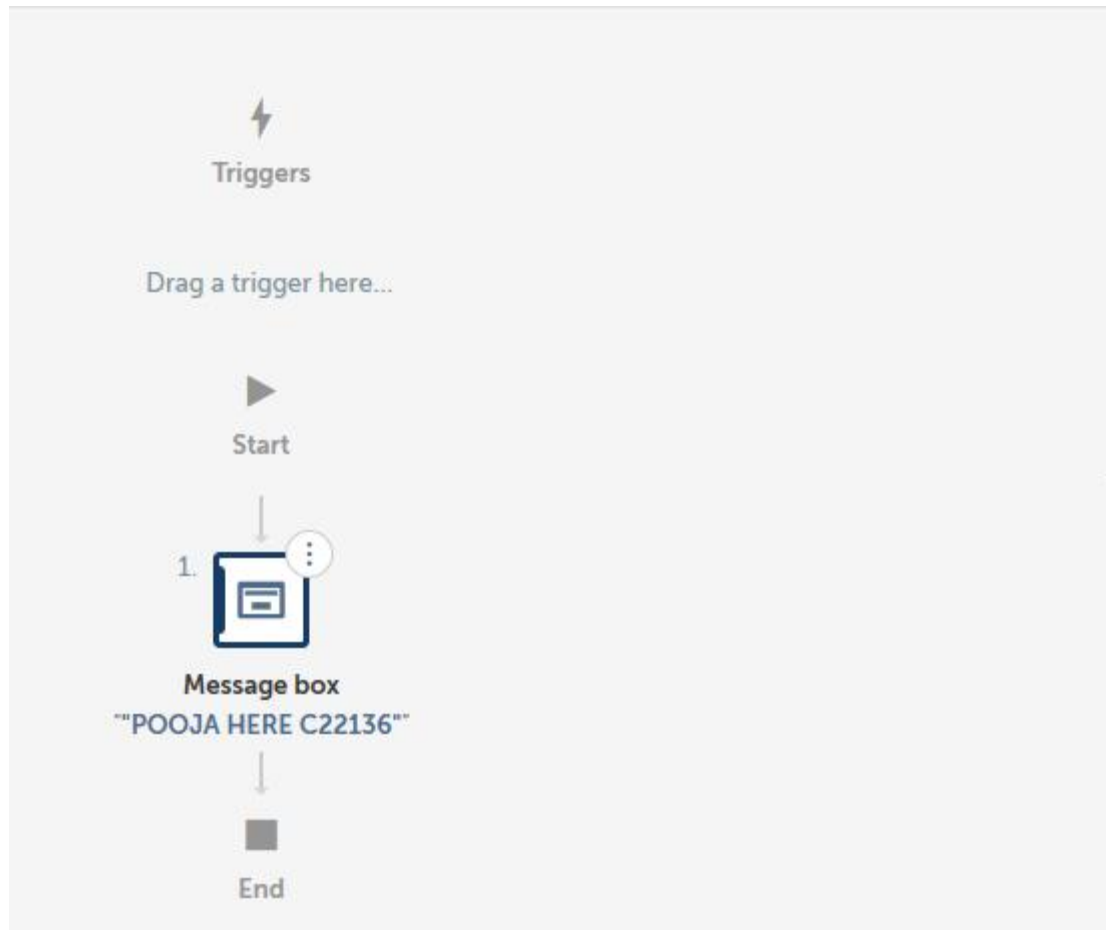
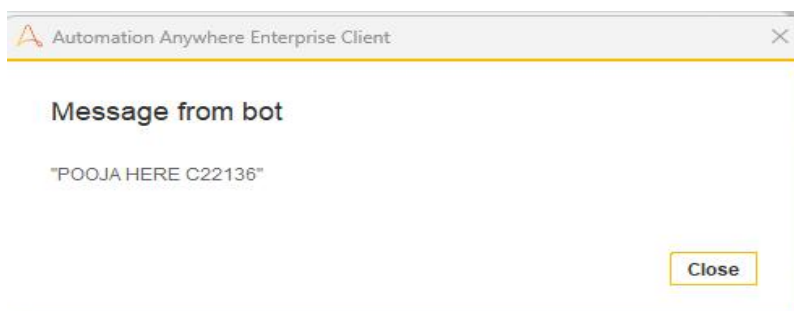
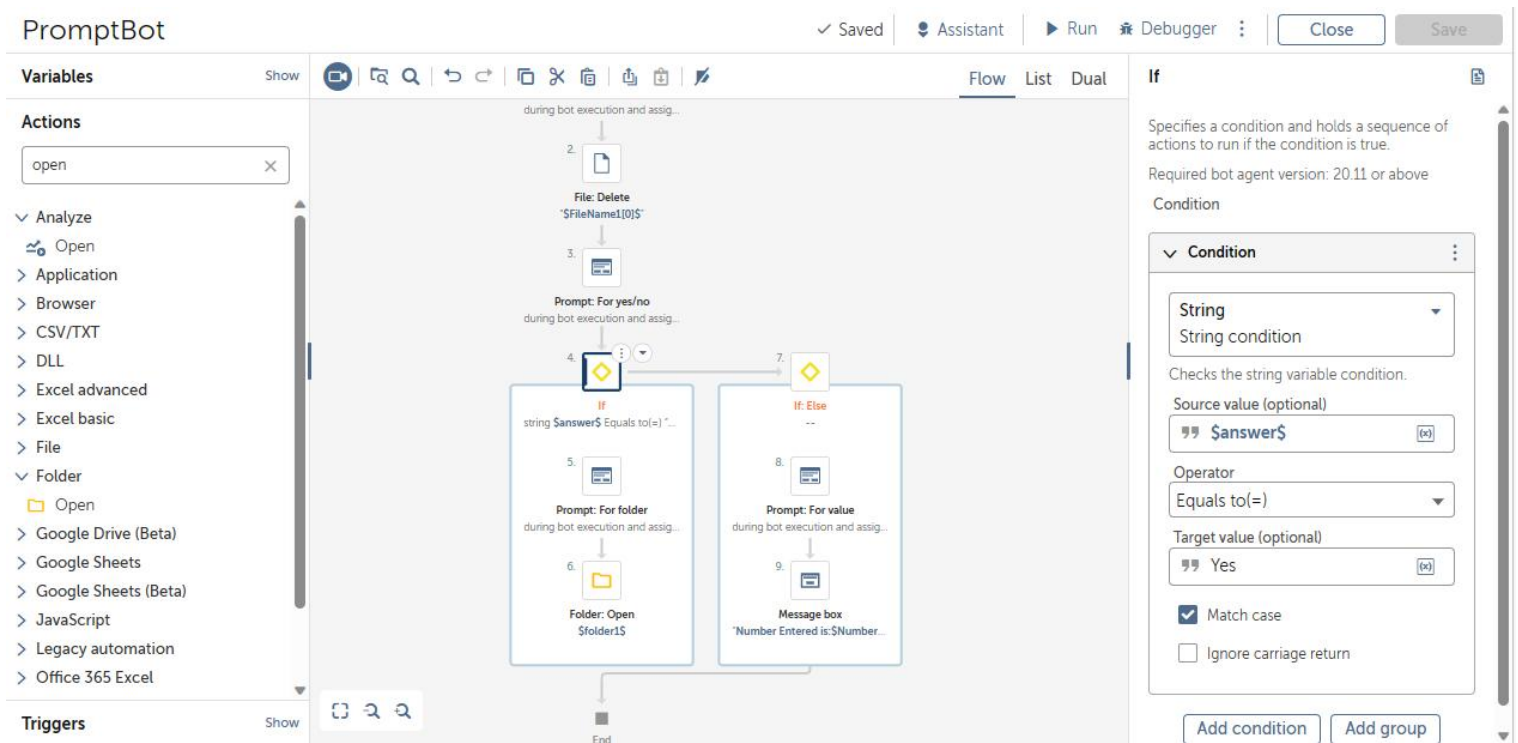
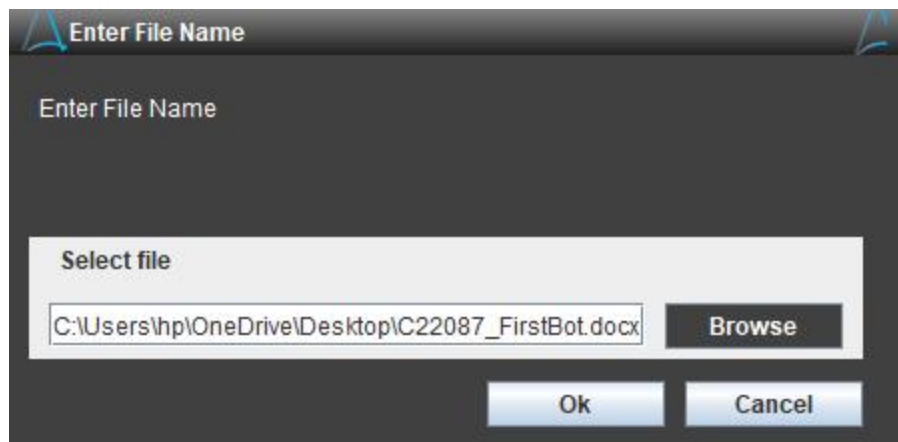
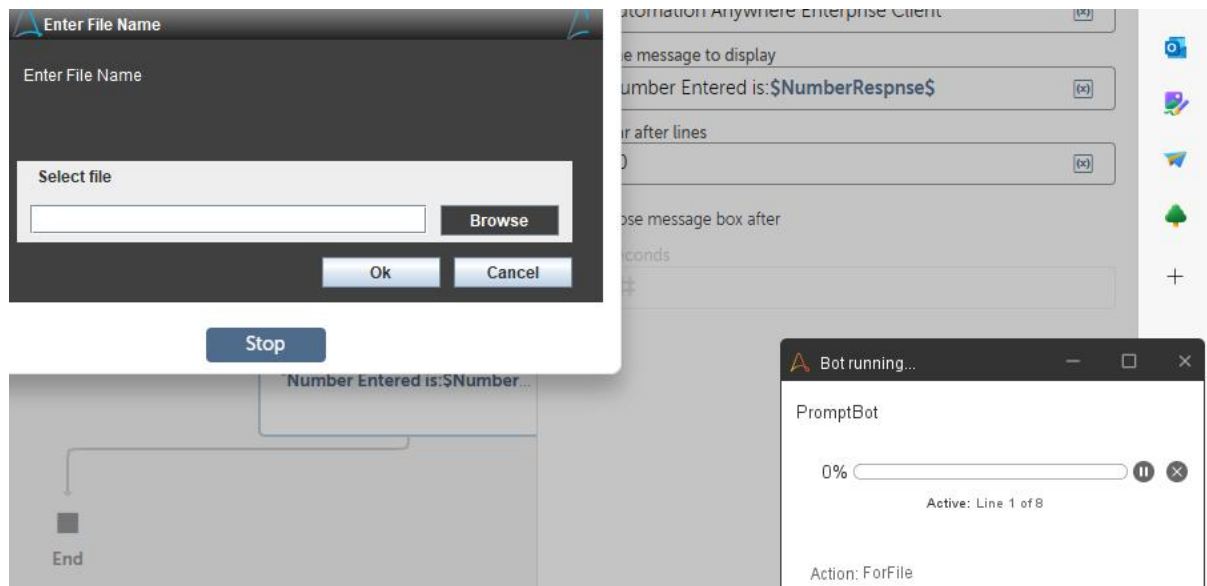
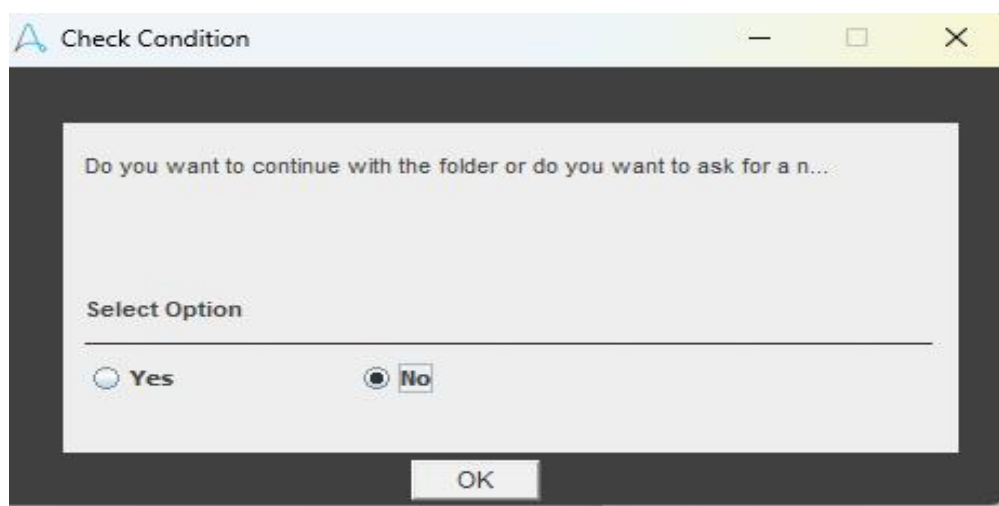


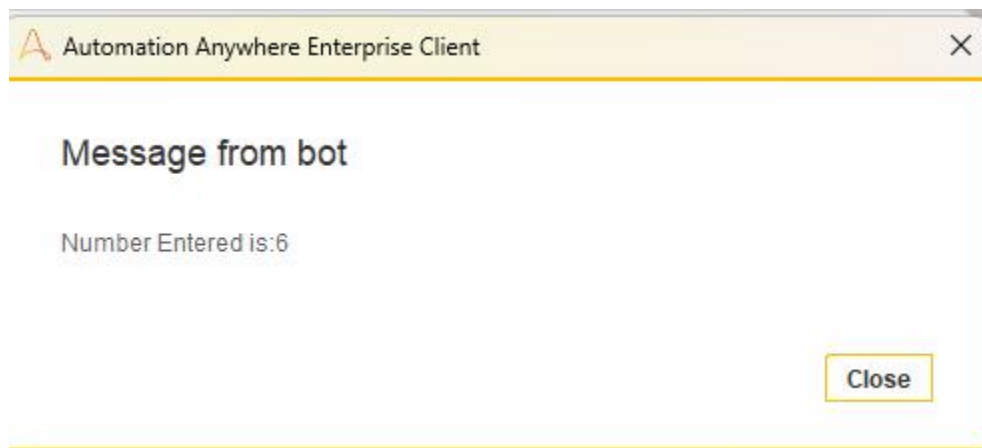
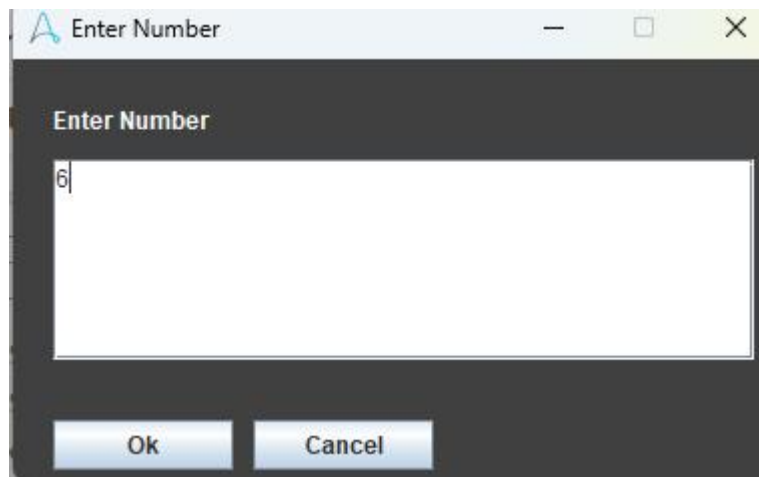
Category-1**AUTOMATION ANYWHERE(AA):*****1) Demonstrate message-box:*****OUTPUT:**

2) Demonstrate prompts:***Screenshot of Bot:******OUTPUT:***

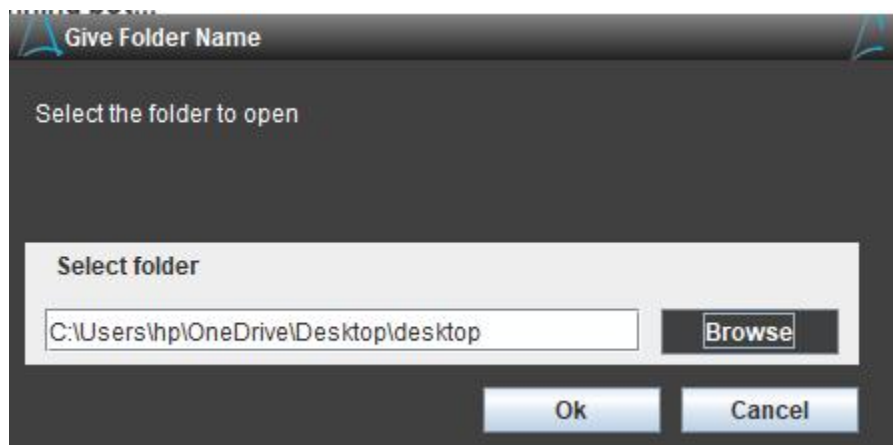
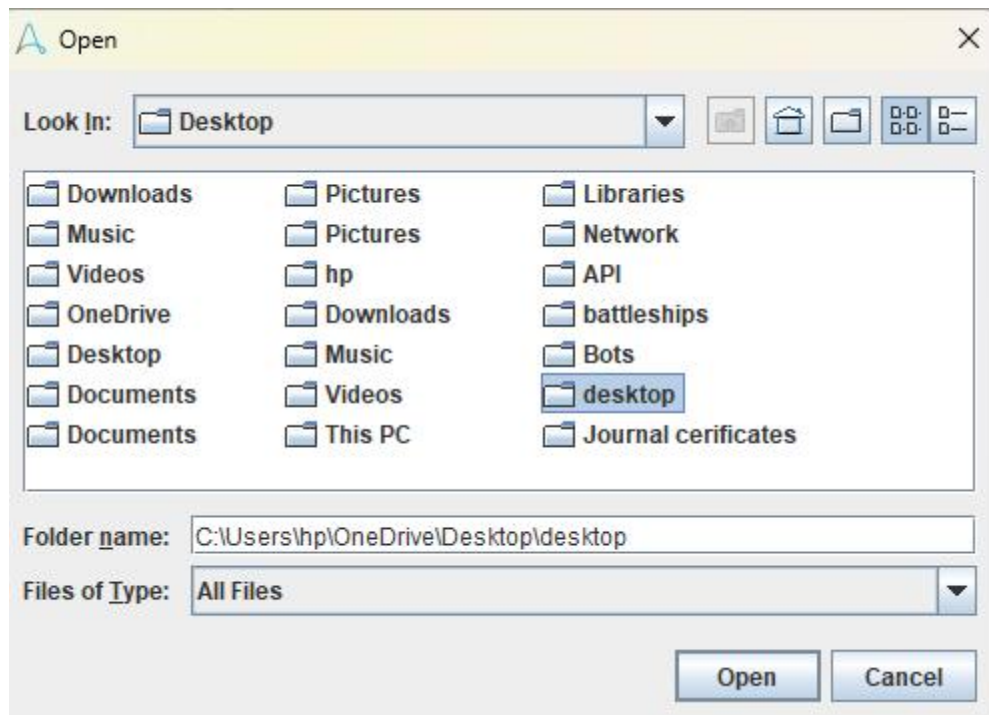


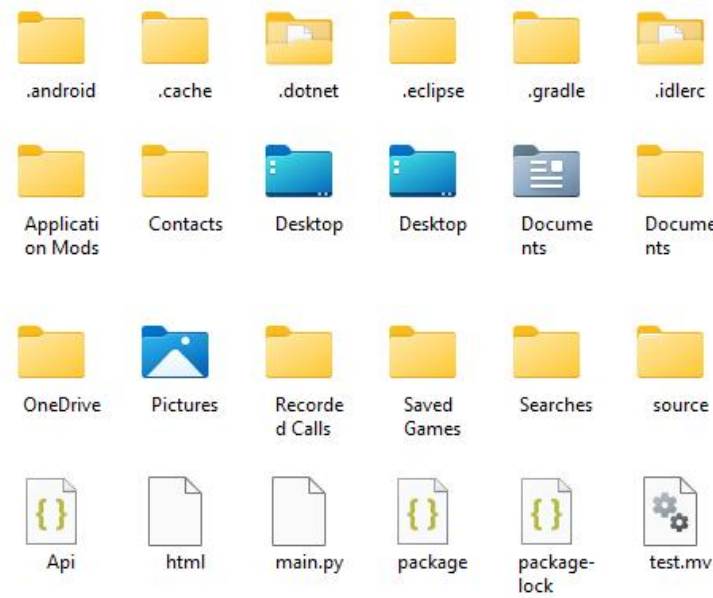
IF IT IS A NO:





IF IT IS A YES:





Triggers

Drag a trigger here...

Start

1.



Loop
: 5 times

2.



Message box
\$myListVariable[\$Counter]\$

End

Runs a sequence of actions for condition is met.

Required bot agent version: 20.

Label (optional)



Loop Type

☒ Iterator

Iterator

Loop

For n times

Iterator for n times

times

5

Assign the current value

Counter

☐ While

Condition

Condition

Add condition

Check the condition

3) Demonstrate list variables:

Screenshot Of Bot:

Edit variable

Type

☒ List

Name

myListVariable

Maximum = 50 characters

☐ Use as input

☐ Use as output

☐ Constant (read-only)

Default value (optional)

Value at 0

Value at 1

Apple

Value at 2

Mango

Value at 3

Banana

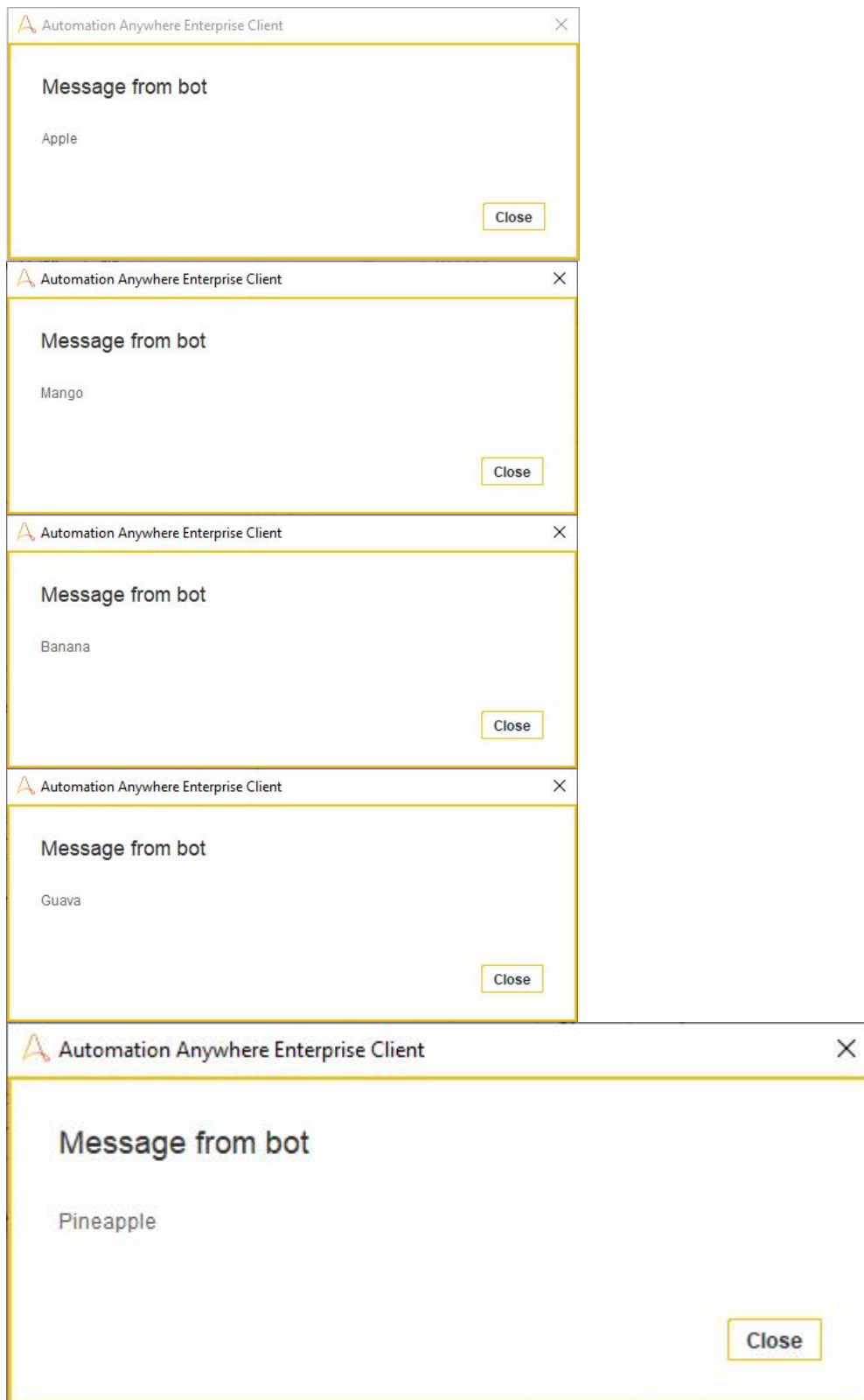
Value at 4

Guava

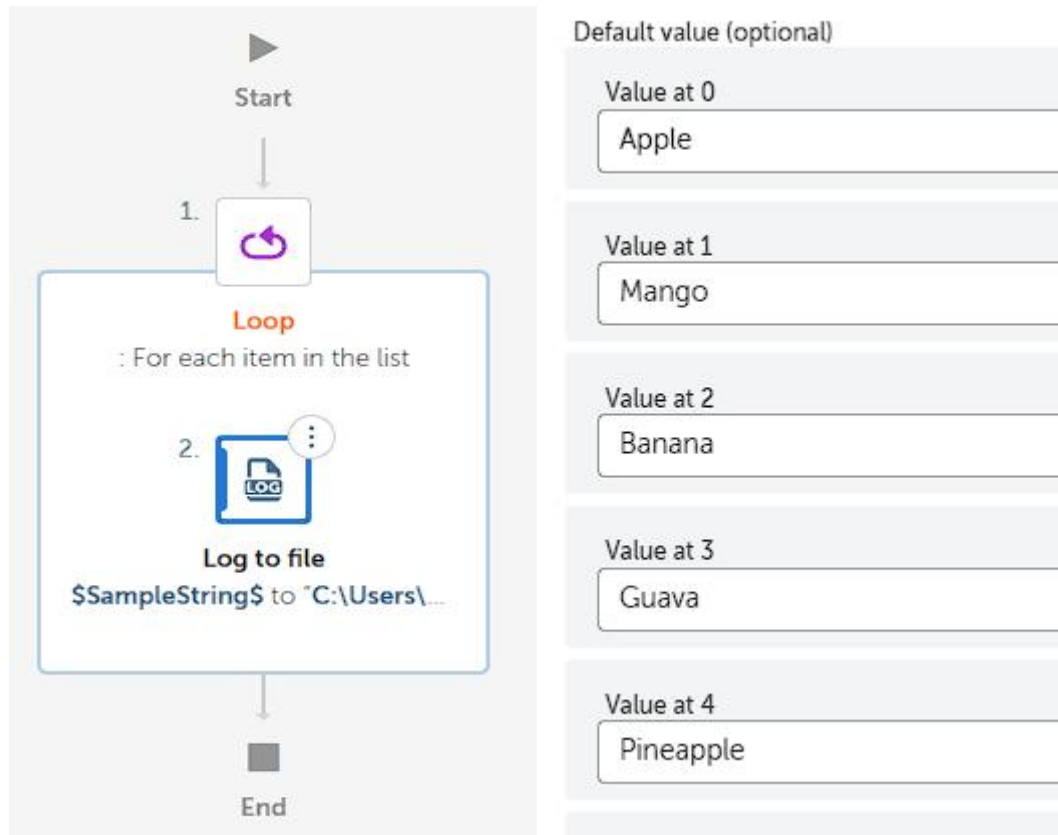
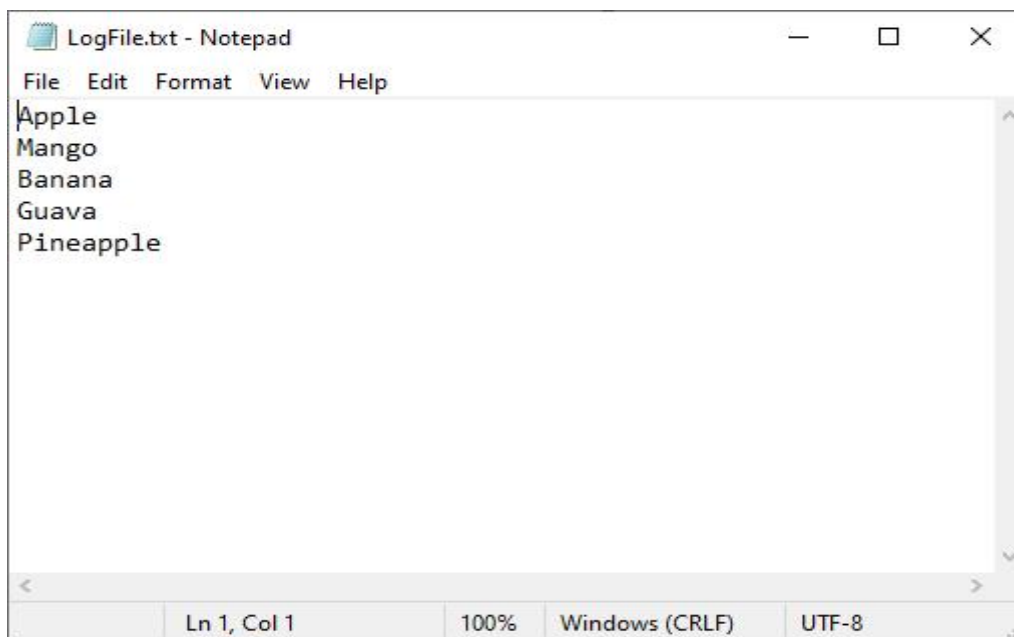
Value at 5

Pineapple

OUTPUT:



***4) Demonstrate logfile:
Screenshot of Bot:***


***OUTPUT:******5) Demonstrate keystroke:******Screenshot of Bot:***

required bot agent version: 21.200 or above

Triggers

Drag a trigger here...

Start

1. 

Simulate keystrokes

"Hello World!!!!!!!!!!!![RETUR..."

End

Select window

Browser Application Variable

☐ \$Window1\$ (x)

Window title

☐ BotsToSubmit.txt - Notepad

☒ Case sensitive

Window application path

C:\Windows\System32\notepad.exe

☐ Resize window
May improve bot accuracy

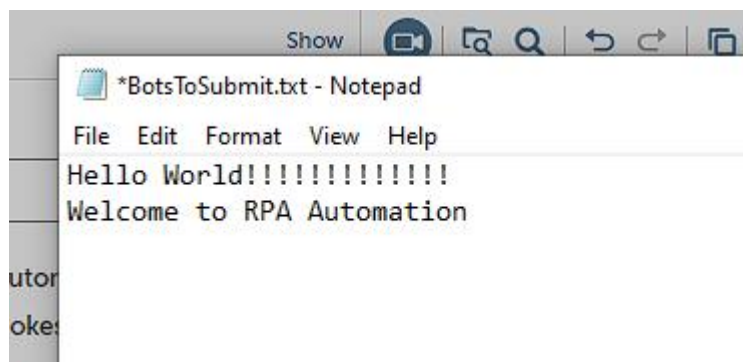
Keystrokes

☒ Enter keystrokes here or use the on-screen keyboard

☐ Select a credential

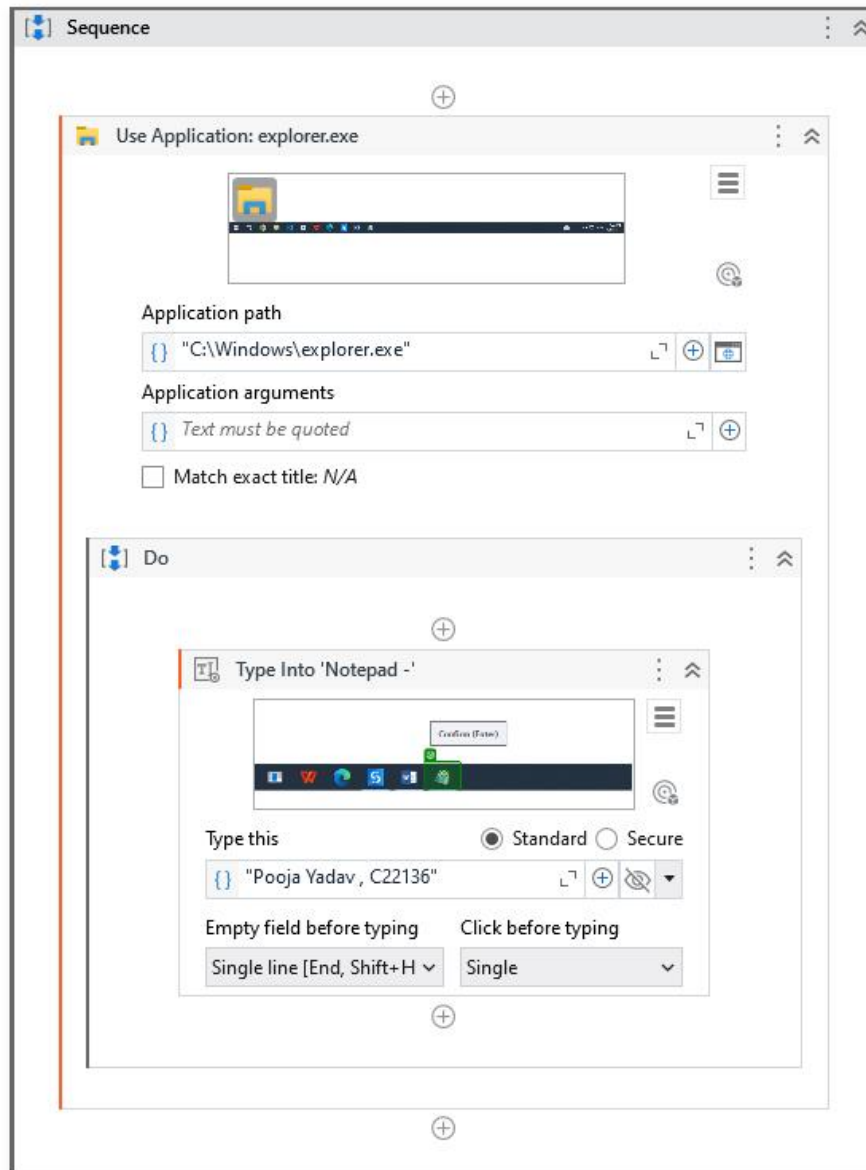
Credential Variable

“ Hello World!!!!!!!!!!!! [RETURN] Welcome to F

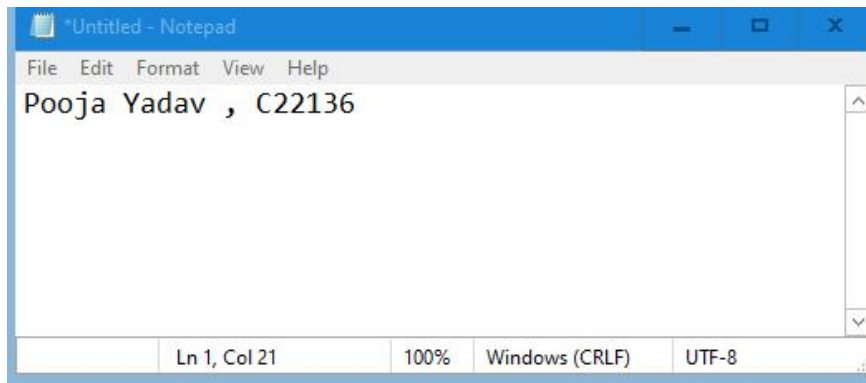
OUTPUT:

UI-Path:

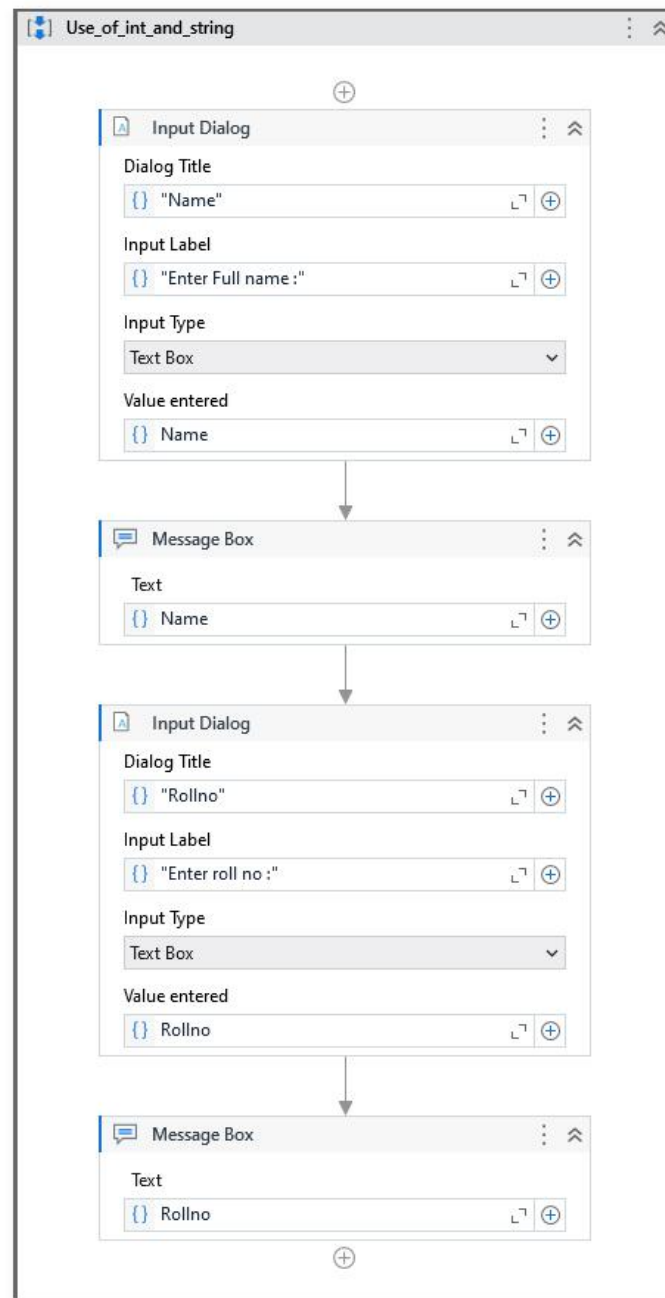
- 1) Demonstrate basic recording in Bot:
Screenshot of Bot:***




OUTPUT:



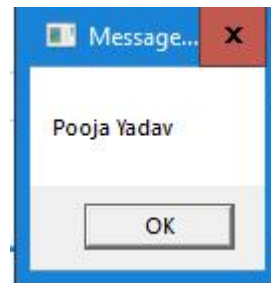
2) Demonstrate declaration and use of int and String Variables.

Screenshot of Bot:

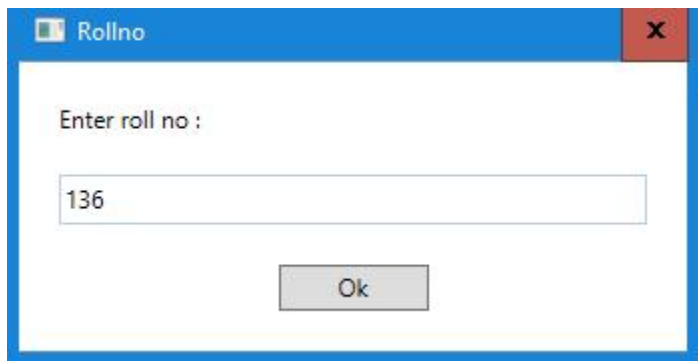
Name	Variable type	Scope	Default
Name	String	Use_of_int_and_st...	Enter a VB expression
Rollno	Int32	Use_of_int_and_st...	Enter a VB expression

OUTPUT:

A screenshot of a Windows-style dialog box titled "Name". It has a blue title bar with a close button (X) on the right. The main area is white and contains the text "Enter Full name :". Below this text is a text input field containing the text "Pooja Yadav". At the bottom center of the dialog is a button labeled "Ok".



A screenshot of a Windows-style message box titled "Message...". It has a blue title bar with a close button (X) on the right. The main area is white and displays the text "Pooja Yadav". At the bottom center of the dialog is a button labeled "OK".

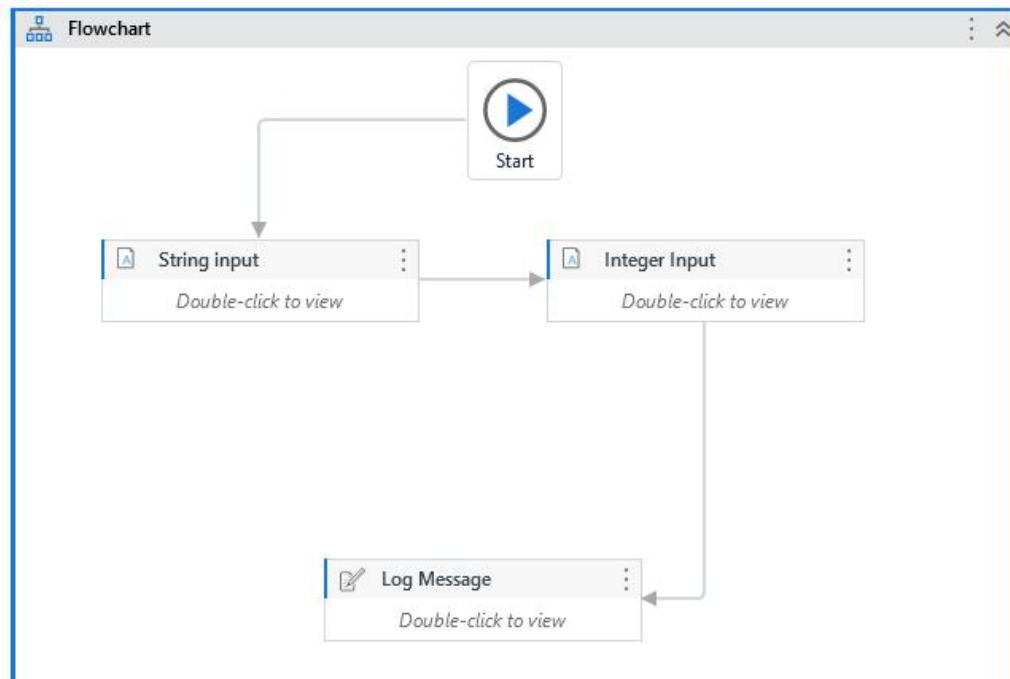


A screenshot of a Windows-style dialog box titled "Rollno". It has a blue title bar with a close button (X) on the right. The main area is white and contains the text "Enter roll no :". Below this text is a text input field containing the text "136". At the bottom center of the dialog is a button labeled "Ok".



A screenshot of a Windows-style message box titled "Message...". It has a blue title bar with a close button (X) on the right. The main area is white and displays the text "136". At the bottom center of the dialog is a button labeled "OK".

**3) Demonstrate Flowchart:
Screenshot of Bot:**

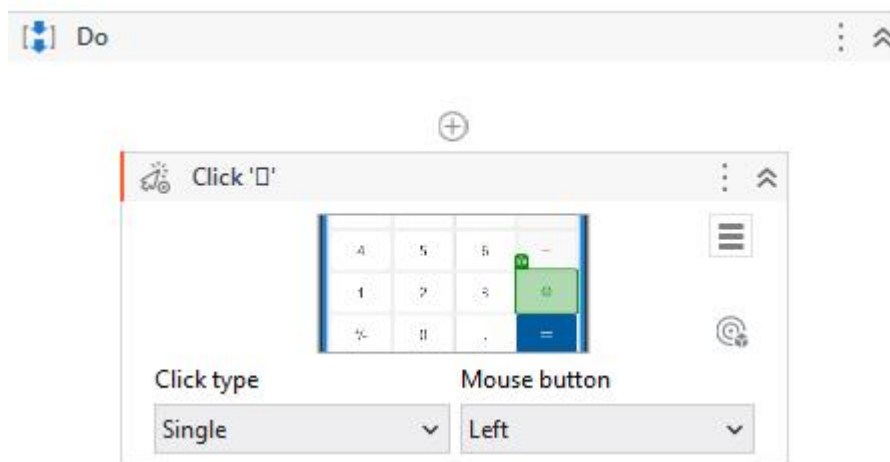
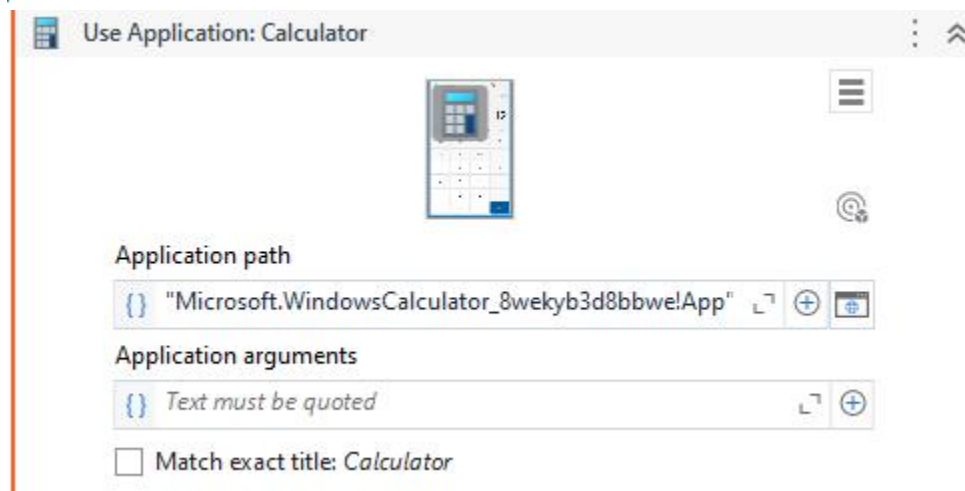
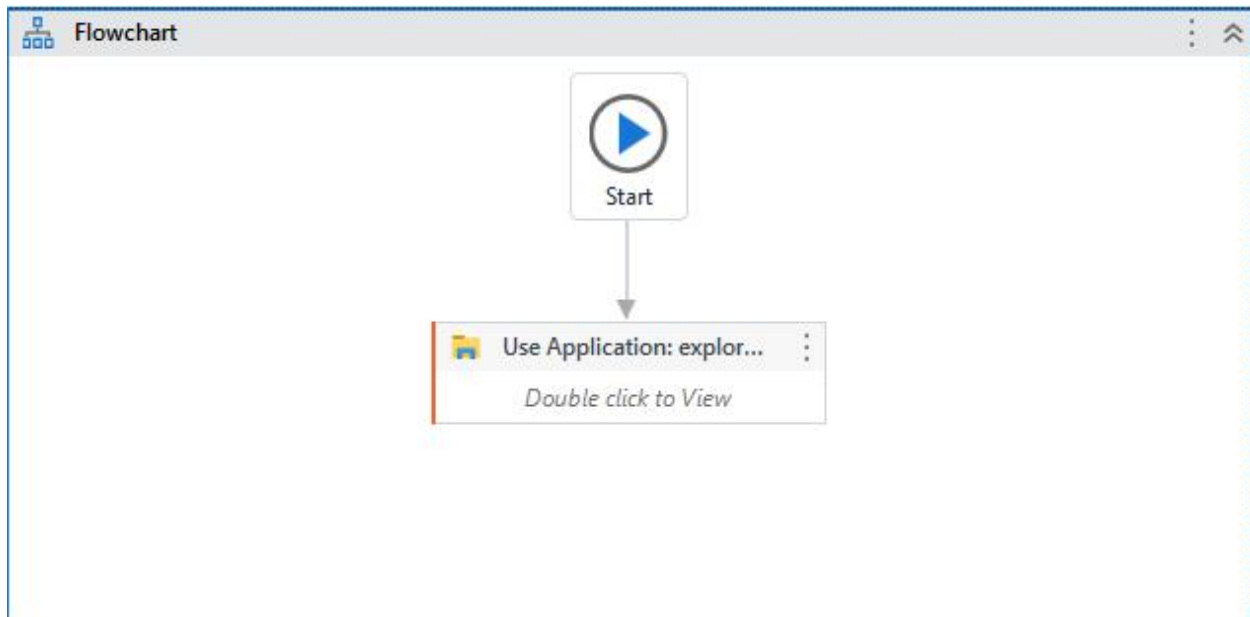


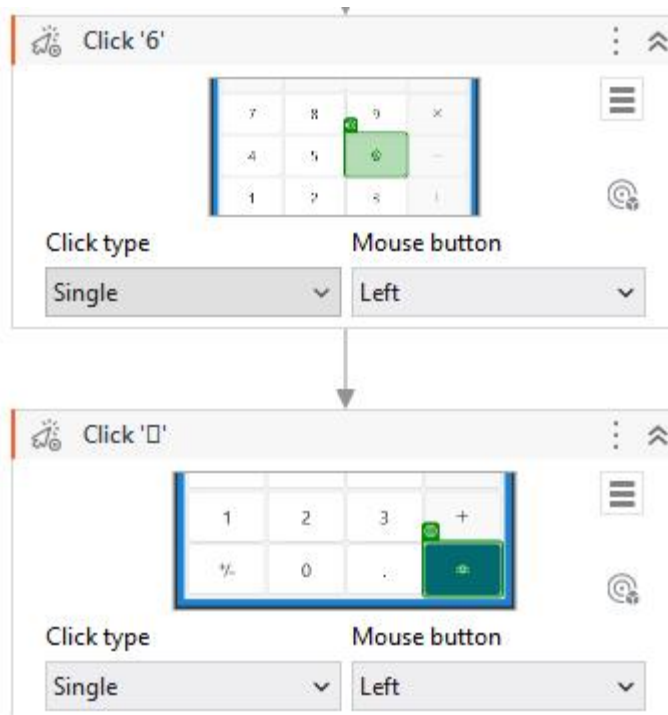
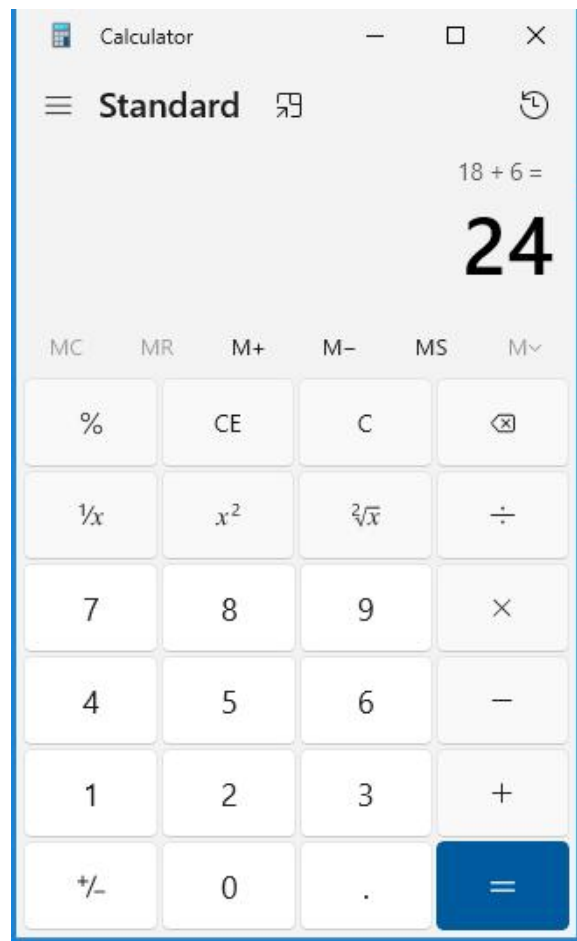
OUTPUT:

- Lab1_24March execution started
- Pooja Yadav136
- Lab1_24March execution ended in: 00:03:10

The image shows two side-by-side windows. The left window is titled "name" and contains the text "Enter Full name" above a text input field. The field contains the text "Pooja Yadav". Below the field is an "Ok" button. The right window is titled "rollno" and contains the text "Enter the roll no" above a text input field. The field contains the text "136". Below the field is an "Ok" button.

**4) Demonstrate Flowchart with Recording:
Screenshot of Bot:**

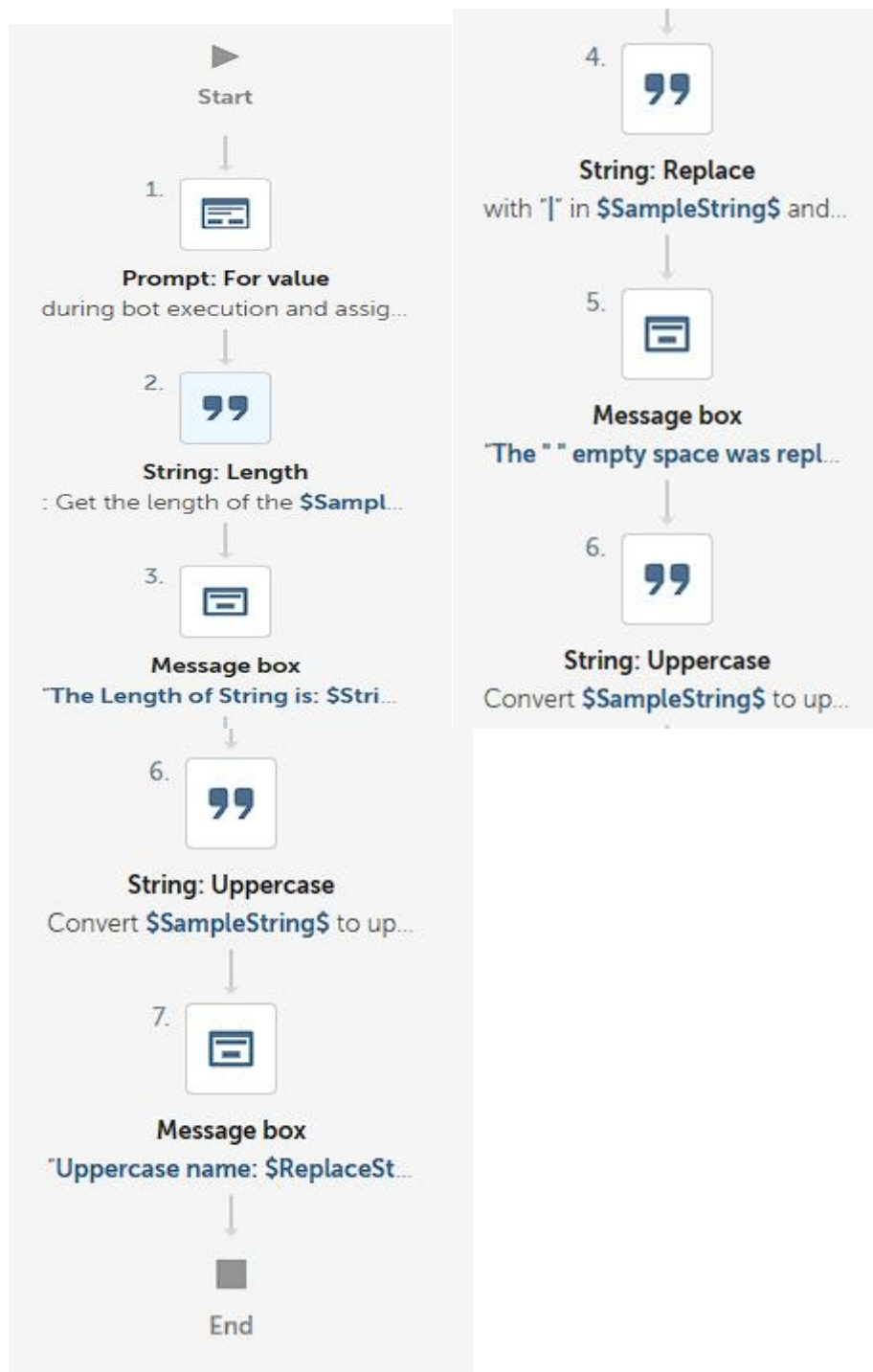


**OUTPUT:**

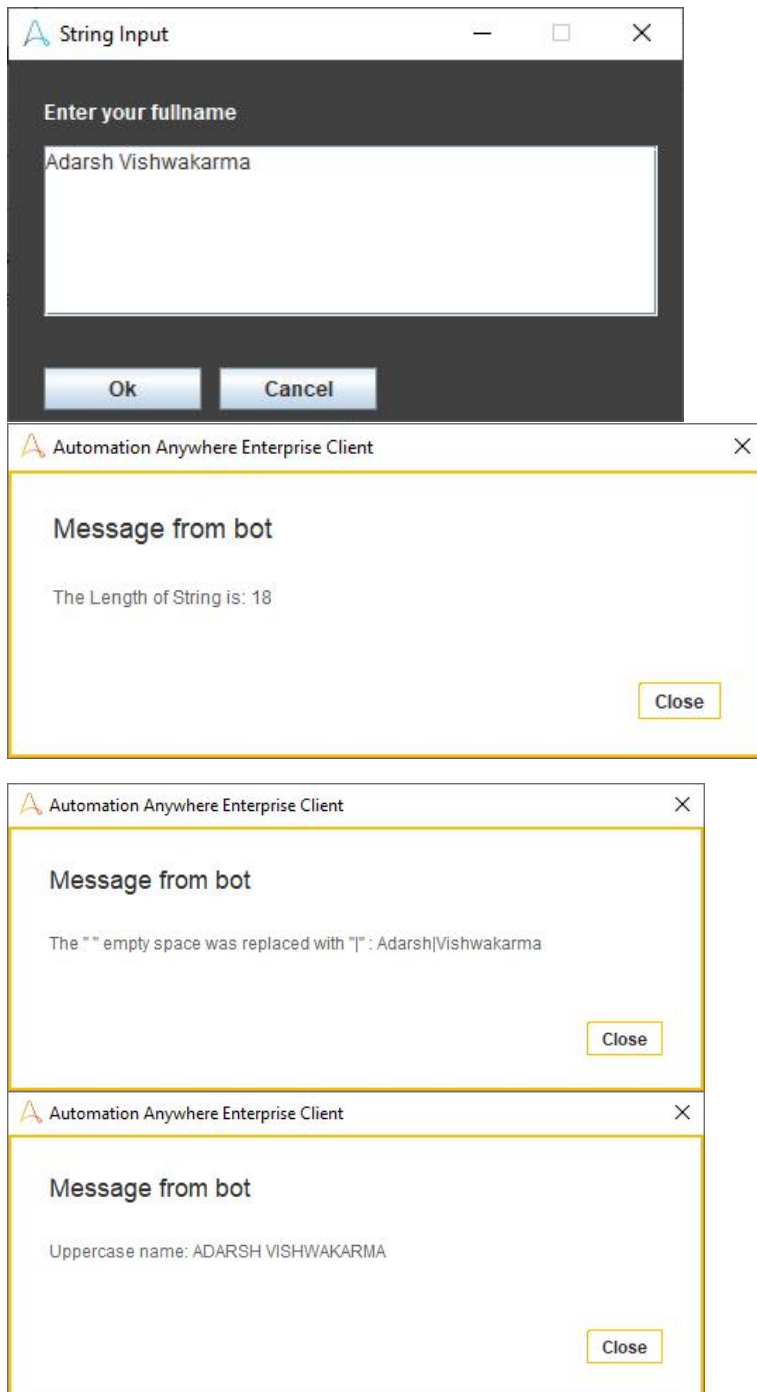
CATEGORY-2

AUTOMATION ANYWHERE(AA):

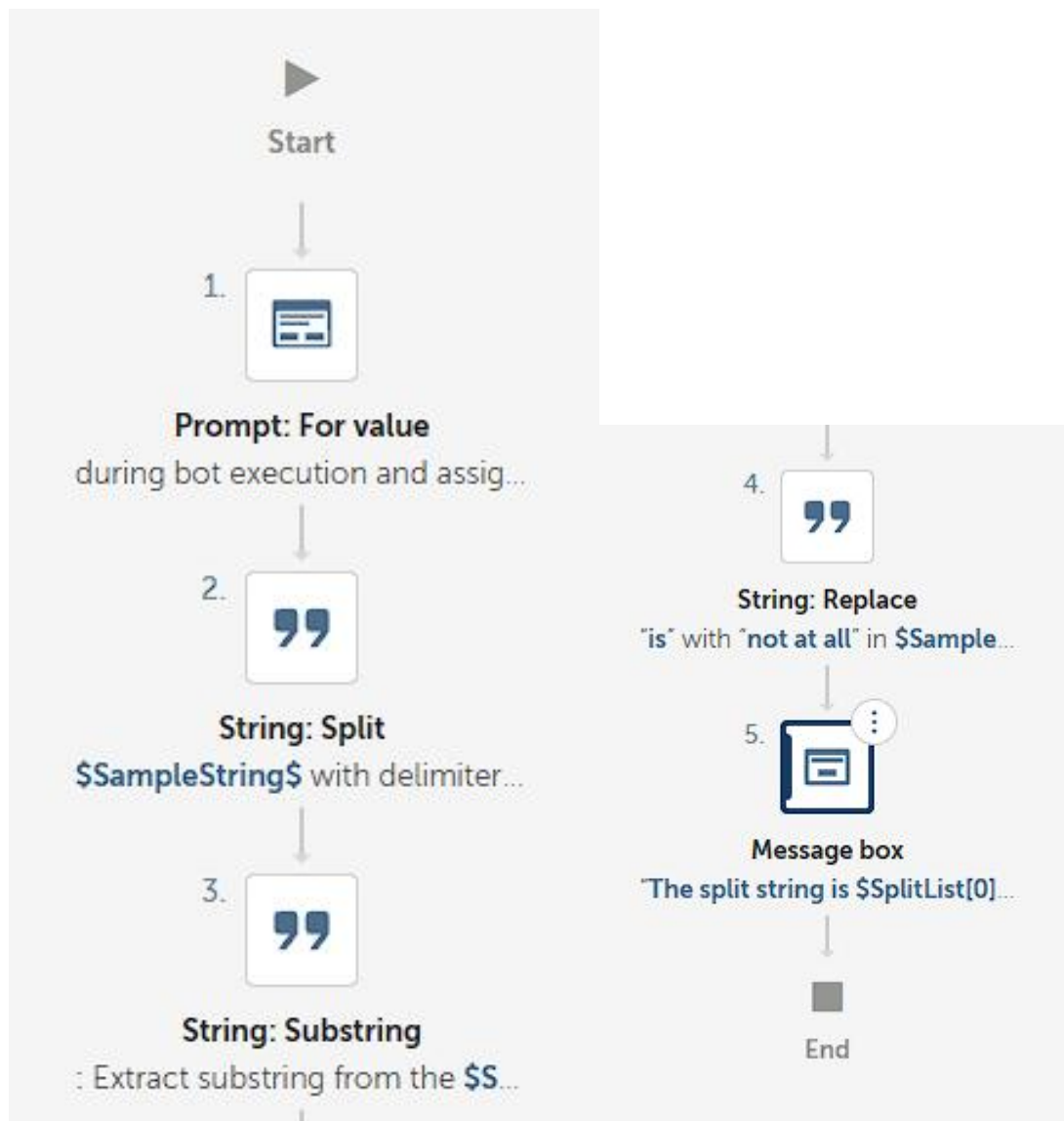
1) Bot to demonstrate string functions:

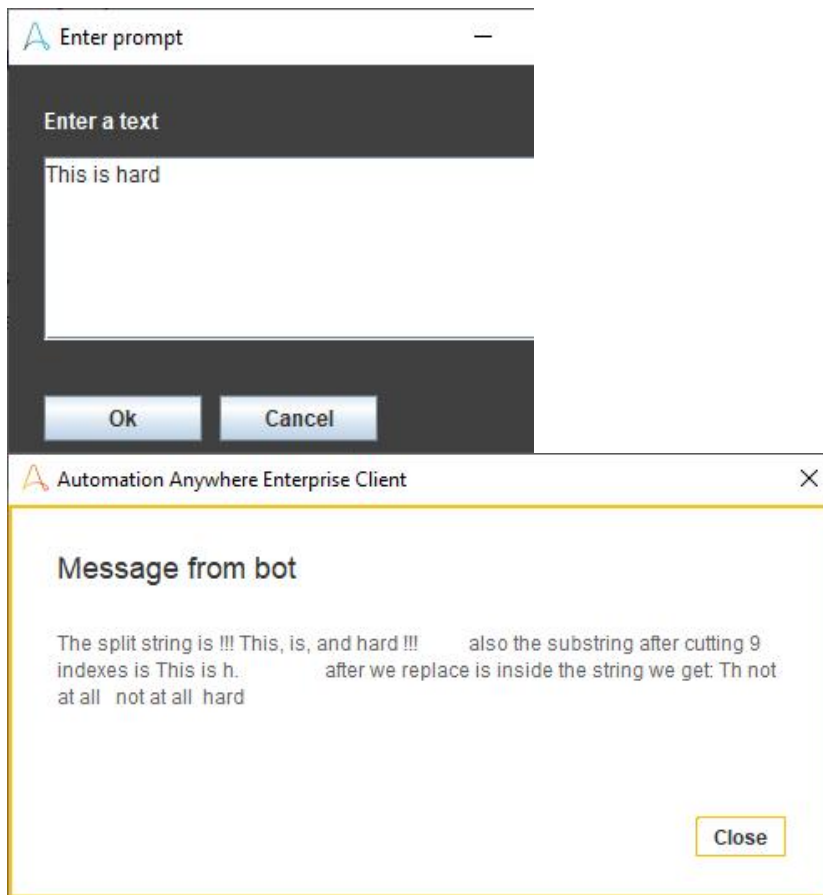


OUTPUT:

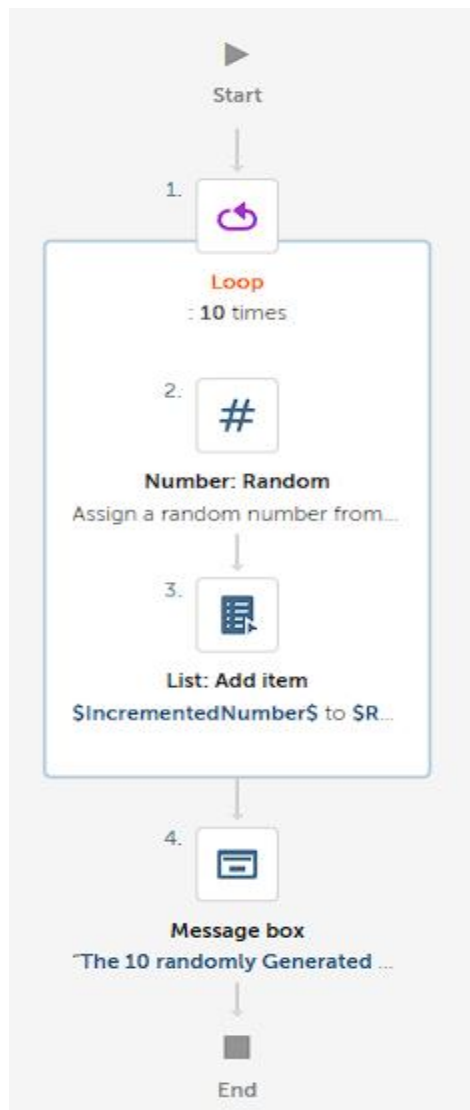


2) Bot for string operations: split, extract a substring, replace.

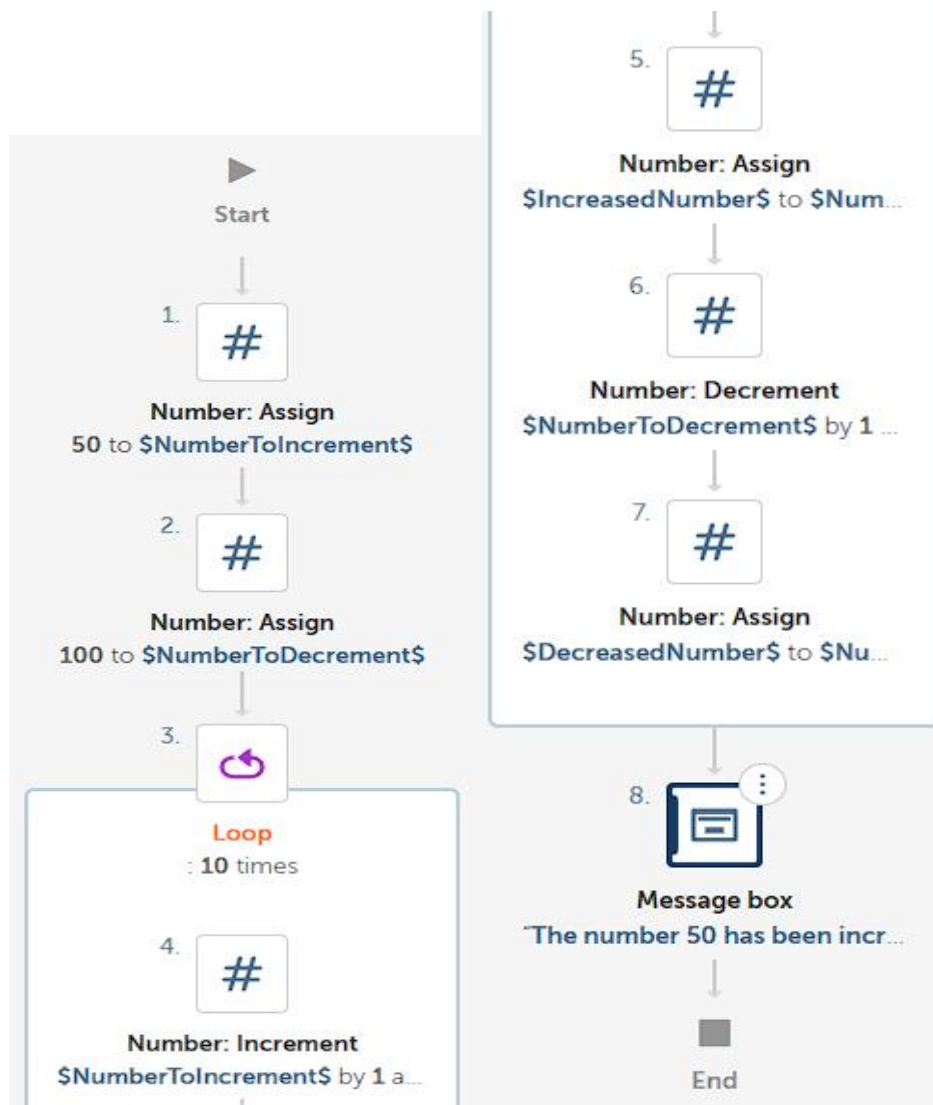
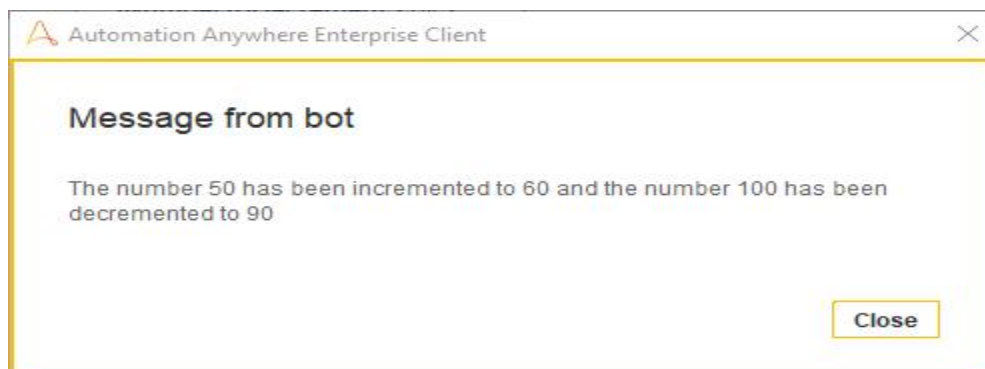
Screenshot of Bot:***OUTPUT:***

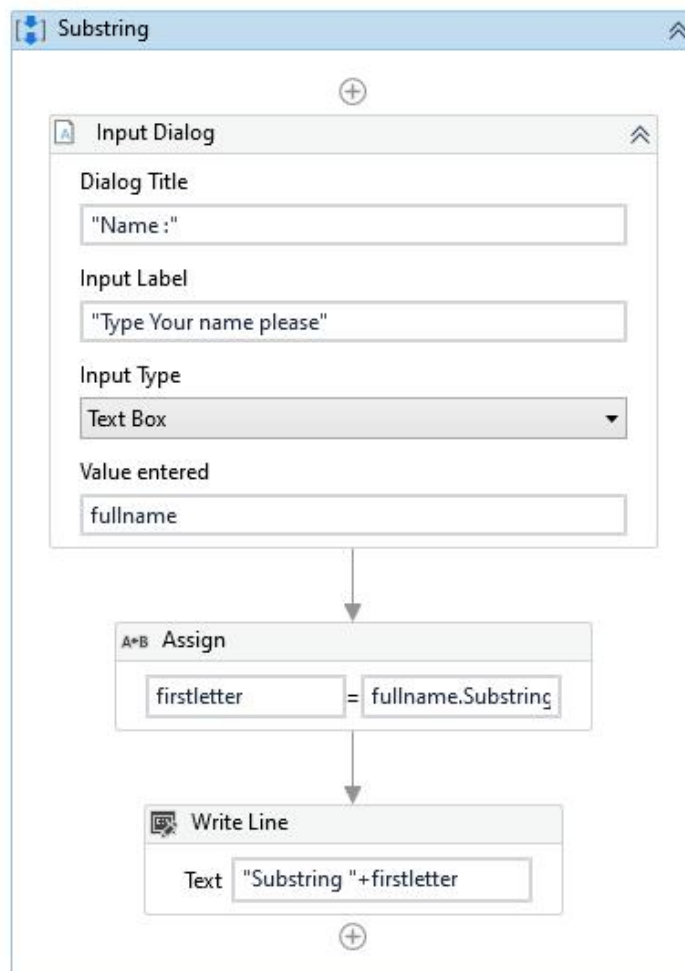
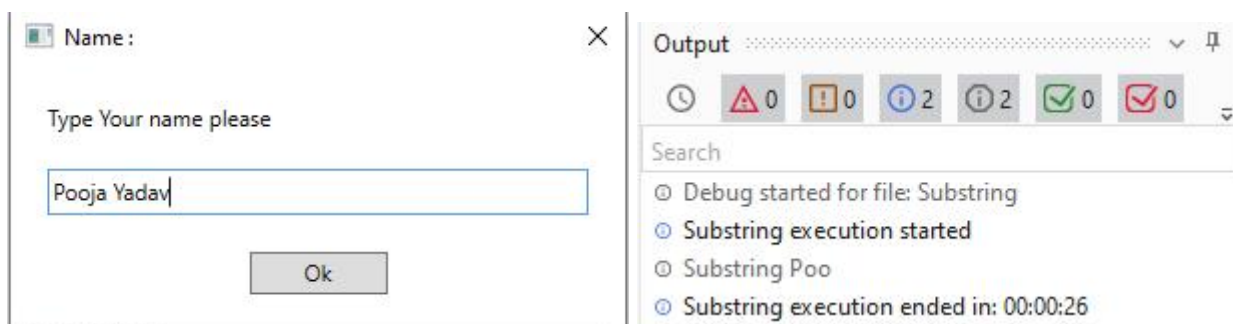


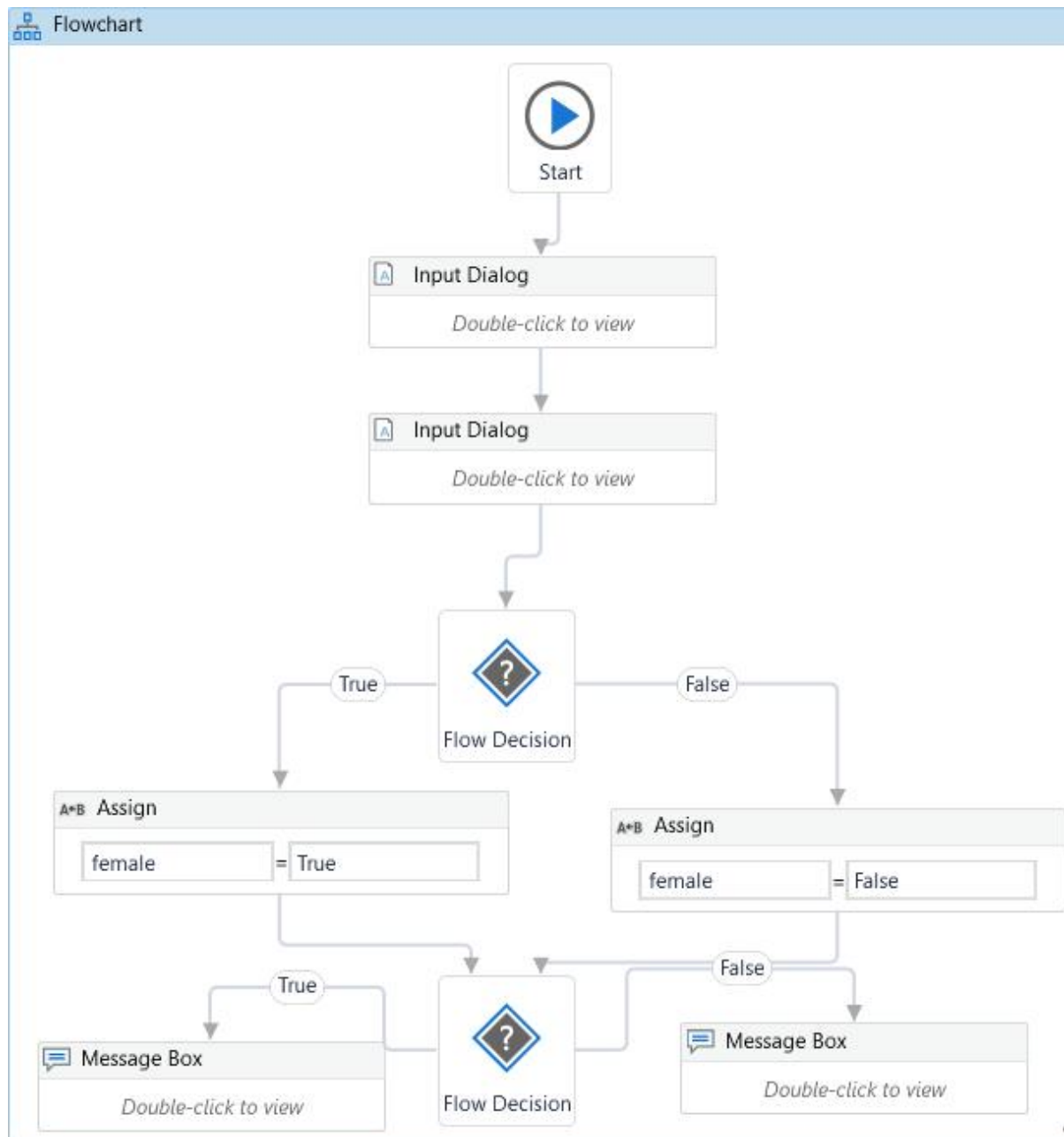
3) Bot to demonstrate loop using Number package:

Screenshot of Bot:***OUTPUT:***

3) Bot to demonstrate increment and decrement actions on number and datetime variables:

Screenshot of Output:**OUTPUT:*****UI-Path:***

1) Demonstrate substring using string variable.***Screenshot of Bot:******Output:***

2) Demonstrate Boolean Variable using Flow Decision:**Screenshot of Bot:**


Output:

 Enter ×

Your Name

Rasika


Ok

 Enter ×

Your Gender

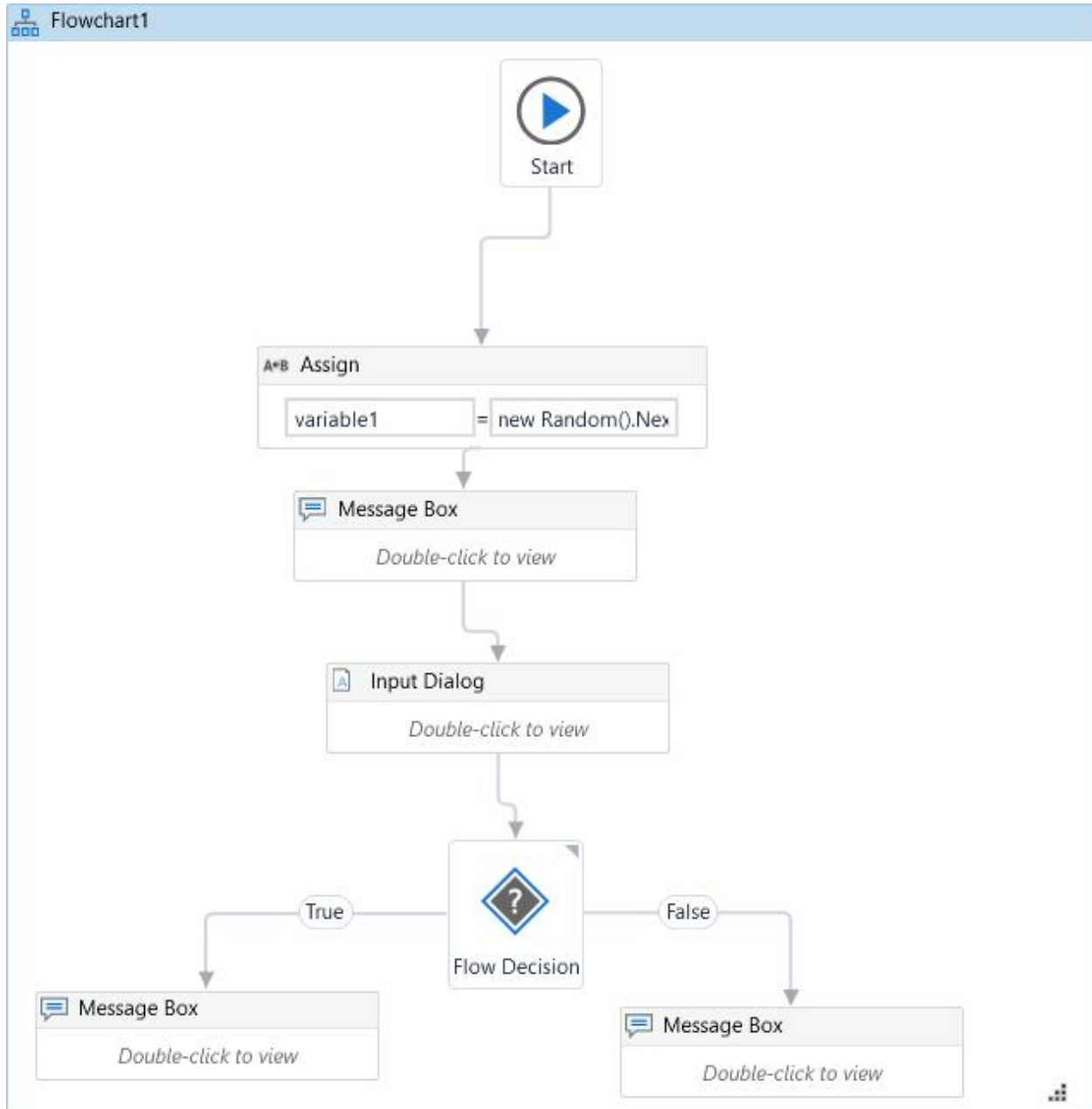
Female

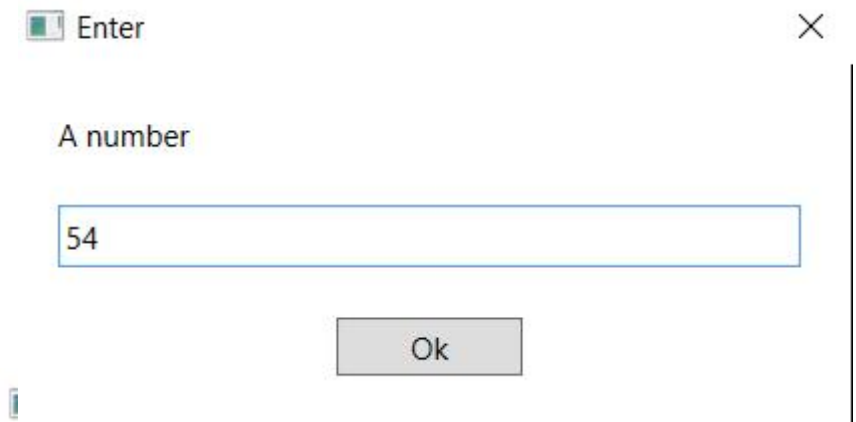
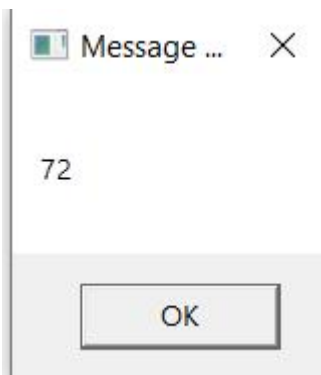
Ok

 C22126 ×

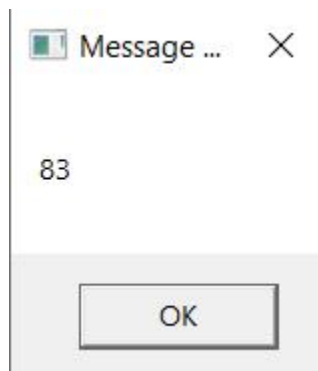
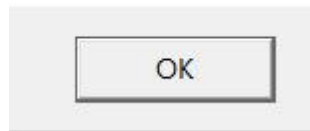
Rasika is Female

OK

3) Demonstrate flow Decision for Random Variable:**Screenshot of Bot:**

Output:

Its not a match

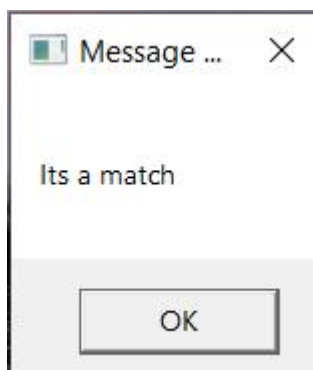


Enter

A number

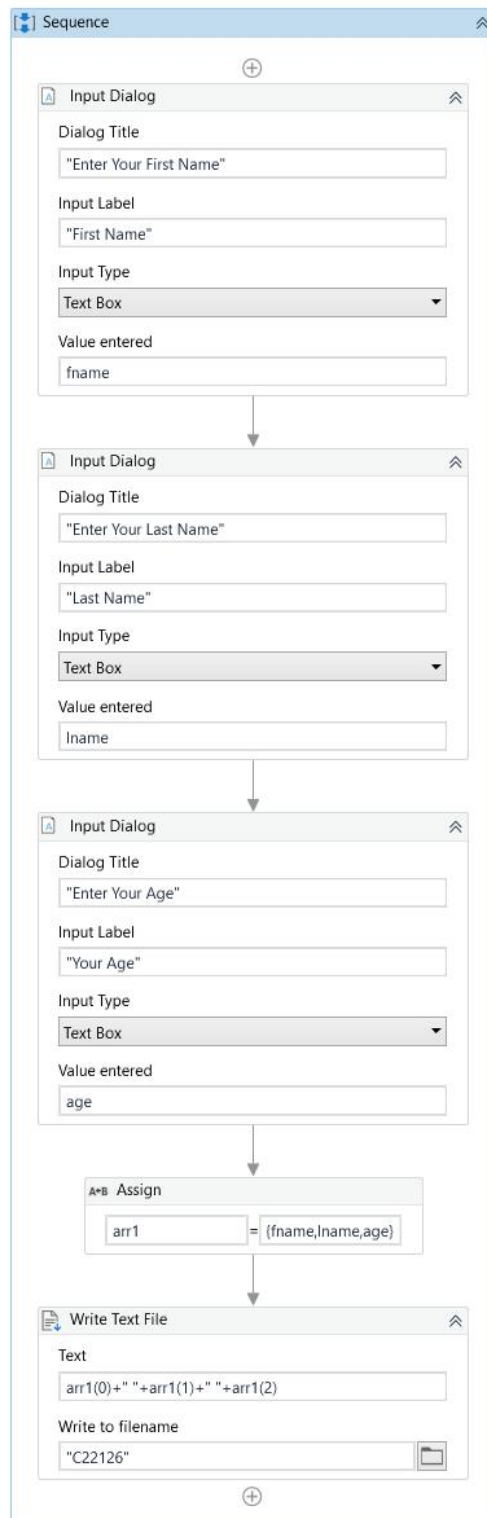
83

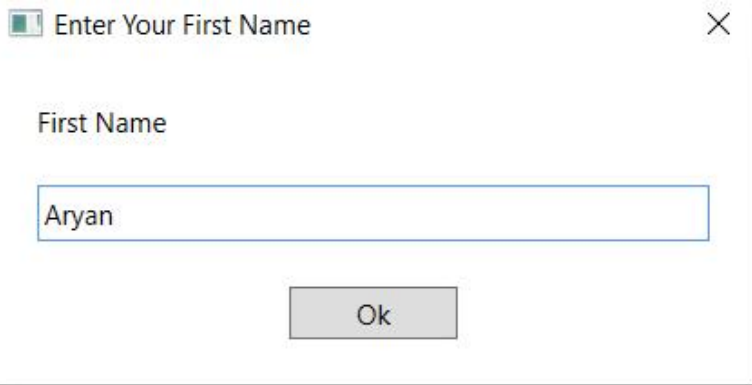
Ok



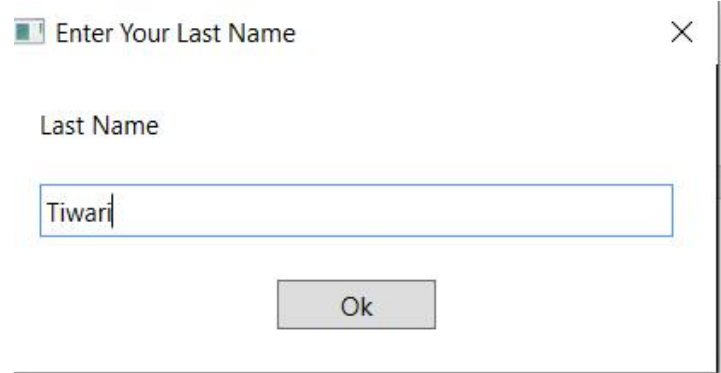
4) Design a bot to accept two strings: name and age, store it into array and write It into notepad file.

***Screens
hot of
Bot:***

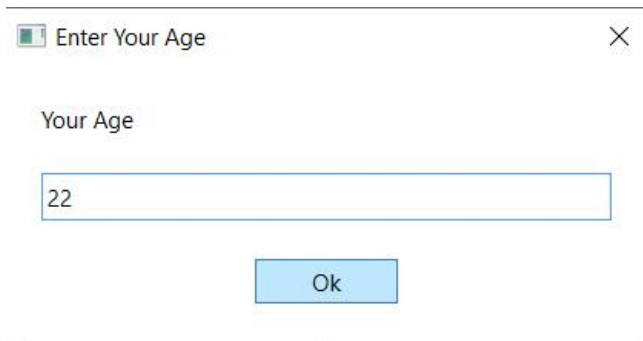


OUTPUT:

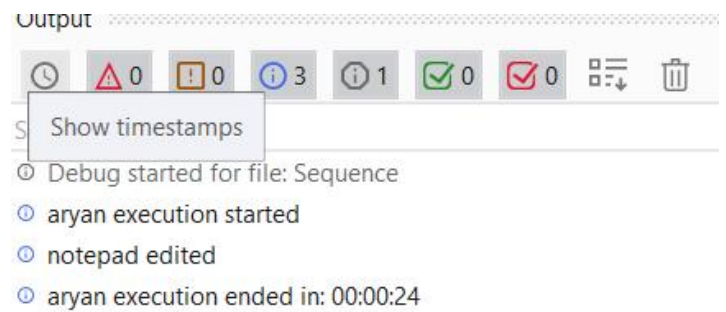
A screenshot of a dialog box titled "Enter Your First Name". It has a close button (X) in the top right corner. The label "First Name" is above a text input field containing the text "Aryan". Below the input field is an "Ok" button.



A screenshot of a dialog box titled "Enter Your Last Name". It has a close button (X) in the top right corner. The label "Last Name" is above a text input field containing the text "Tiwari". Below the input field is an "Ok" button.

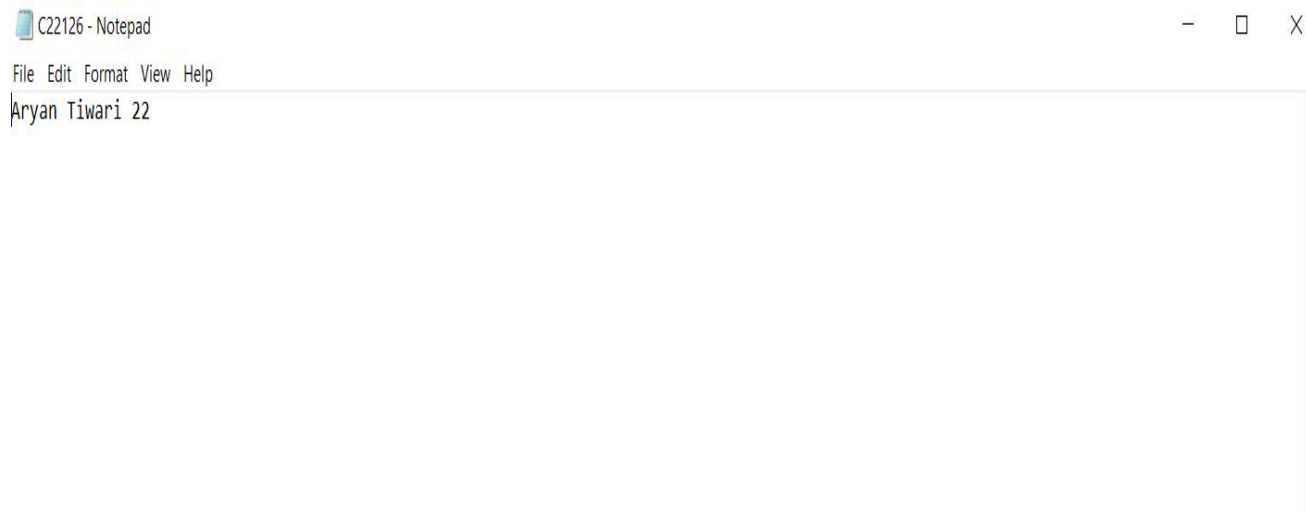


A screenshot of a dialog box titled "Enter Your Age". It has a close button (X) in the top right corner. The label "Your Age" is above a text input field containing the text "22". Below the input field is an "Ok" button.



A screenshot of the "Output" window in a software application. The window has a toolbar with icons for clock, error, warning, info, debug, success, and completion, each with a count. A "Show timestamps" button is visible. The output log shows the following entries:

- Debug started for file: Sequence
- aryan execution started
- notepad edited
- aryan execution ended in: 00:00:24



A screenshot of a Notepad window titled "C22126 - Notepad". The menu bar includes "File", "Edit", "Format", "View", and "Help". The text content of the notepad is "Aryan Tiwari 22".

5) Demonstrate generic value variables:

The screenshot shows the UiPath Studio interface with a workflow named 'GenericValues'. The workflow consists of two 'Write Line' activities. The first 'Write Line' activity has a text field containing 'A+B'. The second 'Write Line' activity has a text field containing 'A+C'. Below the workflow, a table lists the variables used:

Name	Variable type	Scope	Default
A	String	GenericValues	"123"
B	String	GenericValues	"456"
C	GenericValue	GenericValues	123
D	GenericValue	GenericValues	456

At the bottom of the table, there is a 'Create Variable' button. The right sidebar shows the 'Object Repository' with 'Descriptors' and 'UI Activities' tabs. A tooltip titled 'Introducing Object Repository' is visible on the right side.

The screenshot shows the UiPath Studio interface with a workflow named 'GenericValues'. The workflow consists of four 'Write Line' activities. The first 'Write Line' activity has a text field containing 'A+B'. The second 'Write Line' activity has a text field containing 'A+C'. The third 'Write Line' activity has a text field containing 'C=A'. The fourth 'Write Line' activity has a text field containing 'C=A'. Below the workflow, a table lists the variables used:

Name	Variable type	Scope	Default
A	String	GenericValues	"123"
B	String	GenericValues	"456"
C	GenericValue	GenericValues	123
D	GenericValue	GenericValues	456

At the bottom of the table, there is a 'Create Variable' button. The right sidebar shows the 'Object Repository' with 'Descriptors' and 'UI Activities' tabs. A tooltip titled 'Introducing Object Repository' is visible on the right side.

Name	Variable type	Scope	Default
A	String	GenericValues	"123"
B	String	GenericValues	"456"
C	GenericValue	GenericValues	123
D	GenericValue	GenericValues	456

Introducing Object Repository

With Object Repository you can manage your project's UI Descriptors, add them to UI repositories and re-publish them as UI libraries for global reuse.

UI Descriptors can be part of the project for project wide reuse, part of a global repositories for testing purposes or part of UI libraries for global, cross project sharing.

[Learn More](#)

OUTPUT:

Name	Variable type	Scope	Default
A	String	GenericValues	"123"
B	String	GenericValues	"456"
C	GenericValue	GenericValues	123
D	GenericValue	GenericValues	456

Output

Debug started for file: GenericValues

GenericValues execution started

123456

123123

246

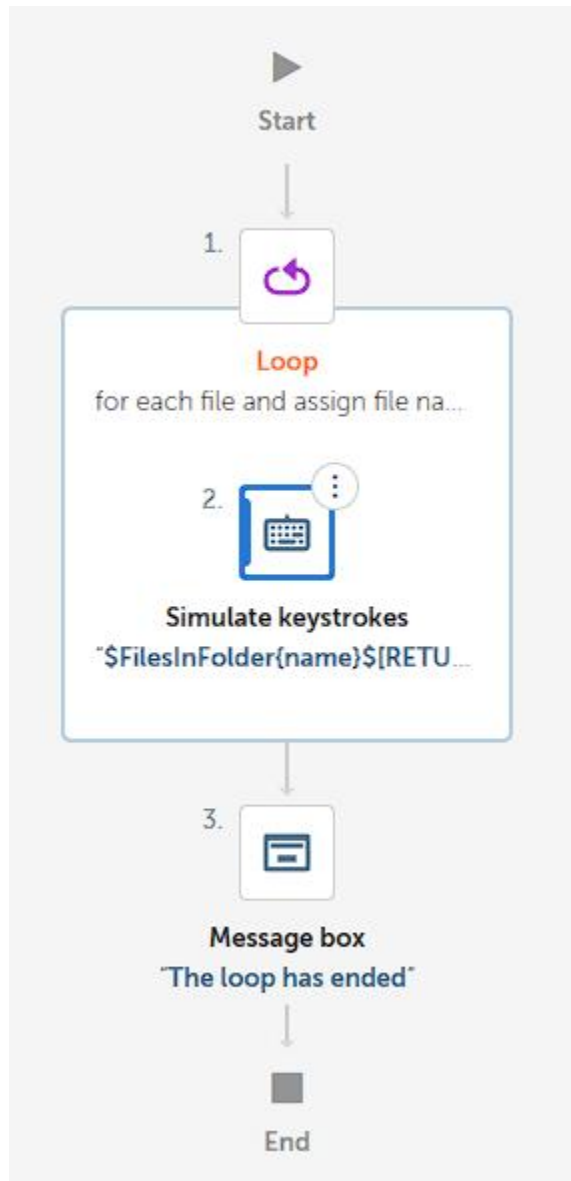
579

GenericValues execution ended in: 00:00:00

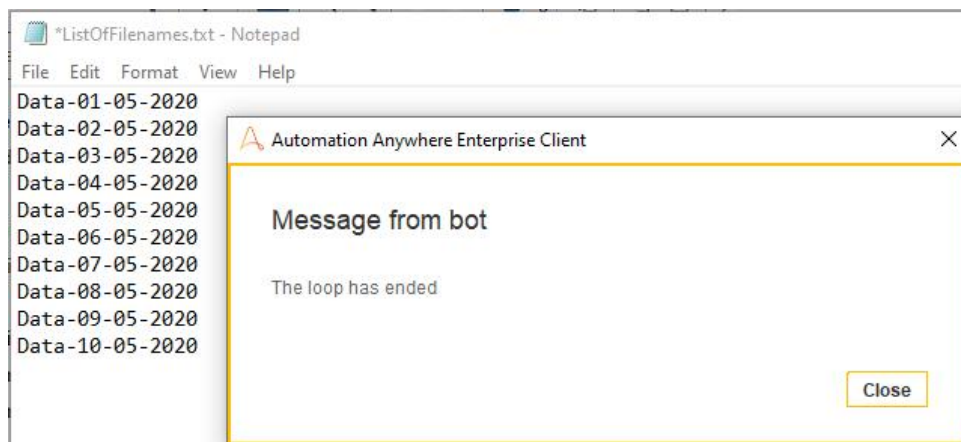
CATEGORY 3:***AUTOMATION ANYWHERE(AA):***

1) Bot to demonstrate loop for reading file names in a folder and writing those into notepad file:

Screenshot of Bot:



Organize	
> MCA > SEM II > RPA > rpaTesting	
Name	Date modified
excelWriting	14-05-2023 23:48
output	14-05-2023 23:32
Data-01-05-2020.txt	14-05-2023 21:21
Data-02-05-2020.txt	14-05-2023 21:21
Data-03-05-2020.txt	14-05-2023 21:22
Data-04-05-2020.txt	14-05-2023 21:22
Data-05-05-2020.txt	14-05-2023 21:22
Data-06-05-2020.txt	14-05-2023 21:22
Data-07-05-2020.txt	14-05-2023 21:22
Data-08-05-2020.txt	14-05-2023 21:22
Data-09-05-2020.txt	14-05-2023 21:22
Data-10-05-2020.txt	14-05-2023 21:22

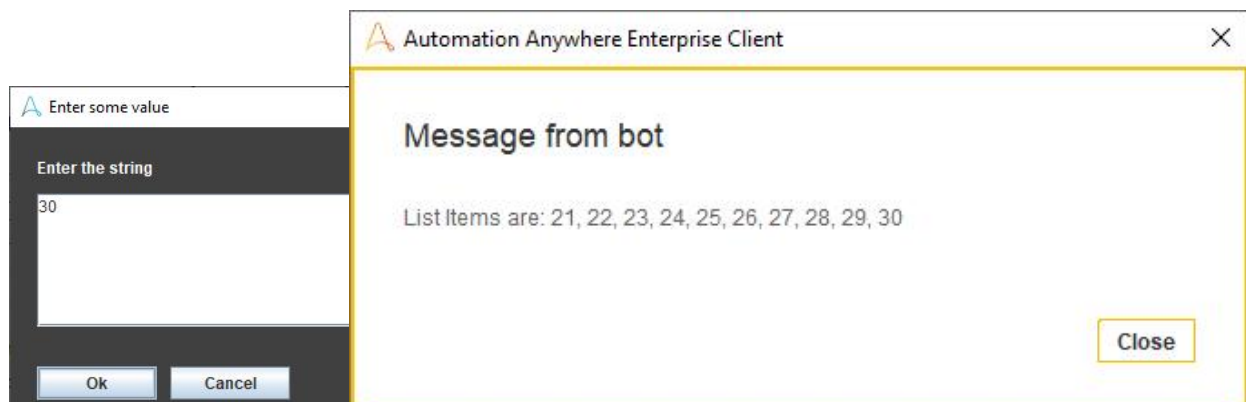
OUTPUT:

2) Bot to demonstrate list variable using list package

Screenshot of Bot:



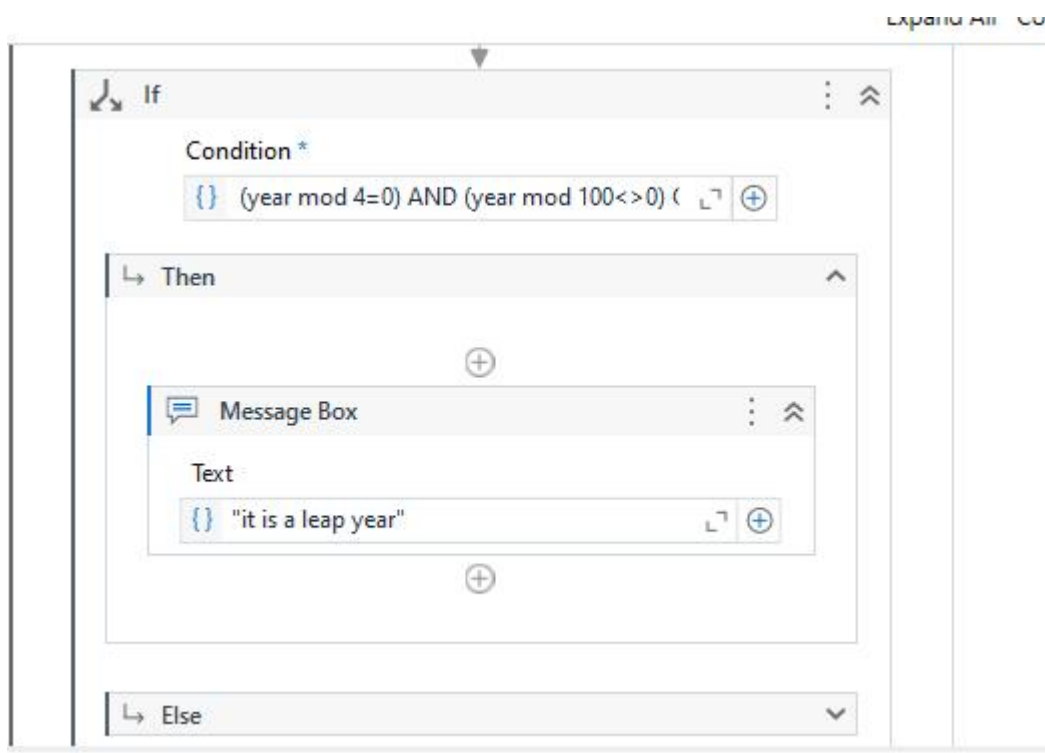
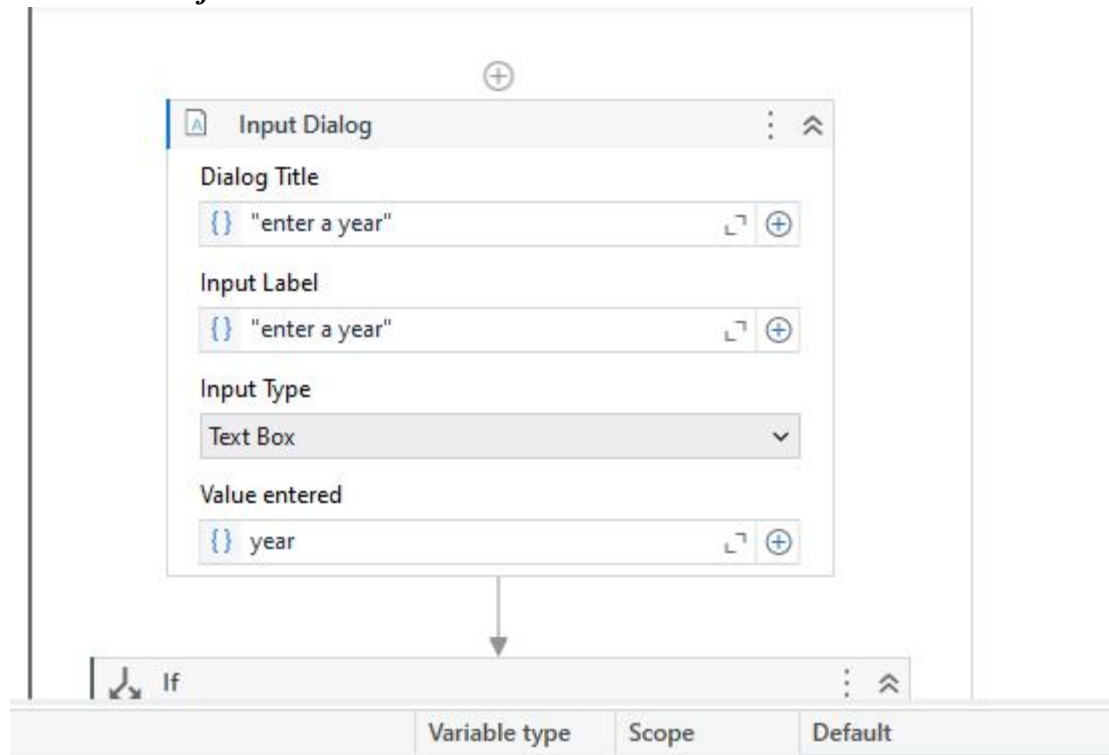
OUTPUT:

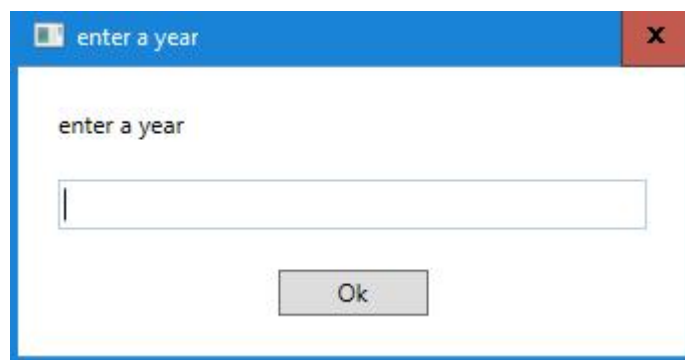
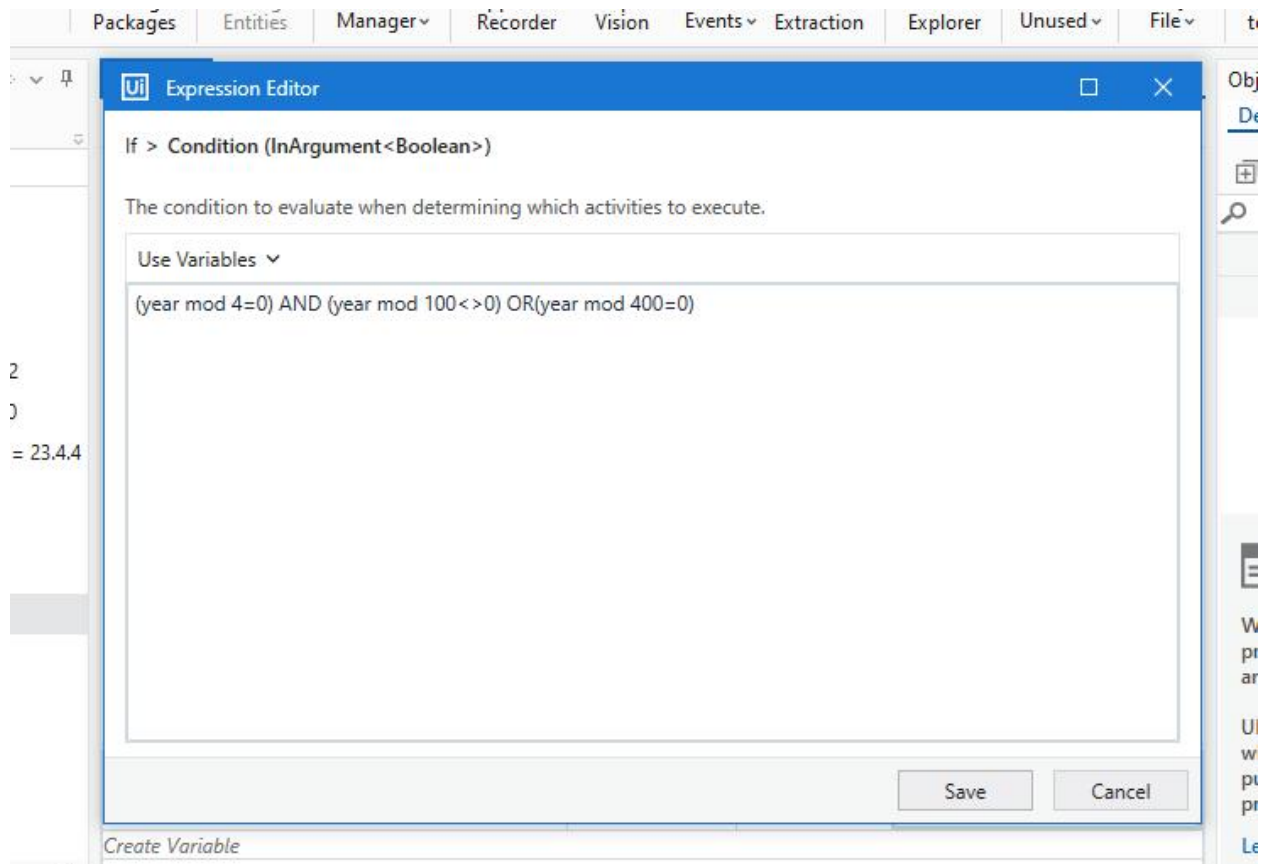


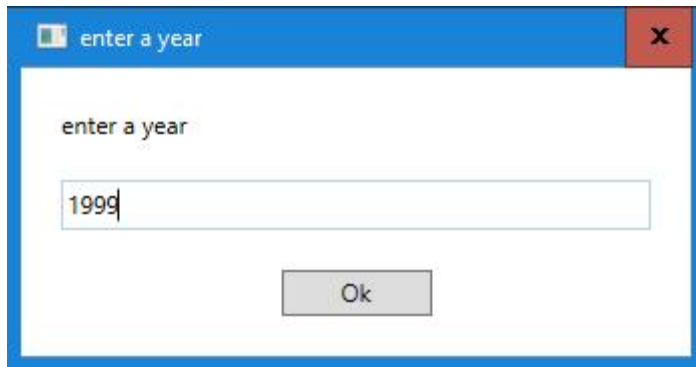
UI-Path:

1) Write a bot to check year is a leap year.

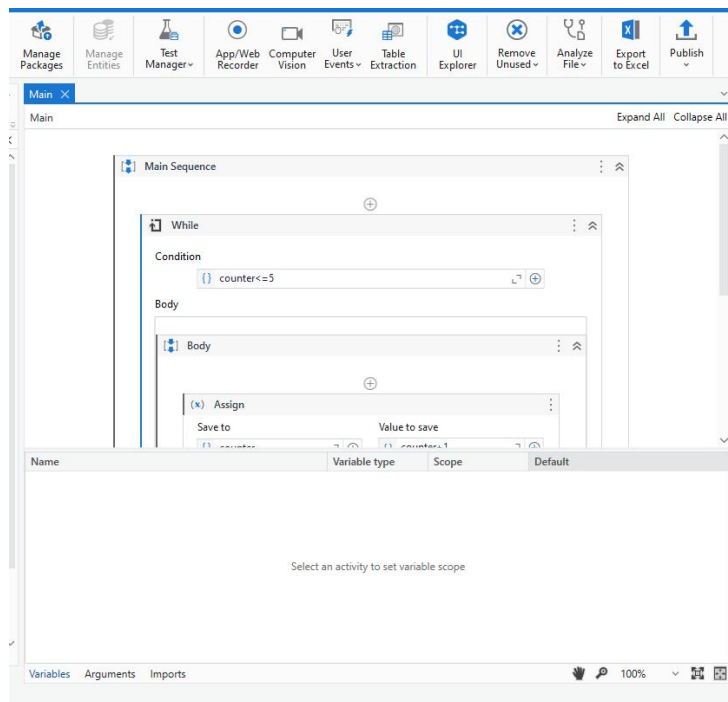
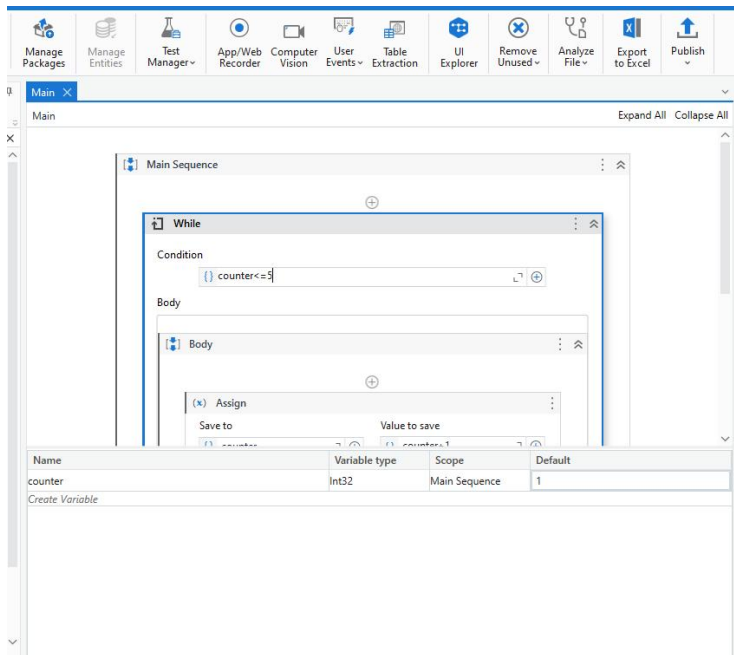
Screenshot Of Bot:

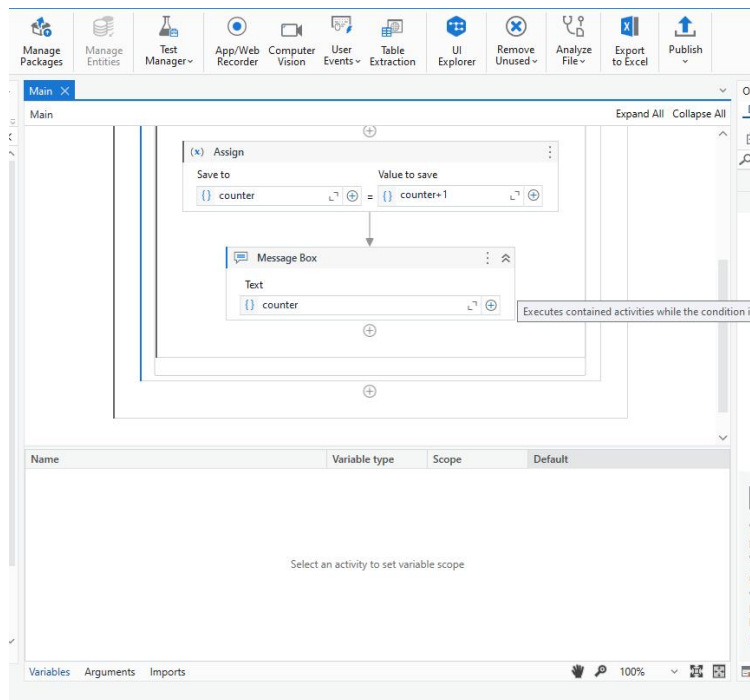
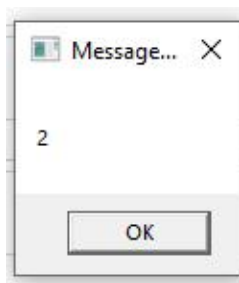




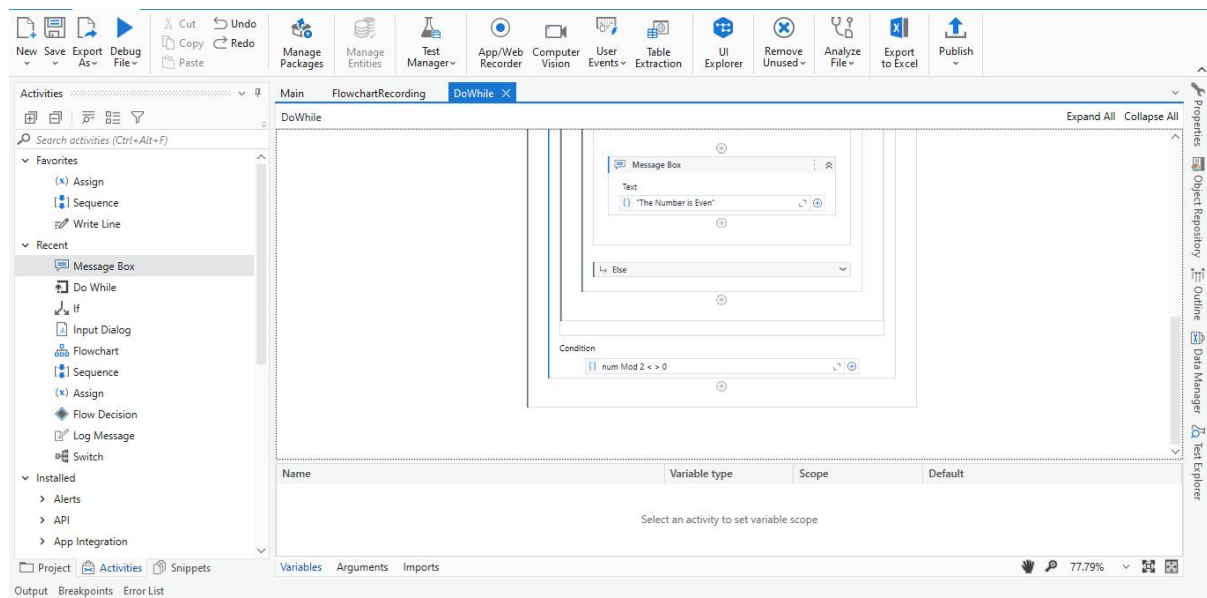
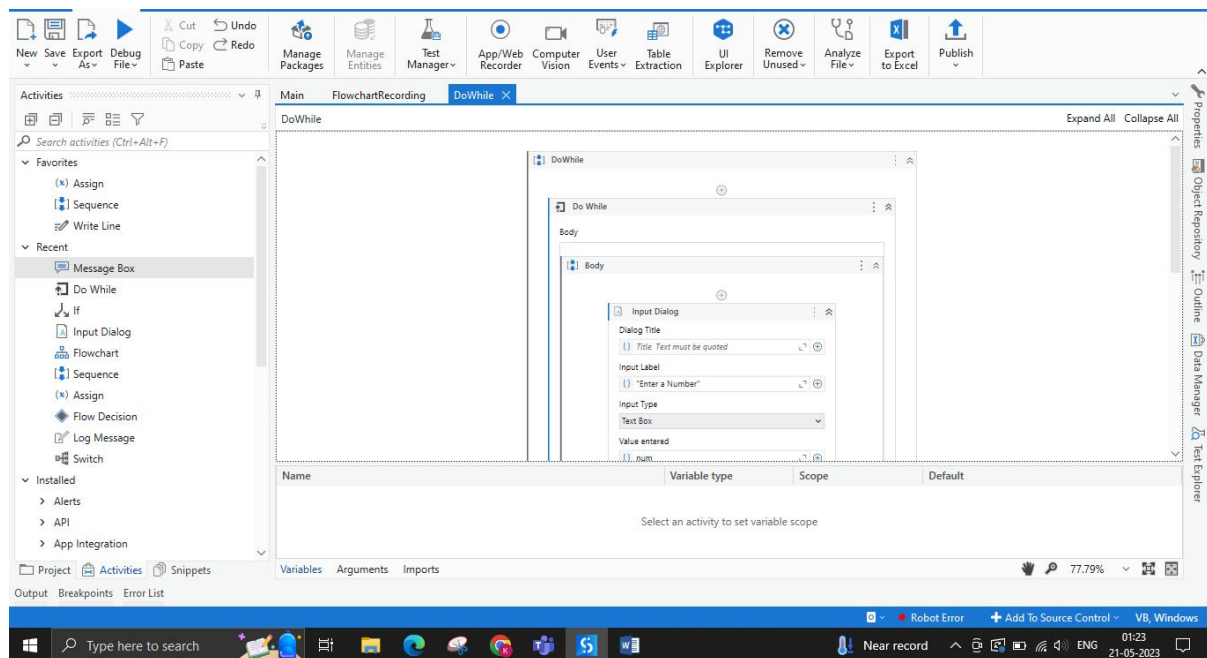


***2) Demonstrate while loop:
Screenshot of Bot:***

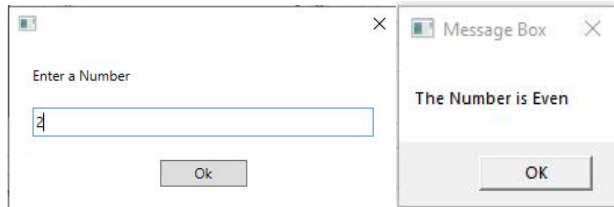


**OUTPUT:**

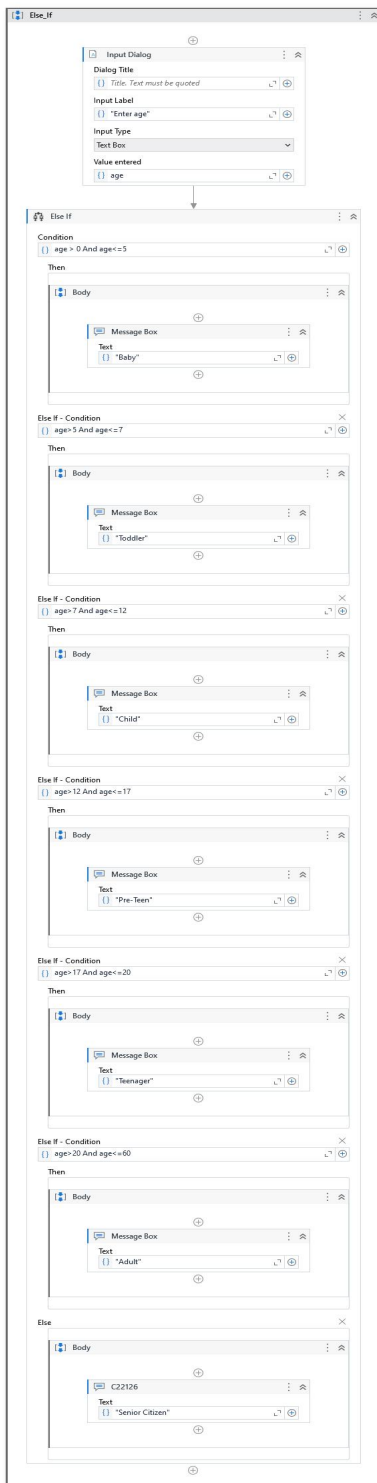
3) Demonstrate do while loop:
Screenshot of Bot:

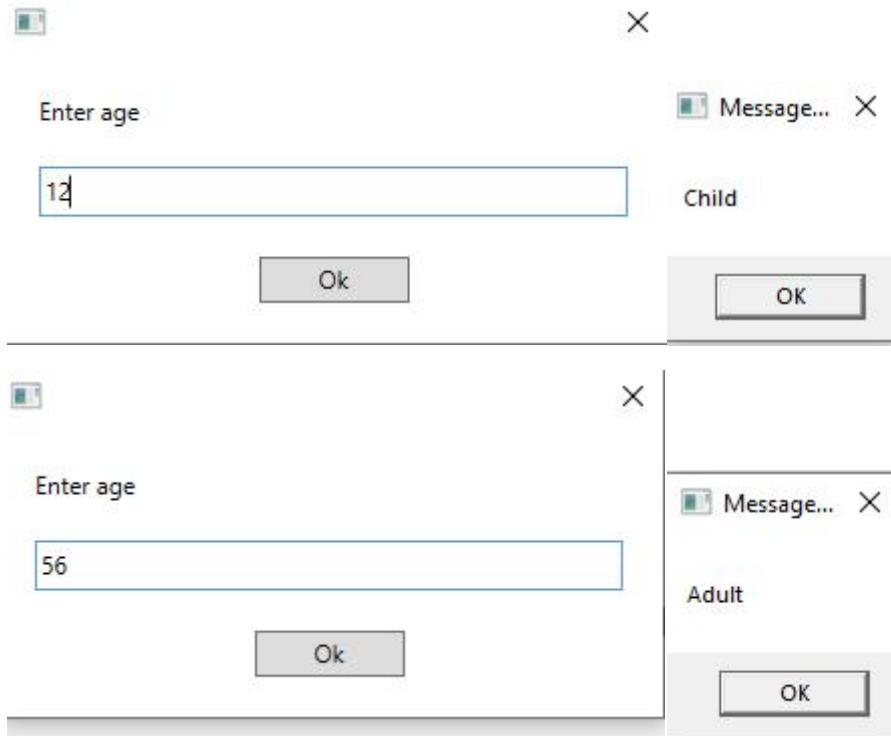


OUTPUT:

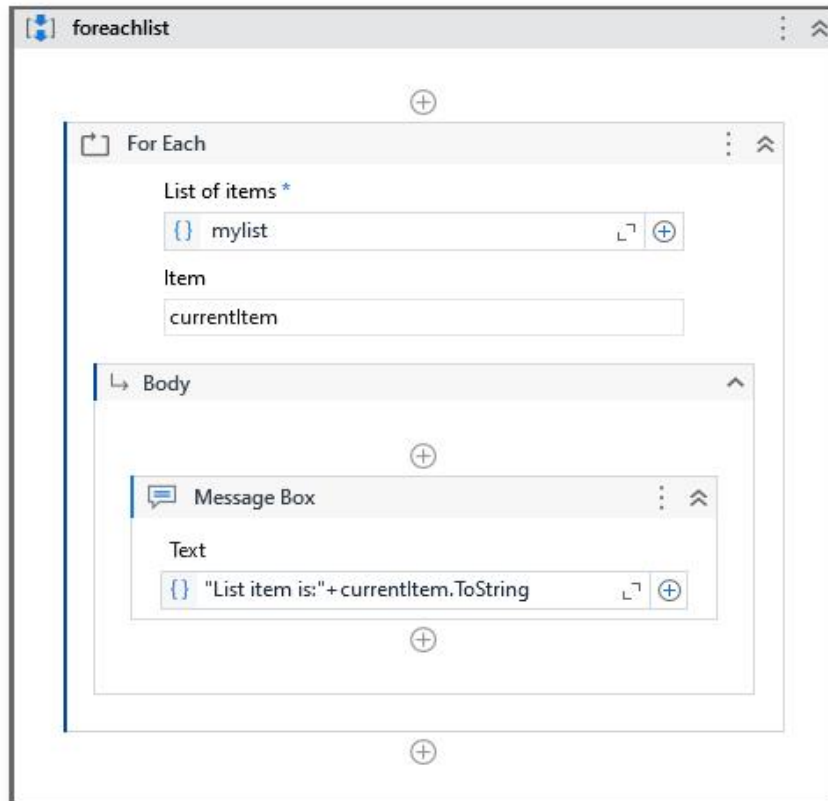
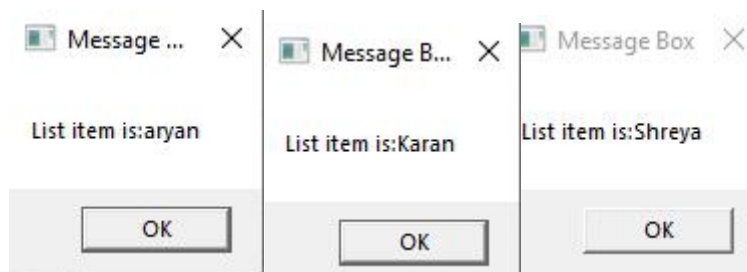


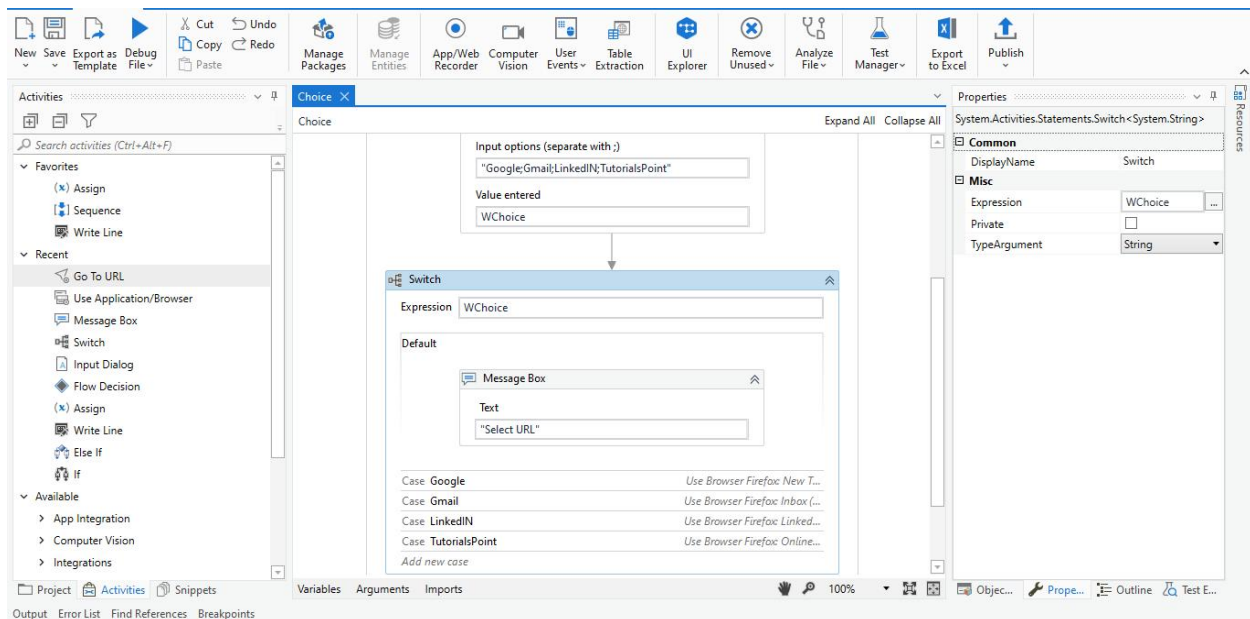
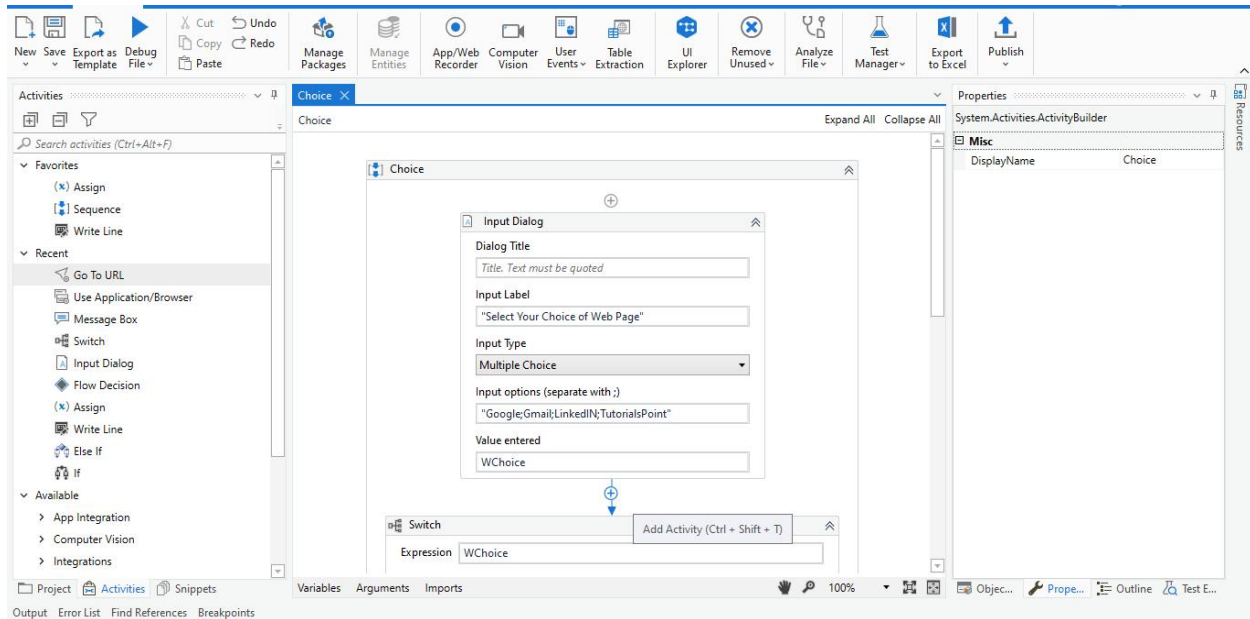
4) Demonstrate elseIF:

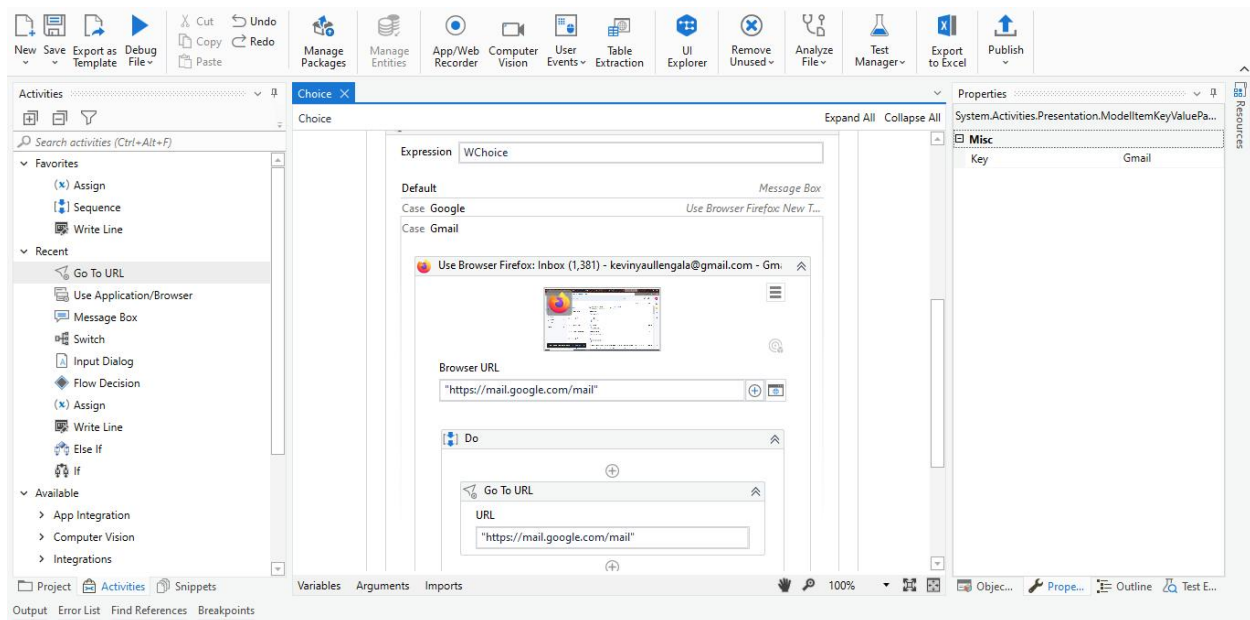
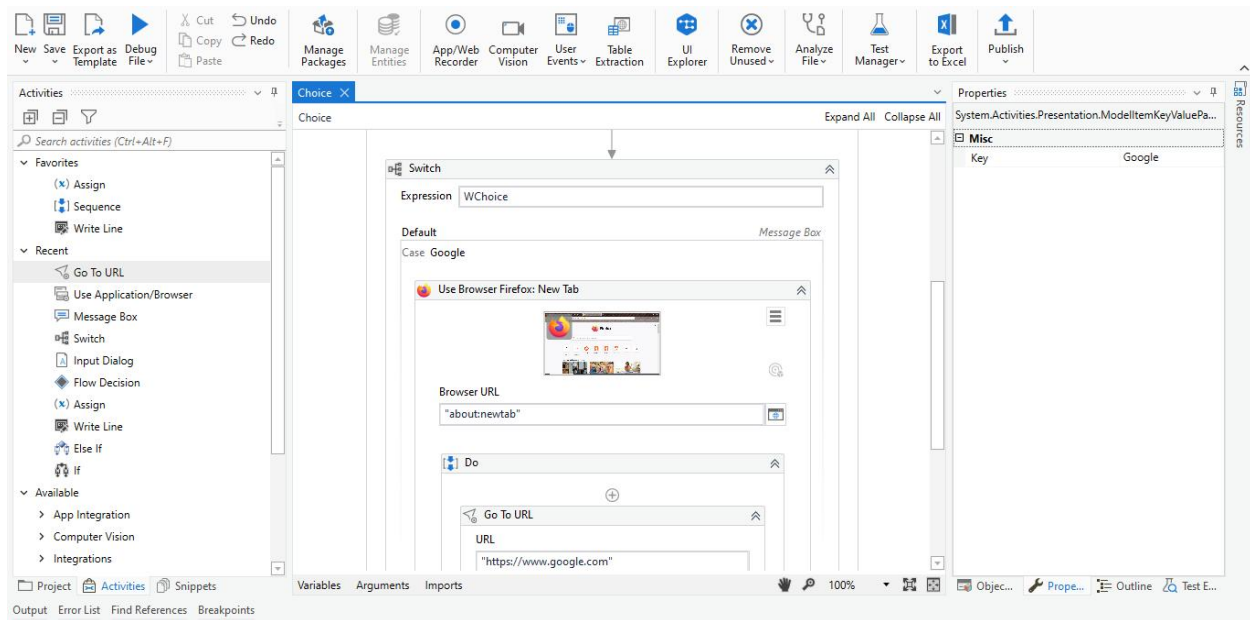
Screenshot Of Bot:***Output:***

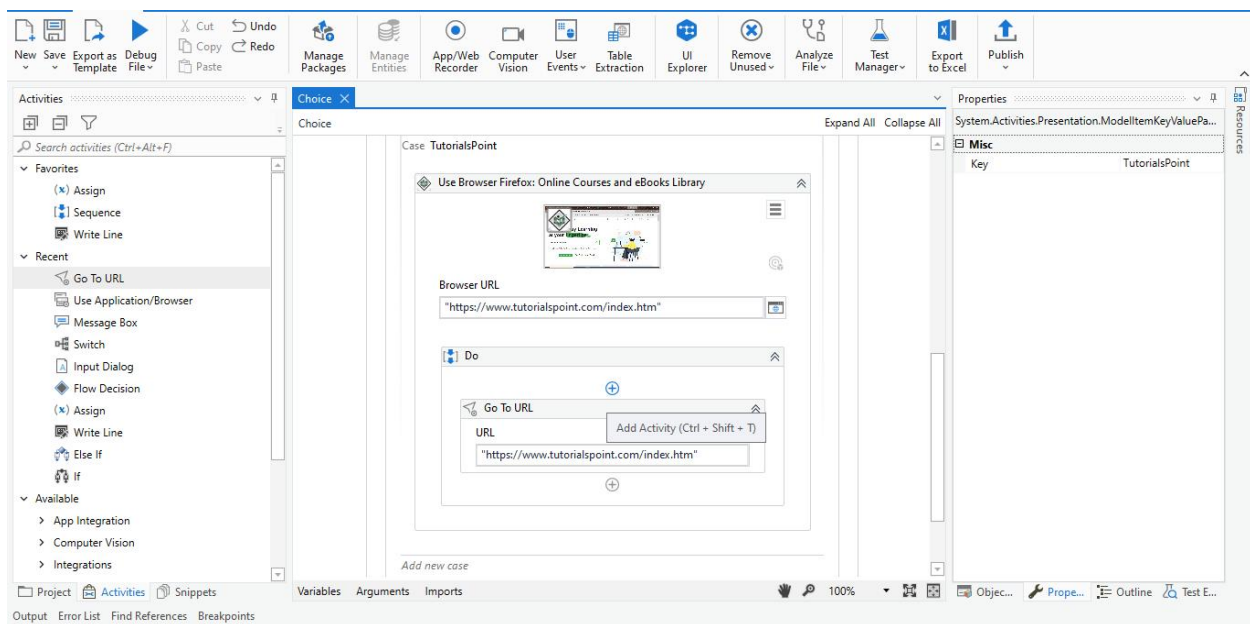
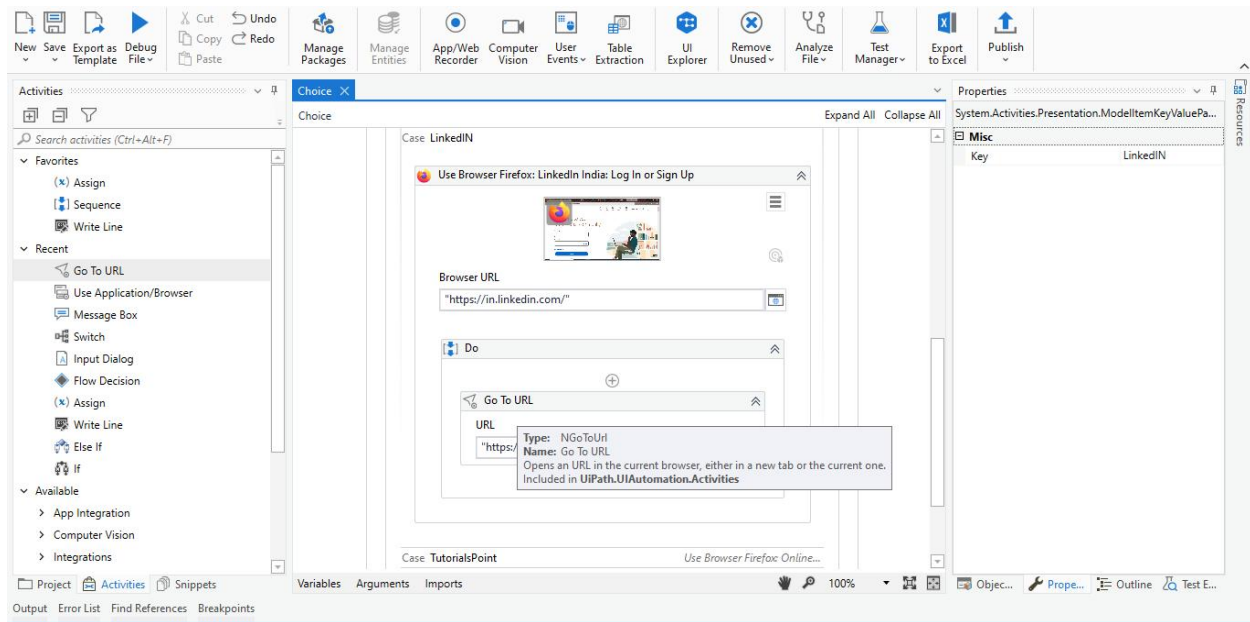


5) Demonstrate ForEach for list variable:

Screenshot Of Bot:***OUTPUT:******7) Demonstrate switch:***

Screenshot of Bot:





OUTPUT: