**Practical No: 9**

**Email Automation**

**AIM: A) Automate the process of send mail event (on any email)**

1. Drag and drop the "Use Gmail" activity into your sequence.  
   A screenshot of a computer

   Description automatically generated
2. Choose the default option and authenticate your Gmail account in the browser pop-up.  
   A screenshot of a computer

   Description automatically generated
3. Inside the "Do" section of the "Use Gmail" activity, add a "Send Email" activity.
4. Select Gmail as the account.
5. Enter the recipient’s email address, subject, body, and attach a file if necessary.  
   A screenshot of a computer

   Description automatically generated

**OUTPUT:**

**A screenshot of a computer

Description automatically generated**

**Learnings:**

**A. Automate Send Mail Event:**

We learned to use the "Use Gmail" activity to send an email, authenticate a Gmail account, and configure email details, enabling automation of the email-sending process.

**B:** **Automate the process of launching an assistant bot on a keyboard event**

1. Drag and drop the "Trigger Scope" activity into your sequence.
2. Inside the Trigger Scope, add a "Hotkey Trigger" activity.
3. Configure the Hotkey Trigger by selecting the desired hotkey combination.  
   A screenshot of a computer

   Description automatically generated
4. In the Action section of the Trigger Scope, add the sequence of activities that you want to be executed when the specified hotkey combination is pressed.  
   A screenshot of a computer

   Description automatically generated
5. Save your workflow and run the sequence.

**OUTPUT:**

**A screenshot of a computer

Description automatically generated**

**B. Automate Launching Assistant Bot on Keyboard Event:**

We grasped the concept of using the "Trigger Scope" and "Hotkey Trigger" activities to execute a sequence of actions when a specified hotkey combination is pressed, facilitating automation triggered by keyboard events.

**Aim: c) Demonstrate the Exception handing in UiPath.**

1. Drag and drop the "Try Catch" activity into your sequence.
2. Inside the Try block, add an activity that might throw an exception during execution. For example, use a "Type Into" activity to type text into an already running Notepad window. If the Notepad window is not running, it will throw an exception.  
   A screenshot of a computer

   Description automatically generated
3. In the Catch block, handle a specific exception. For example, take a "SelectorNotFoundException" exception.
4. Add a "Message Box" activity inside the Catch block to notify you when an exception is thrown.  
   A screenshot of a computer

   Description automatically generated
5. In the Finally block, add a "Message Box" activity to test if the Try-Catch activity successfully handles the exception.  
   A screenshot of a computer

   Description automatically generated
6. Save your workflow and run the sequence.

**OUTPUT:**

**Case 1:** **Notepad is running**

**A screenshot of a computer screen

Description automatically generated**

**Case 2: Notepad is not in an active state**

(In this situation, the UiPath application will pause for a duration of 30 seconds, attempting to find the Notepad window. If unsuccessful, it will then trigger an exception.)

**A screenshot of a computer message

Description automatically generatedA screenshot of a message box

Description automatically generated**

**Learnings:**

**C. Demonstrate Exception Handling:**

We understood the implementation of the "Try Catch" activity for handling exceptions, demonstrated by attempting to type into a Notepad window and catching a specific exception (SelectorNotFoundException), with notification via a "Message Box" activity. The Finally block tested the overall success of handling exceptions.