

Candidate Onboarding Project

edureka!

edureka!

Index

Key Points	1
Candidate Onboarding Project Part 1	2
Candidate Onboarding Project Part 2	6
Candidate Onboarding Project Part 3	17
Candidate Onboarding Project Part 4	23
Candidate Onboarding Project Part 5	31
Candidate Onboarding Project Part 6	38
Candidate Onboarding Project Part 7	44
Sequences & Variables of The Project	47

Key Points

Note – Due to frequent automatic updates of UiPath, the UI of the tool is subjected to change. All the demos in the course will always be executed on the latest version of the community edition by the instructor. While referring to these documents, we recommend following the logical steps explained in the documents.

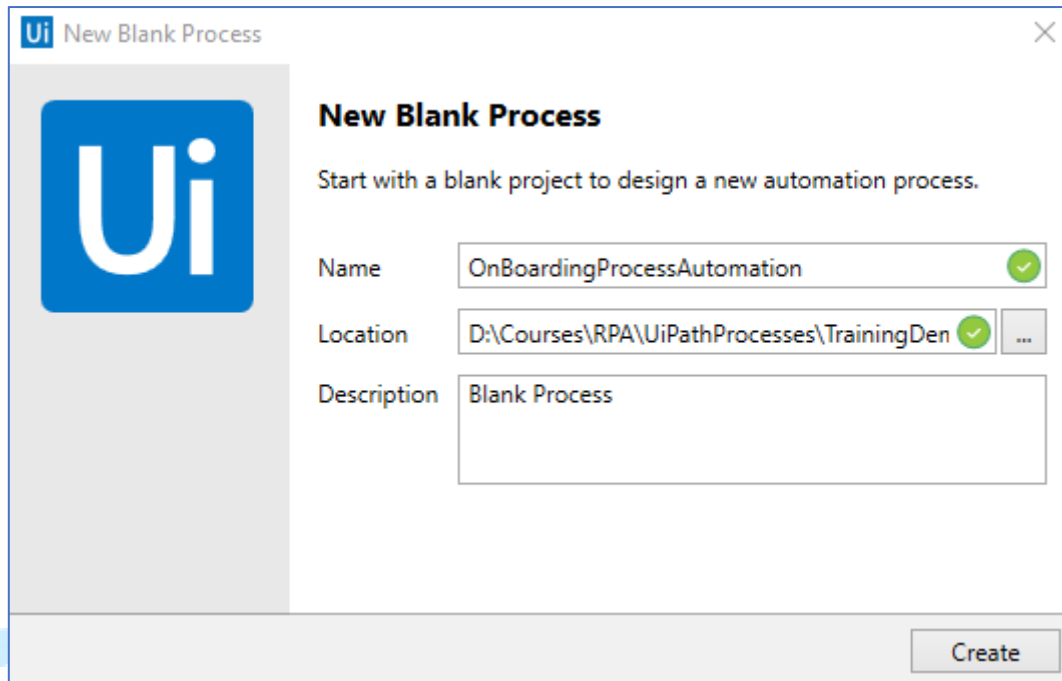
- **Google Chrome** is recommended for all your web actions for this course
- Make sure your browser does not automatically download the excel file into a default folder (Downloads). We need this data in the same folder in which we have our project. If this situation exists, change your browser settings to, “**Ask before downloading**”
- For Gmail users make sure to **enable “Access Less Secure Apps”** from your account settings and **IMAP protocol**, so that UiPath can access it.
- For any other accounts, you can google your SMTP/IMAP credentials and you must make sure that **SMTP/IMAP permissions are granted by your Mail account**
- Credentials for Gmail account are :

Protocol	Port	URL
SMTP	587	“smtp.gmail.com”
IMAP	993	“imap.gmail.com”

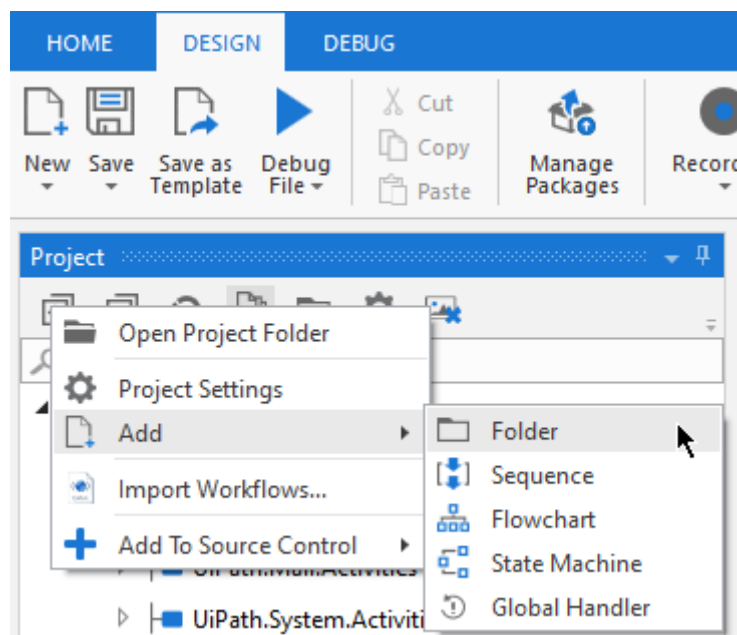
- For all Excel Activities, your excel window and all the further processes of excel should be inside Excel Application Scope
- While reading Excel File, if your workflow gets stuck, change your excel settings to “Enable Editing”
- If the PDF activities don’t appear in the Activities panel, then install **UiPath.PDF.Activities** package using Manage Packages option
- Check the type and scope of your variables and activity blocks
- Update the selectors, if the elements are not detected

Candidate Onboarding Project Part - 1

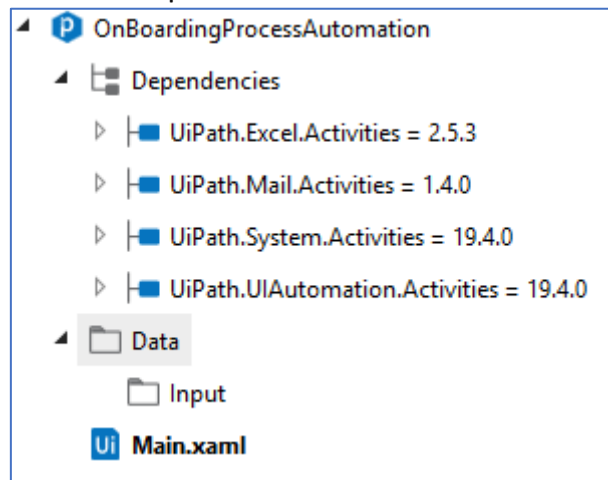
1. Create a Transactional Process named ***OnBoardingProcessAutomation***



- Right click the Expand All icon and choose Add New Folder
- Enter the name as Data



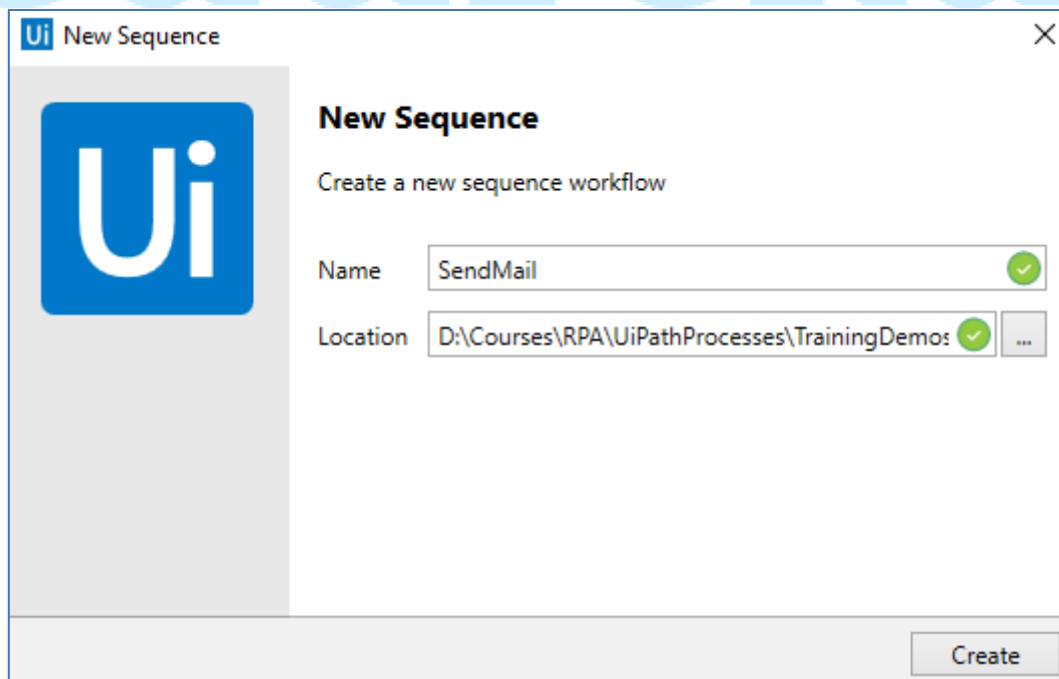
- Right Click Data folder and again select Add New Folder
- Enter the name as Input



2. Create the following sequences:

- SendMail**
- Init**
- KillProcess**
- DownloadCandidateMaster**

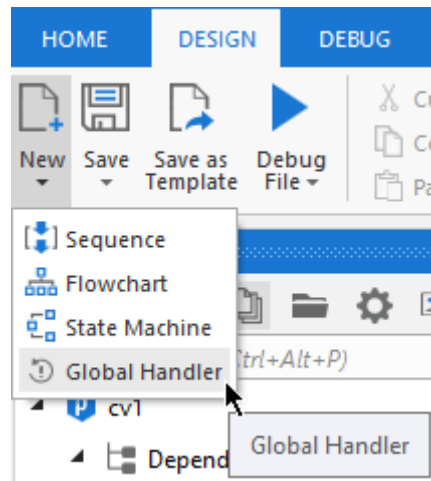
- Go to Design Tab, select new > Sequence, type SendMail in Name and click create



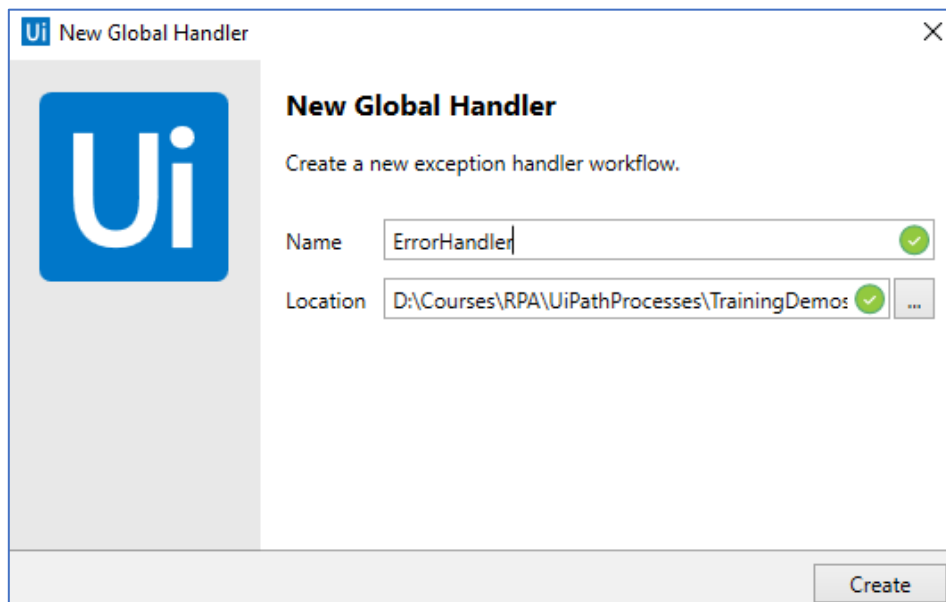
3. Create a Global Handler

- Click the New button
- Select Global Handler

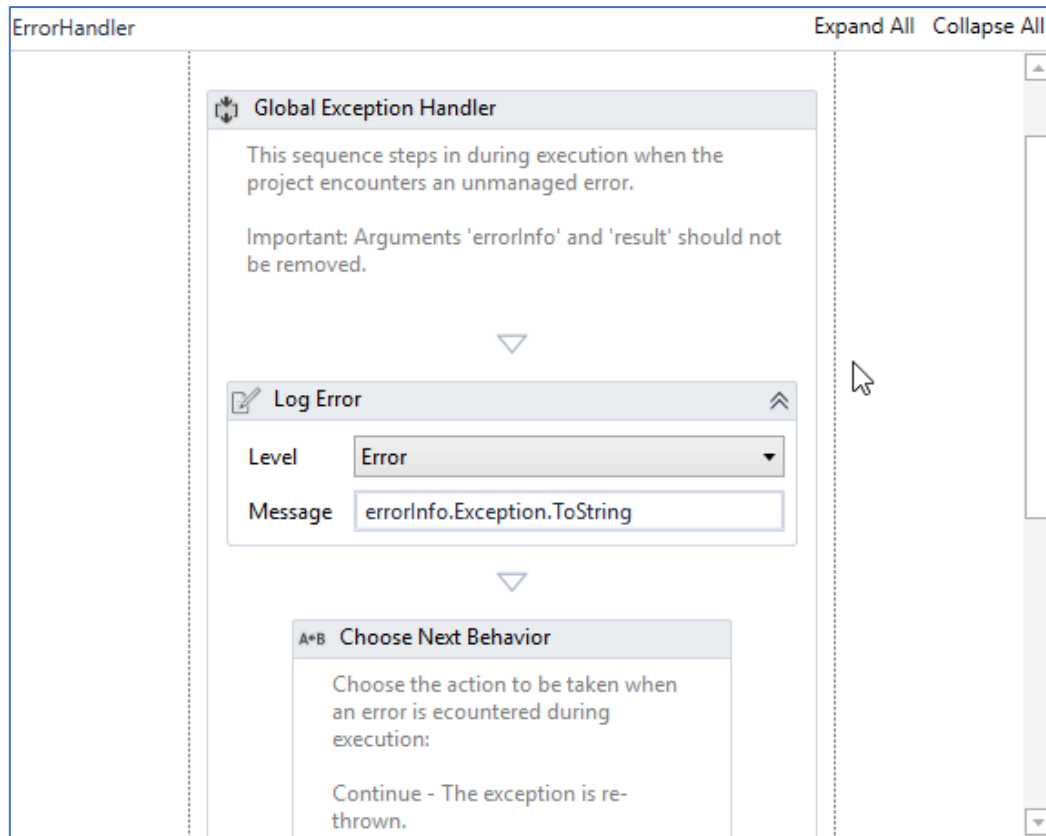
- Enter the name as ErrorHandler
- Click Create Click the New button
- Select Global Handler
- Enter the name as ErrorHandler
- Click Create



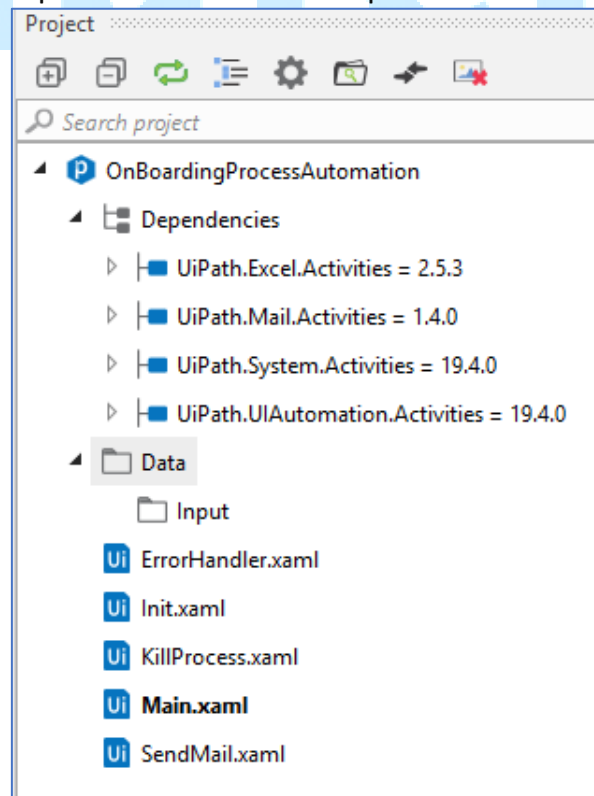
edureka!



- The Global Handler looks like this

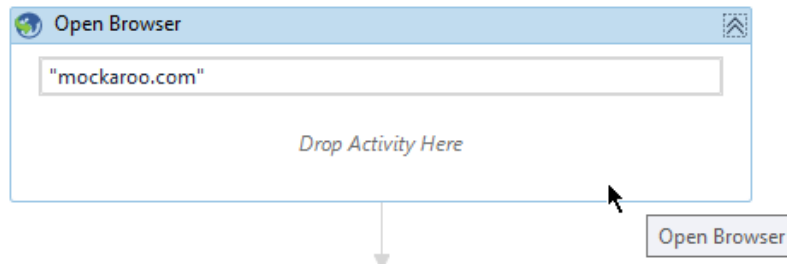


- The project panel should look as depicted below

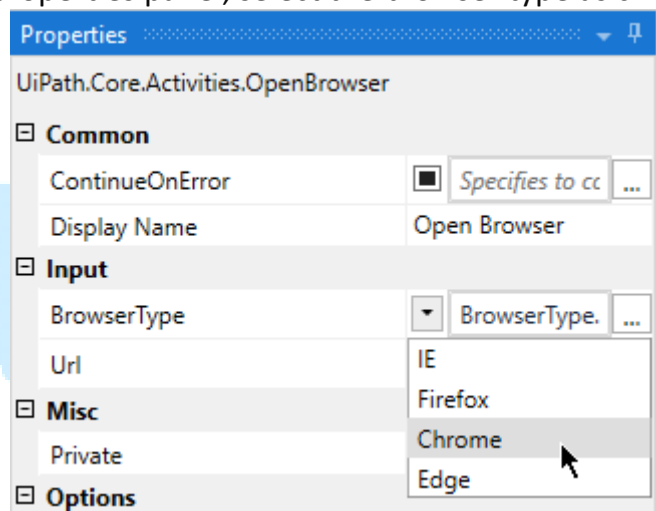


Candidate Onboarding Project Part - 2

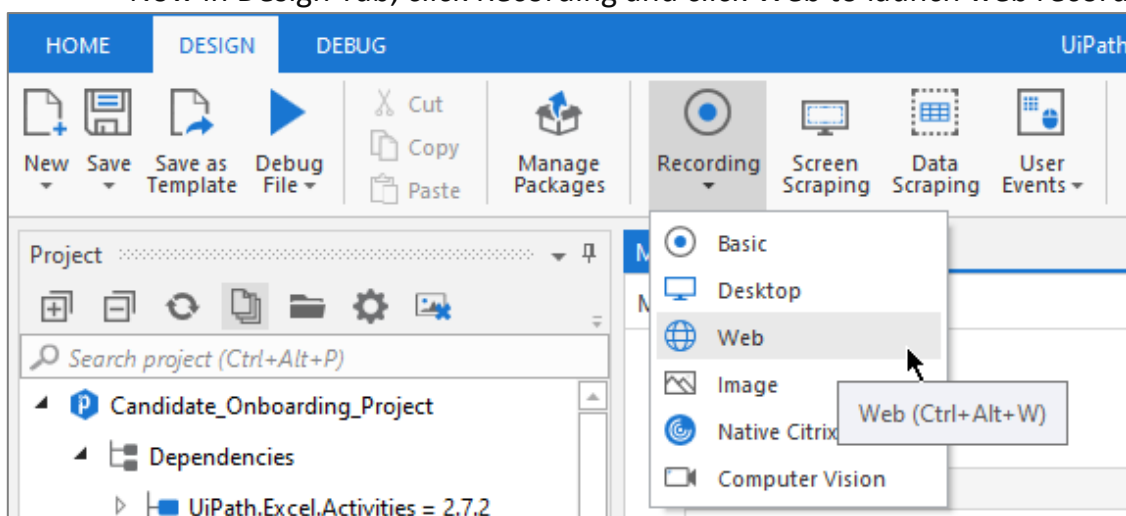
4. We will design our workflow to pull the candidate data from Mockaroo website
5. Open Google Chrome and open www.mockaroo.com
6. Double click DownloadCandidateMaster Sequence and drag an open browser activity and enter the url (In Double Quotes)



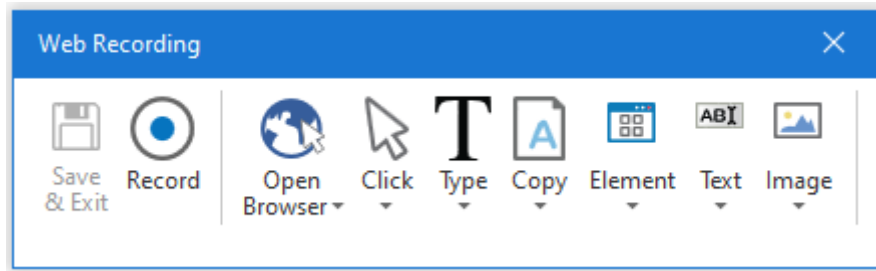
- In the properties panel, select the browser type as chrome



- Now in Design Tab, click Recording and click Web to launch web recorder



- Web Recording options will be displayed as shown below



- Click x beside IP address field to remove it
- In the indicate anchor window, click No

The image shows the Mockaroo web application interface. At the top, there's a header with a navigation menu: (ESC) - Cancel Selection, (F2) - Select after Delay, (F3) - Select Region, (F4) - UI Framework: Default. Below the header, there's a table with columns: Field Name, Type, and Options. The table contains the following rows:

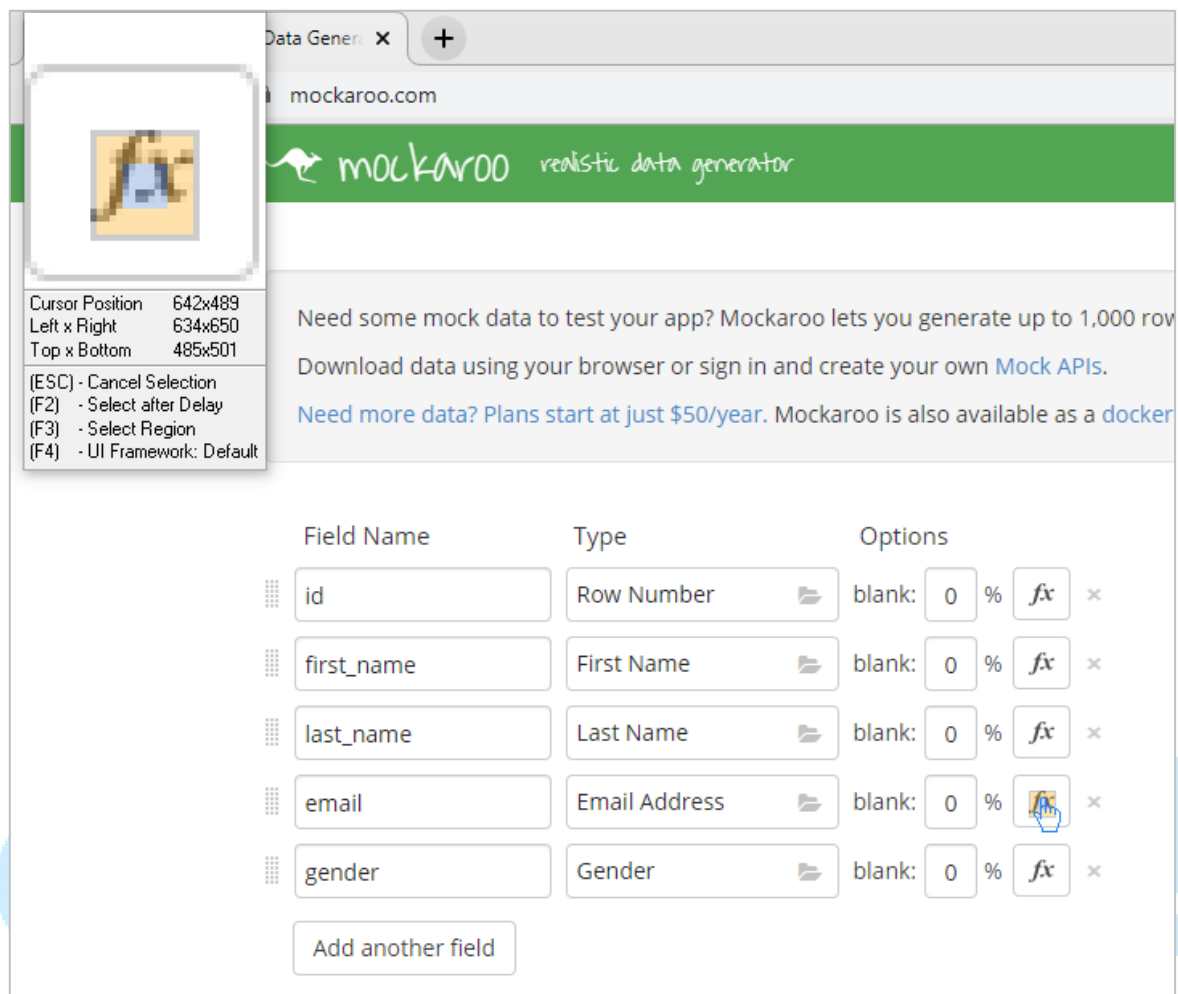
Field Name	Type	Options
id	Row Number	blank: 0 % fx x
first_name	First Name	blank: 0 % fx x
last_name	Last Name	blank: 0 % fx x
email	Email Address	blank: 0 % fx x
gender	Gender	blank: 0 % fx x
p_address	IP Address v4	blank: 0 % fx x

Below the table, there's a button 'Add another field'. At the bottom, there are settings for '# Rows: 1000', 'Format: CSV', 'Line Ending: Unix (LF)', and 'Include: ☒ header ☐ BOM'. There are also buttons for 'Download Data', 'Preview', and 'More'. A link 'Want to save this for later? Sign up for free.' is at the bottom right.

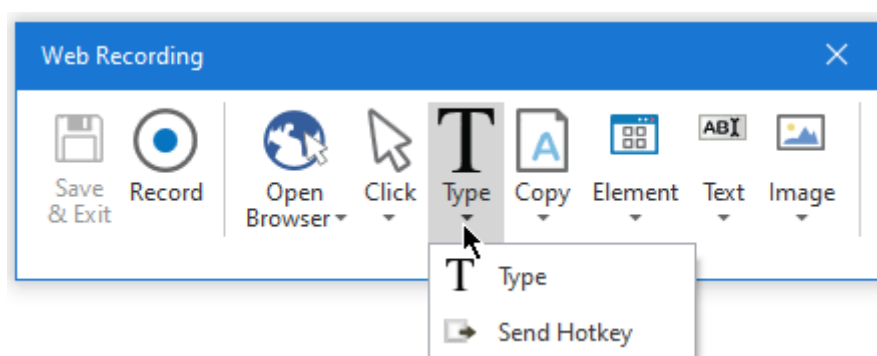
- Use **Click** option in web recorder



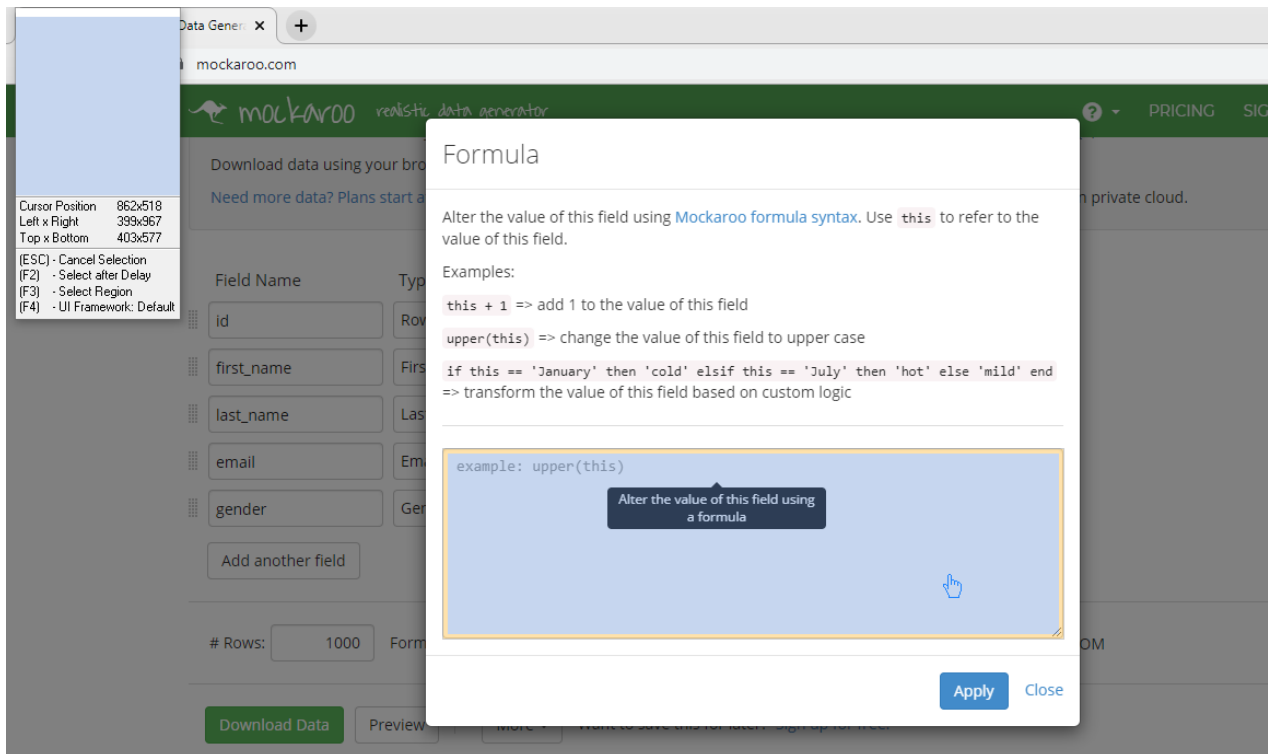
- Click fx against email, this is done to include a valid email address in our input data



- Click **Type** in Web Recorder



- Indicate the box to type email id



- Type your email id in **Single Quotes** for e.g. **'FirstName_LastName@edureka.com'**
- Check Empty Field check box
- Press Enter

Formula

Alter the value of this field using [Mockaroo formula syntax](#). Use `this` to refer to the value of this field.

Examples:

`this + 1` => add 1 to the value of this field

`upper(this)` => change the value of this field to upper case

`if this == 'January' then 'cold' elsif this == 'July' then 'hot' else 'mild' end`
=> transform the value of this field based on custom logic

example: `upper(this)`

Type the desired value

'edureka_training@edureka.com'

add

alt

lalt

ralt

back

break

caps

☐ Type password ☒ Empty field

Press enter when you are done

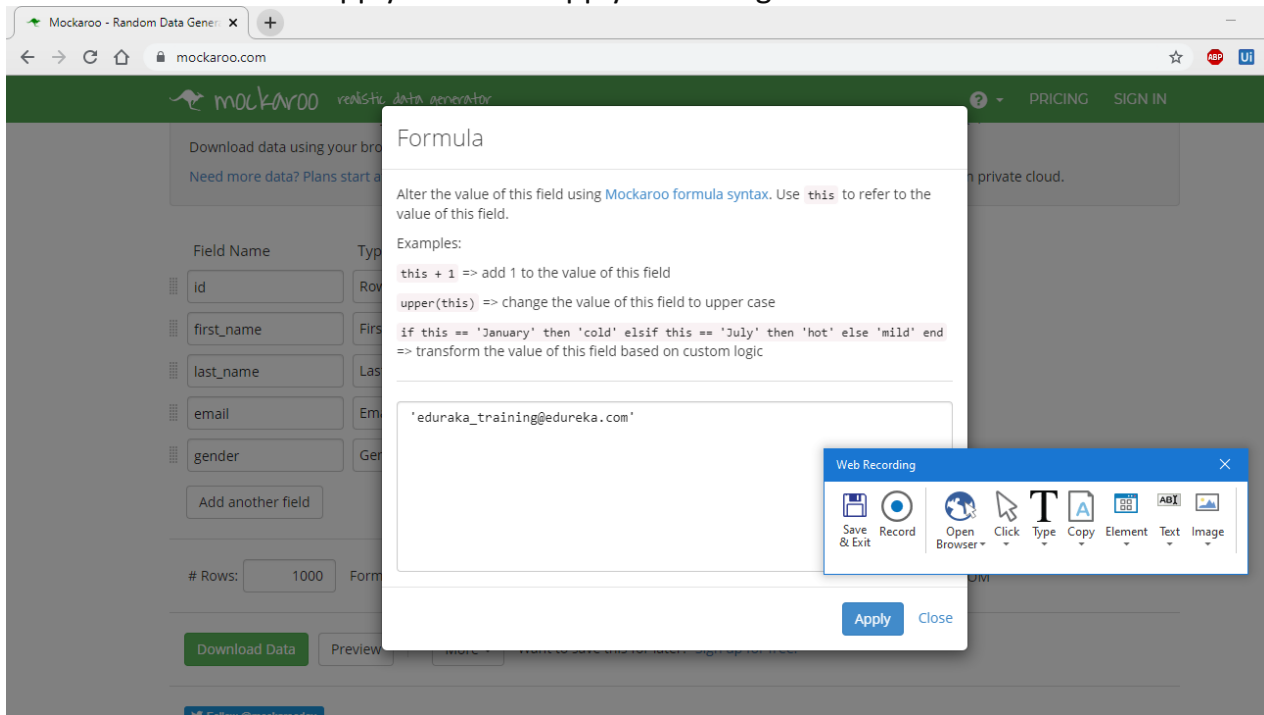
Apply

Close

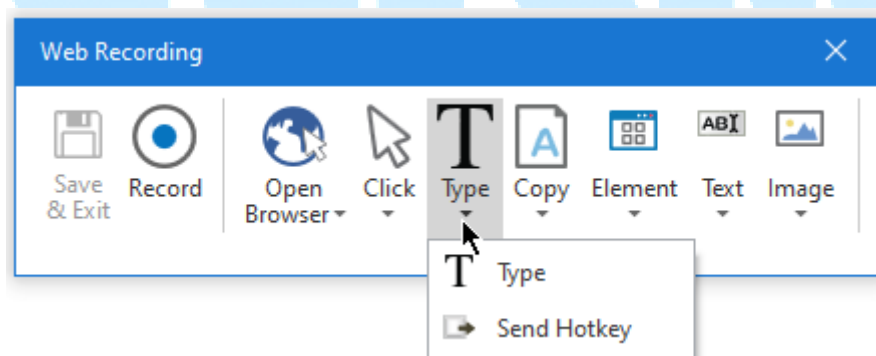
- Choose Click > Click in Web recorder



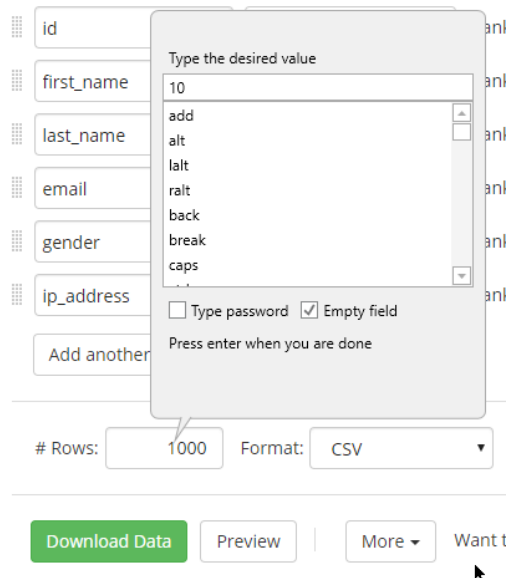
- Indicate Apply button to apply the change



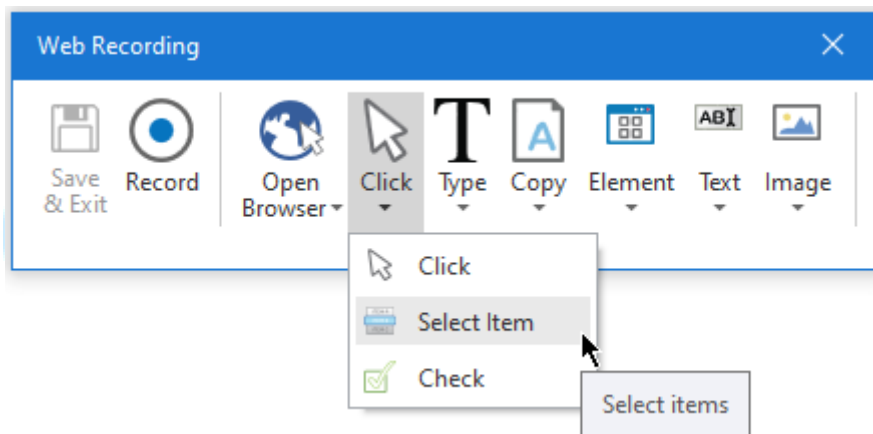
- Similarly, select Click > Click and indicate Close button
- Click **Type** in web recorder



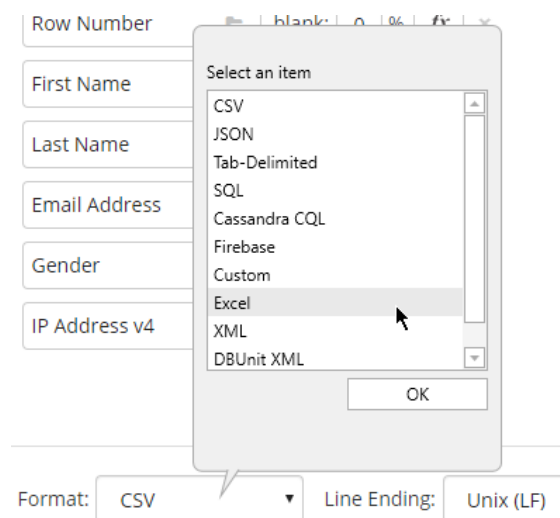
- Type 10 in Rows text box
- Check Empty Field checkbox
- Press Enter



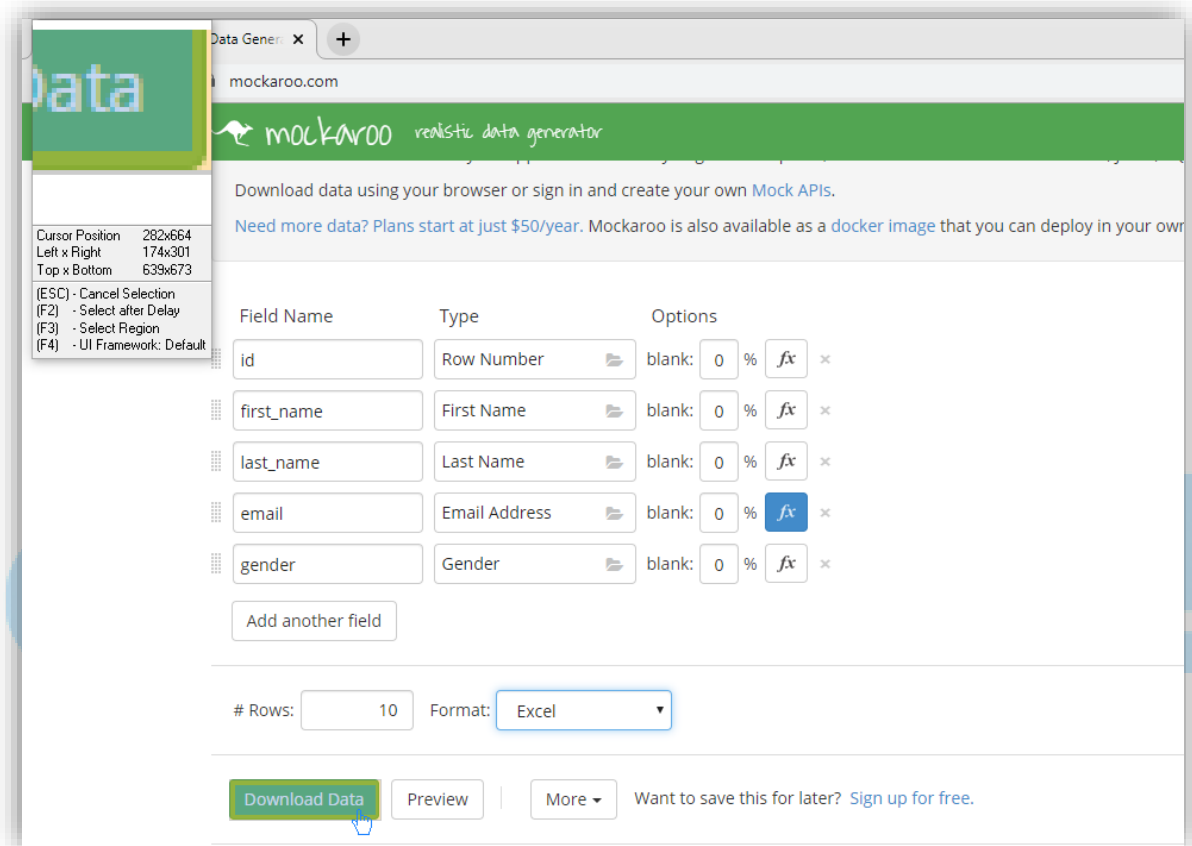
- Choose Click > Select Item in Web Recorder



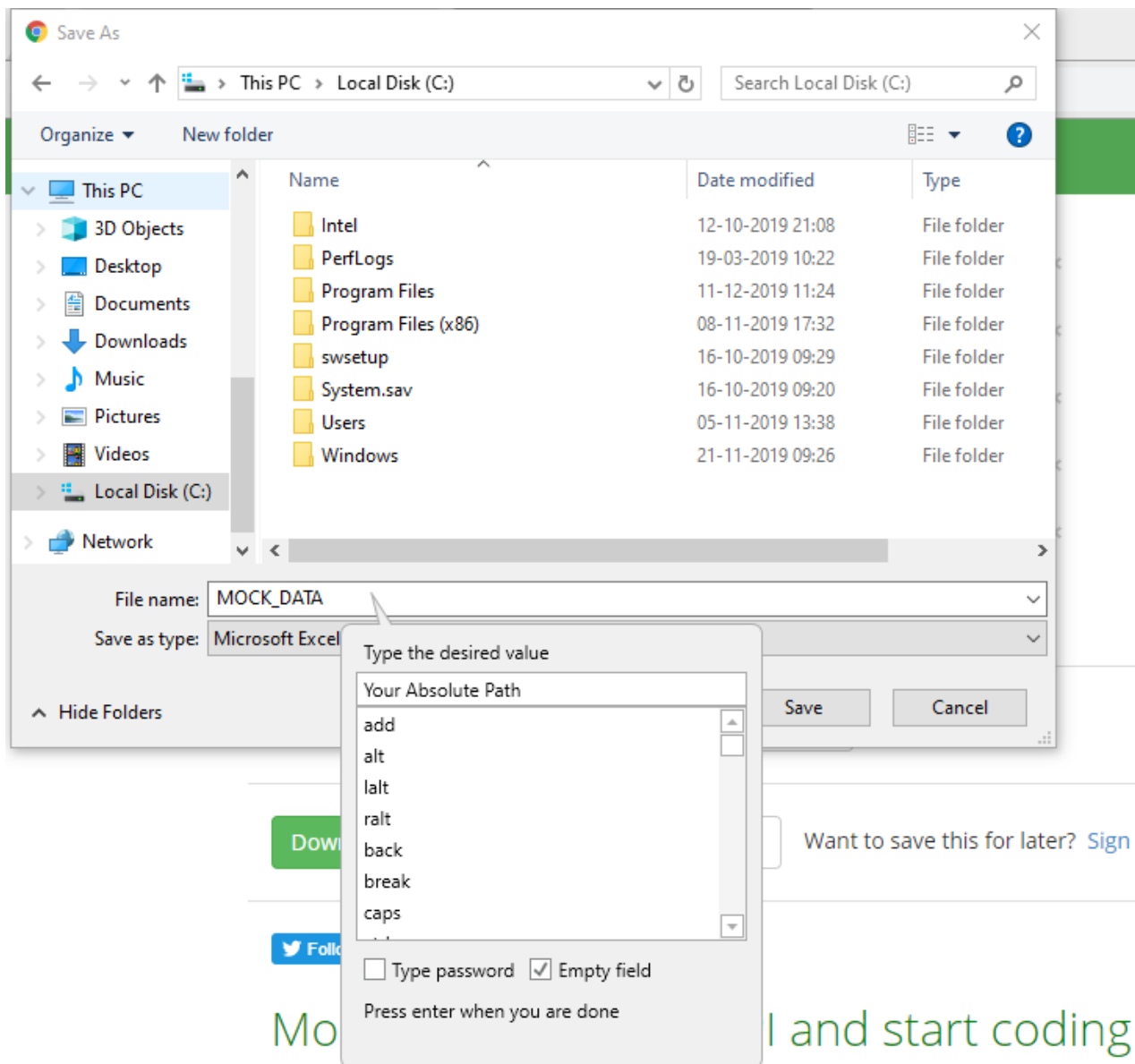
- Select Excel in Drop Down and click OK



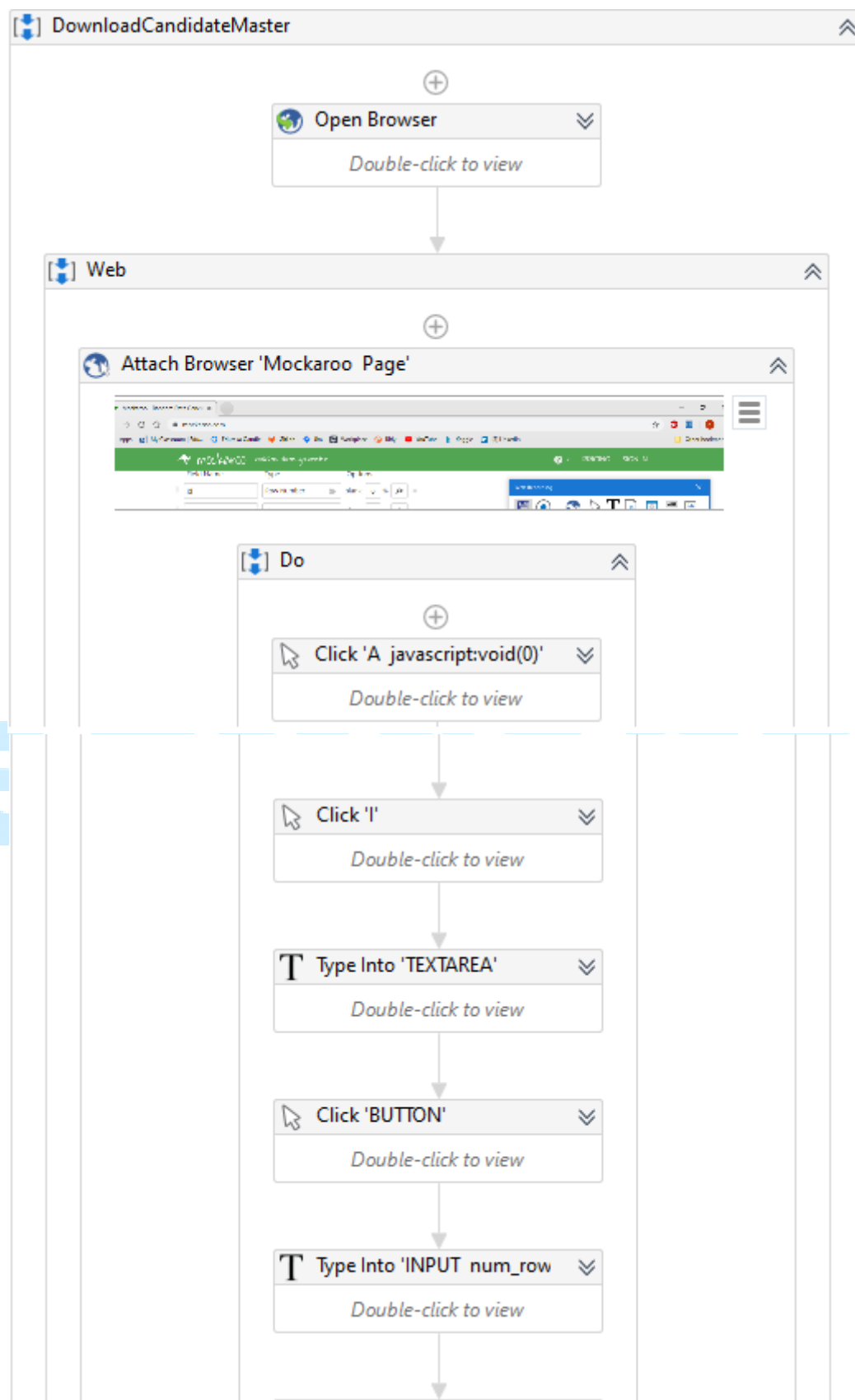
- Scroll down the page a little bit so that the DownloadData button is visible
- Again, select Click > Click in web recorder
- Indicate Download Data button

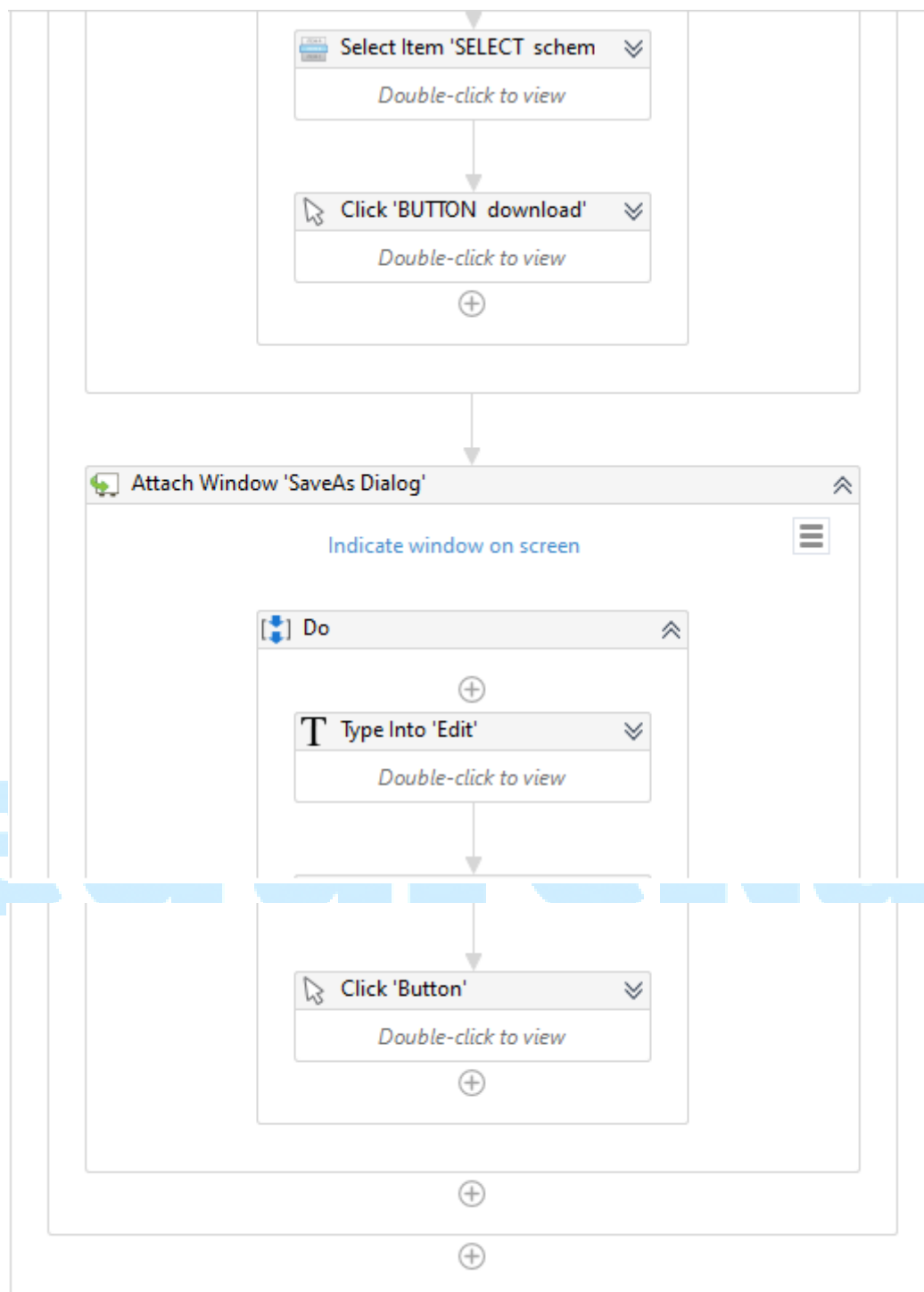


- Select Click > Click in Web Recorder
- Click Type in Web Recorder and indicate File name field
- Enter the file name as the location of your CandidateOnboarding UiPath project followed by **\Data\Input\CandidateMaster.xlsx**
- Example -
 "C:\Users\Documents\UiPath\CandidateOnboarding\Data\Input\Candidate Master.xlsx" (Absolute Path)



- Close the Web Recorder window and select Yes to keep the recorded output or click on Save & Exit
- The final sequence should look like the screenshots shown below

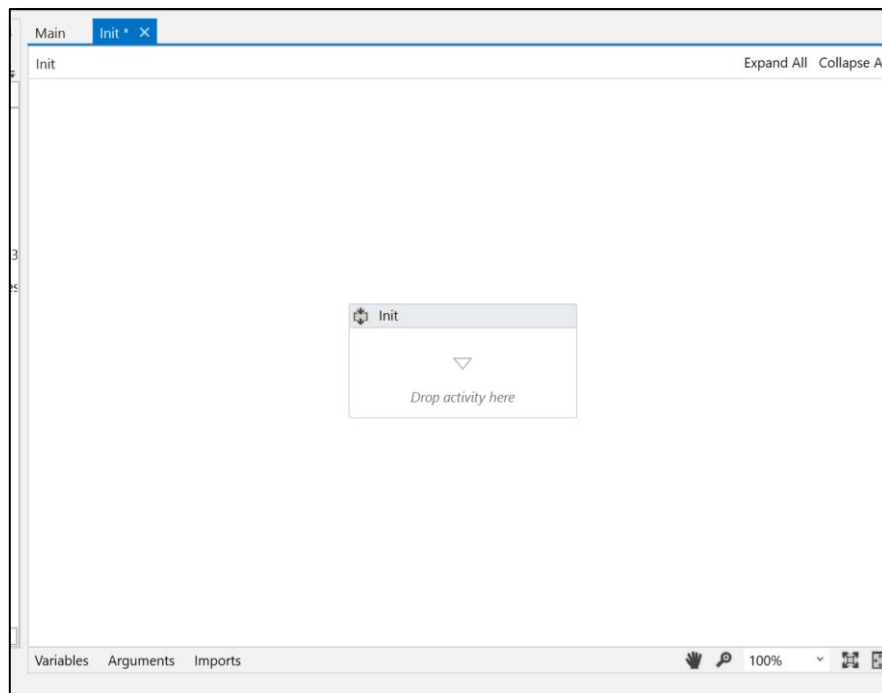




Candidate Onboarding Project Part - 3

We will configure the processes created in this project namely, Init.xaml, Initial_send_mail.xaml KillProcess.xaml. We will mainly use the System activities of UiPath to configure these processes.

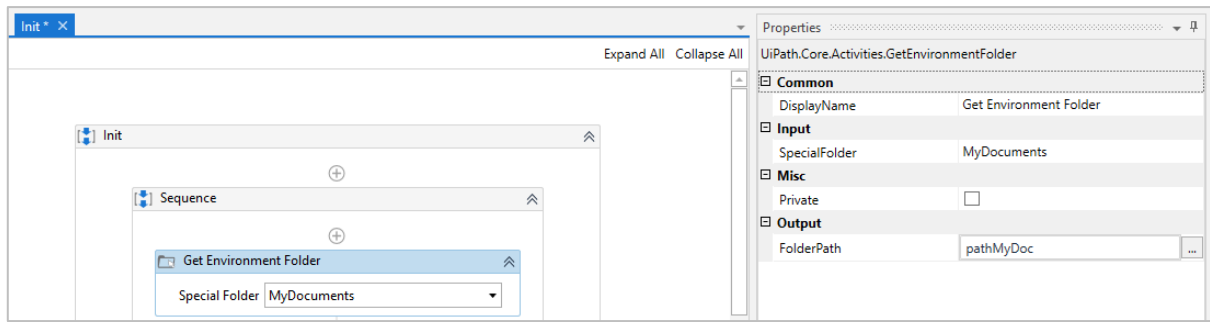
7. Configure Init.xaml file, Open the Init.xaml file



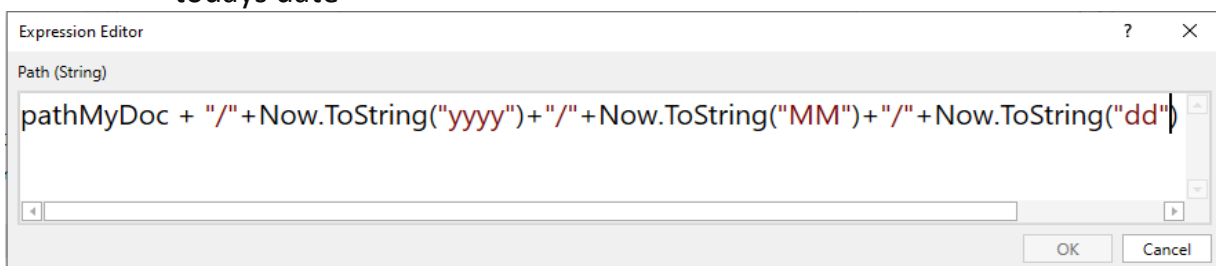
8. Create two variables pathMyDoc and pathCustomLog

Name	Variable type	Scope	Default
pathMyDoc	String	Init	<i>Enter a VB expression</i>
pathCustomLog	String	Init	<i>Enter a VB expression</i>
<i>Create Variable</i>			

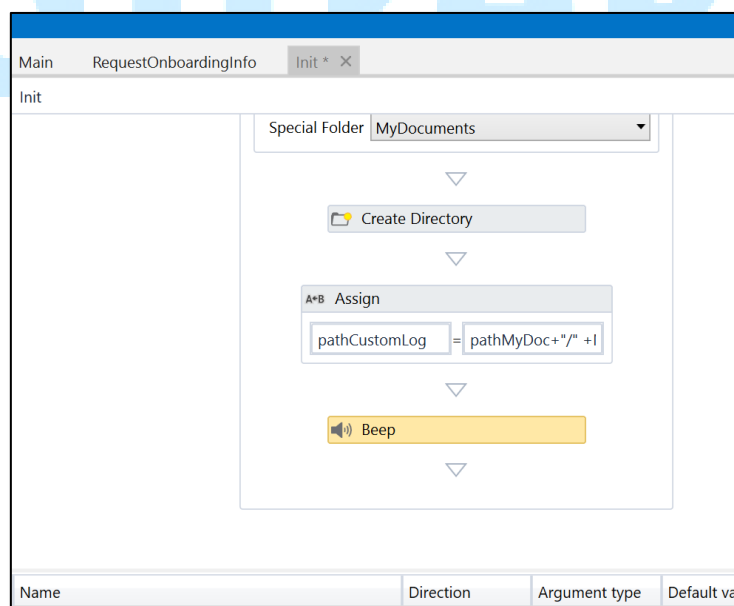
- Go to Activities panel, go to environment and drag Activity Get Environment Folder
- Map variable pathMYDoc to output property



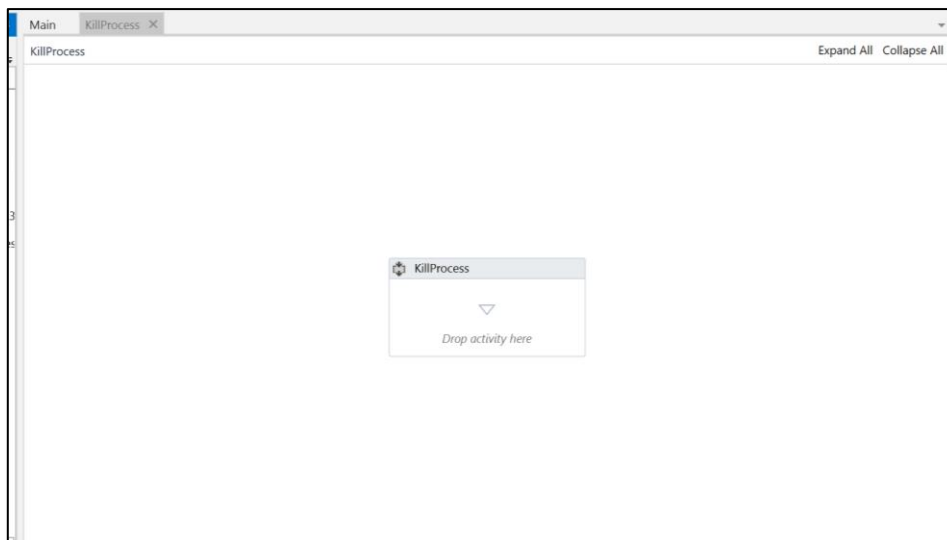
- Go to System > File, drag **Create Directory** activity
- In the Create Directory Activity write the code to create folder structure by todays date



- Drag Beep Activity at the end of the file to play beep to signal process start

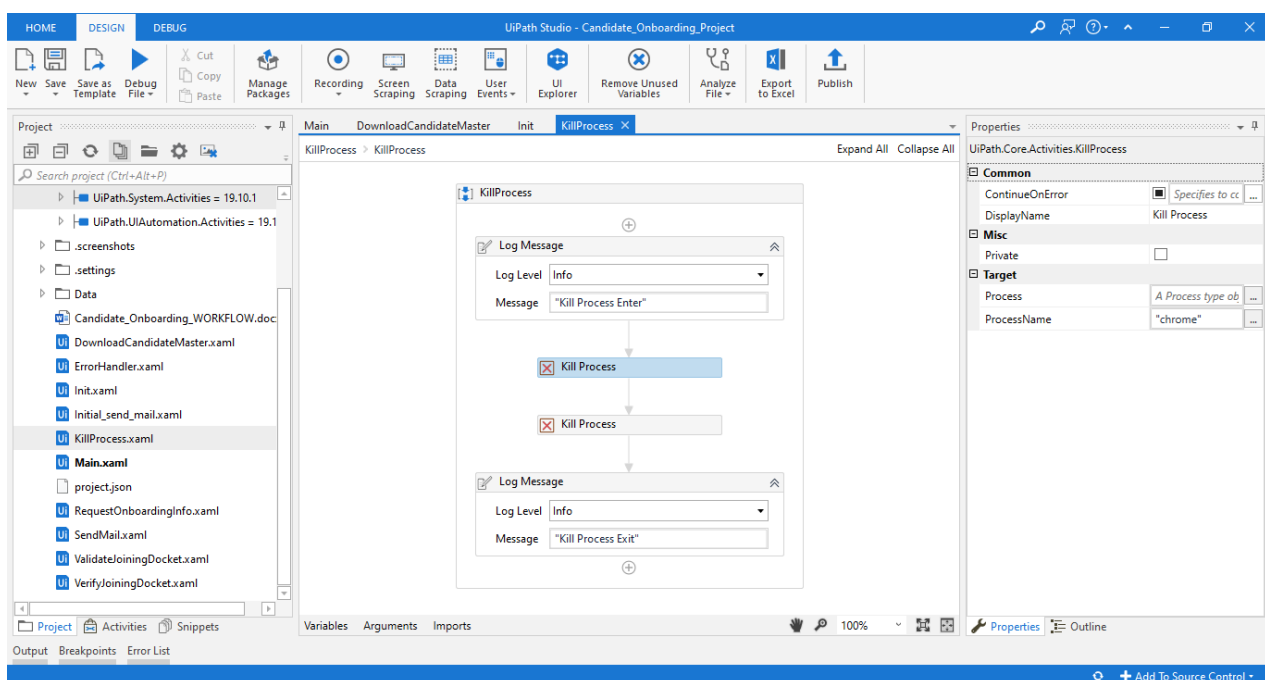


9. Configure KillProcess.xaml, Open the **KillProcess.Xaml** file

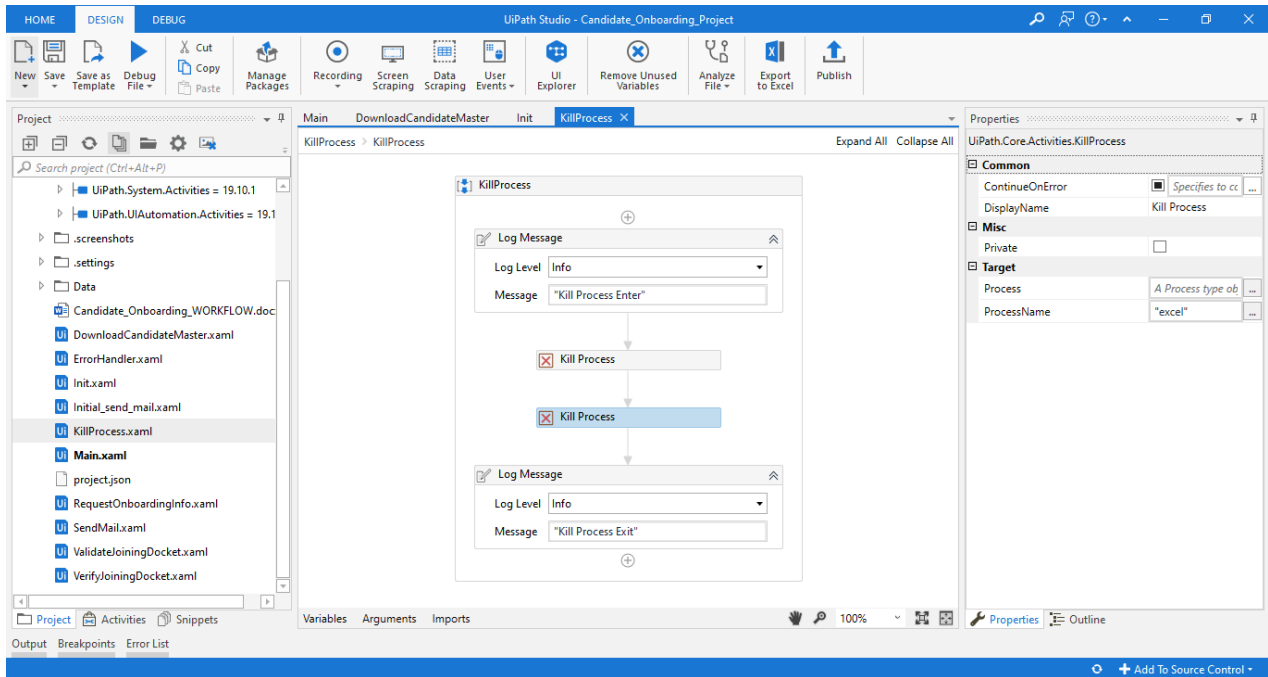


- Drag two Log Message activities in the KillProcess.xaml file
- Type the messages as shown below

- Now in Activities Panel, in the search box, type Kill to get **Kill Process** Activity
- Drag two kill process activities as shown below
- We will kill chrome and excel process to ensure they start as new instances
- Select the first Kill process activity, in the properties panel, in **Process Name**, enter "chrome"

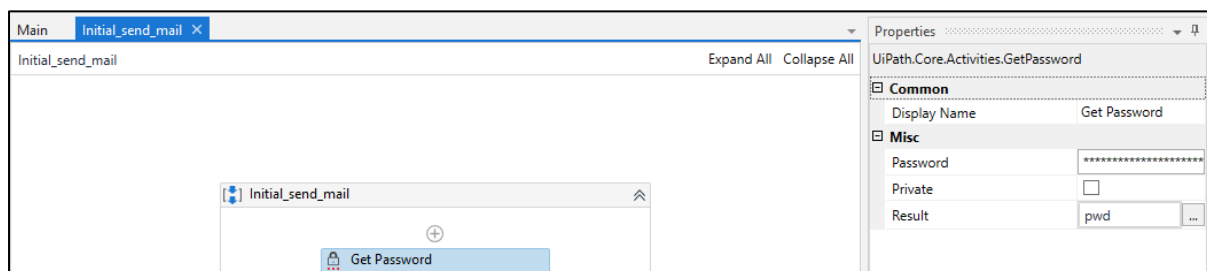


- Select the second Kill process activity, in the properties panel, in process name write “excel”



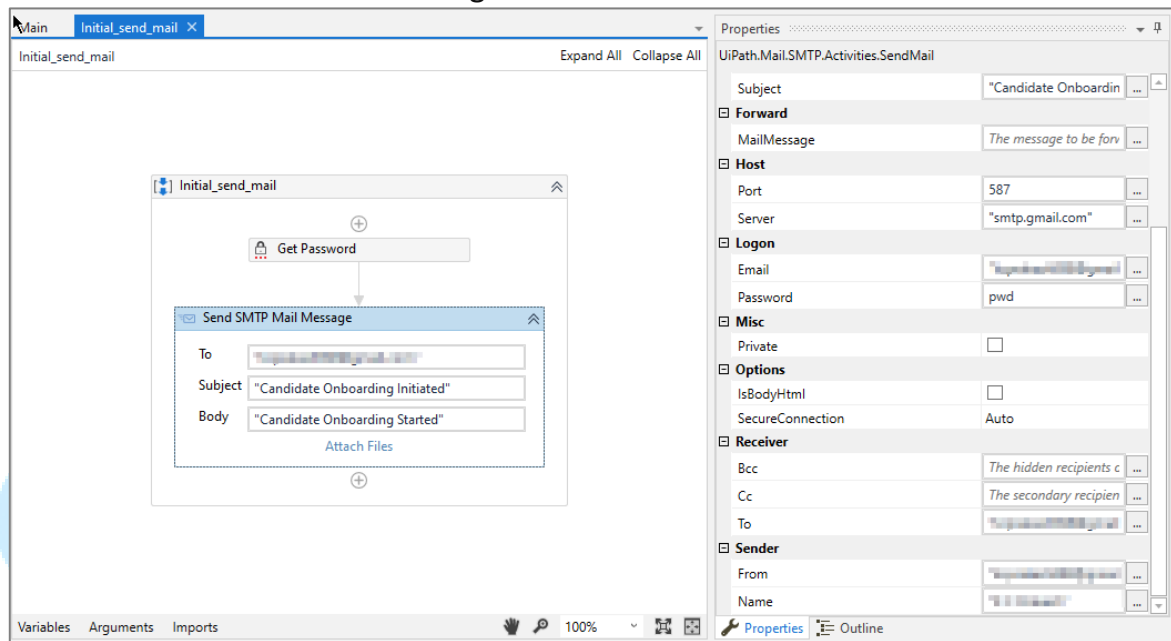
- Save both the processes
- Create a Sequence named Initial_send_mail

10. Drag Get Password Activity, in the properties panel under miscellaneous section type your password and set its output value (ctrl + k) to a variable named **pwd**

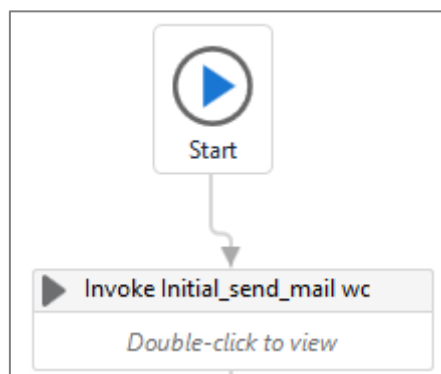


- For Mail Activities, SMTP and IMAP Protocol is recommended
- Drag SMTP Mail Message Activity
- SMTP server URL for gmail is smtp.gmail.com
- SMTP Port Number for gmail is 587

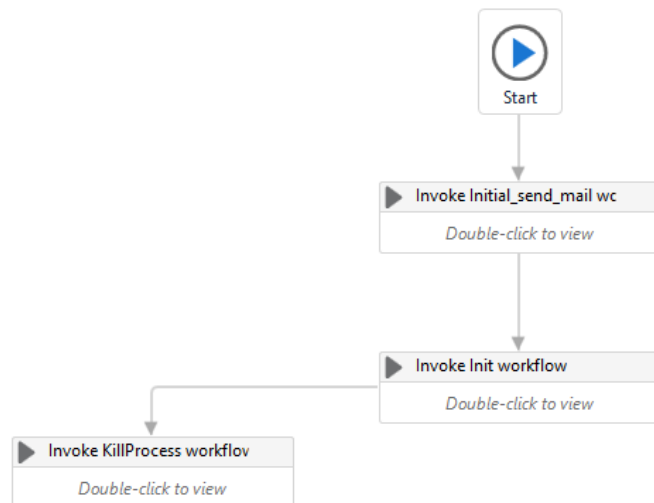
- For any other accounts, you can directly google your SMTP/IMAP credentials and you have to make sure that SMTP/IMAP permissions are granted by your Mail account
- For Gmail users make sure to enable “Access Less Secure Apps” from your account settings and IMAP protocol, so that UiPath can access it.
- Enter your email in the properties panel and in the password pass in the variable pwd
- You can fill the details as per your choice, but in this demo, we have filled the subject – “Candidate Onboarding Initiated” and fill the body field - “Candidate Onboarding Started”



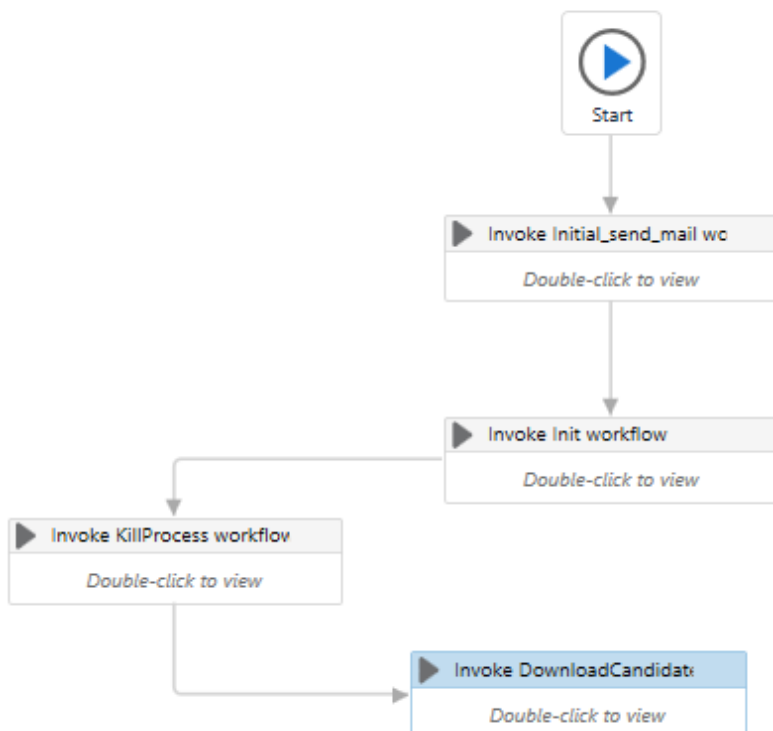
11. Go to Main workflow, Drag Initial_send_mail Sequence and set it as first node



12. Connect Initial_send_mail.xaml to Init.xaml & connect Init.xaml to KillProcess.xaml



13. Connect KillProcess to DownloadCandidateMaster



Candidate Onboarding Project Part - 4

We will configure the processes in the OnBoardingProcessAutomation project by using Excel and Mail Activities

14. For sending mail, we will use “Send SMTP Mail” activity in the SendMail.xaml file

- To use “Send SMTP Mail” activity, you must check these things. They are:
 - ✓ Drag Get Password activity and map your mail’s log in password to a variable (ctrl + k) so it is not visible to anyone
 - ✓ Write your password in password field in misc section in properties panel and map it to a variable named pwd
 - ✓ Go to your email account settings, under security, enable access less secure apps
 - ✓ Google your account’s SMTP port number and server name
 - ✓ For Gmail, the SMPT port number is 587 and SMTP server name is “smtp.gmail.com”

15. Create the following Argument

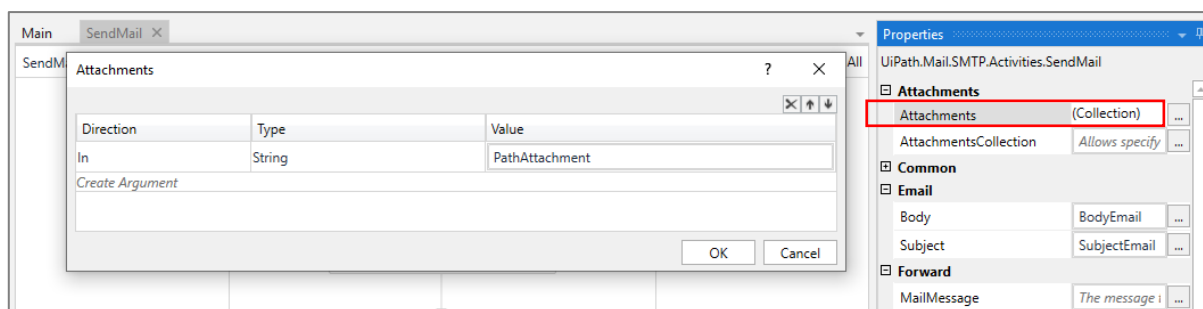
- I. ToEmailAddress
- II. SubjectEmail
- III. BodyEmail
- IV. PathAttachment

16. Pass these arguments into their corresponding fields

Name	Direction	Argument type	Default value
ToEmailAddress	In	String	Enter a VB expression
SubjectEmail	In	String	Enter a VB expression
BodyEmail	In	String	Enter a VB expression
PathAttachment	In	String	Enter a VB expression
Create Argument			

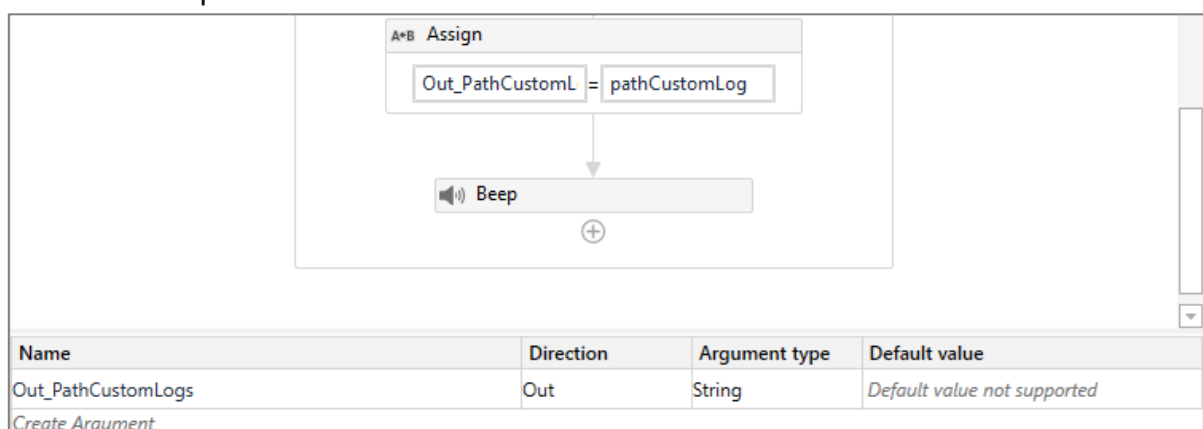
17. Enter your Email id from which you want to send the mail, type it into the email field under log on section in the properties panel

18. In the attachments, pass the PathAttachment argument



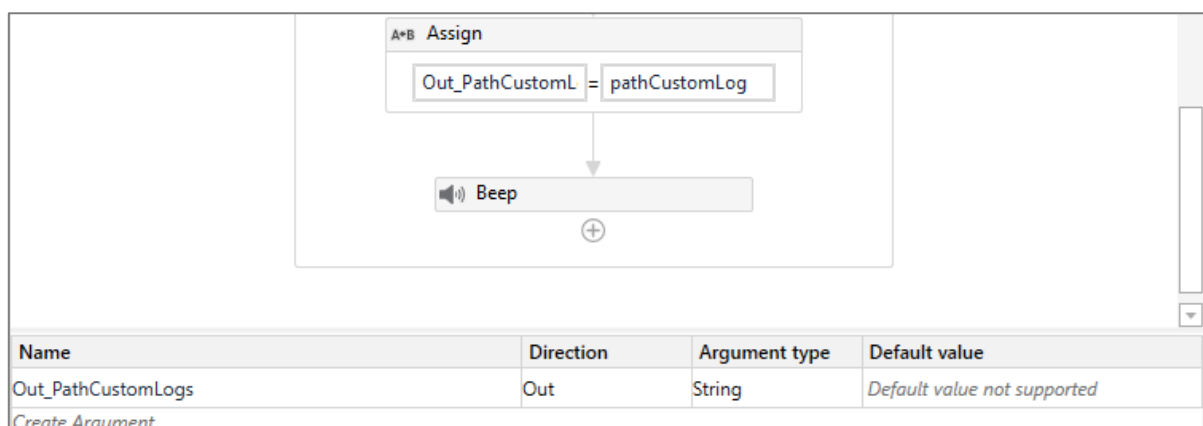
19. Go to Init.xaml sequence create an argument named Out_PathCustomLog and set its direction as OUT

20. Use another Assign Activity and assign Out_PathCustomLog to pathCustomLog in Init.xaml sequence

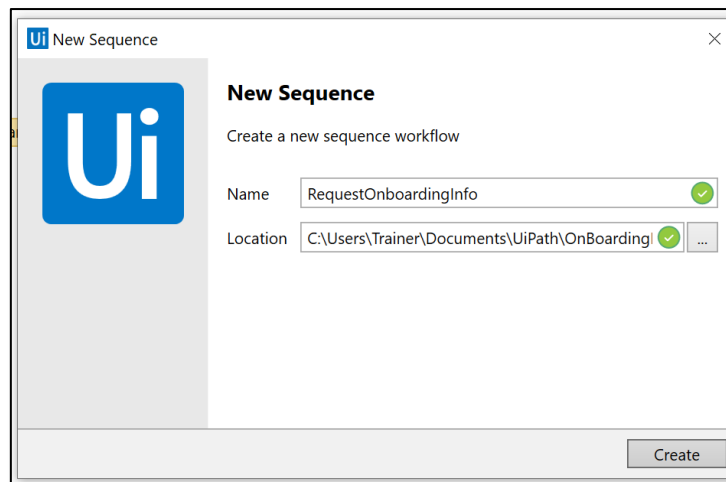


21. Now in the Main workflow create a variable named CustomLogPath

22. Now double click Invoke Init.xaml workflow, click import arguments assign CustomLogPath in the value corresponding to Out_PathCustomLog



23. Create a new Sequence file **RequestOnboardingInfo**



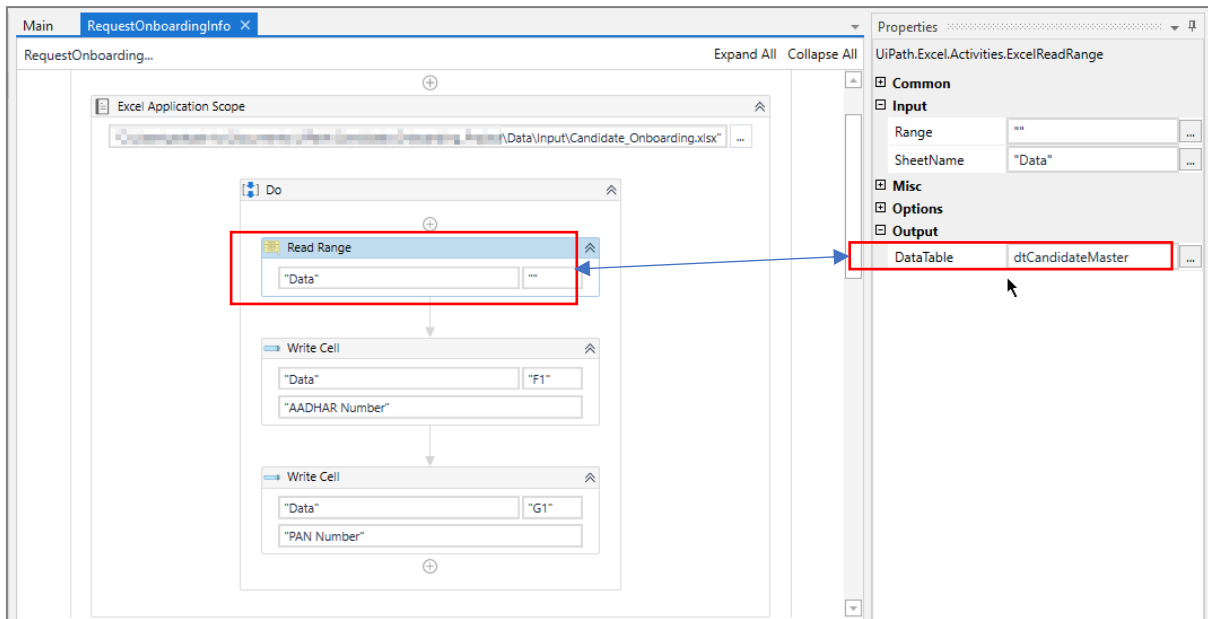
24. Drag Excel Application scope in the sequence and navigate to the candidate onboarding file downloaded from mockaroo and select it

NOTE

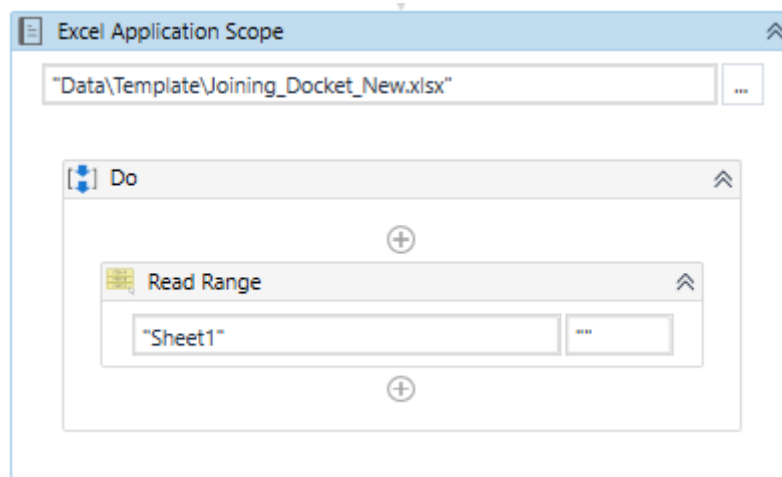
- Make sure you have downloaded Joining Docket Dataset from Module 4 on LMS
- Create a folder in Data folder named Template and, in that folder, save it with the name of **Joining Docket**

25. In do block of Excel Application scope, drag Read Range activity to read the file and change sheet name from "Sheet 1" to "Data". Map Read Range activity's output to a variable (ctrl +k), named **dtCandidateMaster** and its type should be "**DataTable**" and change its scope to **RequestOnboardingInfo**

- Drag Write Cell below or above Read Range and to add column for aadhar number, set the range "F1" and write the text as "AADHAR Number"
- Drag one more write cell activity, change the range to "G1" and set the text as "PAN Number"



- Drag another excel application scope activity and navigate to Joining Docket.xlsx and add read range activity to the do block



26. To send mail to people we need to run a for loop, but before that you need to create the following variables:

Scope of all these variables must be RequestOnboardingInfo

- ***CandidateID***
- ***CandidateEmail***
- ***CandidateName***
- ***CandidateMailSubject***
- ***CandidateMailBody***

RequestOnboardingInfo

Excel Application Scope

"C:\Users\prakash kc\Documents\UiPath\Candidate_Onboarding_Project\Data\Input\Candidate_Onboarding.xlsx"

Do

Read Range

"Data"

Name	Variable type	Scope	Default
dtCandidateMaster	DataTable	RequestOnboarding	Enter a VB expression
CandidateID	String	RequestOnboarding	Enter a VB expression
CandidateEmail	String	RequestOnboarding	Enter a VB expression
CandidateName	String	RequestOnboarding	Enter a VB expression
CandidateMailSubject	String	RequestOnboarding	Enter a VB expression
CandidateMailbody	String	RequestOnboarding	Enter a VB expression
PathJoiningDocket	String	RequestOnboarding	Enter a VB expression

Create Variable

27. Drag For Each Row Activity set variable **dtCandidateMaster** to iterate in the file

RequestOnboardingInfo

Excel Application Scope

Double-click to view

Excel Application Scope

Double-click to view

For Each Row

ForEach row in dtCandidateMaster

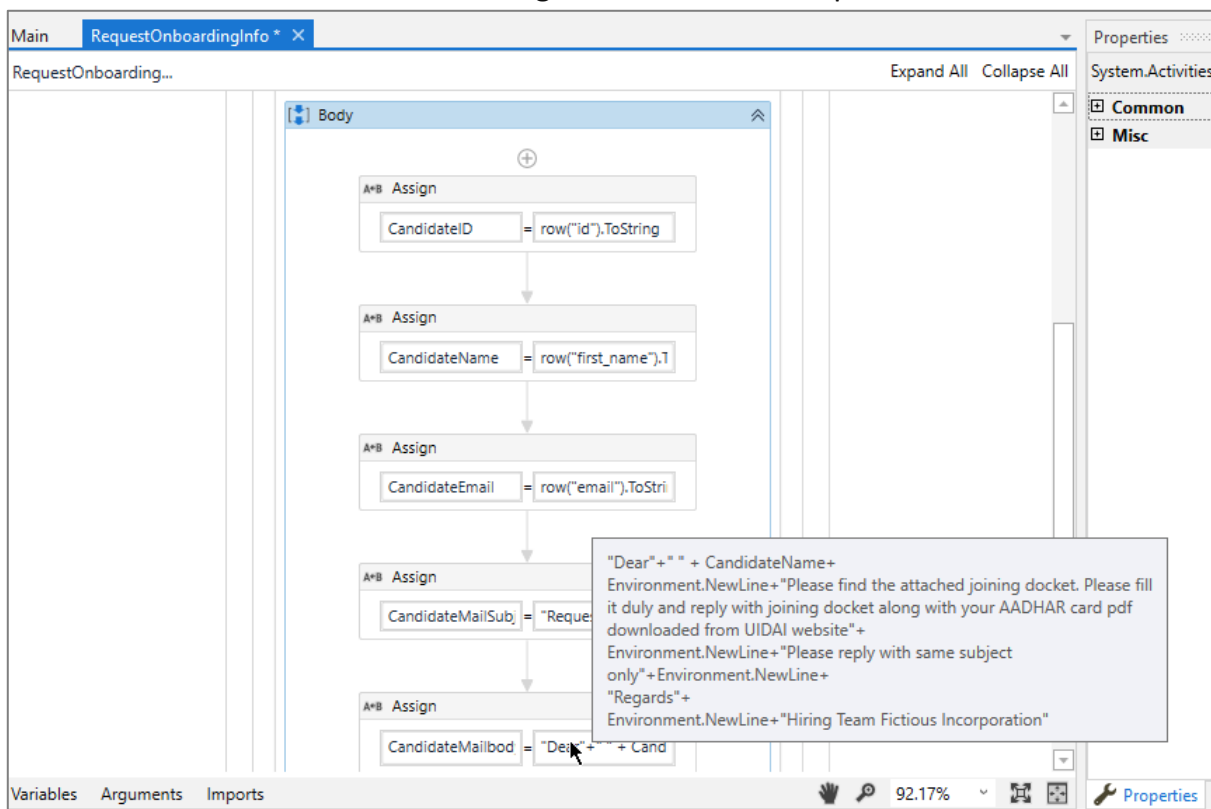
Body

Body

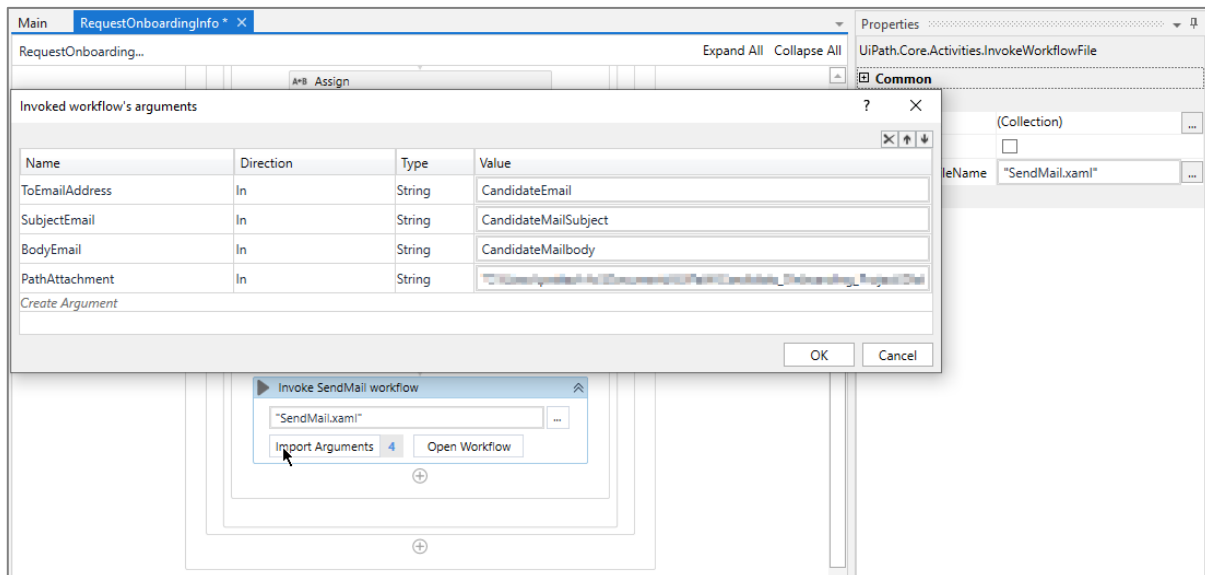
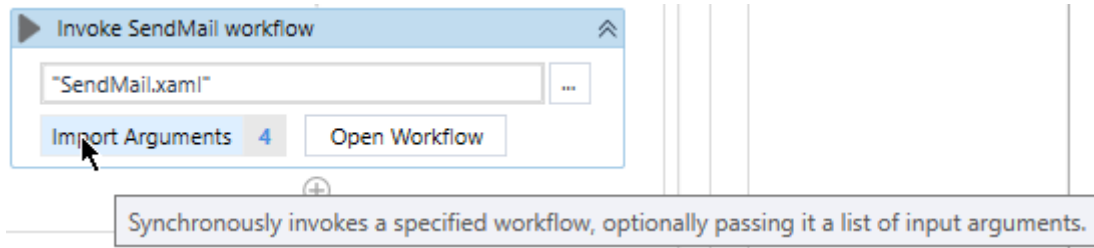
6 action(s)

Double click to View

- In the body of For Each Row Activity drag 4 Assign Activities or drag one and copypaste it 4 times and perform the following:
- CandidateID = row("id").ToString
- CandidateEmail = row("email").ToString
- CandidateName = row("first_name").ToString+" "+row("last_name").ToString
- CandidateMailSubject = "Request to Fill the joining docket for candidate Id:"
- CandidateMailBody = "Dear"+" " + CandidateName+
Environment.NewLine+"Please find the attached joining docket. Please fill it
duly and reply with joining docket along with your AADHAR card pdf
downloaded from UIDAI website"+
Environment.NewLine+"Please reply with same subject
only"+Environment.NewLine+
"Regards"+
Environment.NewLine+"Hiring Team Fictious Incorporation"

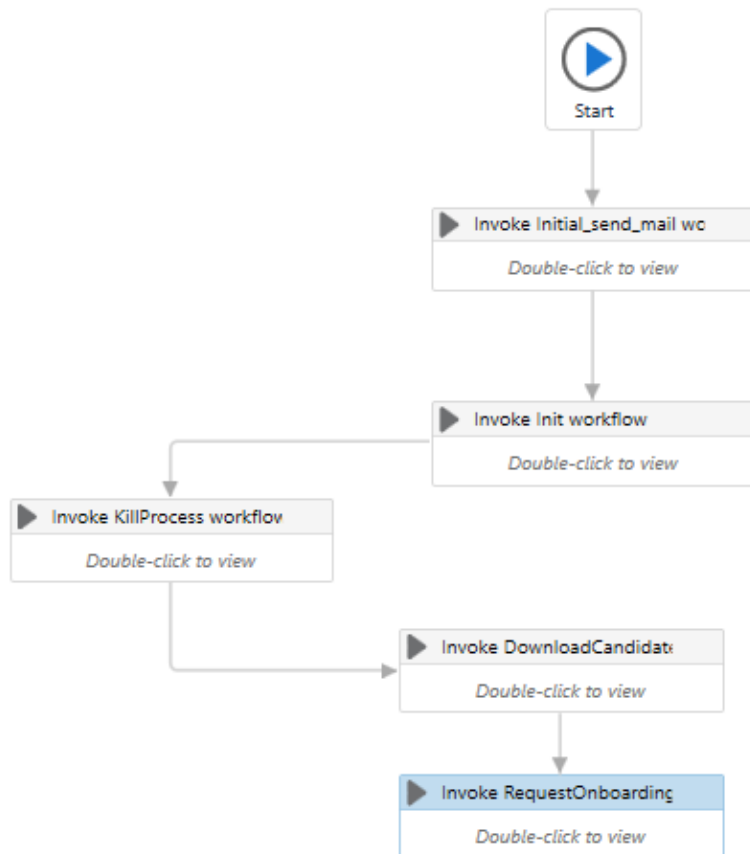


28. Drag **SendMail.xaml** sequence at the end of these assign activities, click on import Arguments and map the arguments to the corresponding variables



29. In **PathAttachment** put the path of Joining Docket.xlsx

30. Connect DownloadCandidateMaster.xaml to RequestOnboardingInfo.xaml

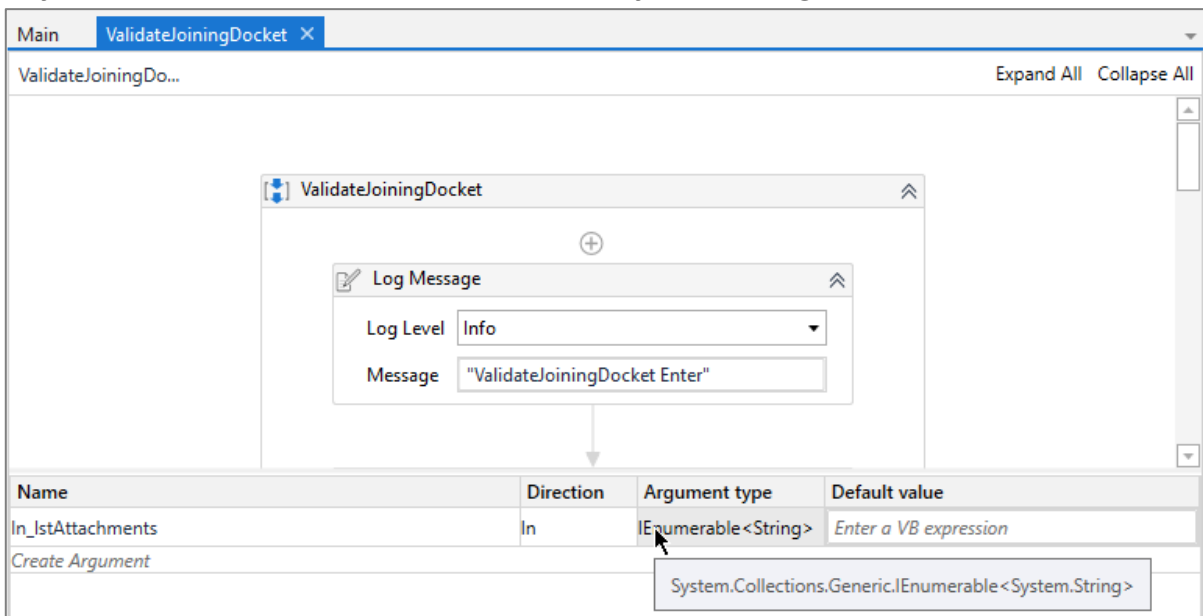


*Try running the project, it's always a good practice to run the project in **Debug Mode**, and if you get stuck when reading the excel file, try enable editing of excel file and then continue running the project

Candidate Onboarding Project Part – 5

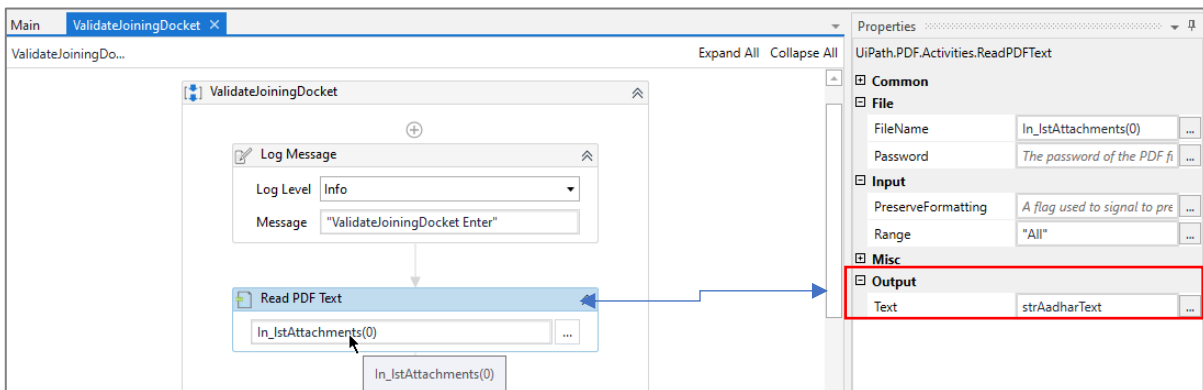
We will create processes to verify the Aadhar details from the Aadhar card PDF received by Mail

31. In the data folder create 2 folders – **Working & Processed**
32. Create a sequence named **ValidateJoiningDocket**
33. Create 2 variables – strAadharText & strVIDDocket
34. Create an argument **In_IstAttachments** with IN direction & it's type should be **System.Collections.Generic.IEnumerable<System.String>**

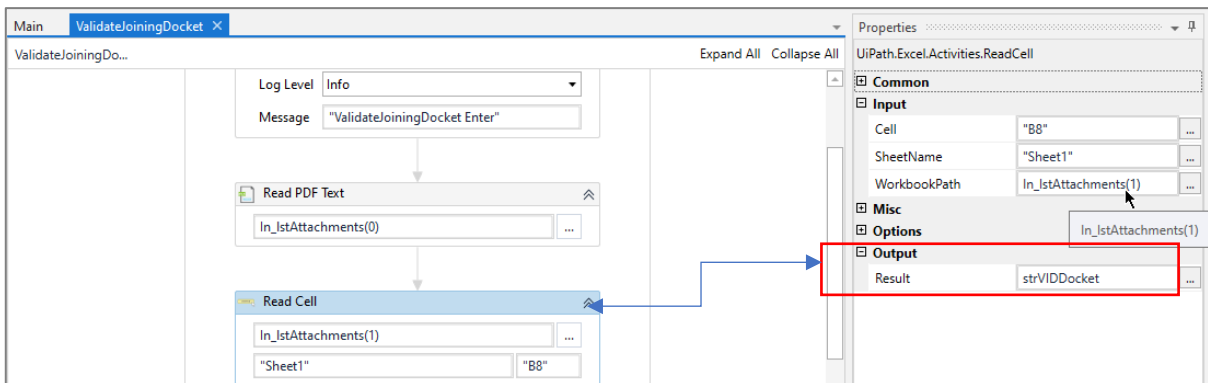


35. Drag Read PDF text activity, in file name input **In_IstAttachments(0)** and set the output variable to **strAadharText**

Note: If the PDF activities don't appear in the Activities panel, then install **UiPath.PDF.Activities** package using Manage Packages option

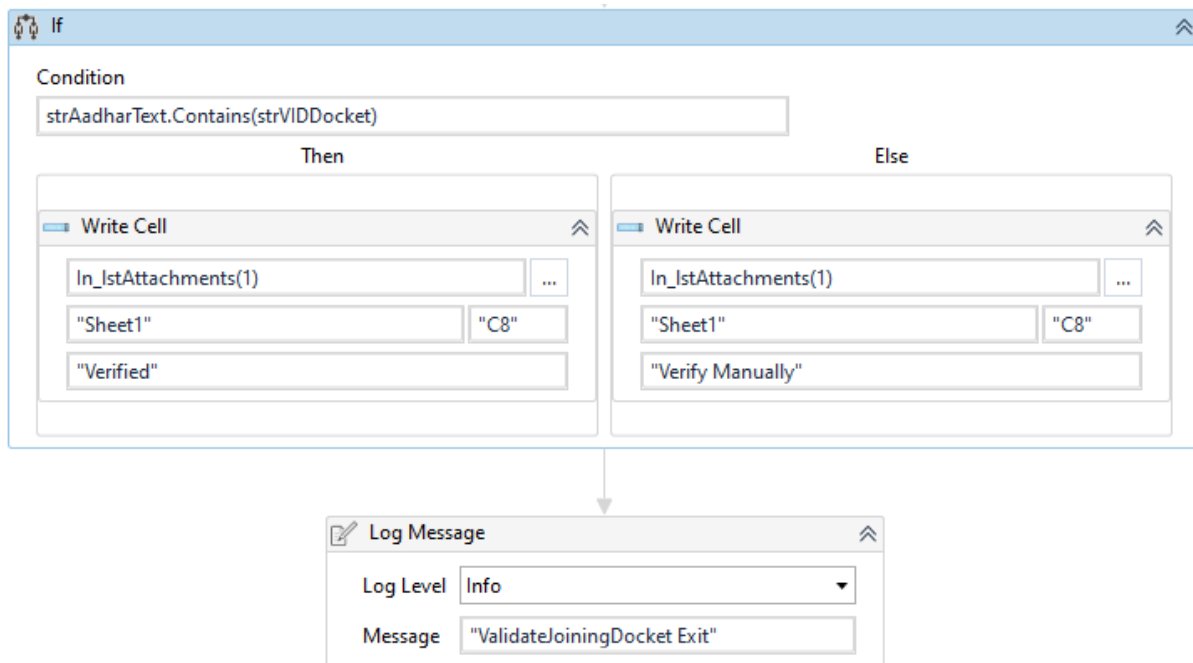


36. Drag Read Cell Activity, below Read PDF text Activity. Input ***In_IstAttachments(1)*** in the filename field and set the output variable to ***strVidDocket***



37. Drag If activity, set the condition ***strAadharText.Contains(strVIDDocket)***

- In THEN block add write cell activity and input the following:
 Cell = "C8"
 Sheet Name = "Sheet 1"
 Text = "Verified"
 Workbook Path = In_IstAttachments(1)
- In ELSE block add write cell activity and input the following:
 Cell = "C8"
 Sheet Name = "Sheet 1"
 Text = "Verified"
 Workbook Path = In_IstAttachments(1)



38. Now we have to move the file from Working folder to Processed folder. Drag Move File Activity

From - `In_IstAttachments(0)`

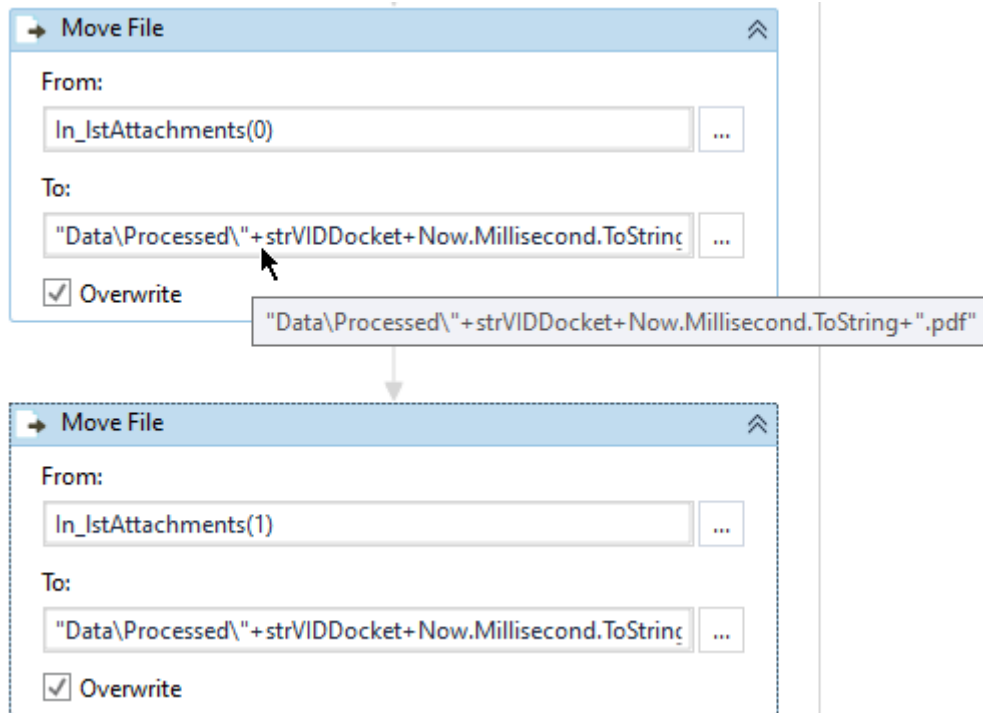
To- `"Data\Processed\"+strVIDDocket+Now.Millisecond.ToString+".pdf"`

- Check Overwrite button
- Drag another Move File Activity

From - `In_IstAttachments(1)`

To- `"Data\Processed\"+strVIDDocket+Now.Millisecond.ToString+".xlsx"`

- Check Overwrite button



39. Create new Sequence, **VerifyJoiningDocket** and the following variables:

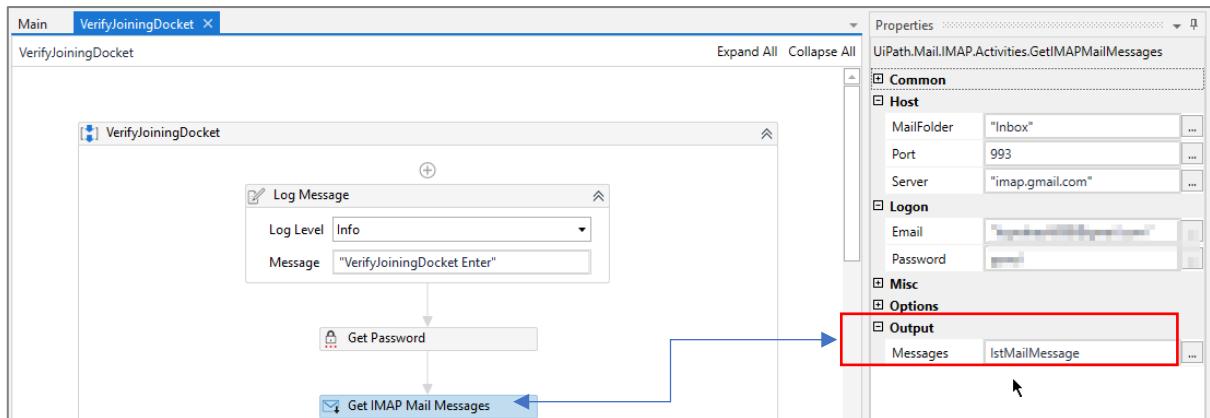
- i. **strEmailSubject**
- ii. **IstMailMessage of type `System.Collections.Generic.List<System.Net.Mail.MailMessage>`**

40. Use Get Password to hide your password or you can copy paste it from SendMail.xaml sequence

41. Send an email filling aadhar details in the joining docket.xlsx and aadhar card.pdf, so IMAP can retrieve the latest mails

42. Drag Get IMAP Mail Activity under Get Password and set the following Values:

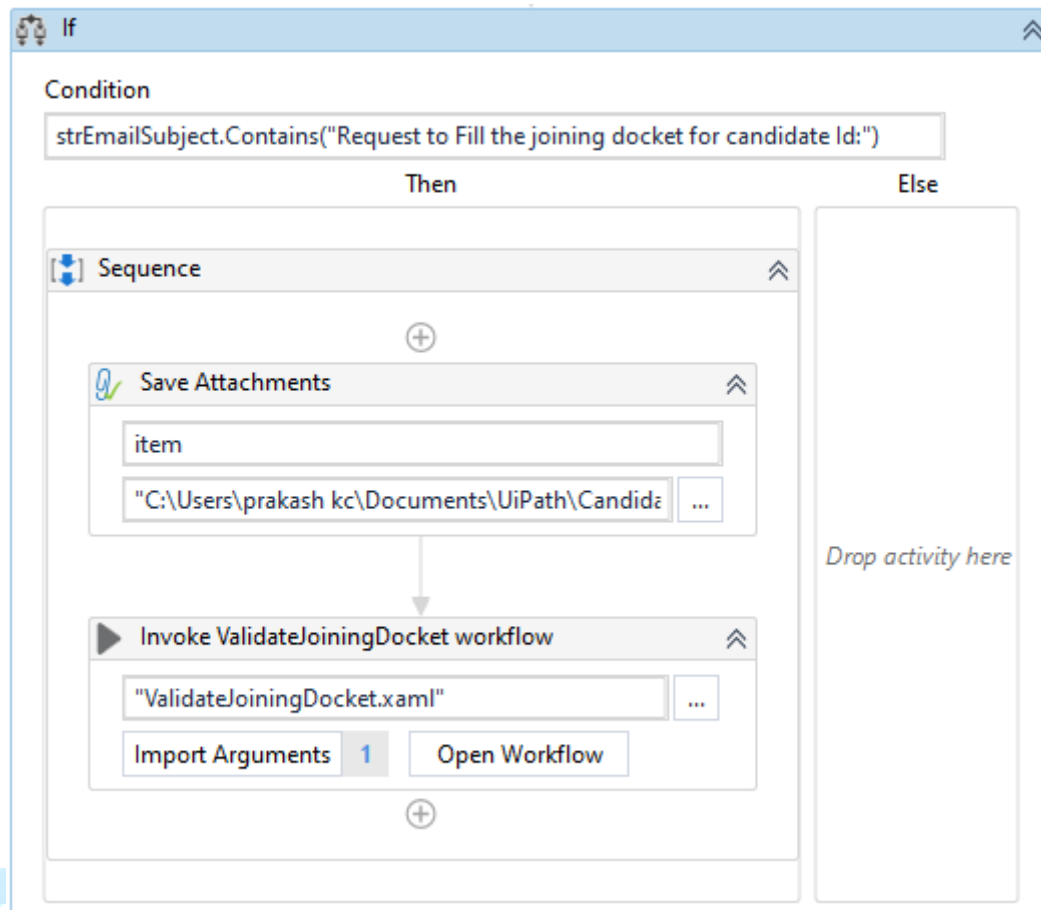
- Enable IMAP from your email account settings
- Port number for IMAP activity is 993
- Server URL is "imap.gmail.com"
- Enter your mail id from which you want to retrieve your mails
- Under Options in the Properties Panel, in the top field set the value of top number of emails you want to retrieve. For e.g. if you want to retrieve top 5 emails then input 5 in the top field
- In the output field set the variable to **IstMessage** and change its type to



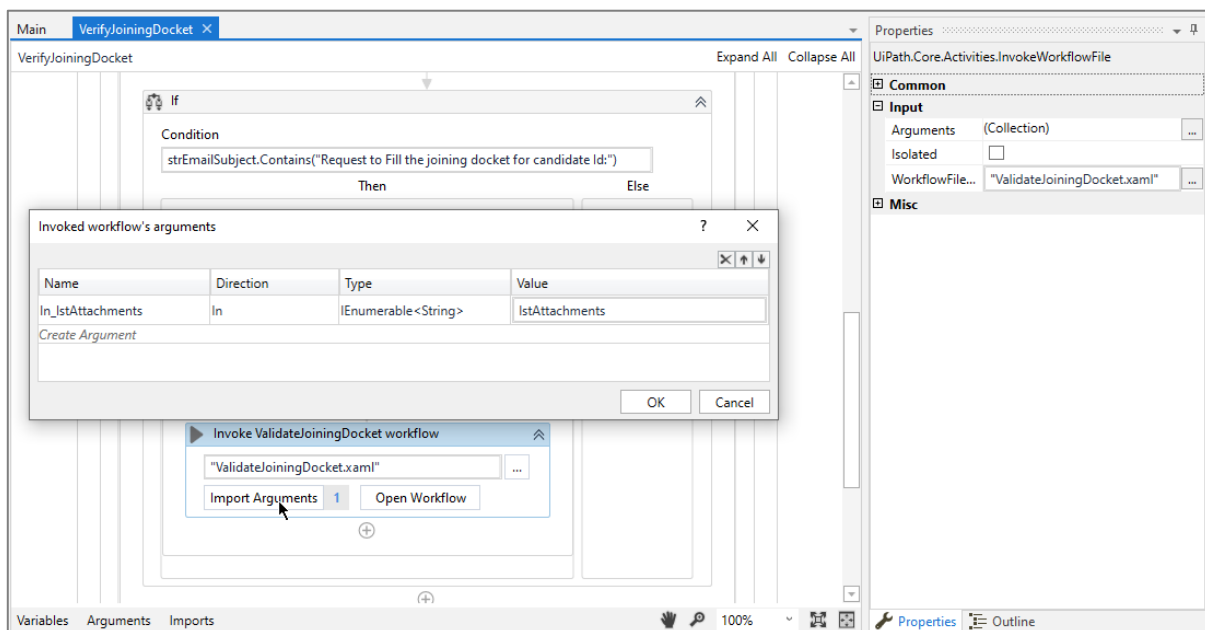
43. Under Get IMAP Activity, Drag For Each Activity. Run the loop in **IstMessage**

44. In body of for each activity, we will perform the following activities:

- Drag assign activity, and assign **strEmailSubject** to item.subject
- Add if condition below assign activity and set the condition-
strEmailSubject.Contains("Request to Fill the joining docket for candidate Id:")
- In THEN block of if activity drag save attachments, set the folder path to Working Folder and in the message field input item and set its output to a variable named **IstAttachments**

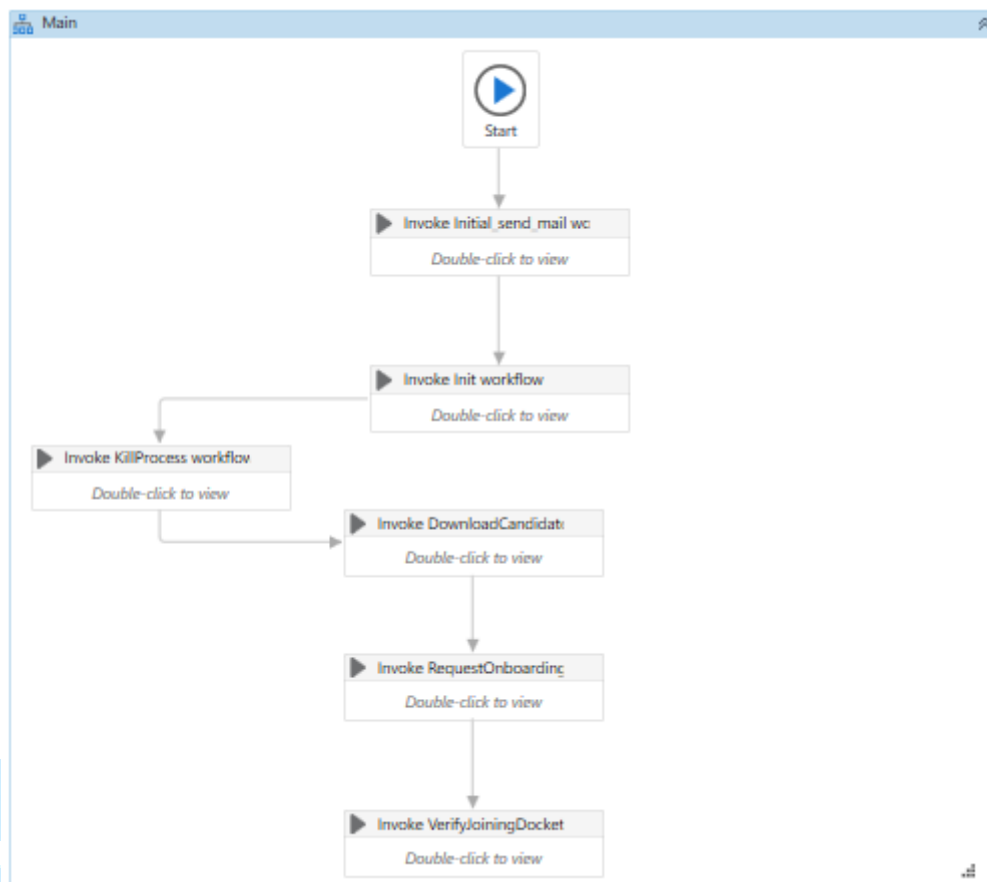


- After saving the attachment drag **ValidateJoiningDocket.xml** and click import arguments and map **In_IstAttachments** to **IstAttachments**



45. Connect RequestOnboardingInfo to VerifyJoiningDocket

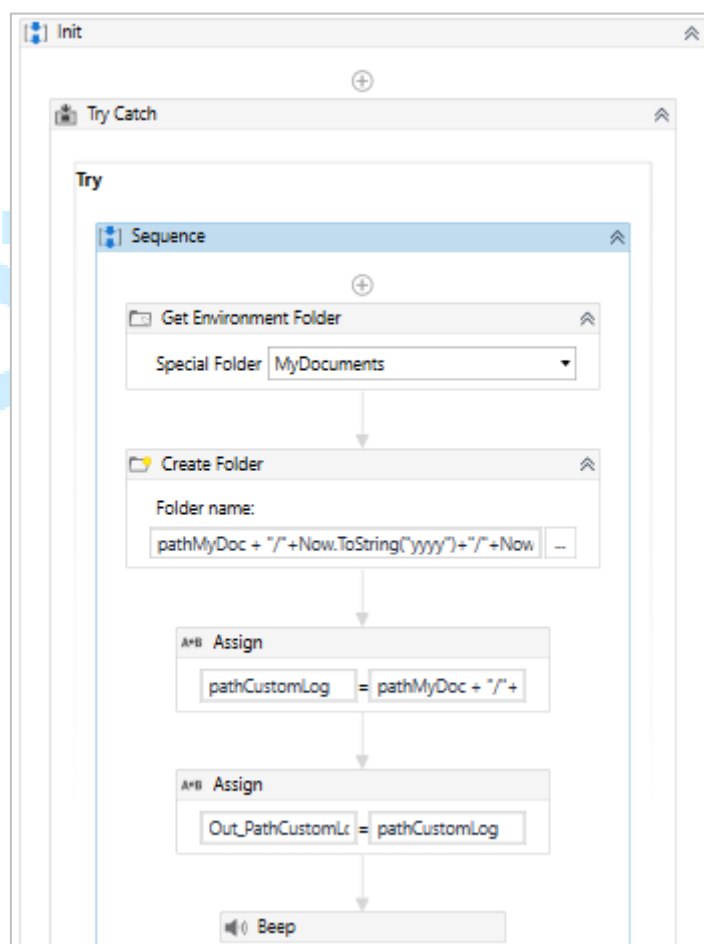
The final workflow should look like this



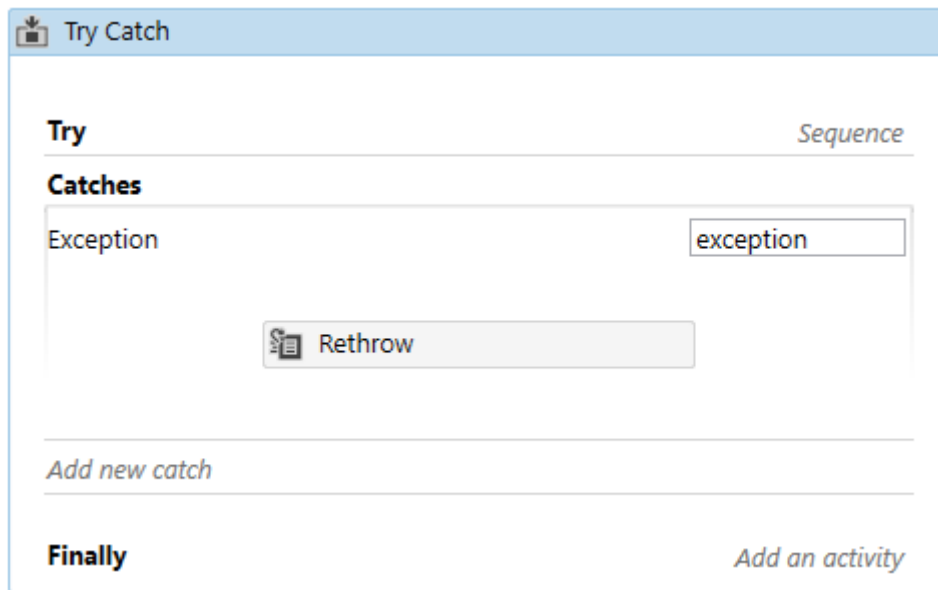
Candidate Onboarding Project Part – 6

We will include Try Catch block to handle exceptions in each of the workflows. We will also Annotate our code.

46. Open Init.xaml file, then in Activities Panel, in search field, type try to get Try Catch Activity
47. Drag Try Catch Activity into the workflow, select all the other activities together and drag them inside the try block

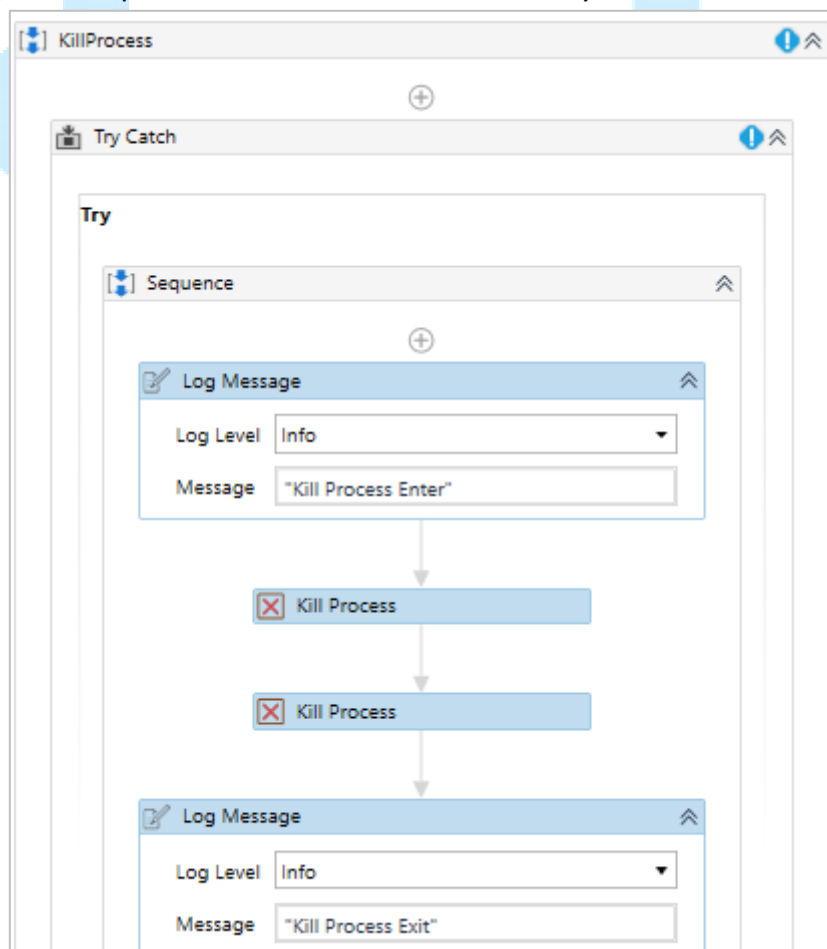


48. Under Catches, click Add New Catch to show exception
 - Select System.Exception from the drop down and click besides the dropdown to show up Catch block
 - Add Rethrow Activity in Catches Block

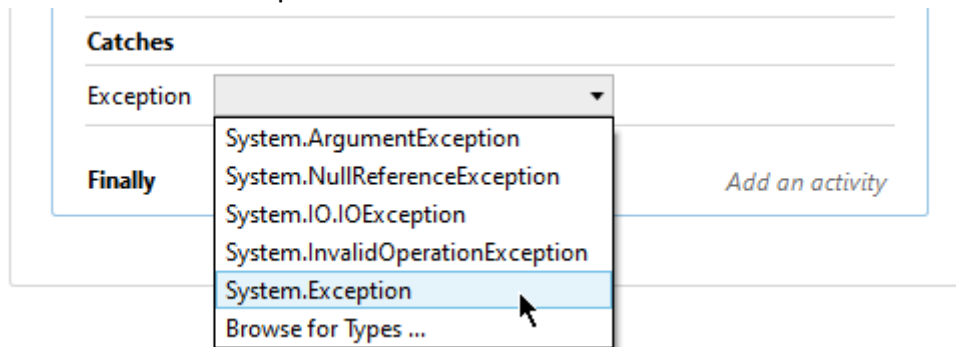


49. Open KillProcess.xaml file

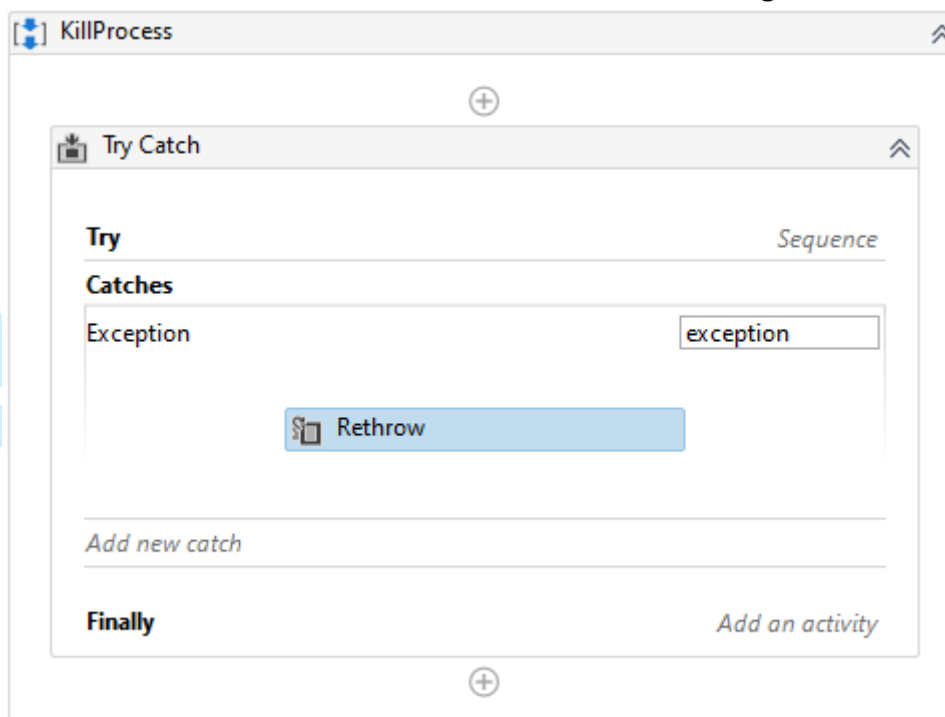
- Drag Try Catch block into the workflow
- Drag and drop all other activities inside the Try block as shown below



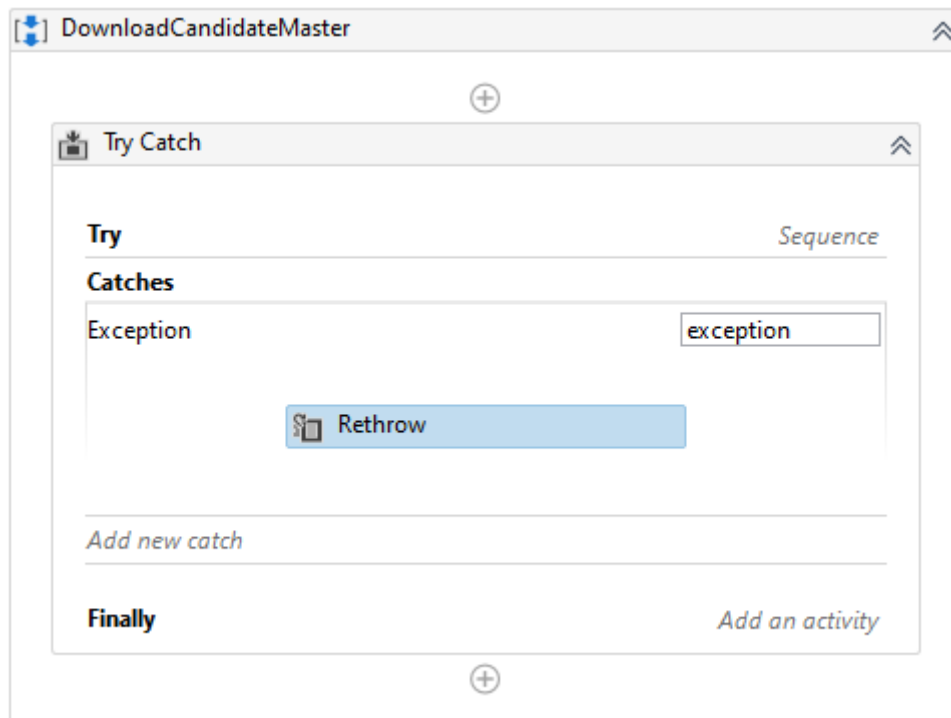
- Click Add New Catch under Catches, select System.Exception and then click beside the DropDown



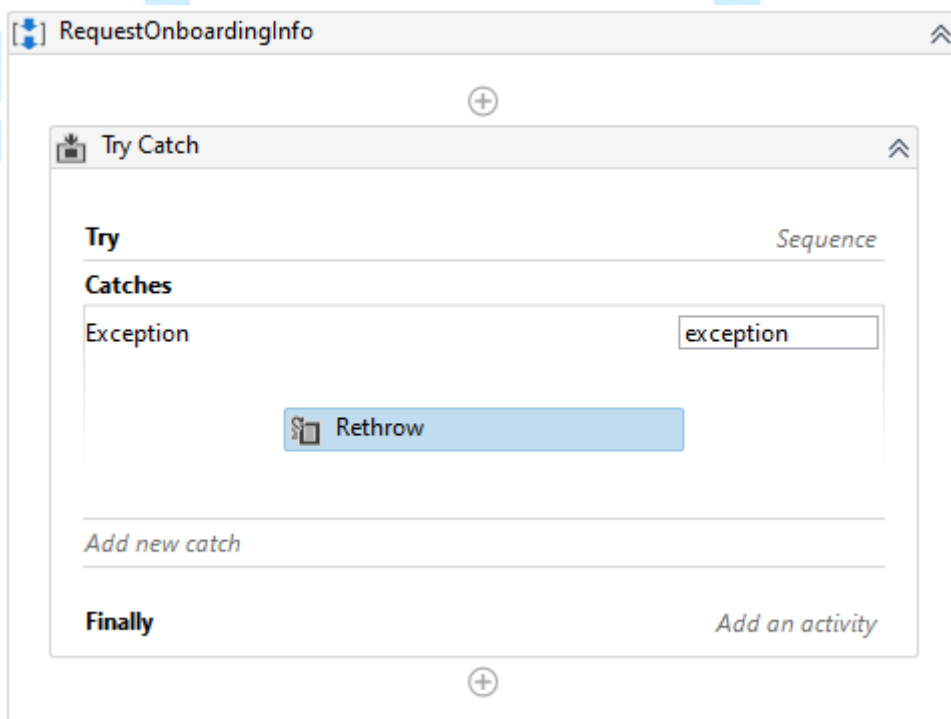
- From the Activities panel, drag Rethrow activity
- The workflow should resemble with the below image-



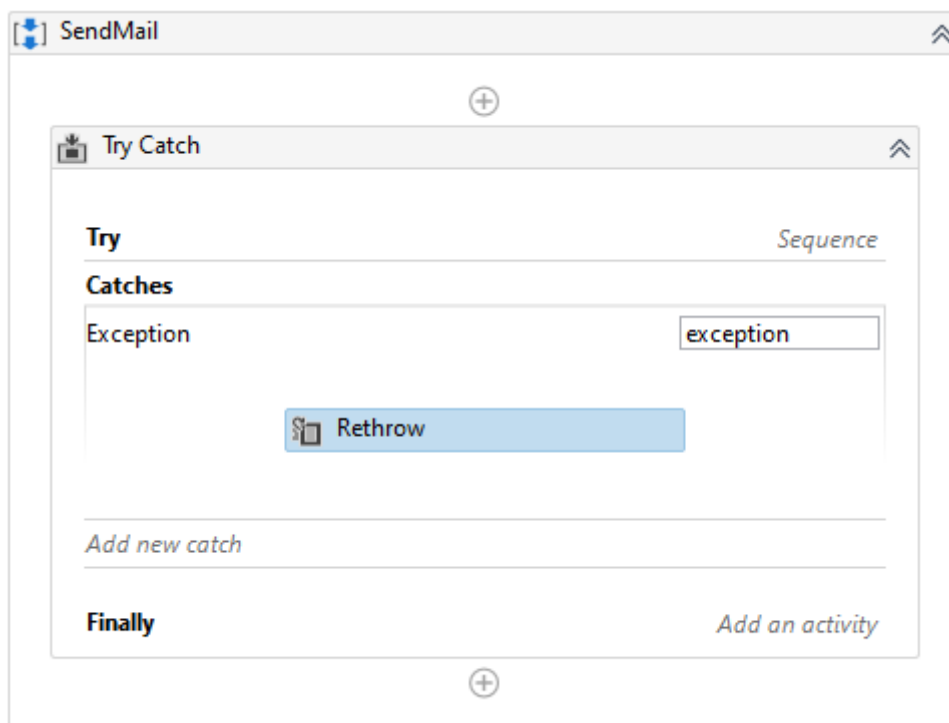
50. Follow the same Process for DownloadCandidateMaster.xml . The file would look like below-



51. Follow the same Process for RequestOnBoardingInfo.xaml. The file would look like below-

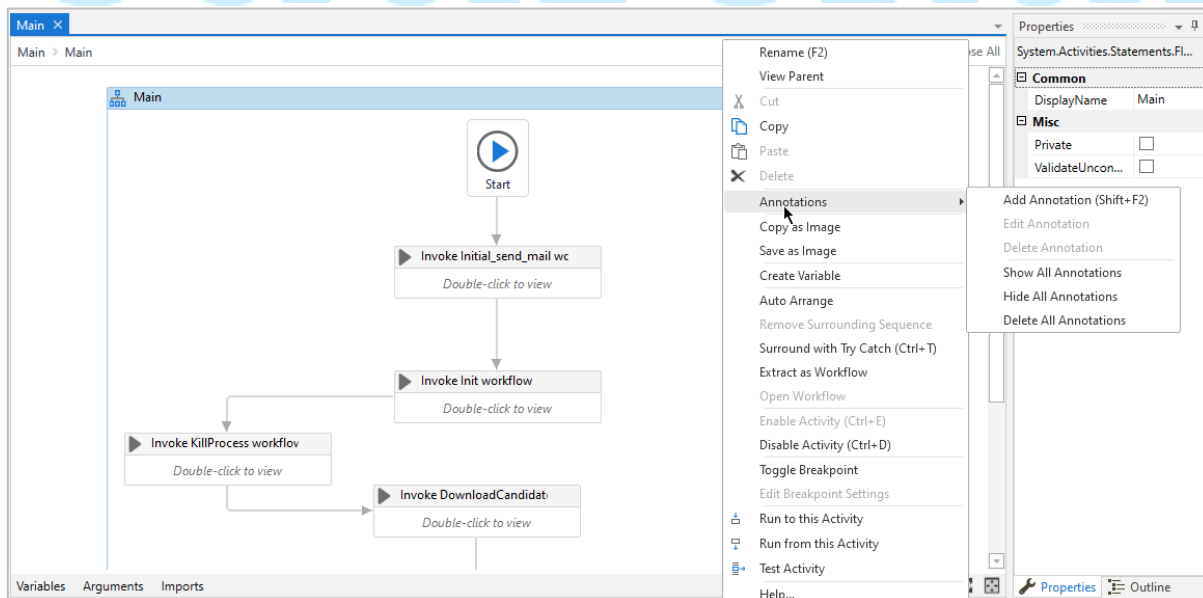


52. Follow the same Process for SendMail.xaml. The file would look like below-




53. Add Annotation

- Open the Main.xaml file
- Right click the flowchart and select Annotations > Add Annotation



- The code after adding annotations will look like below-

 Main

Process Name : Candidate Onboarding

Process Description :

Created By :

Created Data :

Modified By :

Modified Date :

I

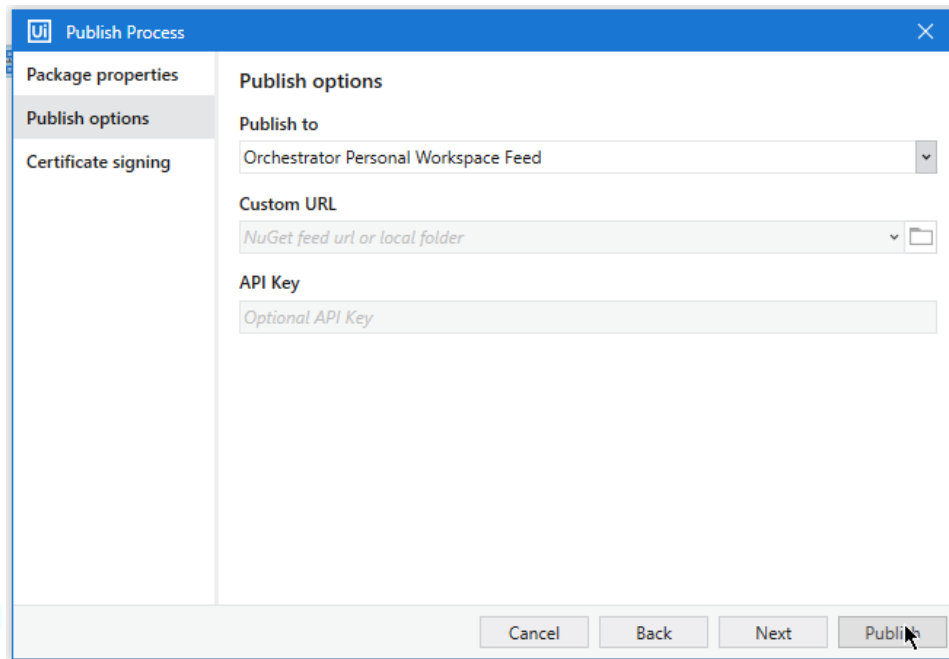
- Save all the workflows

edureka!

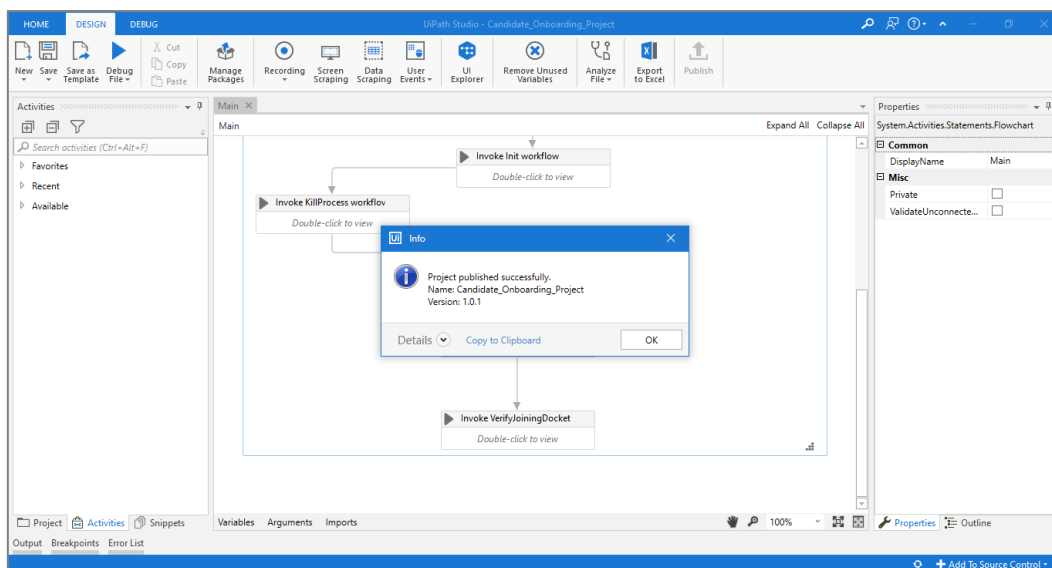
Candidate Onboarding Project Part – 7

54. We will publish the project to our Personal Workspace

55. Click Publish button on the ribbon bar. Select the Publish Option – “Orchestrator Personal Workspace Feed”



56. The message box shown confirms the solution is published to Orchestrator



- ✓ This completes the project. Connect UiPath Robot/Assistant to UiPath Orchestrator. You can run the process from Orchestrator or UiPath Assistant.

Sequences and Variables of The Project

Sequence	Variable	Type	Argument
1. InitialSend Mail	pwd	String	
2. Init	pathMyDoc	String	Out_PathCustomLog of String Type with Out direction
	pathCustomLog	String	
3.Download Candidate Master			
4. SendMail		String	ToEmailAddress with In direction of String Type
		String	SubjectEmail with In direction of String Type
		String	BodyEmail with In direction of String Type
		String	PathAttachment with In direction of String Type
5. Request Onboarding Info	dtCandidateMaster	DataTable	
	CandidateId	String	
	CandidateEmail	String	
	CandidateName	String	
	CandidateMailSubject	String	
	CandidateMailBody	String	
	PathJoiningDocket	String	

6. ValidateJoining Docket	strAadharText	String	In_IstAttachments with In direction of type System.Collections.Generic.IEnumerable<System.String>
	strVIDDocket	String	
7. VerifyJoining Docket	Ist_MailMessage	System.Collections.Generic.List<System.Net.Mail.MailMessage>	
	strEmailSubject	String	
	IstAttachments	In_IstAttachments with In direction of type System.Collections.Generic.IEnumerable<System.String>	