4 Print “Hello”

4.1 Print "Hello" by using Sequence activity

Objective: To code a Robot in UiPath Studio to print "Hello" in a message box and write line by using **“Sequence”** activity.

Learning Outcomes
After completion of this exercise you will get familiar with the following:

- ✓ **“Sequence”** and **“Assign”** activity.
- ✓ **“Comment”** and "**Annotation”.**
- ✓ **“Message Box”** activity.
- ✓ **“Write Line”** activity.

Algorithm:

Step 1: START
Step 2: Add Message Box activity and write " Hello" into Flow Chart Activity
Step 3: Add Write Line activity and write "Hello" in Flow Chart Activity
Step 4: STOP

Step by Step process:

Step 1: Open UiPath Studio.

Step 2: Create the process and name it.

Step 3: Drag the **"Sequence"** activity from the activity panel and drop it in the workflow.

Step 4: Name the **"Sequence"** activity as **Sequence - 'This code is an example of Sequence activity to display Hello using message box and write line activities'.**

Step 5: Drag and Drop the "**Comment"** activity in the **"Sequence"** activity and write the comment **'// Write Hello using message box & write line activities'**.

Step 6: Drag and Drop the Message Box in **"Sequence"** activity and name it as **Message Box – 'To display Hello in a message box pop up'**.

Step 7: In the **Text** property of "**Message Box – 'To display Hello in a message box pop up'**" write **"hello"**.

Step 8: Drag and drop **"Write line"** activity and name it as **Write Line – 'To display Hello in the output panel'.**

Step 9: In the **Text** property of "**Write line – 'To display Hello in the output panel'**" write **"hello"**.

4.2 Print "Hello" by using Flowchart activity

Objective: To code a Robot using UiPath Studio Print "Hello" display "Hello" in a message box and write line by using **"Flowchart"** activity.

Learning Outcomes

After completion of this exercise you will get familiar with the following:

- ✓ **"Flowchart"** activity.
- ✓ **"Comment"** and "**Annotation".**
- ✓ **"Message Box"** activity.
- ✓ **"Write Line"** activity.

Algorithm:

Step 1: START
Step 2: Add Message Box activity and write " Hello" in **"Sequence"** activity
Step 3: Add Write Line activity and write "Hello" in **"Sequence"** activity
Step 4: STOP

6 Displaying a Sun Sign

Objective: Code a Robot in UiPath Studio to display the sun sign of an individual by entering the Date of birth of an individual and produce the output in the message box.

Learning Outcomes
After completion of this exercise you will get familiar with the following:

- ✓ **"Sequence"** and **"Assign"** activity.
- ✓ **"Comment"** and "**Annotation".**
- ✓ **"Open Browser"** and **"Maximize Window"** activity.
- ✓ **"If"** activity and how to set conditions.
- ✓ **"Input Dialog**', **"Type Into",** and **"Click"** activity and how to set **variables** in a code using **"Variable**' panel.
- ✓ Display output in **"Message Box**'.

Algorithm:

Step 1: START
Step 2: Open the URL using Open Browser Activity
Step 3: Declare the variables as 'Dates' , 'Months' , 'Years' , 'SunSign' ,'ElementExist'
Step 4: Use ElementExist variable to check text boxes are avilable in the browser
Step 5: Add If-Else block activity and add ElementExist in the condition text box
Step 6: In the If block use variables 'Dates' , 'Months' , 'Years' to store them
Step 7: In the Else block write "Sunshine couldn't be generated, Page not Found"
Step 8: STOP

Step by Step process:

Step 1: Open UiPath Studio.

Step 2: Create the process and name it.

Step 3: Drag the **"Sequence"** activity from the activity panel and drop it in the workflow.

Step 4: Name the **"Sequence"** activity as **Sequence – 'This code is for identifying the Sun Sign of an individual by entering Date of Birth.'**

Step 4: Add a "**Comment"** activity from the activity panel and write '**// To generate Sun Sign of an individual by getting the date of birth as an input**.'

Step 5: Drag and drop the **"Open Browser"** activity from the activity panel and name it as **Open Browser – 'Opens a specific browser using specified URL'** and inside the URL **text field** write:

"**https://astrologyfutureeye.com/astro-calculators/sun-sign-calculator**"

Step 6: Change the name of "**Do"** sequence from the **"Open Browser"** activity as **Do – 'Get date of birth as input from user, entering it in the browser text boxes and display the sun sign.'**

Step 7: Drag the **"Maximize Window"** activity from the activity panel and drop it in the workflow.

Step 8: Name the **"Maximize Window"** activity as **Maximize Window – 'To maximize the browser window.'**

Step 9: Drag and Drop "**Input Dialog"** activity and name it as **Input Dialog – 'Date of birth entered by user'** and write the values as under:

Title	Label
"Date of Birth'	"Enter Date in numbers like -1, 2 etc."

Step 10: Create a variable through "**Variables**' panel for the **"Input Dialog"** activity as under:

Name	Variable Type	Scope	Default
Dates	String	Do - 'Getting user input, entering it to the browser text boxes, and generating sun sign.'	

Step 11: Declare the variable **Dates** in the **"Output"** Property of "**Input Dialog"** activity.

Step 12: Drag and Drop "**Input Dialog"** activity and name it as **Input Dialog – 'Month entered by user'** and write the values as under:

Title	Label
"Month of Birth'	"Enter Month e.g., April."

Step 13: Create a variable through "**Variables**' panel for the **"Input Dialog"** activity as under:

Name	Variable Type	Scope	Default
Months	String	Do - 'Getting user input, entering it to the browser text boxes, and generating sun sign.'	

Step 14: Declare the variable **Months** in the **"Output"** Property of "**Input Dialog"** activity**.**

Step 15: Drag and Drop "**Input Dialog"** activity and name it as **Input Dialog – 'Year entered by user'** and write the values as under:

Title	Label
"Year of Birth'	"Enter year in YYYY eg, 1995."

Step 16: Create a variable through "**Variables**' panel for the **"Input Dialog"** activity as under:

Name	Variable Type	Scope	Default
Years	String	Do - 'Getting user input, entering it to the browser text boxes, and generating sun sign.'	

Step 17: Declare the variable **Years** in the **"Output"** Property of "**Input Dialog"** activity.

Step 18: Drag the **"Element Exists"** activity from the activity panel and drop it in the workflow.

Step 19: Name the **"Element Exists"** activity as **Element Exists – 'To check if the sun sign finder block exists in the website.'**

Step 20: Create a variable through "**Variables**' panel for the "**Element Exists"** activity **as under:**

Name	Variable Type	Scope	Default
ElementExist	Boolean	Do - 'Getting user input, entering it to the browser text boxes and generating sun sign.'	

Step 21: Declare the variable **ElementExist** in the **"Output"** Property of "**Element Exists"** activity**.**

Step 22: Drag the **"IF"** activity from the activity panel and drop it in the workflow.

Step 23: Name the **"IF"** activity as **IF – 'Condition to check if the Sun sign finder exists or no. If exists then the code will continue, and if not then display an error message.'**

Step 24: Use the **ElementExist** Variable in the condition box of "**IF"** activity**.**

Step 25: Drag the **"Sequence"** activity from the activity panel and drop it in the **"Then"** workflow.

Step 26: Name the **"Sequence"** activity as **Sequence – 'Enter Date of Birth (Date, Month, Year).'**

Step 27: Drag the **"Sequence"** activity from the activity panel and drop it in the workflow.

Step 28: Name the **"Sequence"** activity as **Sequence – 'Date.'**

Step 29: Right-click on **"Sequence"** activity select **Annotation** > **Add Annotation 'Entering and selecting the Date by identifying if in the dropdown menu.'**

Step 30: Drag the **"Click"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 31: Name the **"Click"** activity as **Click - 'To click and select the date from the date block.'**

Step 32: Drag the **"Send Hotkey"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 33: Name the **"Send Hotkey"** activity as **Click - 'To make the dropdown date visible.'**

Step 34: In the **"Send Hotkey"** activity select the **"enter"** key from the "**Key"** dropdown menu.

Step 35: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 36: Name the **"Type Into"** activity as **Type into - 'To select the date box.'**

Step 37: Declare the variable **"Dates"** in the **"Input"** Property of **"Type Into"** activity.

Step 38: Drag the **"Sequence"** activity from the activity panel and drop it in the **Sequence – 'Enter Date of Birth (Date, Month, Year)'** workflow.

Step 39: Name the **"Sequence"** activity as **Sequence – 'Month.'**

Step 40: Right-click on **"Sequence"** activity select **Annotation** > **Add Annotation 'Entering and selecting the month by identifying it from the dropdown menu.'**

Step 41: Drag the **"Click"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 42: Name the **"Click"** activity as **Click - 'To Click and select the month from the month block.'**

Step 43: Drag the **"Send Hotkey"** activity from the activity panel and drop it in the "**Sequence".**

Step 44: Name the **"Send Hotkey"** activity as **Click - 'To make the dropdown month visible.'**

Step 45: In the **"Send Hotkey"** activity select the "**enter**' key from the **"Key"** dropdown menu.

Step 46: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 47: Name the **"Type Into"** activity as **Type into - 'To Select the month box.'**

Step 48: Declare the variable **"Months"** in the **"Input"** Property of **"Type Into"** activity.

Step 49: Drag the **"Sequence"** activity from the activity panel and drop it in the **Sequence – 'Enter Date of Birth (Date, Month, Year)'** workflow.

Step 50: Name the **"Sequence"** activity as **Sequence – 'Year.**

Step 51: Right-click on **"Sequence"** activity select **Annotation** > **Add Annotation 'Entering and selecting the year by identifying it from the dropdown menu.'**

Step 52: Drag the **"Click"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 53: Name the **"Click"** activity as **Click - 'To Click and select the year from the year block.'**

Step 54: Drag the **"Send Hotkey"** activity from the activity panel and drop it in the **"Sequence"**.

Step 55: Name the **"Send Hotkey"** activity as **Click - 'To make the dropdown Year visible.'**

Step 56: In the "**Send Hotkey"** activity select the "**enter**' key from the **"Key"** dropdown menu.

Step 57: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 58: Name the **"Type Into"** activity as **Type into - 'To Select the year box.'**

Step 59: Declare the variable **"Years"** in the **"Input"** Property of "**Type Into"** activity.

Step 60: Drag the **"Click"** activity from the activity panel and drop it in the **Sequence – 'Enter Date of Birth (Date, Month, Year)** workflow.

Step 61: Name the **"Click"** activity as **Click – 'Click on the calculate button to show the zodiac dashboard web screen.'**

Step 62: Drag the **"Get Value"** activity from the activity panel and drop it in the "**Sequence"** workflow.

Step 63: Name the **"Get Value"** activity as **Get Value – 'Get the Sun Sign.'**

Step 64: Declare the variable **"Sun sign"** in the **"Output"** Property of "**Get Value"** activity.

Step 65: Drag the **"Message Box"** activity from the activity panel and drop it in the "**Sequence"** workflow.

Step 66: Name the **"Message Box"** activity as **Message Box - 'To print Zodiac sun sign on screen.'**

Step 67: Drag the **"Message Box"** activity from the activity panel and drop it in the "**Else"** workflow of **If** - '**Sun sign finder exists, enter values else print message**' activity.

Step 68: Declare the variable **"Sun Sign couldn't be generated, Page not Found"** in the **"Input"** Property of **"Message box"** activity.

8 Compare two columns of a spreadsheet

Objective: To code a Robot in UiPath Studio to compare two columns in a spreadsheet and display the result in third column if it is a match or not.

Learning Outcomes
After completion of this exercise you will get familiar with the following:

- ✓ **"Sequence"** and **"Assign"** activity.
- ✓ **"Comment"** and "**Annotation".**
- ✓ **"Excel Application Scope"** activity.
- ✓ **"Read Range"** and **"Write Cell"** activity.
- ✓ "**For Each Row"** activity.
- ✓ **"If"** activity and how to set conditions.

Package Prerequisites:

- 🕓 **UiPath.Excel.Activities = 2.6.0**

Algorithm:

Step 1 - START
Step 2 – Read all the rows in Column 1 and Column 2 of Excel
Step 3 – Declare a variable RowNumber
Step 4 - If Column 1 = Column 2 then it is a match else it is not.
Step 5 - RowNumber = RowNumber + 1
Step 6 – STOP

Step by Step process:

Step 1: Open UiPath Studio.

Step 2: Create the process and name it.

Step 3: Drag the **"Sequence"** activity from the activity panel and drop it in the workflow.

Step 4: Name the **"Sequence"** activity as **Sequence - 'This is a code to compare two columns in a spreadsheet'.**

Step 5: Add a "**Comment"** activity from the activity panel and write **'//Code to read an Excel, compare two columns and then write in Result column if it is a match or not.**

Step: 6: Drag and drop the **"Excel Application Scope"** activity from the activity panel and name it as **Excel Application Scope - 'This activity is used for performing any action on an excel sheet.'**

Step 7: In the **"Excel Application Scope"** activity text field specify the name of the excel file to be used in the code. In this code, the name of the file used is **"Compare Columns.xlsx"** so we will write this name in the text field.

Step 8: Change the name of "**Do"** sequence from the **"Excel Application Scope"** activity as **Do – 'To read, compare and write the data in the excel file'**.

Step 9: Right-click on the **"Do**' sequence and select **Annotation >Add Annotation - 'Read an excel file for each row, check if column 1 and column 2 are equal and write in result column if it is a match or not**'

Step 10: Select the **Excel Read Range** activity from the activities panel and drop in the sequence activity workflow.

Step 11: Change the name of **Excel "Read Range"** activity as **Read Range- 'This activity is used to read an excel sheet'** and write the name of excel sheet as mentioned in the excel file used. In this case, the sheet name used is '**Sheet 1'**.

Step 12: Create two variables from the "Variables' panel in the **Excel "Read Range"** activity as under:

Name	Variable Type	Scope	Default
DataTable1	DataTable	Do - 'To read, compare and write the data in the excel file'	

RowNumber	Int32	Do - 'To read, compare and write the data in the excel file'	

Step 13: Declare the variable **DataTable** in the **"Output"** Property of **"Read Range"** activity**.**

Step 14: Drag and drop the **"For Each Row"** activity and change the name it as – **For Each Row - 'Compare both the rows in the DataTable till the last record "**

Step 15: Declare **Row** in **ForEach** and **DataTable** in **in** section.

Step 16: Change the name of "**Body"** sequence from the **"For Each Row"** activity as **Do – 'Compare the columns (row wise)'**.

Step 17: Drag the **"IF"** activity from the activity panel and drop it in the workflow.

Step 18: Name the **"IF"** activity as **IF – 'column 1 equals column 2 then it's a match else it is not a match'.**

Step 19: Right-click on the **"If'** activity and select **Annotation >Add Annotation – If column 1 equals column 2 then write in each corresponding cell as a match, else not a match against each row**.

Step 20: Inside the "**If"** activity write the condition **"cint(row(0)) = cint(row(1))".**

Step 21: In the **Then** section of "**If"** activity add a "**Write Cell"** Excel activity and change the name from **"Write Cell"** as **Write Cell - 'Write "Match" in each cell if condition is met'**.

Step 22: In the "**Write Cell"** activity properties panel**,** write the following values in the property panel as mentioned below:

Sheet Name	***Range***	***Value***
"Sheet1'	"C"+RowNumber.ToString	"Match"

Step 23: In the **Else** section of "**If"** activity add a "**Write Cell"** Excel activity and change the name from **"Write Cell"** as **Write Cell - 'Write "Not a Match" in each cell if condition is not met'**

Step 24: In the "**Write Cell"** activity properties panel**,** write the following values in the property panel as mentioned below:

Sheet Name	*Range*	*Value*
"Sheet1'	"C"+RowNumber.ToString	"Not a Match'

Step 25: Drag and drop the "**Assign"** activity below the **"If"** activity**.**

Step 26: Change the "**Assign"** activity name **Assign - 'Increment the RowNumber variable by 1'**

Step 27: Declare the **'RowNumber'** variable in the **To** box and **'RowNumber+1"** in the value box.

10 Extracting data from a website

Objective: Code a Robot in UiPath Studio to scrape data from a website and store it in .CSV File.

Learning Outcomes
After completion of this exercise you will get familiar with the following:

- ✓ **"Sequence"** activity.
- ✓ **"Comment"** and "**Annotation".**
- ✓ **"Open Browser"** activity.
- ✓ **"Type Into"** activity.
- ✓ **"Browser Scope**' activity.
- ✓ **"Extract Data"** activity.
- ✓ **"Write CSV"** activity.

Package Prerequisites:

- **UiPath.Excel.Activities = 2.6.0**

Algorithm:

Step 1: START
Step 2: Open the URL using Open Browser Activity
Step 3: Declare variables as 'CSVFile' , 'ExtractDataTable' , 'Search_Item' , 'URL'
Step 4: Use the variables in the different activity blocks to search , find the given item
Step 5: Output the result in the write csv file activity
Step 6: STOP

Step by Step process:

Step 1: Open UiPath Studio.

Step 2: Create the process and name it.

Step 3: Drag the "**Sequence**' activity from the activity panel and drop it in the workflow.

Step 4: Name the "**Sequence**' activity as **Sequence - 'This is the code to extract data from a website in a .CSV file**.'

Step 5: Add a "Comment' activity from the activity panel and write '// **To extract data from a website in a .CSV file.'**

Step 6: Drag the "**Open Browser**' activity from the activity panel and drop it in the workflow.

Step 7: Name the "**Open Browser**' activity as **Open Browser – 'To open the specified URL in Internet Explorer.'**

Step 8: Create a variable from the "**Variables**' panel in the "**Sequence**' activity as under:

Name	Variable Type	Scope	Default
URl	String	Sequence - 'This is the code to extract data from a website in a .CSV file.'	www.amazon.in

Step 8: Declare the variable **URL** in the **"Input"** Property of "**Open Browser"** activity.

Step 9: Drag the "**Sequence**' activity from the activity panel and drop it in the workflow.

Step 10: Name the "**Sequence**' activity as **Sequence - 'This block of code will search the data of iPhone (specified product) and extract in a.CSV file**.'

Step 11: Drag the **"Type Into"** activity from the activity panel and drop it in the **"Sequence"** activity.

Step 12: Name the **"Type Into"** activity as **Type into - 'Type iPhone (specified product) in the search bar of Amazon (specified website).'**

Step 13: Create a variable from the "**Variables**' panel in the "**Type Into**' activity as under:

Name	Variable Type	Scope	Default
Search_Item	String	'This block of code will search the data of iPhone (specified product) and extract in a.CSV file.'	"iPhone'

Step 14: Declare the variable **"Search_item"** in the **"Input"** Property of **"Type Into"** activity.

Step 15: Drag the **"Click"** activity from the activity panel and drop it in the **"Sequence"** activity.

Step 16: Name the **"Click"** activity as **Click - 'To search and display the results by clicking the search button on the webpage.'**

Step 17: Drag the **"Sequence"** activity from the activity panel.

Step 18: Name the **"Sequence"** activity as **Sequence - 'To extract the result data from the website.'**

Step 19: Drag the **"Attach Browser"** activity from the activity panel and drop it in the **"Sequence"** activity.

Step 20: Name the **"Attach Browser"** activity as **Attach Browser - 'It will attach the browser which was opened using Open Browser activity.'**

Step 22: Drag the **"Extract Data"** activity from the activity panel.

Step 23: Name the **"Extract Data"** activity as **Extract Data - 'It will extract the data fields of iPhone (specified product).'**

Step 24: Create a variable from the "**Variables**' panel in the "**Extract data**' activity as under:

Name	Variable Type	Scope	Default
ExtractDataTable	DataTable	Sequence - To extract the result data from the website	New System.Data.DataTable

Step 25: Declare the variable **ExtractDataTable** in the **"Output"** Property of "**Extract Data"** activity.

Step 26: Drag the **"Write CSV"** activity from the activity panel.

Step 27: Name the **"Write CSV"** activity as **Write CSV - 'This will extract the data in the .CSV format from the website.'**

Step 28: Create a variable from the "**Variables**' panel in the "**Write CSV**' activity as under:

Name	Variable Type	Scope	Default
CSVFile	String	Sequence - 'This block of code will collect the information from the results and compile it into a.CSV file.'	"AmazonData.csv"

Step 29: Declare the variable **ExtractDataTable** in the **"Input"** Property and **CSVFile** variable in the **"FilePath"** Property of "**Write CSV"** activity.

11 Filling a webform from an excel sheet

Objective: Code a Robot in UiPath Studio to fill a Webform from the data extracted from an excel sheet.

Learning Outcomes
After completion of this exercise you will get familiar with the following:

- ✓ **"Sequence"** and **"Assign"** activity.
- ✓ **"Comment"** and "**Annotation".**
- ✓ **"Open Browser"** and **"Maximize Window"** activity.
- ✓ **"Excel Application Scope"** activity and how to set conditions.
- ✓ **"For Each Row**', **"Excel Read Range", "Type Into," "Click"** and **"Get Row Item"** activity and how to set **variables** in a code using **"Variable**' panel.

Package Prerequisites:

- 🕓 **UiPath.Excel.Activities = 2.6.0**

Algorithm:

Step 1: START
Step 2: Declare the variables as 'FirstName' , 'LastName' ,'Company' , 'Role',' Address',' Email' ,'PhoneNo'
Step 3: Add browser activity and put ""https://forms.gle/BnonGQCaaY8QGzk46"" this link inside the browser activity
Step 4: Add the Excel application scope activity and add the excel file in it and read the excel file
Step 5: Add For Each row activity to read the contents of the excel file
Step 6: Add Get Row activity to read the excel file in row format
Step 7: Add one more sequence and add Type Into activity in it to read 'FirstName' , 'LastName' ,'Company' , 'Role',' Address',' Email' ,'PhoneNo'
Step 8: Drag and drop click activity to submit the form
Step 9: Drag and drop one more click activity to submit the other form response
Step 10: STOP

Step by Step process:

Step 1: Open UiPath Studio.

Step 2: Create the process and name it.

Step 3: Drag the **"Sequence"** activity from the activity panel and drop it in the workflow.

Step 4: Name the **"Sequence"** activity as **Sequence - 'This code is to automatically fill a webform from data stored in an excel sheet.'**

Step 4: Add a "**Comment"** activity from the activity panel and write **'// Open Browser webform, Read Excel sheet row by row and then enter the data in the webform accordingly.'**

Step 5: Drag and drop the **"Open Browser"** activity from the activity panel and name it as **Open Browser – 'Opens the specified URL in the specified web browser'** and inside the URL **text field** write:

"https://forms.gle/BnonGQCaaY8QGzk46"

Step 6: Right-click on **"Open Browser"** activity select **Annotation** > **Add Annotation** '**Open the browser and maximize it.'**

Step 7: Name the "**Do"** sequence in **"Open Browser"** activity as **Do - 'Get information as per user input, enter it in the browser text boxes and display "Your record has been recorded".'**

Step 8: Drag the **"Maximize Window"** activity from the activity panel and drop it in the workflow.

Step 9: Name the **"Maximize Window"** activity as **Maximize Window – 'To maximize the browser window.'**

Step 10: Drag and Drop "**Excel Application Scope"** activity and name it as **Excel Application Scope – 'To work on the specified excel sheet.'**

Step 11: Insert the name of the excel sheet in **WorkbookPath** property of "**Excel Application Scope"** activity.

Note: In this code the name of excel file is: '**Challenge1.xlsx'**

Step 12: Name the "**Do"** sequence of "**Excel Application Scope"** activity as **Do- 'Read the excel sheet and extract the data row by row until the condition is fulfilled**.'

Step 13: Drag and Drop "**Excel Read Range"** activity and name it as **Read Range - 'To read the excel sheet.'**

Step 14: Declare the sheet Name "**Sheet1"** in the **"Input"** Property and in the Range property of "**Read Range"** activity insert **""(double quotes).**

Step 15: Create a variable through "**Variables**' panel for the **"For Read Range"** activity as under:

Name	Variable Type	Scope	Default
DataTable	DataTable	Do- 'Read the excel sheet and extract the data row by row until the condition is fulfilled.'.	

Step 16: Drag and Drop "**For Each Row"** activity and name it as **For Each Row - 'To extract the data and enter it in the webform until the condition is fulfilled.'**

Step 17: Declare the variable **DataTable** in the **"Input"** Property of "**For Each Row"** activity.

Step 19: Name the "**Body"** sequence as **Body - 'Extracting excel data & entering it in the webform'**.

Step 20: Drag and Drop **"Sequence"** activity and name it as **Sequence - 'This block of code is to extract excel data row by row.'**

Step 21: Right-click on **"Sequence"** activity select **Annotation** > **Add Annotation 'Get the values of the excel sheet, row by row and store it in the variables.'**

Step 22: Drag and Drop **"Get Row Item"** activity inside the "**Sequence"** activity.

Step 23: Name it as **Get Row Item - 'To extract the value of specified row and column.'**

Step 24: Create a variable through "**Variables**' panel for the **"Get Row Item"** activity as under:

Name	Variable Type	Scope	Default
FirstName	GenericValue	Sequence - 'This block of code is to extract excel data row by row.'	

Step 25: Declare the variable **FirstName** in the **"Output"** Property of "**Get Row Item"** activity.

Step 26: Declare the **"FirstName"** in the **"Column Name"** property of "**Get Row Item"** activity.

Step 27: Inside the **Data table row** box write the "**row**' of "**Get Row Item"** activity.

Step 28: Drag and Drop **"Get Row Item"** activity inside the "**Sequence"** activity.

Step 29: Name it as **Get Row Item - 'To extract the value of specified row and column.'**

Step 30: Create a variable through "**Variables**' panel for the **"Get Row Item"** activity as under:

Name	Variable Type	Scope	Default
LastName	GenericValue	Sequence - 'This block of code is to extract excel data row by row.'	

Step 31: Declare the variable **LastName** in the **"Output"** Property of "**Get Row Item"** activity.

Step 32: Declare the **"LastName"** in the **"Column Name"** property of "**Get Row Item"** activity.

Step 33: Inside the **Data table row** box write the "**row**' of "**Get Row Item"** activity.

Step 34: Drag and Drop **"Get Row Item"** activity inside the "**Sequence"** activity.

Step 35: Name it as **Get Row Item - 'To extract the value of specified row and column.'**

Step 36: Create a variable through "**Variables**' panel for the **"Get Row Item"** activity as under:

Name	Variable Type	Scope	Default
Company	GenericValue	Sequence - 'This block of code is to extract excel data row by row.'	

Step 37: Declare the variable **Company** in the **"Output"** Property of "**Get Row Item"** activity.

Step 38: Declare the **"Company Name"** in the **"Column Name"** property of "**Get Row Item"** activity.

Step 39: Inside the **Data table row** box write the "**row**' of "**Get Row Item"** activity.

Step 40: Drag and Drop **"Get Row Item"** activity inside the "**Sequence"** activity.

Step 41: Name it as **Get Row Item - 'To extract the value of specified row and column.'**

Step 42: Create a variable through "**Variables**' panel for the **"Get Row Item"** activity as under:

Name	Variable Type	Scope	Default
Role	GenericValue	Sequence - 'This block of code is to extract excel data row by row.'	

Step 43: Declare the variable **Role** in the **"Output"** Property of "**Get Row Item"** activity.

Step 44: Declare the **"Role in Company"** in the **"Column Name"** property of "**Get Row Item"** activity.

Step 45: Inside the **Data table row** box write the "**row**' of "**Get Row Item"** activity.

Step 46: Drag and Drop **"Get Row Item"** activity inside the "**Sequence"** activity.

Step 47: Name it as **Get Row Item - 'To extract the value of specified row and column.'**

Step 48: Create a variable through "**Variables**' panel for the **"Get Row Item"** activity as under:

Name	Variable Type	Scope	Default
Address	GenericValue	Sequence - 'This block of code is to extract excel data row by row.'	

Step 49: Declare the variable **Address** in the **"Output"** Property of "**Get Row Item"** activity.

Step 50: Declare the **"Address"** in the **"Column Name"** property of "**Get Row Item"** activity.

Step 51: Inside the **Data table row** box write the "**row**' of "**Get Row Item"** activity.

Step 52: Drag and Drop **"Get Row Item"** activity inside the "**Sequence"** activity.

Step 53: Name it as **Get Row Item - 'To extract the value of specified row and column.'**

Step 54: Create a variable through "**Variables**' panel for the **"Get Row Item"** activity as under:

Name	Variable Type	Scope	Default
Email	GenericValue	Sequence - 'This block of code is to extract excel data row by row.'	

Step 55: Declare the variable **Email** in the **"Output"** Property of "**Get Row Item"** activity.

Step 55: Declare the **"Email"** in the **"Column Name"** property of "**Get Row Item"** activity.

Step 56: Inside the **Data table row** box write the "**row**' of "**Get Row Item"** activity.

Step 57: Drag and Drop **"Get Row Item"** activity inside the "**Sequence"** activity.

Step 58: Name it as **Get Row Item - 'To extract the value of specified row and column.'**

Step 59: Create a variable through "**Variables**' panel for the **"Get Row Item"** activity as under:

Name	Variable Type	Scope	Default
PhoneNo	GenericValue	Sequence - 'This block of code is to extract excel data row by row.'	

Step 60: Declare the variable **PhoneNo** in the **"Output"** Property of "**Get Row Item"** activity.

Step 61: Declare the **"Phone Number"** in the **"Column Name"** property of "**Get Row Item"** activity.

Note: From step 21 to 71, we have been created and declared **7 variables for 7 "Get Row Items".**

Step 62: Inside the **Data table row** box write the "**row**' of "**Get Row Item"** activity.

Step 63: Drag and Drop **"Sequence"** activity inside the "**Body" section of "For each row."**

Step 64: Name it as **Sequence - 'This block of code will enter the extracted data in the webform.'**

Step 65: Right-click on **"Sequence"** activity select **Annotation > Add Annotation 'Pick the variable and enter it in the required text box of webform.'**

Step 66: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 67: Name the **"Type Into"** activity as **Type Into - 'To insert the First Name in the Text Box.'**

Step 68: Use the **FirstName** variable in the text box of **"Type Into"** activity.

Step 69: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 70: Name the **"Type Into"** activity as **Type Into - 'To insert the Last Name in the Text Box.'**

Step 71: Use the **LastName** variable in the text box of **"Type Into"** activity.

Step 72: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 73: Name the **"Type Into"** activity as **Type Into - 'To insert the Company Name in the Text Box.'**

Step 74: Use the **Company** variable in the text box of **"Type Into"** activity.

Step 75: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 76: Name the **"Type Into"** activity as **Type Into - 'To insert the role in the Text Box.'**

Step 77: Use the **Role** variable in the text box of **"Type Into"** activity.

Step 78: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 79: Name the **"Type Into"** activity as **Type Into - 'To insert the Address in the Text Box.'**

Step 80: Use the **Address** variable in the text box of **"Type Into"** activity.

Step 81: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 82: Name the **"Type Into"** activity as **Type Into - 'To insert the Email in the Text Box.'**

Step 83: Use the **Email** variable in the text box of **"Type Into"** activity.

Step 84: Drag the **"Type Into"** activity from the activity panel and drop it in the "**Sequence"** activity.

Step 85: Name the **"Type Into"** activity as **Type Into - 'To insert the Phone Number in the Text Box.'**

Step 86: Use the **PhoneNo** variable in the text box of **"Type Into"** activity.

Step 87: Drag and drop the **"Click"** activity.

Step 88: Name it as **Click – 'to click submit button to fill the details.'**

Step 89: Drag and drop the **"Click"** activity.

Step 90: Name it as **Click – 'to submit another response'**

17 Checking data mismatch using “Try catch” mechanism

Objective: Code a Robot to build a data table and then fill the data from the .CSV file. Once filled check for the mismatching columns using the try catch mechanism.

Learning Outcomes
After completion of this exercise you will get familiar with the following:

- ✓ **“Sequence”** activity.
- ✓ **“Try Catch”** activity.
- ✓ **“Write CSV”** activity.

Package Prerequisites:

- **UiPath.Excel.Activities = 2.6.2**

Algorithm

Step 1: START
Step 2: Declare the variables as 'Name' , 'Age' ,'DataTable1'.
Step 3: Add **try catch** block inside the sequence and input name and wrong format age from the user.
Step 4: Add Build Data Table Activity for building table for name and age add data row activity.
Step 5: Display message as "Exception : you added an invalid data" in the exception activity panel.
Step 6: Write the Data into CSV file.
Step 7: STOP

Step by Step process:

Step 1: Open UiPath Studio.

Step 2: Create the process and name it.

Step 3: Drag the **"Sequence"** Activity from the activity panel and drop it in the workflow.

Step 4: Name the **Sequence – 'This is a code to build a data table and then fill the data from the .CSV file. Once filled check for the mismatching columns using the try catch mechanism.'**

Step 5: Drag the **"Try Catch"** activity from the activity panel.

Step 6: Name the **"Try Catch"** activity as: **Try catch – 'It catches the exception and continue the workflow.'**

Step 7: Drag the **"Sequence"** activity from the activity panel inside the **'Try'** block.

Step 8: Name the **"Sequence"** activity as: **Sequence – 'To get the input and enter the data into datatable.'**

Step 9: Drag and Drop "**Input Dialog"** activity and name it as **Input Dialog - 'Ask user to enter the name as input.'** and write the values as under:

Title	Label
Name box	"Please enter your name:'

Step 10: Create a variable through "**Variables**' panel for the **Input Dialog - 'Ask user to enter the name as input.'** activity as under:

Name	Variable Type	Scope	Default
Name	String	Sequence - 'This is a code to build a data table and then fill the data from the .CSV file. Once filled check for the mismatching columns using the try catch mechanism.'	

Step 11: Declare the variable **Name** in the **"Output"** Property of **Input Dialog - 'Ask user to enter the name as input.'**

Step 12: Drag and Drop "**Input Dialog"** activity and name it as **Input Dialog - 'Ask user to enter the age in wrong format.'** and write the values as under:

Title	Label
Age box	"Please Enter your age in wrong format i.e. Alphabets"

Step 13: Create a variable through "**Variables**' panel for the **Input Dialog - 'Ask user to enter the age in wrong format.'** activity as under:

Name	Variable Type	Scope	Default
Age	String	Sequence - 'This is a code to build a data table and then fill the data from the .CSV file. Once filled check for the mismatching columns using the try catch mechanism.'	

Step 14: Declare the variable **Age** in the **"Output"** Property of **Input Dialog - 'Ask user to enter the age in wrong format.'**

Step 15: Drag the **"Build Data Table"** activity from the activity panel.

Step 16: Name as the **"Build Data Table"** activity as **Build Data Table - 'To create a data table.'**

Step 17: Create a variable through "**Variables**' panel for the "**Build Data Table"** activity **as under:**

Name	Variable Type	Scope	Default
DataTable1	DataTable	Sequence - 'This is a code to build a data table and then fill the data from the .CSV file. Once filled check for the mismatching columns using the try catch mechanism.'	

Step 18: Declare the variable **DataTable1** in the **"Output"** Property of **"Build Data Table"** activity**.**

Step 19: Drag the **"Add Data Row"** from the activity panel.

Step 20: Name as the **"Add Data Row"** activity as **Add Data Row - 'To add a row in the datatable.'**

Step 21: Select "**System.Exception"** in the exception box property.

Step 22: Drag the **"Write CSV"** activity from the activity panel.

Step 23: Name the **"Write CSV"** activity as **Write CSV - 'To write the input data into CSV file.'**

Step 24: Declare the variable **DataTable1** in the **"Input"** Property and **Untitled.csv** variable in the **"FilePath"** Property of "**Write CSV"** activity.

Step 25: Drag the **"Sequence"** activity from the activity panel inside the **'Catches'** block.

Step 26: Name as the **"Sequence"** activity as **Sequence - 'To catch the exception and print in the message box.'**

Step 27: Drag the **"Message Box"** activity from the activity panel.

Step 28: Name as the **"Message Box"** activity as: **Message Box - 'To print the exception message.'**

Step 29: In the **Text** property of "**Message Box**' write **"Exception: you added an invalid data in the datatable"**

Step 30: Drag the **"Message Box"** activity from the activity panel.

Step 31: Name as the **"Message Box"** activity as: **Message Box - 'To print the exception message.'**

Step 32: In the **Text** property of "**Message Box**' write **"Exception: Cannot write data in csv as value entered is wrong"**