Automation Engineer Level 1

Exercise 1a

Create a new Excel and write data in it

Objective

By the end of this exercise, you will be able to make use of the Excel Engine 3.0 standard modules in Tosca. This will enable you to create an end-to-end Excel test case that creates a new excel file and enters data in it.

Why is this Important?

Tricentis Tosca supports steering Excel files using Excel Engine 3.0 and Excel UI Engine 3.0 out of the box. This exercise will enable you to leverage the standard modules available for the Excel Engine 3.0

Project Perspective

Excel is used all across to store data and is integrated with many SUTs. There are applications that take excel files as an input to upload data into the system. Excel files can be created to enter data which would later be uploaded into the systems using the Tosca Excel Engine 3.0.

Instructions

1. Log in to your Tosca Commander workspace and navigate to the path
   AE1 Exercises>>TestCases>>Excel Engine and create a new **TestCase** in this folder
   and name it as **Create new Excel and write data**

2. Within this TestCase, create three **TestStepFolders** – 'Precondition', 'Process', and
   'Postcondition'

3. Add the Standard Module **TBox Open Excel Workbook** to the folder **Precondition** and
   rename it as **Create and open new Excel workbook**

4. Input **Values** as shared in the table below:

TestStep Value	Value	ActionMode
Workbook Name	Employee_Data_2022	Input
Path	C:\Tosca_Projects\Employee_Data_2022.xlsx	Input
Create New	True	Input

5. Add the Standard Module **TBox Create Excel Worksheet** to the folder **Precondition** and rename it as **Create new Excel worksheet**.

6. Input **Values** as shared in the table below:

TestStep Value	Value	ActionMode
Workbook Name	Employee_Data_2022	Input
Worksheet Name	Employee_Personal_Records	Input

7. Add the Standard Module **TBox Define Excel Range** to the folder **Precondition** and rename it as **Define Excel range**.

8. Input **Values** as shared in the table below:

TestStep Value	Value	ActionMode
Workbook Name	Employee_Data_2022	Input
Worksheet Name	Employee_Personal_Records	Input
Range Name	EmployeeDataRange	Input
Start Cell	A1	Input
End Cell	F7	Input

9. Add the Standard Module **TBox Excel Range Manipulation** to the folder **Process** and rename it as **Define headers**.

10. Input **Values** as shared in the table below:

TestStep Value	Value	ActionMode
Range Name	EmployeeDataRange	Input
Data Table		Select
Enter values in $header row to define header		
$header		Select
$1	SrNo	Input
$2	FirstName	Input
$3	LastName	Input
$4	EmailId	Input
$5	EmployeeId	Input

11. Add the Standard Module **TBox Excel Range Manipulation** to the folder **Process** and rename it as **Enter data in Excel**.
12. Input **Values** as shared in the table below:

TestStep Value	Value	ActionMode
Range Name	EmployeeDataRange	Input
Data Table		Select
Enter values in $1 row to enter data in 1st row		
$1		Select
SrNo	1	Input
FirstName	John	Input
LastName	Doe	Input
EmailId	John.Doe@test.com	Input
EmployeeId	1011	Input
Enter values in $2 row to enter data in 2nd row		
$2		Select
SrNo	2	Input
FirstName	Jane	Input
LastName	Doe	Input
EmailId	Jane.Doe@test.com	Input
EmployeeId	1012	Input

13. Add the Standard Module **TBox Close Excel Workbook** to the folder **Postcondition** and rename it as **Close Excel workbook**.
14. Input **Values** as shared in the table below:

TestStep Value	Value	ActionMode
Workbook Name	Employee_Data_2022	Input
Path	C:\Tosca_Projects	Input
Save	True	Input

15. Mark the TestCase **Completed** and run it in ScratchBook.

Expected outcome

The TestCase should be executed successfully, and a new Excel file would be created post execution. The newly created Excel will have the data we entered using the TestCase.

Hints

1. The Excel sheets can be steered like tables

2. You can save the Excel sheet in various formats like PDF, CSV etc