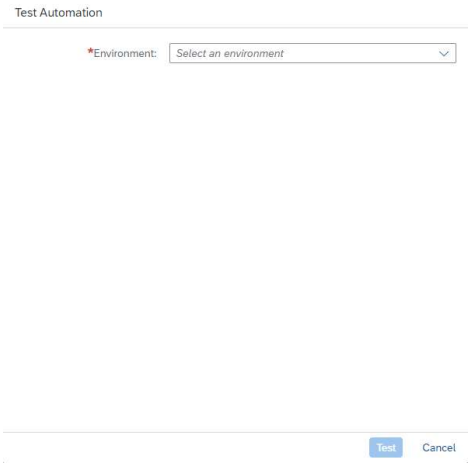
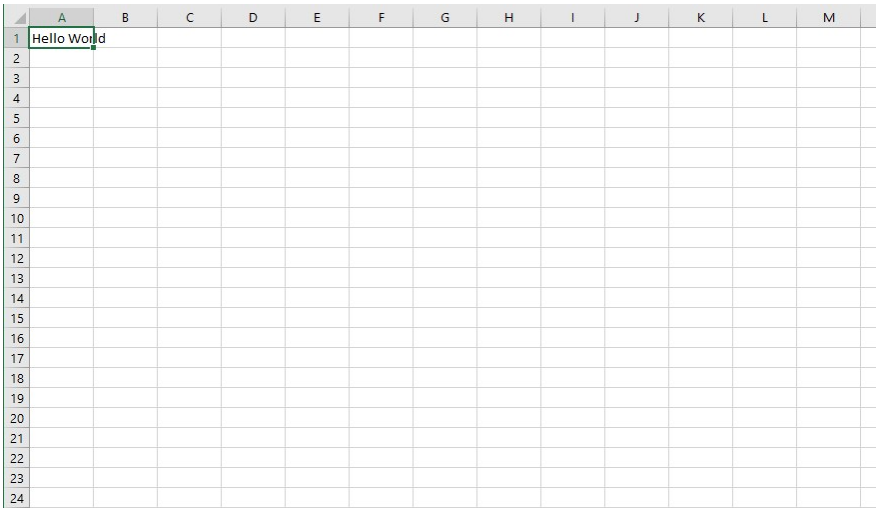
Now search for “add worksheet” in the search bar on the right side. Drag & drop the Add Worksheet activity from the Tools panel to the workflow, between the green Start and red End node.	
You will notice a small red triangle in the corner of the activity. It indicates that something is missing. Let’s fix that. **Click** on the Add Worksheet activity to open the Statement Details panel on the right.	
Choose a name for the Excel sheet in the field labeled worksheetName. For our demo, let's go with the title *Hello World.*	

Click on the grey canvas to deselect the Statement Details panel and see the Activities panel again. From there, search for “set values” and **drag & drop** the Set Values activity onto the canvas, below the Add Worksheet activity.	
Click on the Set Values activity to open the Statement Details panel on the right.	
In the rangeDefinition field, you can now **choose** a cell where you want to set your input. Here *A1* is chosen.	

In the values field, **enter** a string which you want to insert into the cell in the Excel file. For the purpose of this demo, we will enter *Hello World*.	
Finally, **click** on the canvas again to get back your Activities panel and **search** "release" to find one last activity. **Drag & drop** the *Release Excel Instance* activity onto your automation, below the *Set Values* activity.	

Test your project

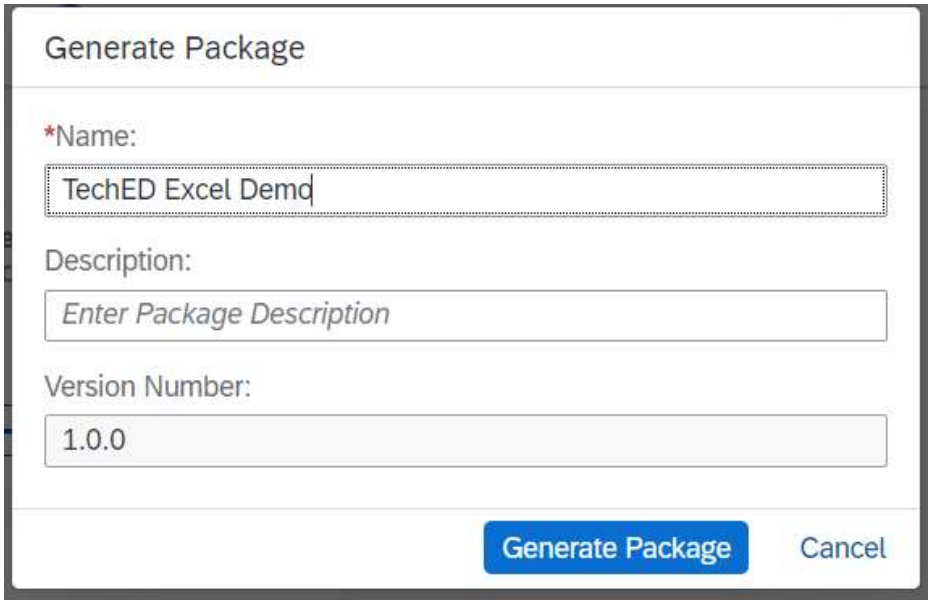
Explanation	Screenshot
Once everything is done, **click on the Save button** on the top right of the screen.	
Now **click on the Test button** to launch the automation.	

Select the Environment you have previously created.	
After a few seconds of loading, an Excel file will open which will show the *Hello World* string set in the cell A1. When you are finished, don't forget to close the tester session by clicking on the red Stop icon that appeared on your screen.	

Run your project

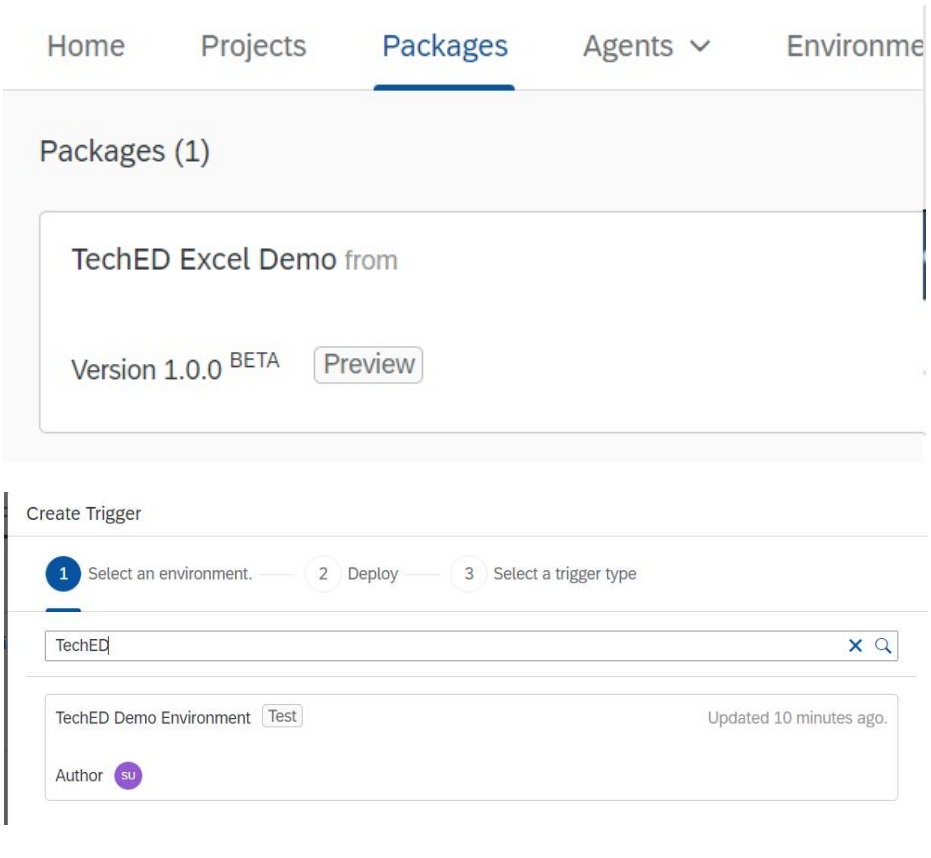
Explanation	Screenshot

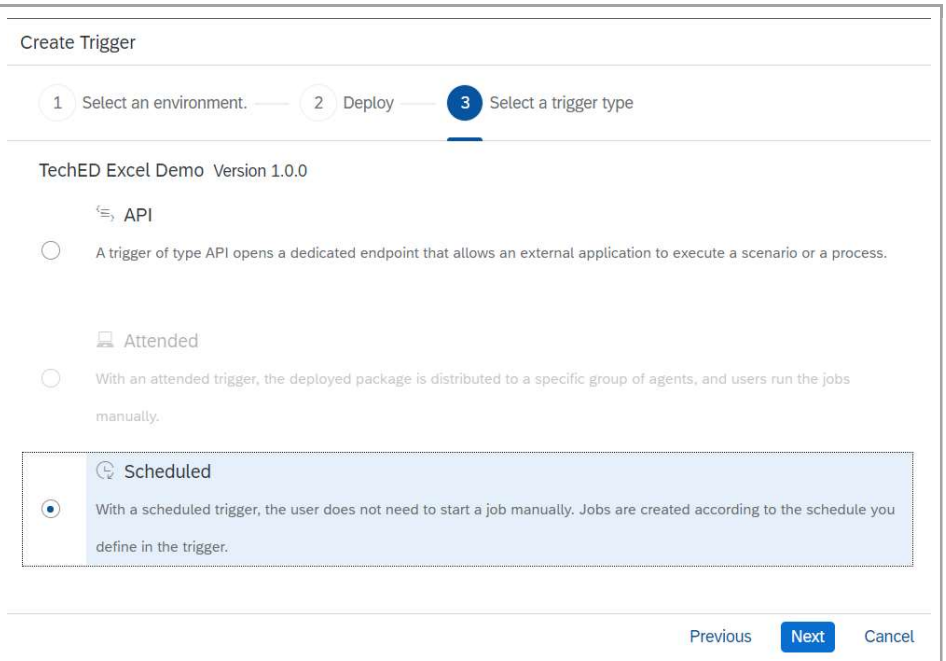
- Go to your project.
- Click on the **Home** button of the project
- Check that the conditions to generate a Package are fulfilled
  (i.e. All values for error and warning are set to 0)
- Click on the **Generate Package** button on the right
  A dialog will open, again click on **Generate Package**
  Close the Project navigator tab



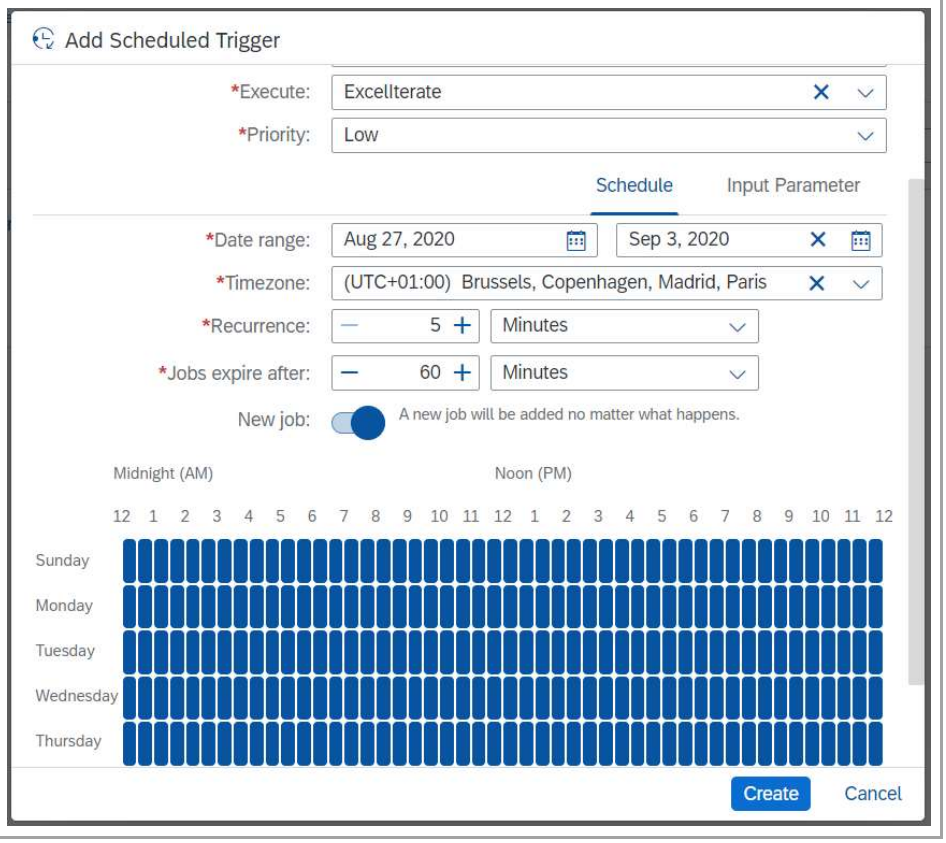
Locate the generated package in the **Packages** tab. **Click** on **Add Trigger** to create a scheduled trigger:

- In “Select an Environment”, select the Environment you have previously created
- In “Select a trigger type”, chose **Scheduled** trigger



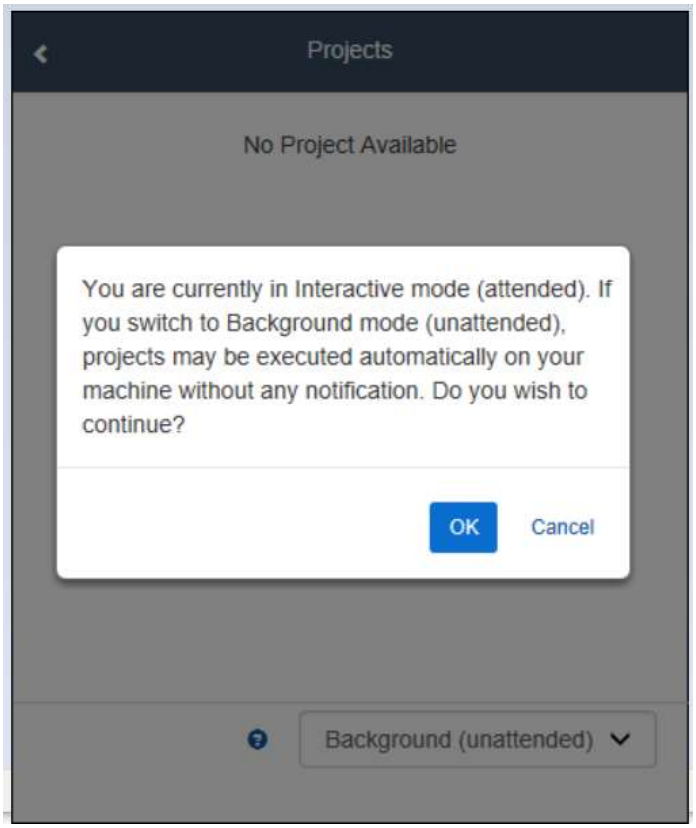


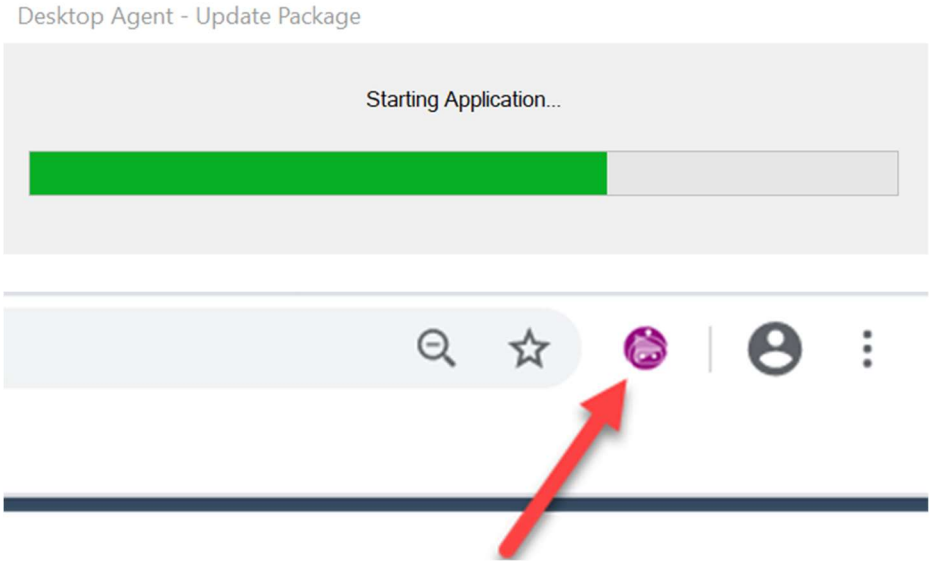
- For Execute, select from dropdown your automation to execute
- Select **Date Range** from yesterday until tomorrow and some Time period including today
- Click **Create**



<table>
<tr>
<td>

- Start Desktop Agent (You will see a popup which indicates that the package is downloading)
  - From the **Factory**, check in Agents/Agents table the default mode is Attended by default.
  - From the **systray**, click on projects, then select 'Background (unattended) and click on OK.

</td>
<td> 
</td>
</tr>
<tr>
<td>Execute unattended process
Wait until job gets distributed and scenario gets executed successfully (3 – 5 min)

You will observe in the systray that the desktop agent becomes active.</td>
<td> 
</td>
</tr>
</table>

CONCLUSION

You have completed the exercise!

You are now able to:

- create an automation using the Cloud Studio
- use automations to manipulate Excel sheets
- test and run your project

REFERENCES

Help portal for SAP Intelligent Robotic Process Automation: https://help.sap.com/viewer/p/IRPA
SAP Intelligent Robotic Process Automation on SAP Community:
https://community.sap.com/topics/intelligent-rpa

www.sap.com/contactsap

THE BEST RUN SAP®