Robotic Process Automation
Practical #3

Name	Ninad Karlekar	Roll Number	22306A1012
Class	MSCIT part 2		Class A
Subject/Course:	Robotic Process Automation		
Topic	Decision Making and Looping statements.		

Q1. Create an automation UiPath Project using Decision Making statements.

1. Drag and drop an "Input Dialog" activity into the sequence.

2. Drag and drop an "If" activity below the "Input Dialog" activity.
3. Add a "Message Box" Activity (Then Branch):

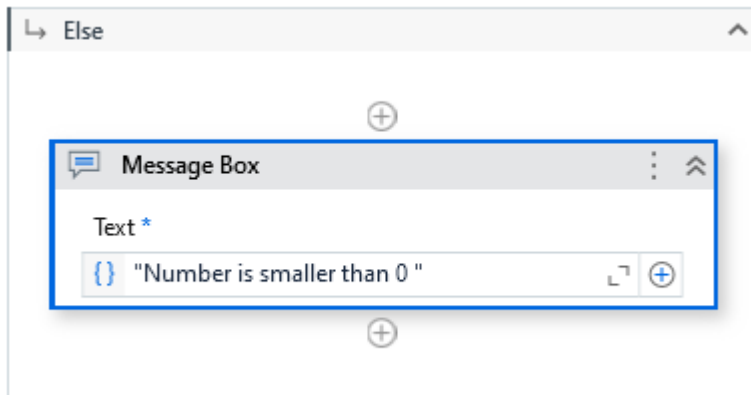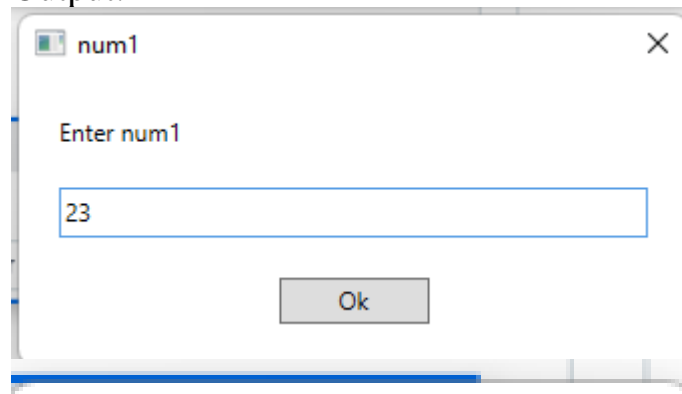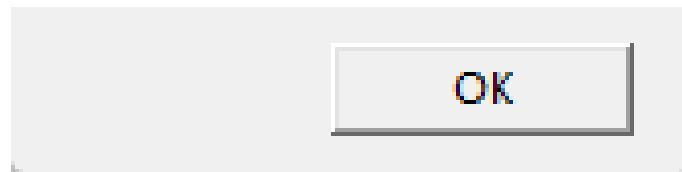4. Add a "Message Box" Activity (Else Branch)

Output:

Else IF

1. Drag and drop an "Input Dialog" activity into the sequence.

[↕] elseif

⊕

| [A] Input Dialog | ⋮ ≫ |

Dialog Title

{} "Num 1"

Input Label

{} "Enter Num 1"

Input Type

Text Box ⌄

Value entered
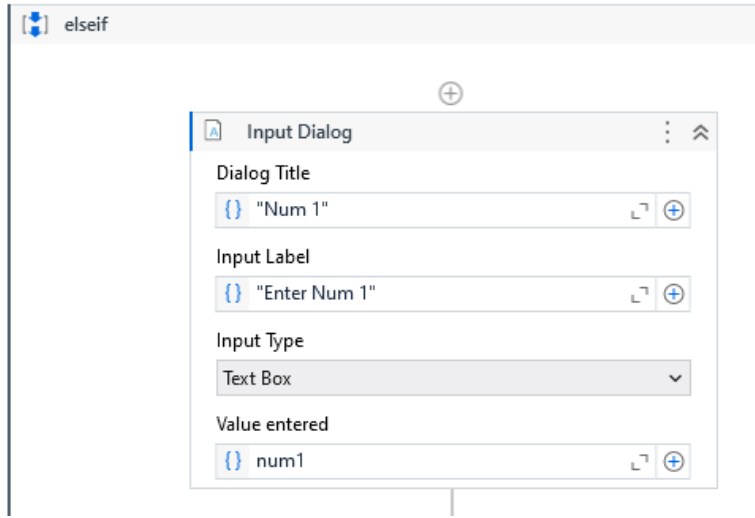
{} num1

2. Add a "Message Box"

⬁ Else If

Condition *

{} num1 = 0

Then

[↕] Body ⋮

⊕

💬 Message Box ⋮ ≫

Text *

{} "Number is zero"

⊕

Else If - Condition *

{} num1 >0

Then

[✦] Body

⊕
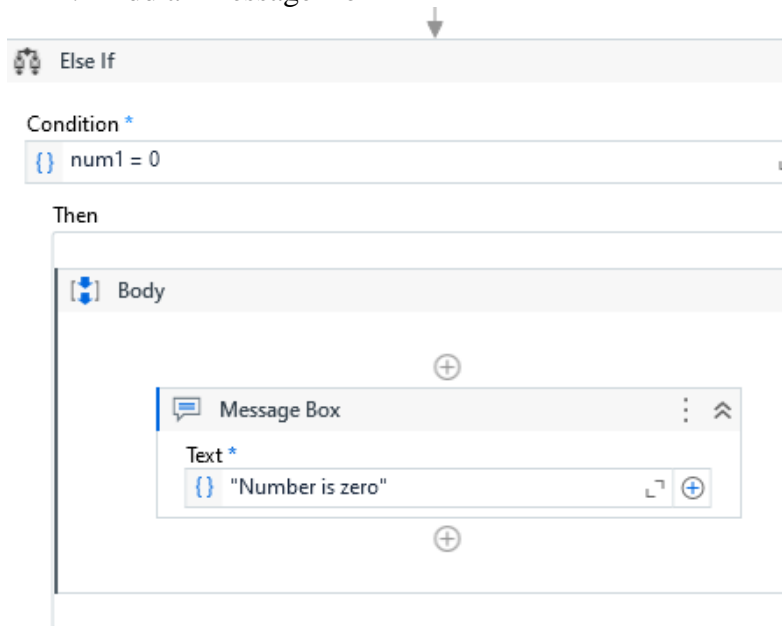
💬 Message Box   ⋮ ⌄

Text *
{} "Number is positive"   ⌐⌐ ⊕

⊕

Else   ✕

[✦] Body   ⋮ ⌄

⊕

💬 Message Box   ⋮ ⌄

Text *
{} "Number is negative"   ⌐⌐ ⊕

⊕

OUTPUT:

◻ Num 1   ✕
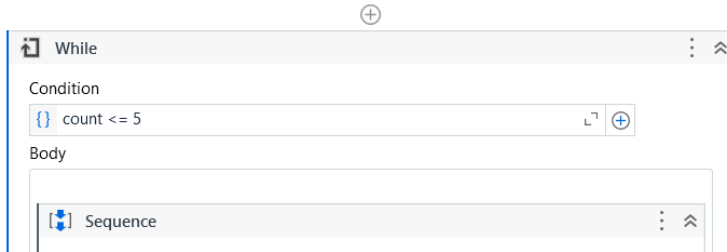
Enter Num 1

25

Ok

◻ Message Box   ✕

Number is positive

OK

## Q2. Create an automation UiPath Project using looping statements.

While Loop

1. Add while activity and set condition to ->   count <=5
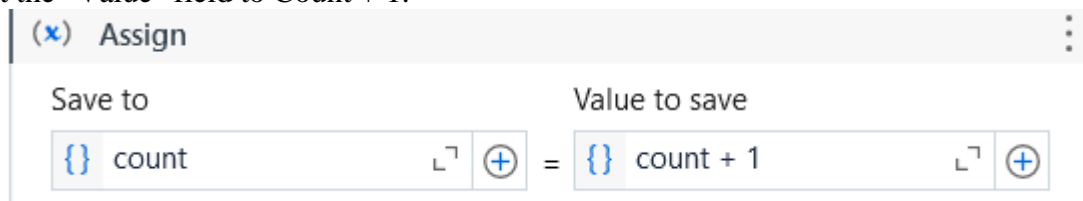
While
Condition
{} count <= 5
Body
[⋮] Sequence

2. Create variable count and set variable type to int32

Name	Variable type	
count	Int32	R
Create Variable		

3. Inside while body create message box and write message -> "Current Count: " + count.ToString()

💬 Message Box
Text *
{} "Current Count: " + count.ToString()
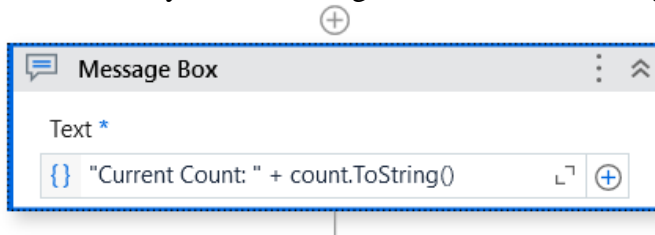
4. Drag and drop assign activity inside while
   Set the "To" field to Count.
   Set the "Value" field to Count + 1.

(x) Assign		
Save to		Value to save
{} count	=	{} count + 1

output

■ Message B... ✕	■ Message B... ✕	■ Message B... ✕	■ Message B... ✕
Current Count: 0	Current Count: 1	Current Count: 2	Current Count: 3
OK	OK	OK	OK

Current Count: 4

OK

Current Count: 5

OK
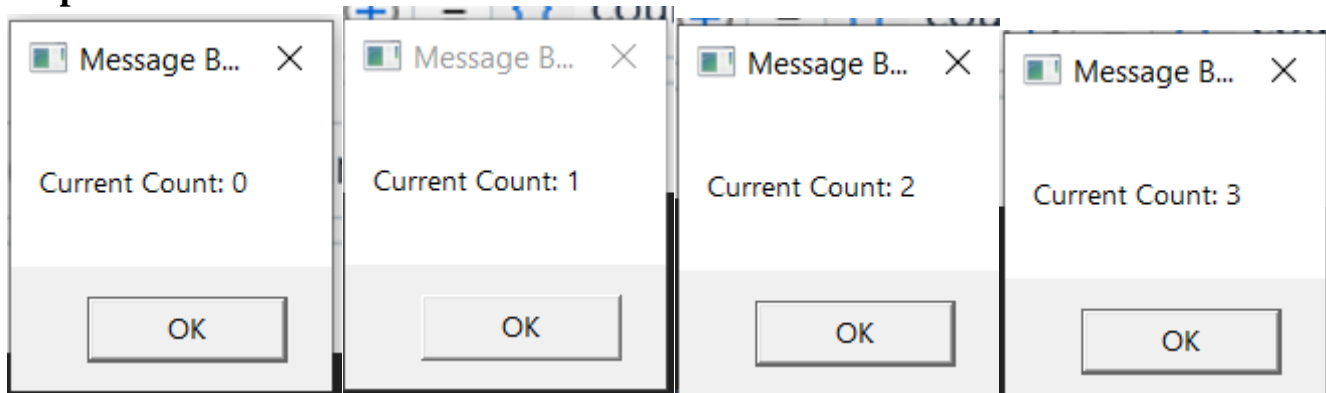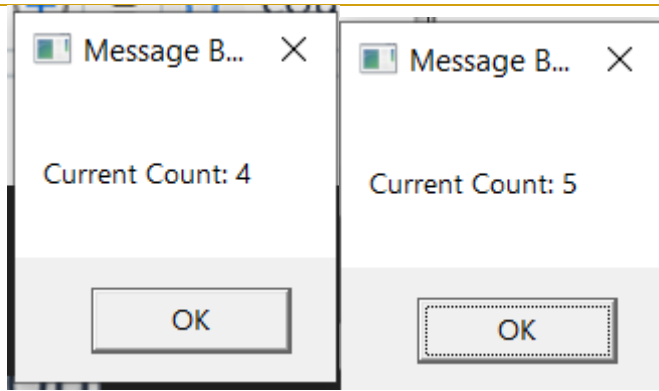
Do While

1. Do While Loop: Drag and drop a "Do While" activity from the "Activities" panel into your workflow.

Do While

Body

Drop activity here

Condition

{} Enter a VB expression

2. Create a variable named 'Count' of type "Int32" to keep track of the current count.

Name	Variable type	Scope
Count	Int32	Do While
Create Variable		

3. we'll use the condition: Count <= 5. This means the loop will continue as long as the Count variable is less than or equal to 5.
4. Inside the "Do While" activity, drag and drop a "Message Box" activity.

Do While

Body

Message Box

Text *

{} Text must be quoted

Condition

{} Count <= 5

5. Configure the Message Box Activity:

```
💬 Message Box                        ⋮  ⌃

   Text *
   {}  "Current Count: " + Count.ToString()    ⌐  ⊕
```

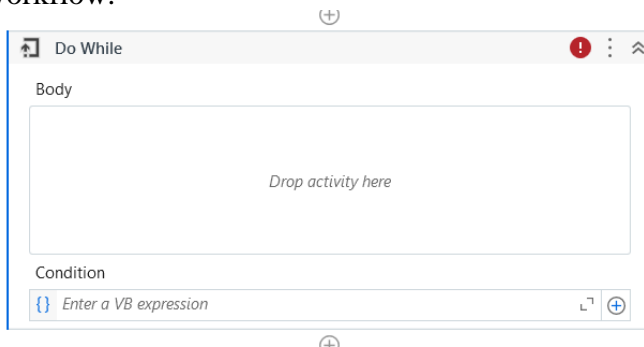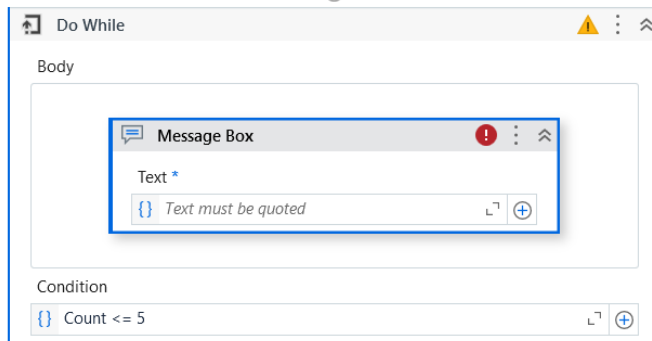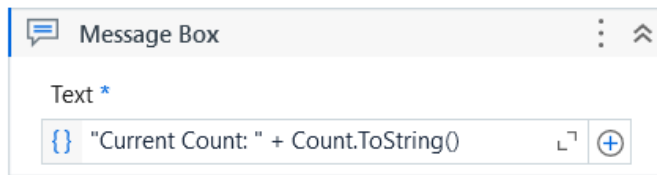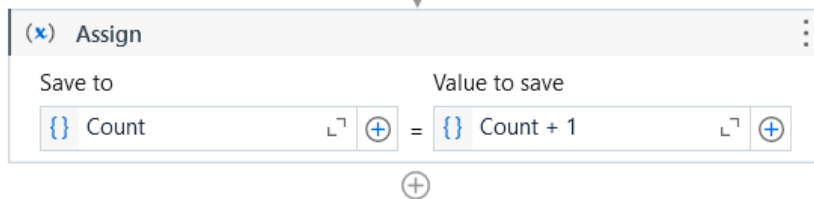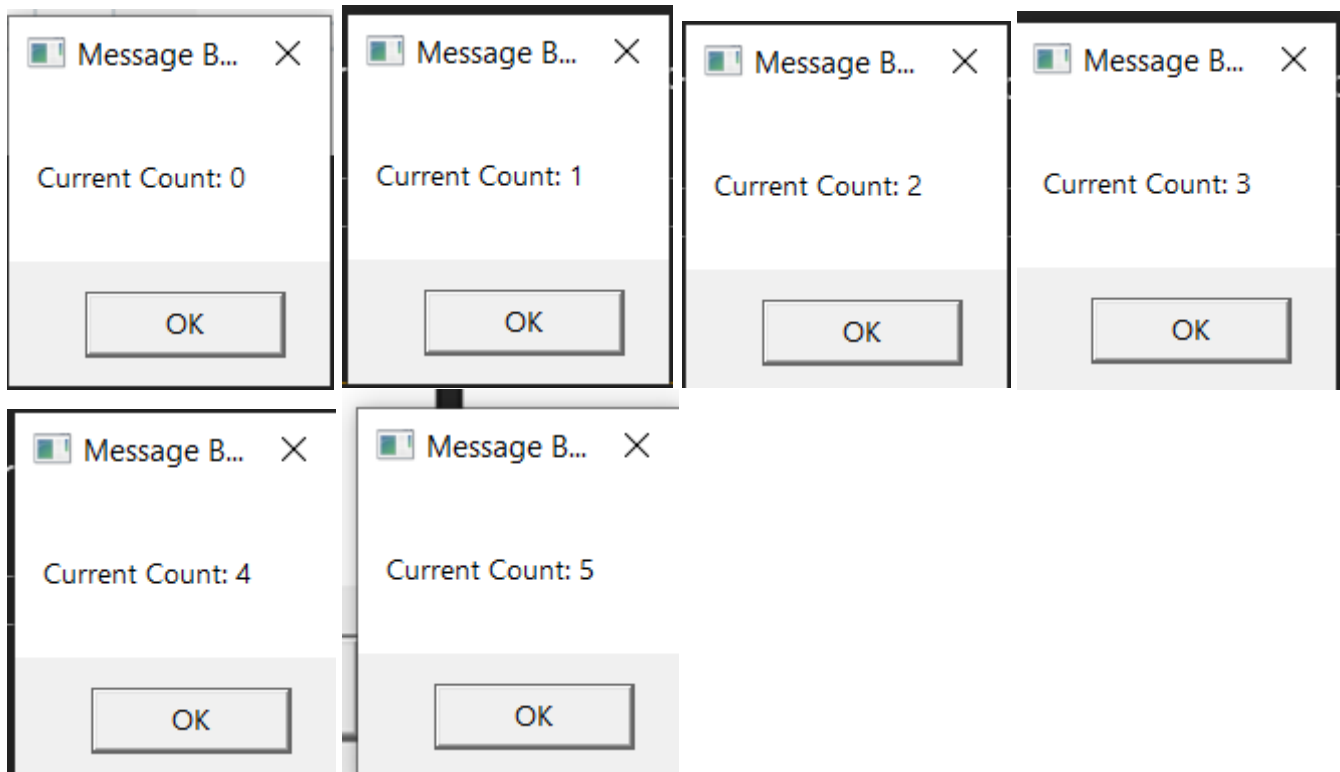6. Add an "Assign" activity inside the "Do While" loop:
   Set the "To" field to Count.
   Set the "Value" field to Count + 1.

```
(x)  Assign                                    ⋮

   Save to                    Value to save
   {}  Count          ⌐ ⊕  =  {}  Count + 1      ⌐ ⊕

                          ⊕
```

OUTPUT

Message B... ✕	Message B... ✕	Message B... ✕	Message B... ✕
Current Count: 0	Current Count: 1	Current Count: 2	Current Count: 3
OK	OK	OK	OK

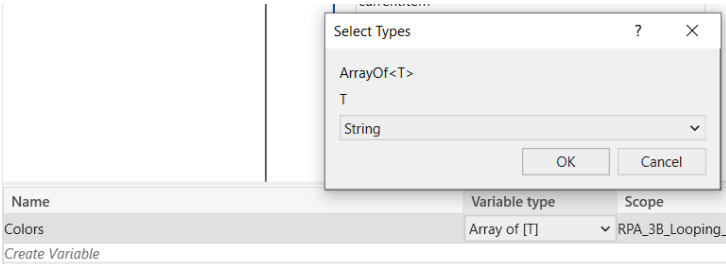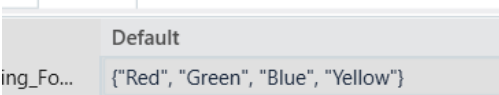| Message B... ✕	Message B... ✕
Current Count: 4	Current Count: 5
OK	OK

FOR EACH

we'll create a "For Each" loop to iterate through a list of names and display each name using a message box.

1. Add a List of Names:

	Select Types	?	×
ArrayOf<T>			
T			
String			
	OK	Cancel	

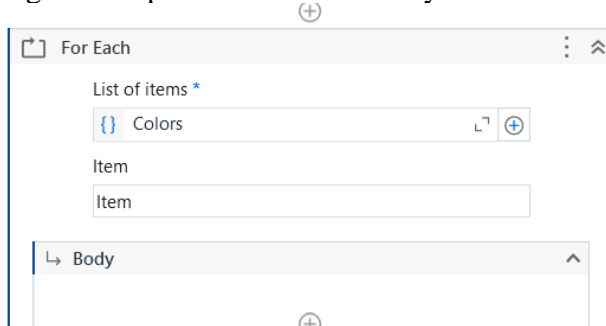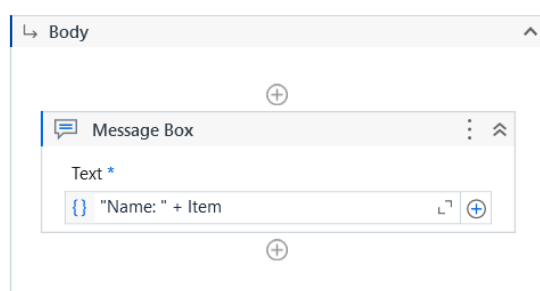| Name | | Variable type | Scope |
| Colors | | Array of [T] | RPA_3B_Looping_F |
| Create Variable | | | |

2. In the "Default" value field of the variable, enter the list of colors enclosed in curly braces { } and separated by commas.

| | Default |
| ing_Fo... | {"Red", "Green", "Blue", "Yellow"} |

3. Drag and drop a "For Each" activity from the "Activities" panel into your workflow.

For Each

List of items *
{} Colors

Item
Item

↳ Body

4. Inside the "For Each" activity, drag and drop a "Message Box" activity.

↳ Body

Message Box

Text *
{} "Name: " + Item

Name		Variable type	Scope	Default
Colors		String[]	RPA_3B_Looping_Fo...	{"Red", "Green", "Blue", "Yellow"}
Create Variable				

OUTPUT:

Message ... ×	Message ... ×	Message ... ×	Message ... ×
Name: Red	Name: Green	Name: Blue	Name: Yellow
OK	OK	OK	OK