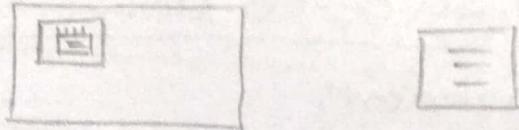


## 11. Deployment

- Deploy the recorded workflow to the desired environment (e.g., UiPath Studio, Orchestrator) for execution

### SOURCE CODE

Use Application New Text Document.txt - Notepad

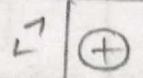


Application Path

{ "Microsoft.Windows NotePad - Week5/app" } +

Application arguments

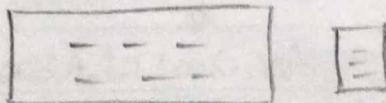
{ "Text must be quoted" }



L → DO



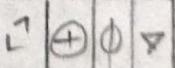
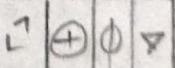
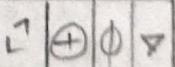
Type Into "Robot Process automation"



Type this

Standard/Secure

{ "Robotic Process automation" }



Empty field before typing

click before typing

Single line field shift+V

None

Verify that "Robotic Process - typed" [ ]

Explore.exe()

Application path  
`{}` "c:\windows\explorer.exe"

Application arguments  
`{}`

DO

A click 'RPA'

Click type: Double  Mouse button: Left

Indicate verification-target on screen

Click 'TIPS for users of Assess...'

Click type: Double  Mouse button: Left

Indicate verification-target on screen

② Chrome New Tab

Browser URL

23 "Chrome://New-tab/"

L → DO

"2= Click 'Gmail'

Gmail

Click type

Single ✓

Mouse button

left ✓

Indicate verification target on screen

"2= Click 'sent'

> Sent

Click type

Single ✓

Mouse button

left ✓

Indicate verification target on screen

## SOURCE CODE

W Browser

Browser URL: "https://www.wikipedia.org"

→ DO

Type Into "INPUT SearchInput"

Type this      Standard      Secure

{ } "Robotic Process Automation"      { } { } { }

Empty field before typing, click before typing

Single Line (Shift + V)      Single v

Verify that RPA is type

II Click Search

Click type      mouse button  
Single v      Left v

II Get Text "Robotic Process"

Save to      { } { }

{ } extracted.txt      { } { }

III Write Text file

Text      { } { }

{ } Extracted text      { } { }

write to file name

{ } "extracted.txt"      { } { }

@ Chrome All Products Books-toScrape - Send-to

Browser URL

{3 "https://books-toScrape.com/" ↵ +0}

↳ DO

Extracted Table Data

Extract to

{3 Extract ↵ +0}



{3 "Scraped.xlsx" ↵ +0}

{3 "Sheets" ↵ +0

{3 Extract ↵ +0

Browser

Application path  
{3 "Microsoft.Windows.Notepad - Pwewk" }

Application arguments  
{3 }

DO

Type Annotation

Type Into 'India, officially the R'

Type this

Empty Field before typing

Single Field (ctrl+shift+H)

Click before typing

None

Std OSecure

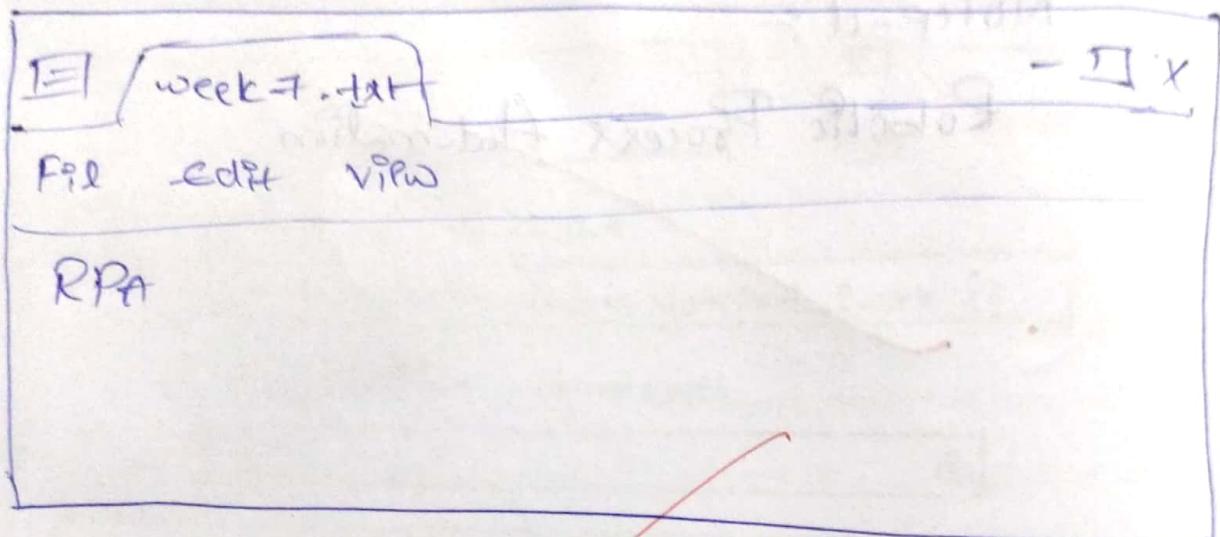
40.

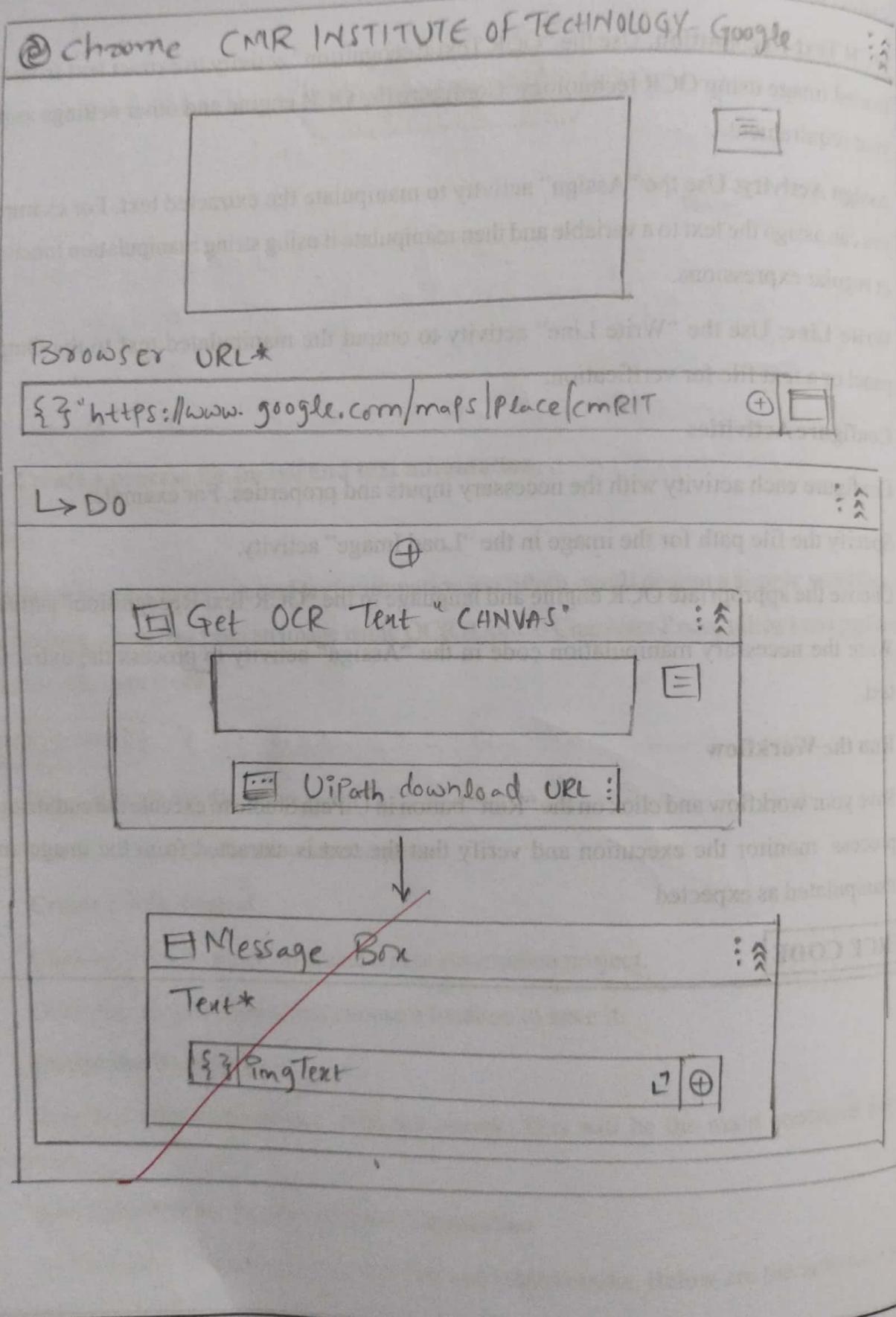
olpe

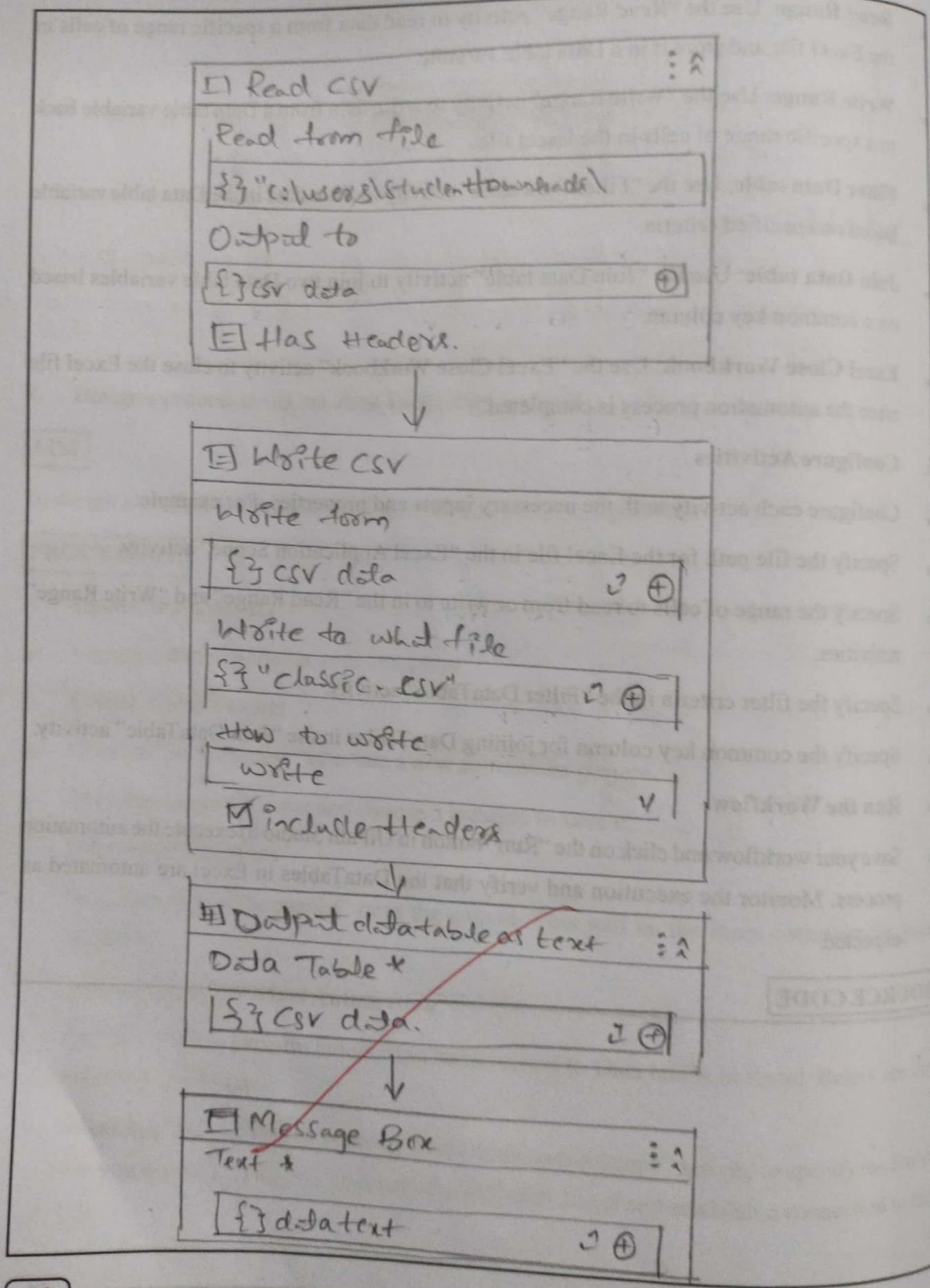
Note pad :-

Robotic Process Automation.

**OUTPUT**







Write CSV

Write form

{3 CSV data

⊕

Write to whtfile

{3 "classic.csv"

⊕

How to write

Append

✓

✓ include headers



Output datatable as text

⋮

Data Table\*

{3 CSV data

⊕



Message Box

⋮

Text\*

{3 data@text

⊕

## Message Box

X

sepal length, Sepal width, Petal length, Petal width

5.1, 3.5, 1.4, 0.2, Setosa

4.9, 3.1, 4.0, 0.2, Setosa

4.6, 3.1, 5.0, 0.2, Setosa

OK

- **For Each:** Use a “For Each” activity to iterate through each retrieved email message.
- **Send SMTP Mail Message:** Use the “Send SMTP Mail Message” activity to send an email. Configure the activity with the necessary details such as sender, recipient, subject, and body.
- **Save Attachment:** Use the “Save Attachment” activity to save attachments from email messages to a specified location on your computer.

## 5. Configure Activities

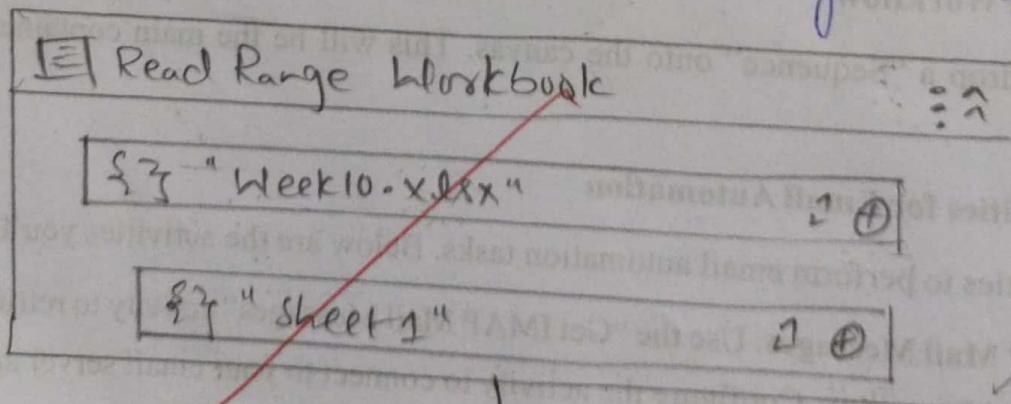
- Configure each activity with the necessary inputs and properties. For example:
- Specify the IMAP server settings and mailbox credentials in the “Get IMAP Mail Messages” activity.
- Specify the email details (sender, recipient, subject, body) in the “Send SMTP Mail Message” activity.
- Specify the folder path to save attachments in the “Save Attachment” activity.

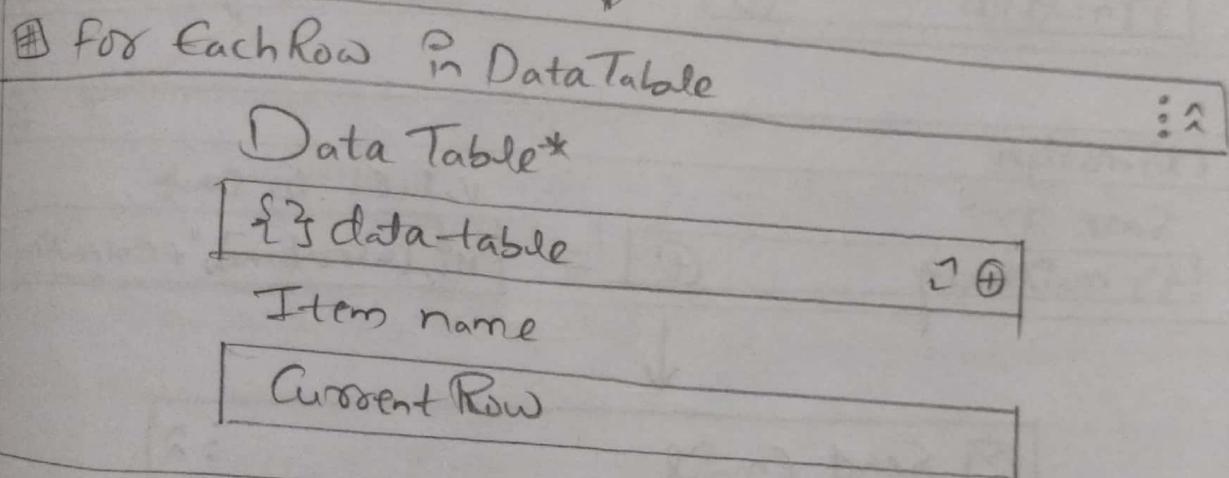
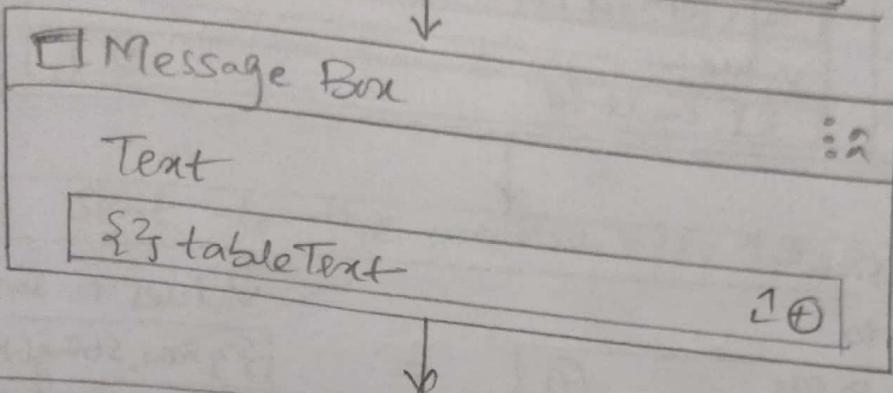
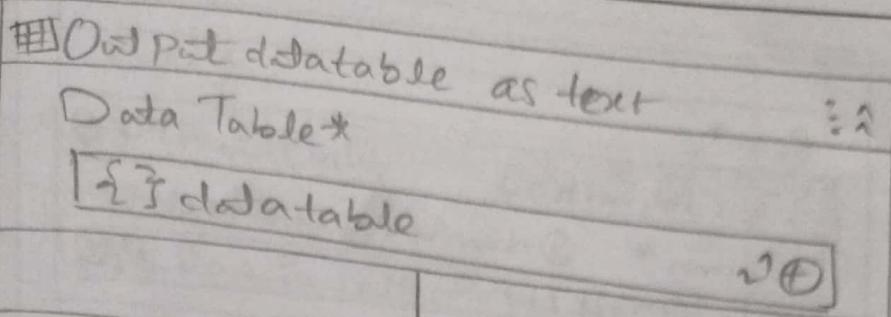
## 6. Run the Workflow

- Save your workflow and click on the “Run” button in UiPath Studio to execute the email automation process.
- Monitor the execution and verify that emails are retrieved, sent, and attachments are saved as expected.

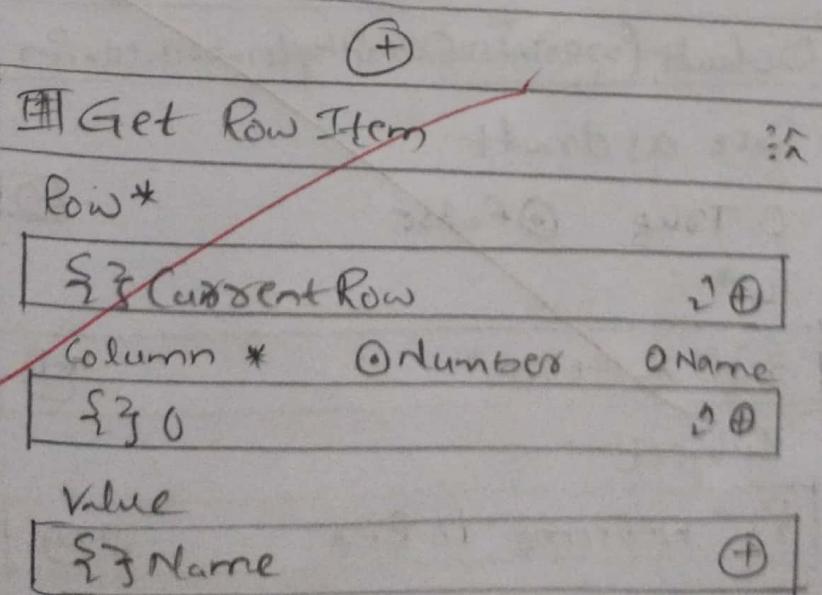
### SOURCE CODE

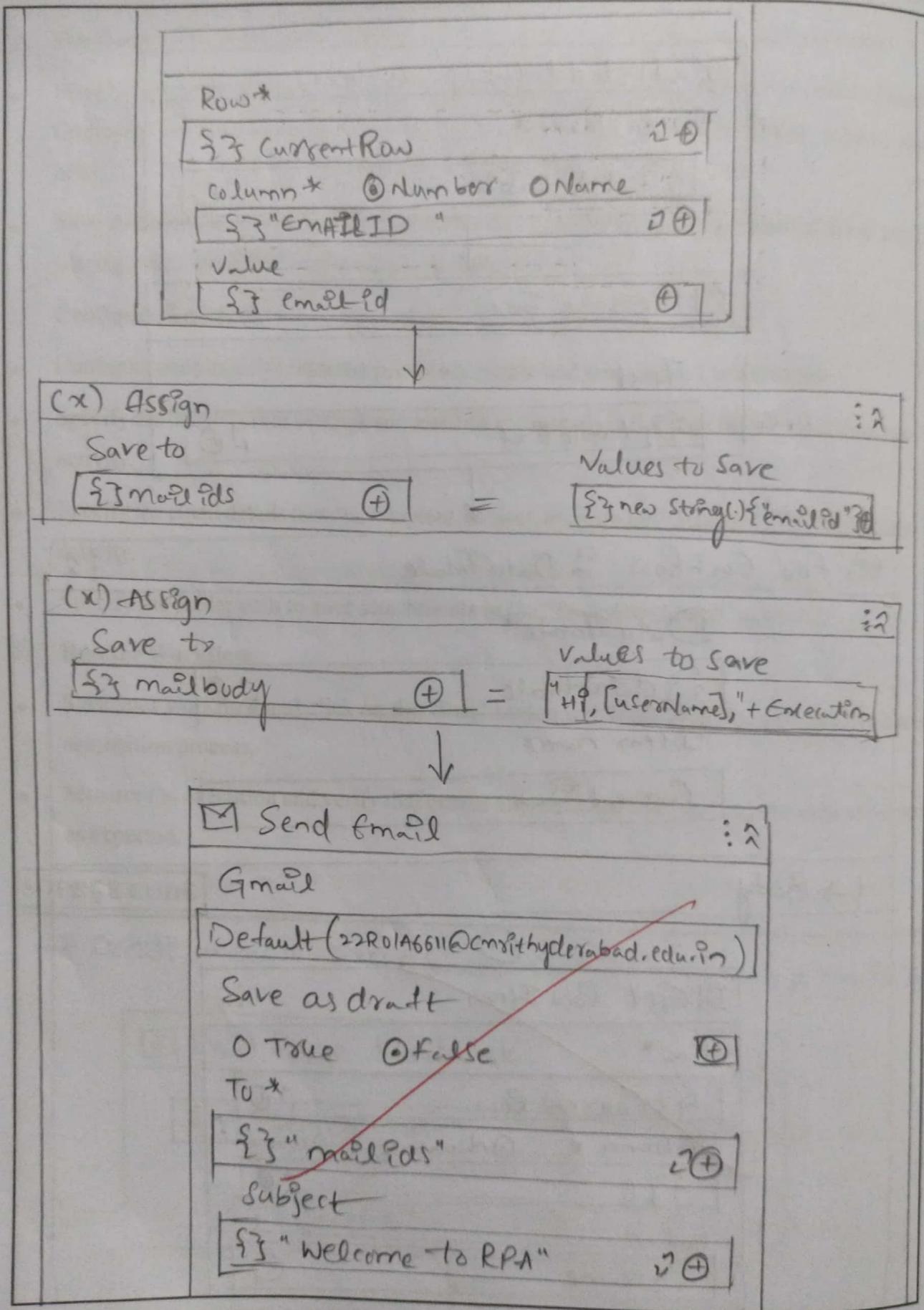
+ Create week10 Excel Sheet containing names & Email Ids





↳ Body





Body	
{? \$mailBody	10
Attachments (8)	
{? \$item in Collection	20

O/P:

mail sent to the week10.xlsx data

- **Directory.GetFiles:** Use the “Directory.GetFiles” method to get a list of all PDF files in the specified folder. Store this list in an array variable.
- **For Each:** Use a “For Each” activity to iterate through each PDF file in the array.
- **Read PDF Text:** Use the “Read PDF Text” activity to read the text from each PDF file.
- **Close Application:** Use the “Close Application” activity to close the PDF reader application (e.g., Adobe Acrobat) after reading each PDF file.

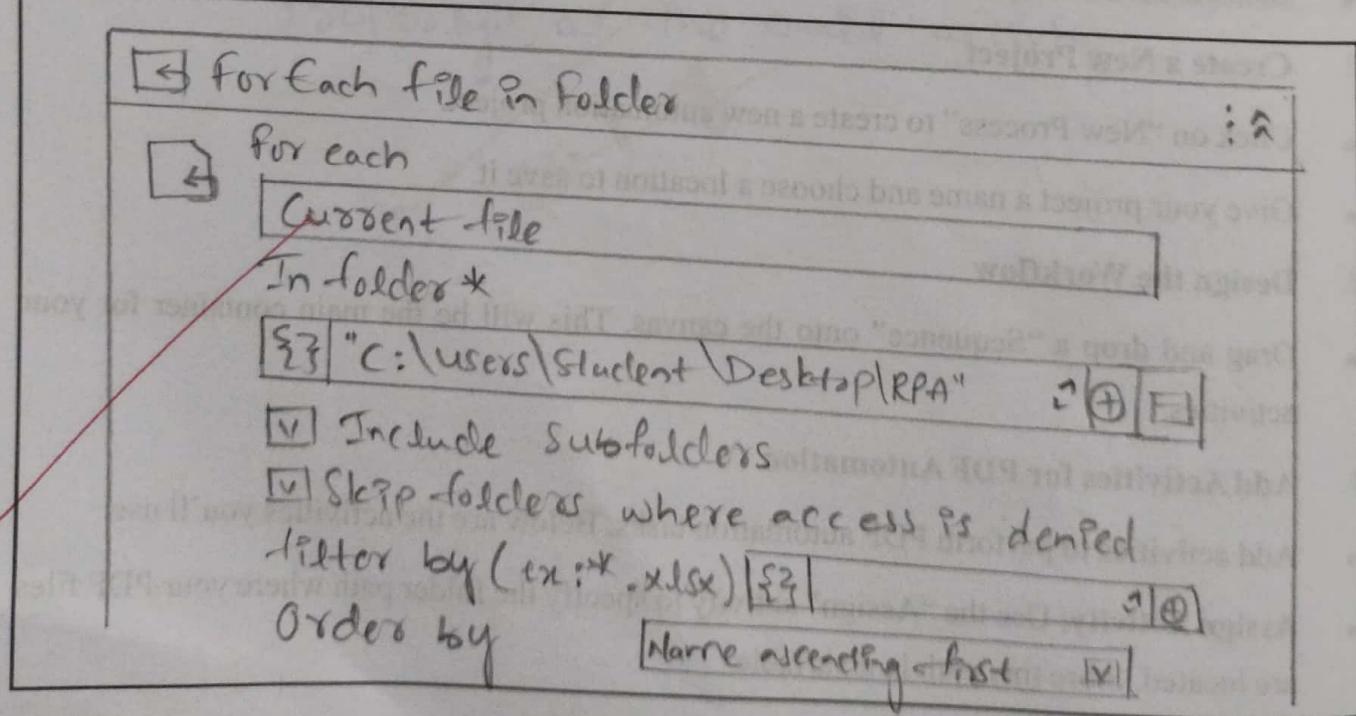
## 5. Configure Activities

- Configure each activity with the necessary inputs and properties. For example:
- Specify the folder path variable in the “Assign” activity to define the location of your PDF files.
- Specify the array variable containing the list of PDF files in the “For Each” activity.
- Configure the “Read PDF Text” activity to read the text from each PDF file.

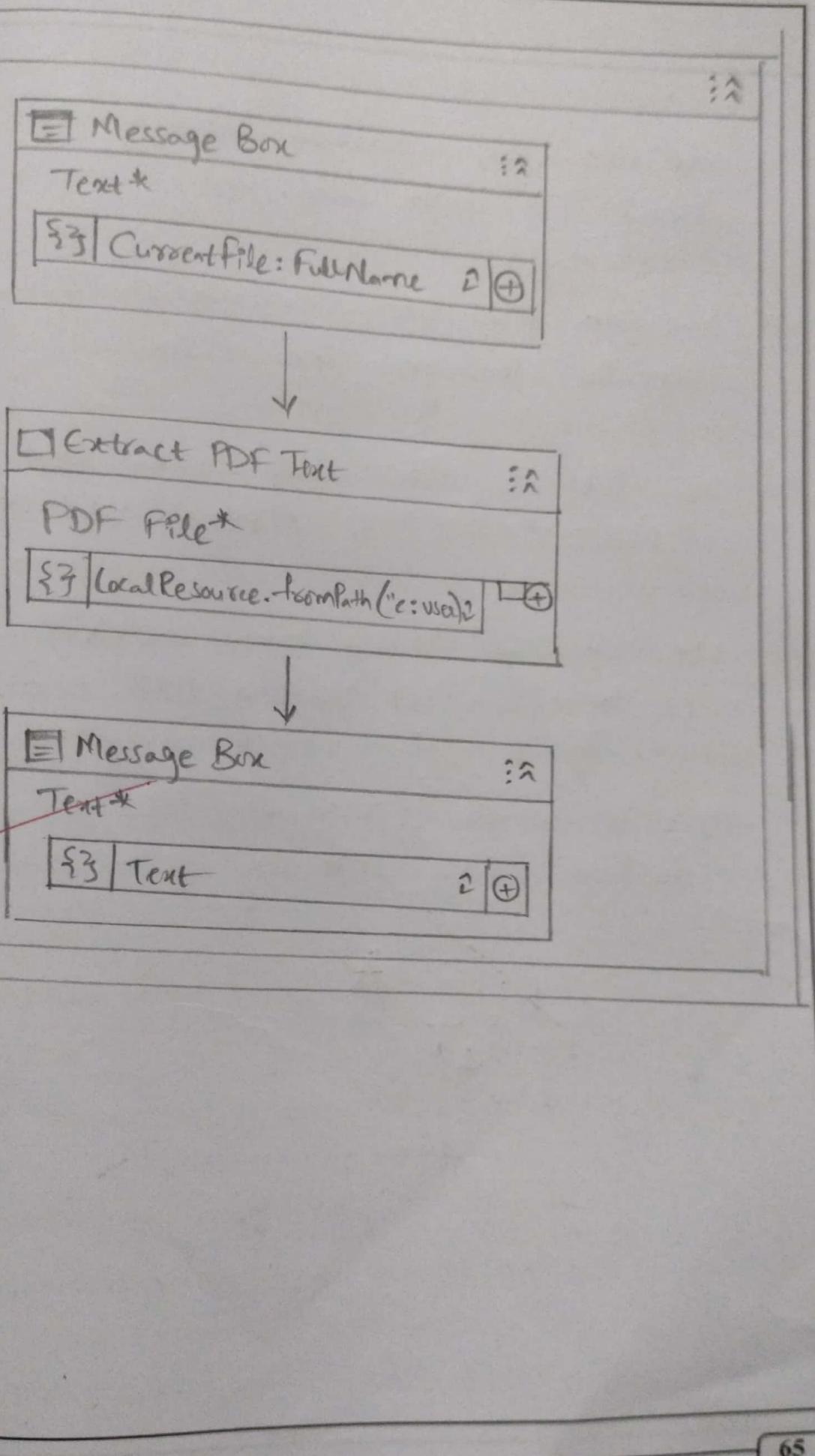
## 6. Run the Workflow

- Save your workflow and click on the “Run” button in UiPath Studio to execute the PDF automation process. monitor the execution and verify that the text is read from each PDF file, and the PDF reader application is closed after processing each file

### SOURCE CODE



L&gt;DO



- **Write Cell:** Use the “Write Cell” activity to write data to a specific cell in the Excel file. This activity can also be used to change the background color of a cell.

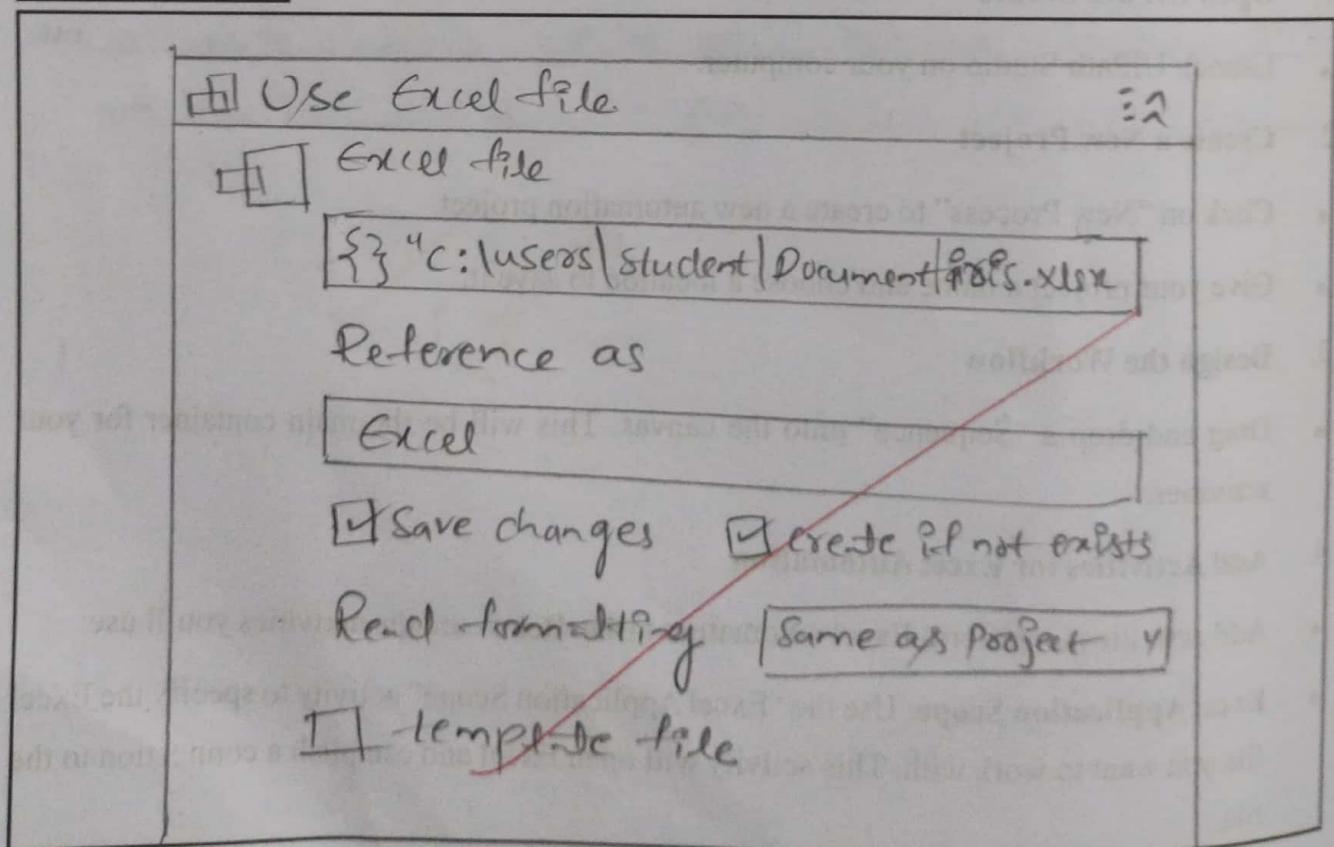
## 5. Configure Activities

- Configure each activity with the necessary inputs and properties. For example:
- Specify the file path for the Excel file in the “Excel Application Scope” activity.
- Specify the cell address (e.g., “A1”) in the “Write Cell” activity where you want to change the background color.
- Use the **Background** property of the “Write Cell” activity to set the desired background color. You can use color codes (e.g., “#FF0000” for red) or named colors (e.g., “Red”).

## 6. Run the Workflow

- Save your workflow and click on the “Run” button in UiPath Studio to execute the Excel automation process.
- Monitor the execution and verify that the background color of the specified cell is changed as expected.

### SOURCE CODE



L&gt;DO

 Set Range Color

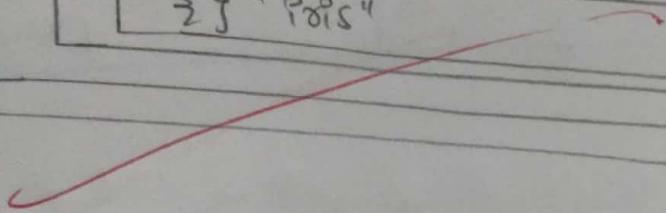
Color \*

{ } System.Drawing.Color.Blue 

Range \*

{ } "A1:A2" 

SheetName

{ } "Posis" 

Sepal length	Sepal width	Petal length	Petal width
5.1			

↙

RP

- **Create Report:** Use activities like “Build Data Table” to structure the extracted data into a report format (e.g., Excel, HTML).

## 5. Add Activities for Email Automation

- Add activities to send the report to the required recipient. Below are the activities you'll use:
- **Send SMTP Mail Message:** Use the “Send SMTP Mail Message” activity to send an email. Configure the activity with the necessary details such as sender, recipient, subject, body, and attachment (attach the generated report).

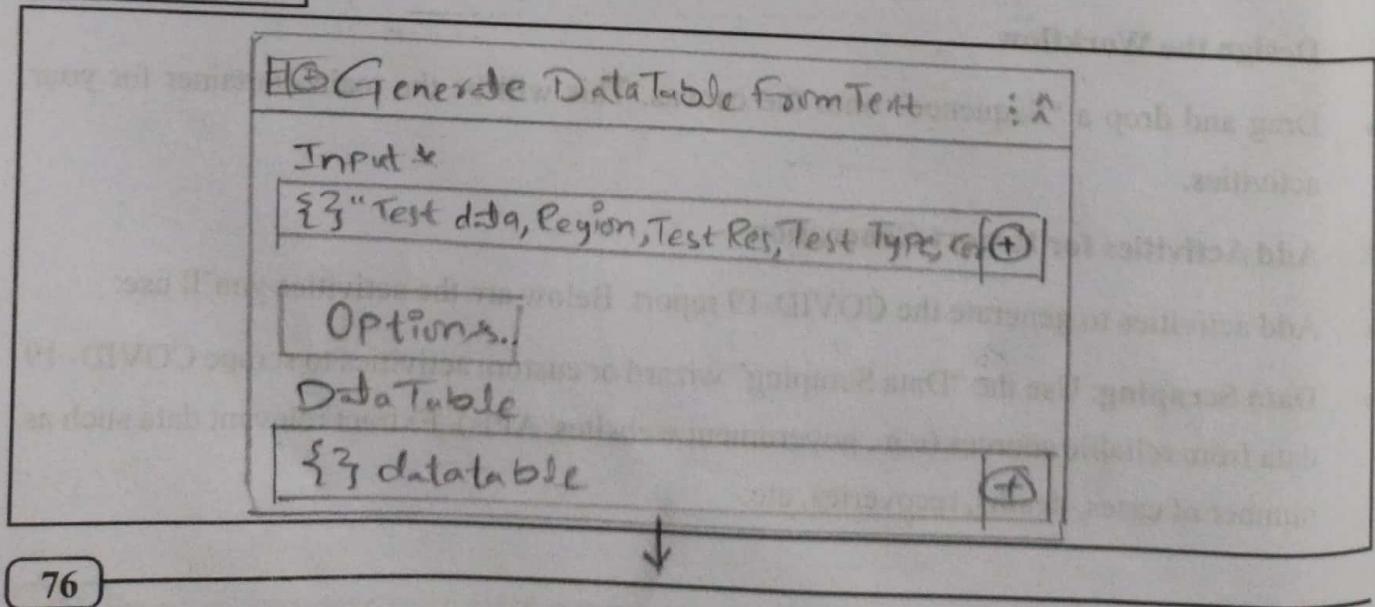
## 6. Configure Activities

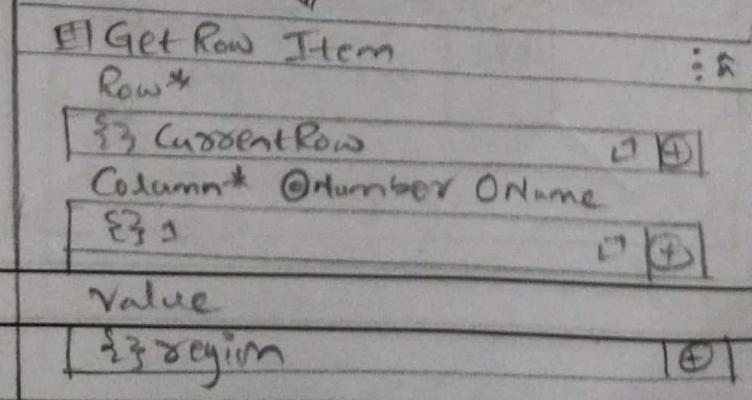
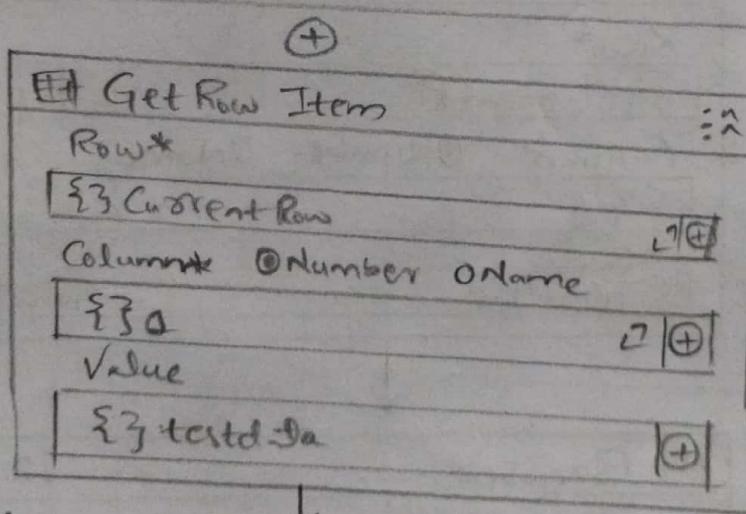
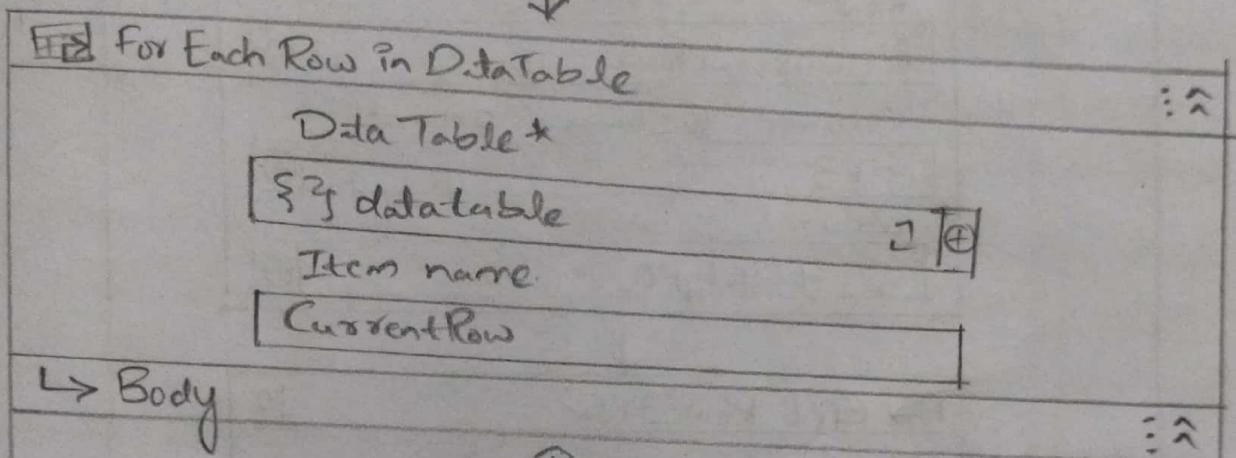
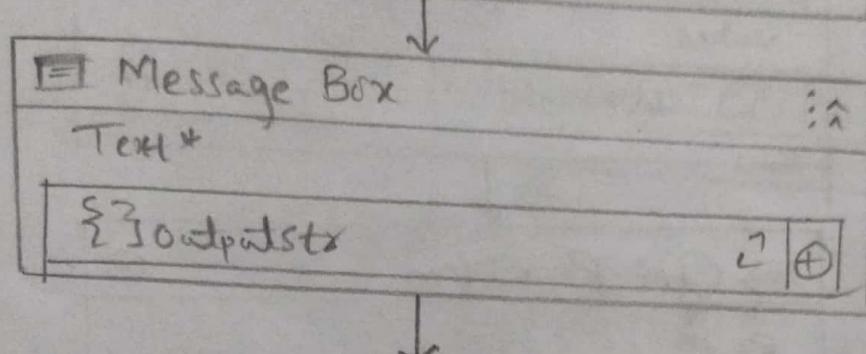
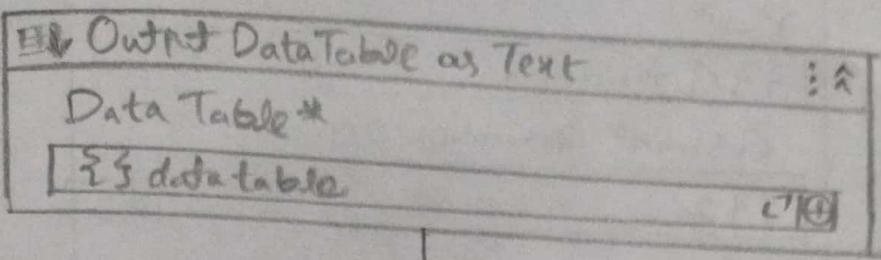
- Configure each activity with the necessary inputs and properties. For example:
- Specify the data source and extraction method in the “Data Scraping” activity.
- Structure the extracted data into a report format using the “Build Data Table” activity.
- Configure the “Send SMTP Mail Message” activity with the appropriate email server settings and recipient details.

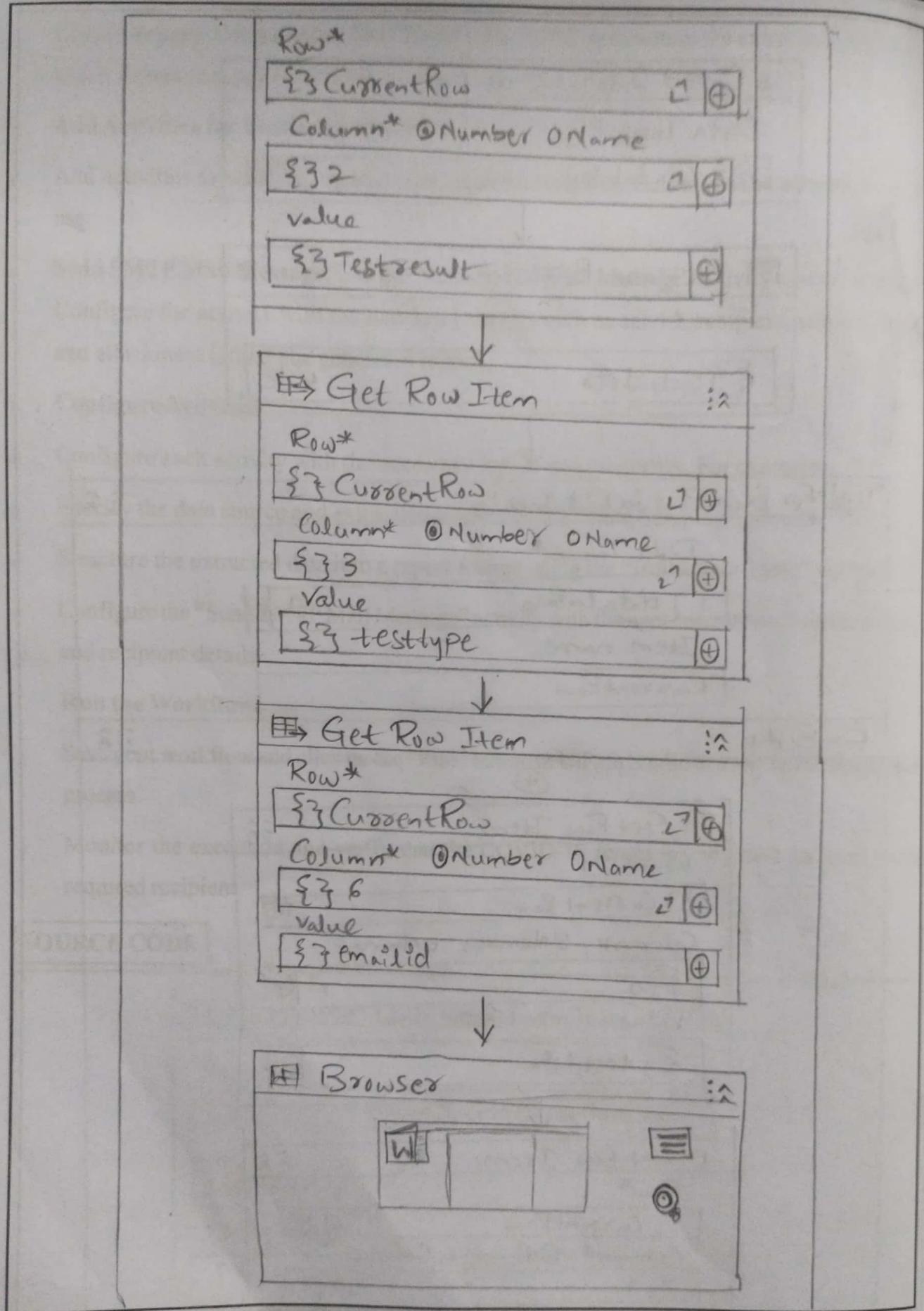
## 7. Run the Workflow

- Save your workflow and click on the “Run” button in UiPath Studio to execute the automation process.
- Monitor the execution and verify that the COVID-19 report is generated and sent to the required recipient.

### SOURCE CODE







## OUTPUT

↳ DO

Word Application Scope

File Path\*

{? "covid19.doc"} 2/⊕

↳ DO

(x) Assign

Save to

{? testastobody} ⊕ = {? "----"} ⊕

Append Text

Text\*

{? testastobody}

New Line\*

On

Save Document as PDF

File Path\*

{? "covid19.pdf"} ⊕

Replace existing

(n) Assign

Save to

{? mailArray} ⊕ = {? NewString{? EmailId3} ⊕ }

Send Email

Gmail\*

Default (28016611@gmail.com)

Save as draft

True or False

To\*

{? mailArray}

Subject

{? "COVID 19"}

Body

{? "HF" + name + ";" + Envir.NewLine} ⊕

Attachments(s)

{? Enter a VB Expression} ⊕

- **Message Box:** Use the “Message Box” activity to display a reminder message to the user. Customize the message to indicate that it’s time to take their medicine.

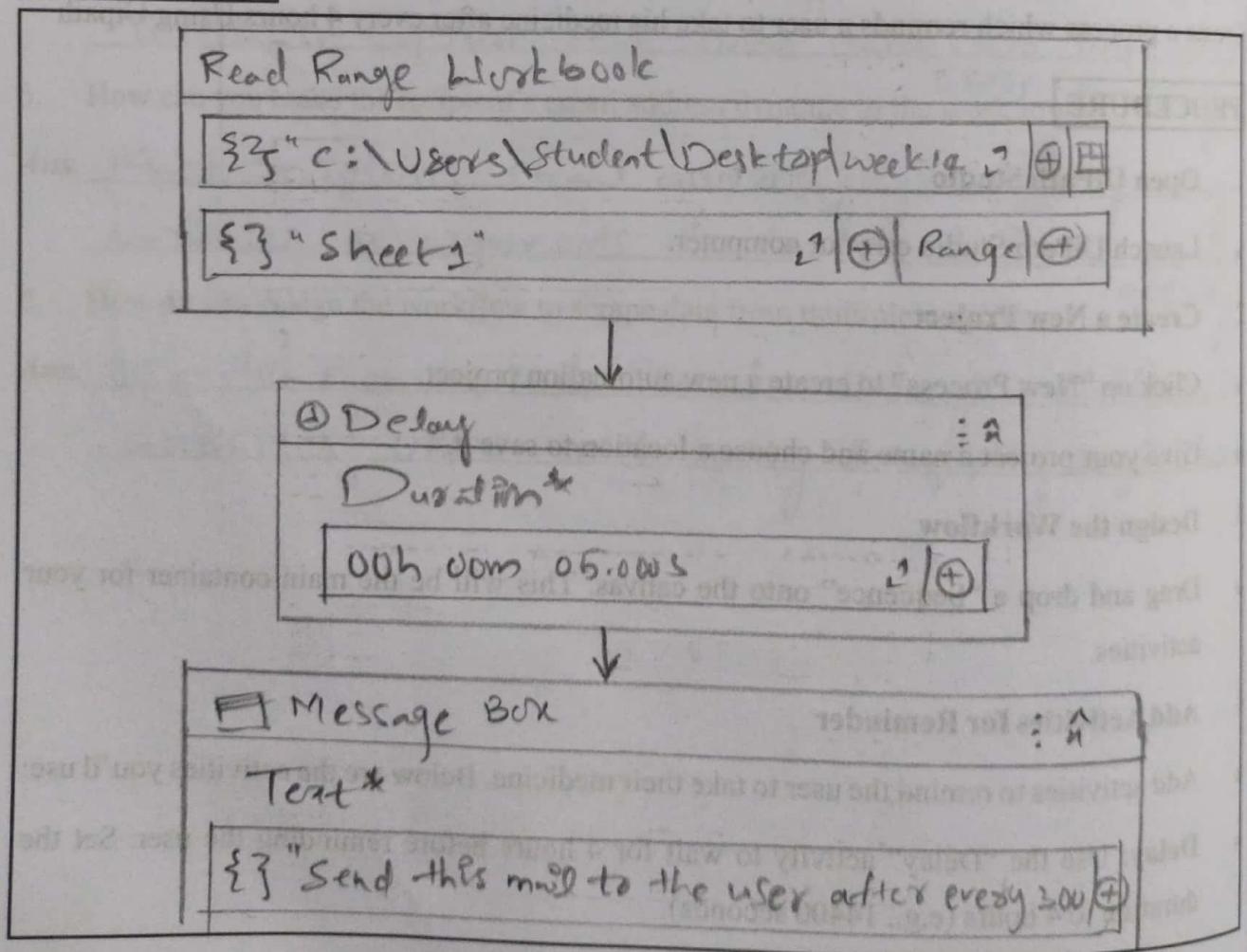
## 5. Run the Workflow

- Save your workflow and click on the “Run” button in UiPath Studio to execute the automation process.
- The automation will wait for 4 hours and then display a message box reminding the user to take their medicine.

## 6. Repeat the Process

- To ensure continuous reminders, you can add a loop around the sequence of activities so that the process repeats indefinitely.
- Use the “While” or “Do While” activity to create a loop that continuously executes the reminder activities every 4 hours

## SOURCE CODE



for EachRow in Datasource

DataTable\*

{ } datatable

2 (+)

Item name

Current Row

↳ Body

El GetRow Item

Row\*

{ } CurrentRow

2 (+)

Column\* @Number @Name

{ } 0

2 (+)

Value

{ } Username

(+)



Get Row Item

:=>

Row\*

{ } CurrentRow

2 (+)

Column\* @Number @Name

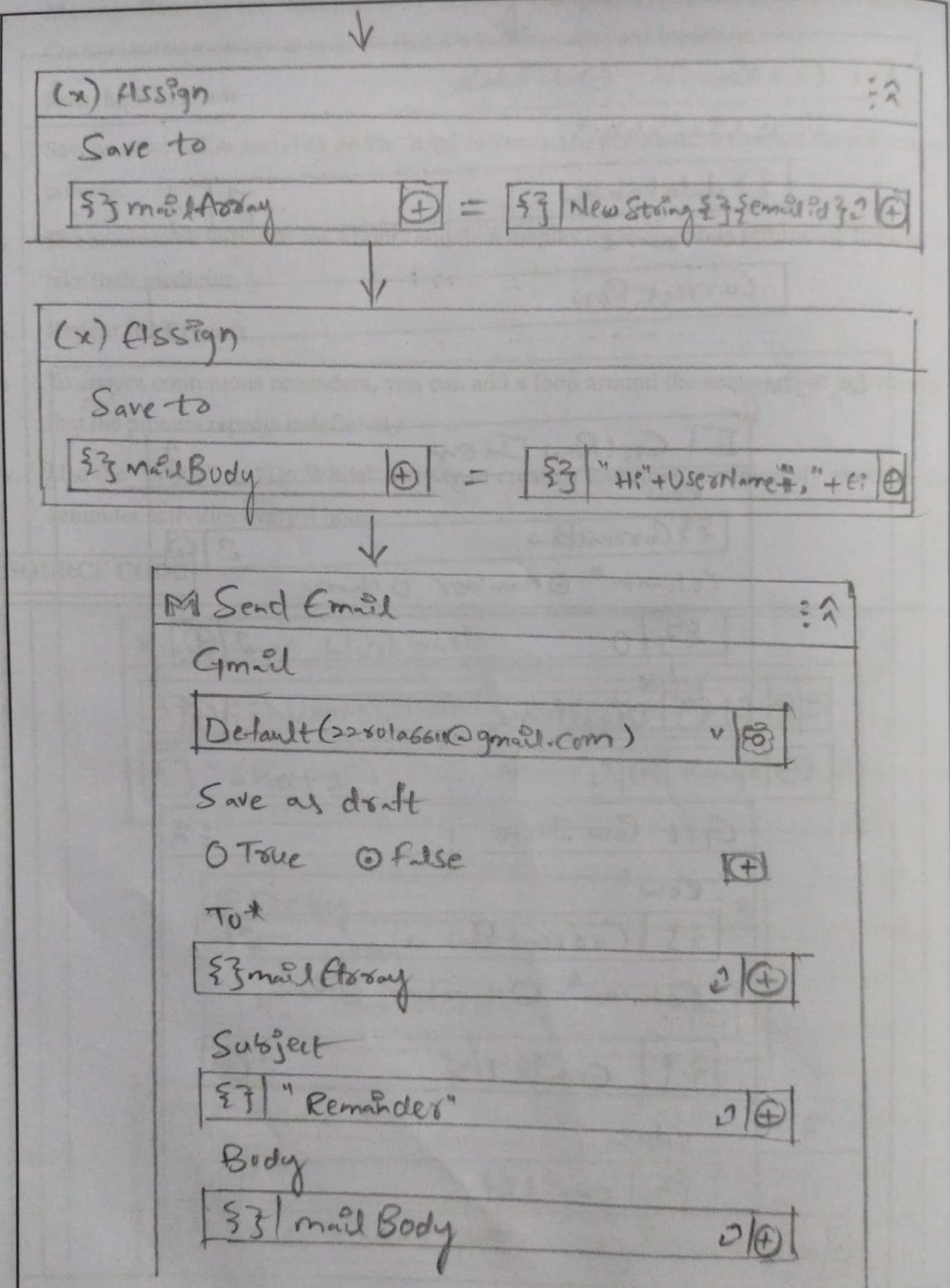
{ } "Email ID"

? (+)

Value

{ } emailId

(+)

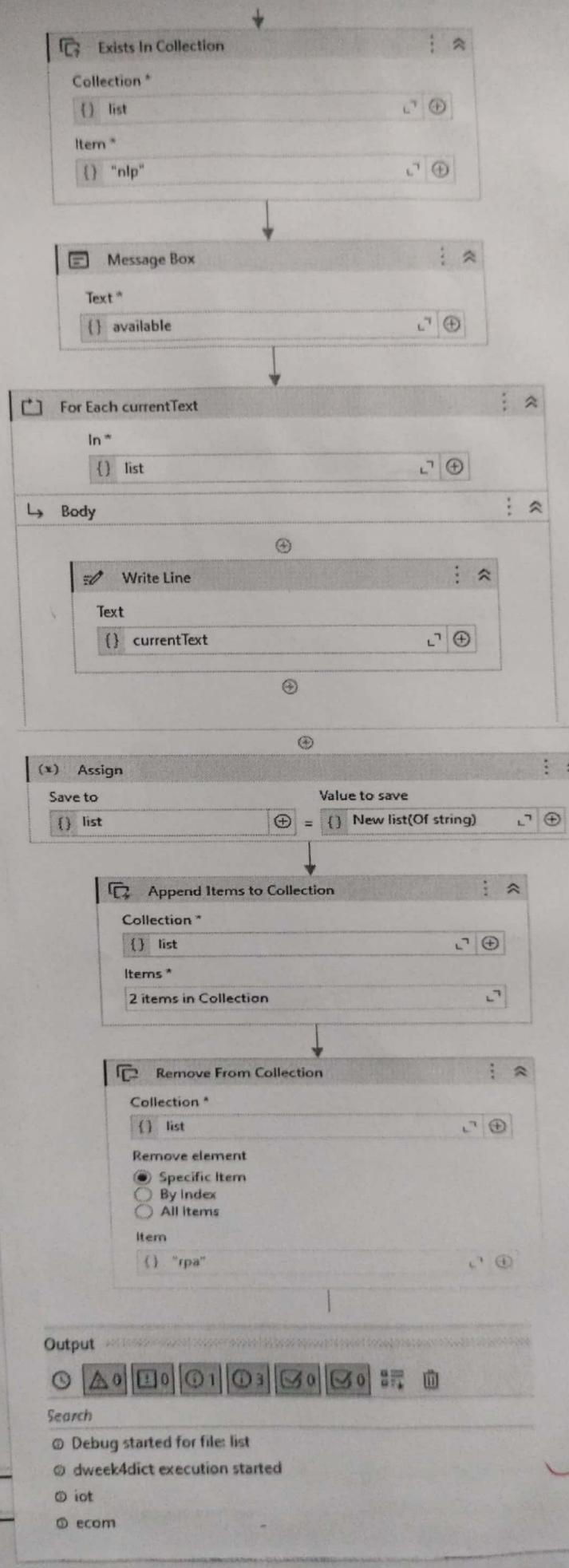


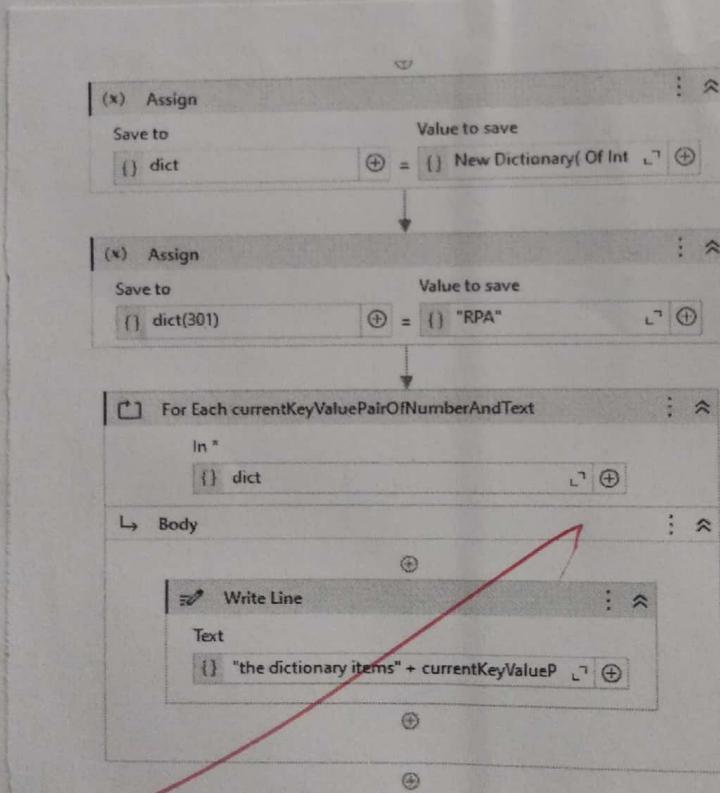
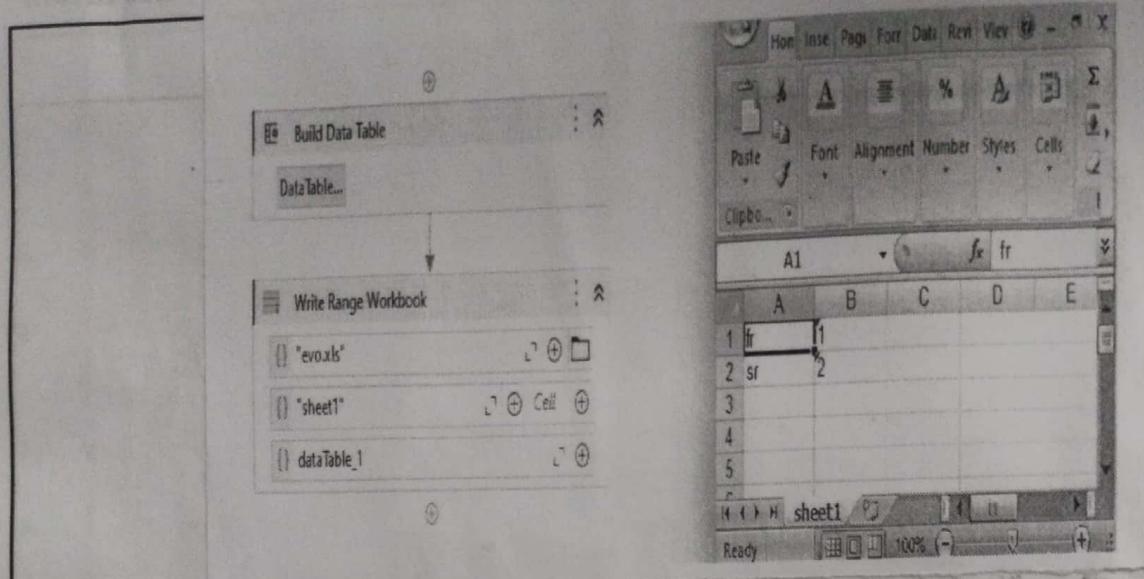
## OUTPUT

## Message Box

Send this email to the user

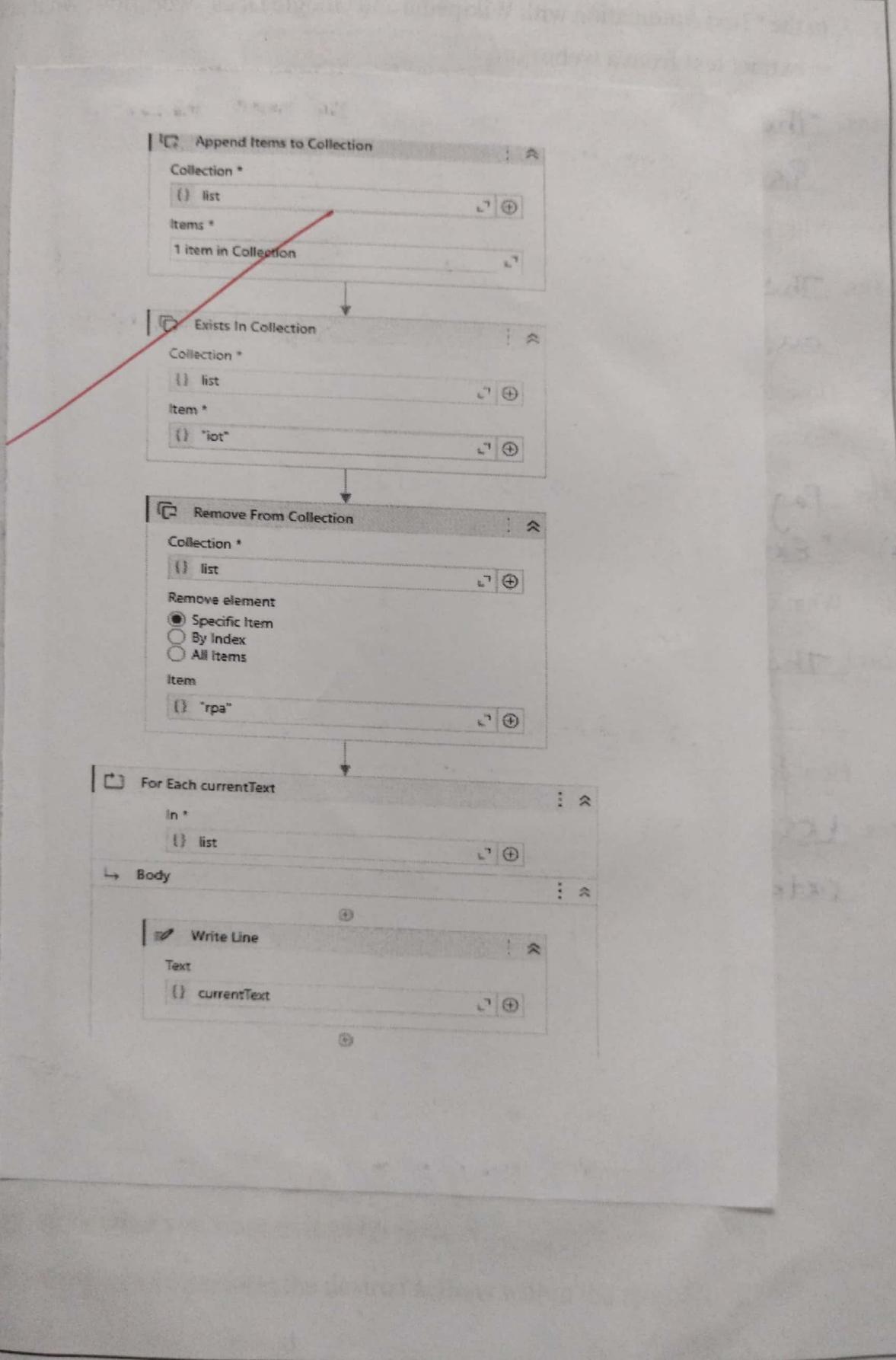
after every 300s





- ① dweek4dict execution started
- ② the dictionary items[301, RPA]
- ③ the dictionary items[302, IOT]
- ④ the dictionary items[303, NLP]
- ⑤ the dictionary items[304, AML]
- ⑥ the dictionary items[305, ECOM]
- ⑦ dweek4dict execution ended in: 00:00:19

OUTPUT



**Input Dialog**

Dialog Title  
{\$} "enter a value" ↗⊕

Input Label  
{\$} "a value is" ↗⊕

Input Type  
Text Box ✓

Value entered  
{\$} a ⊕

**Message Box**

Text \* ↗⊕

{\$} a ↗⊕

**Input Dialog**

Dialog Title  
{\$} "enter b value" ↗⊕

Input Label  
{\$} "b value is" ↗⊕

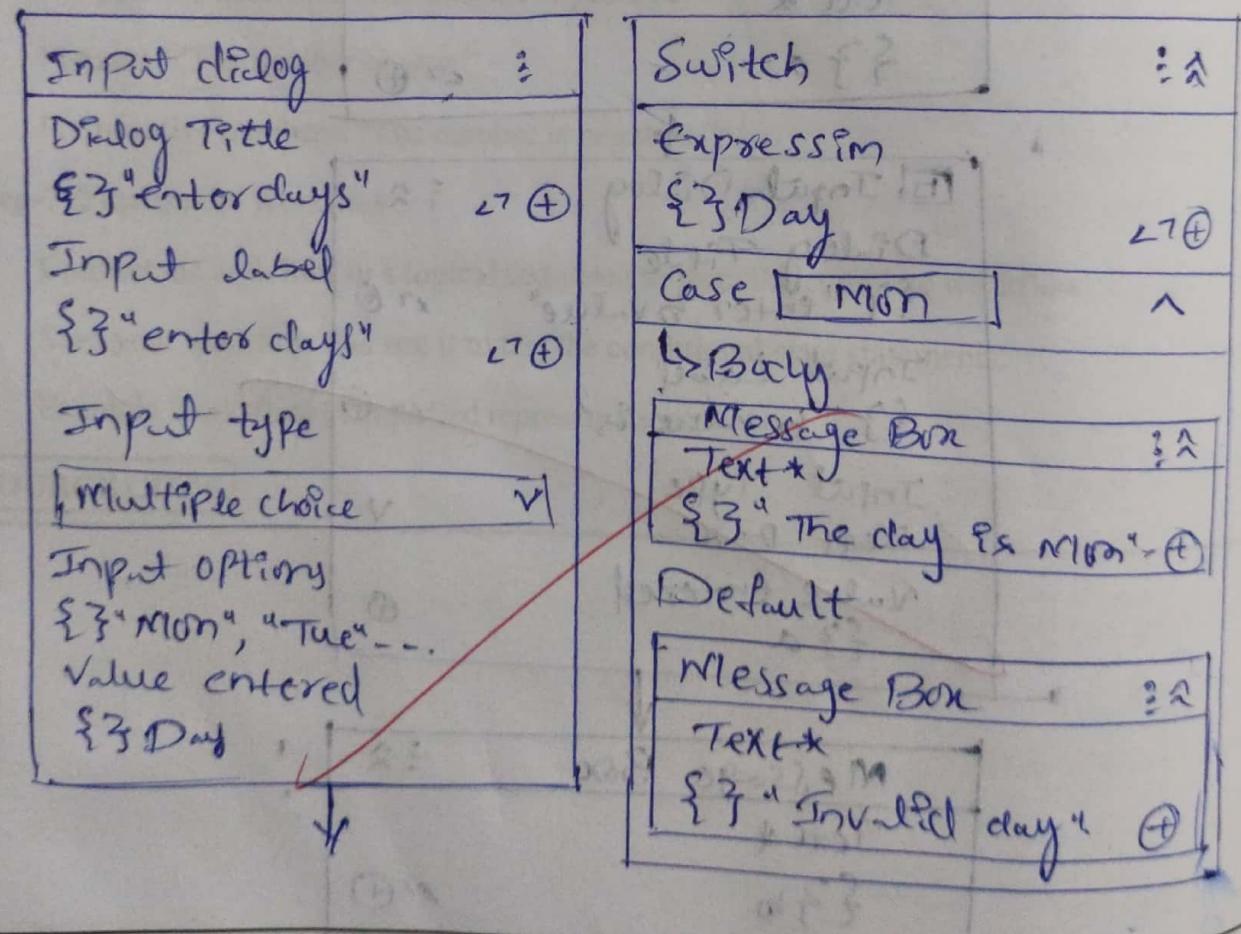
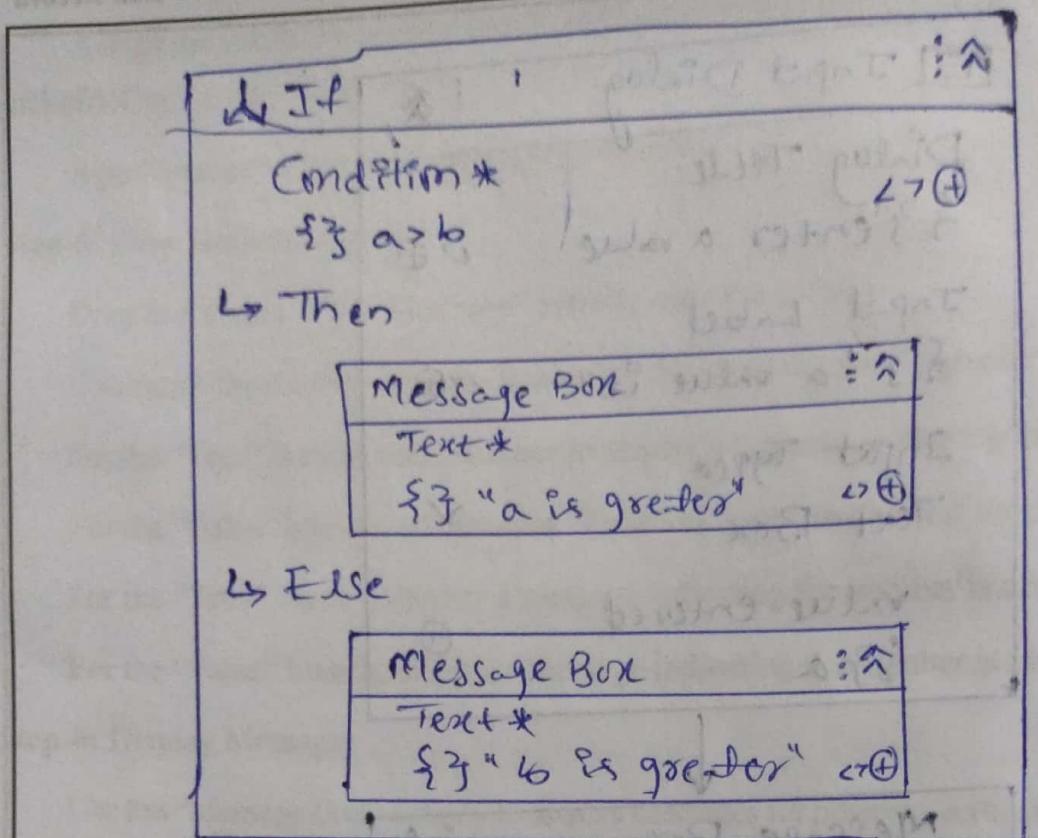
Input Type  
Text Box ✓

Value entered  
{\$} a ⊕

**Message Box**

Text \* ↗⊕

{\$} b ↗⊕



## OUTPUT

