Robotic Enterprise Framework

Robotic Enterprise Framework is a project template based on the State Machine. This is made in accordance with all the best practices regarding

- Logging,
- Handling exceptions,
- Application initialization, and others, ready to overcome complex business scenarios.

❖ Why to use Reframe work in UiPath?

Reframe work makes a developer's tasks much easier. It works as a template and provides all the basic details that you as a developer may need for any process automation. The template lets you read and store the config data, open the required application and close all the unnecessary applications, get the transaction and process it, and when required it allows to retry the transaction.

Along with that it logs the status of all processed, failed & successful transactions. Hence to build a robust and fast process use Reframe work in UiPath. That also facilitates you these super cool features.

❖ How Reframe work works in UiPath?

If you have ever worked at UiPath studio then you might know it has 3 types of data flow representations: sequence, flowchart and state machine.
Though Reframe work contains all the 3 data flows, the main structure of the program is built using a **state machine** as it provides a more clean and functional data flow representation.
So, before knowing how the Reframe work works in the UiPath, let's have a look on it's architecture.

❖ Where do you get this Framework?

Before 2018, the UIPATH version We had this frame in Github from where we had to download the framework, but now UiPath your way has the frame in its UiPath studio.

- Go to UiPath studio
- New from Template ------------> Robotic Enterprise Framework

Once you open the UiPath studio you see the screen like this
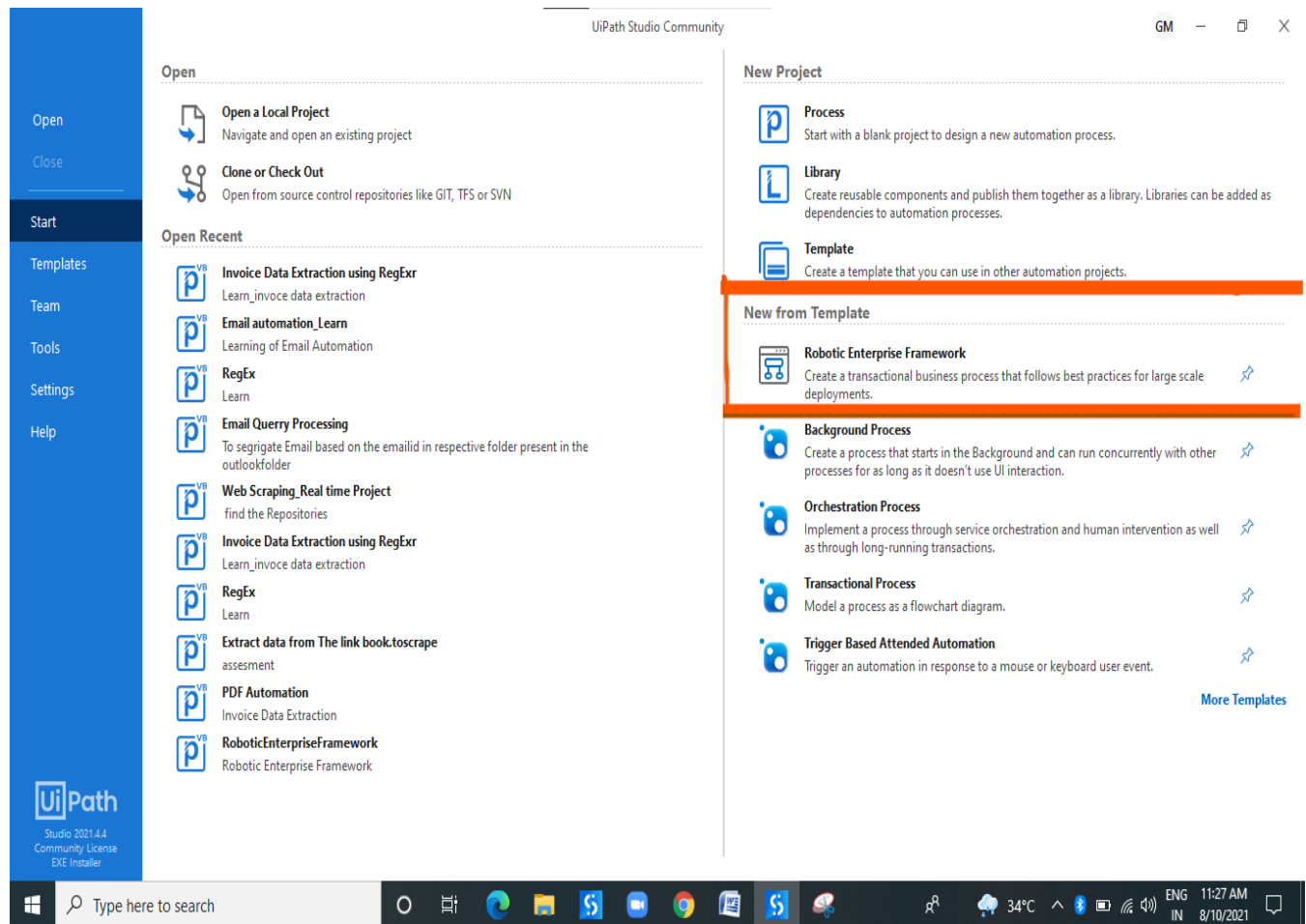
Once you click on it, the 'New robotic Enterprise framework' wizard will open up. Fill up the name, location, and description as per the project and click on the create button. Now you will encounter this screen.
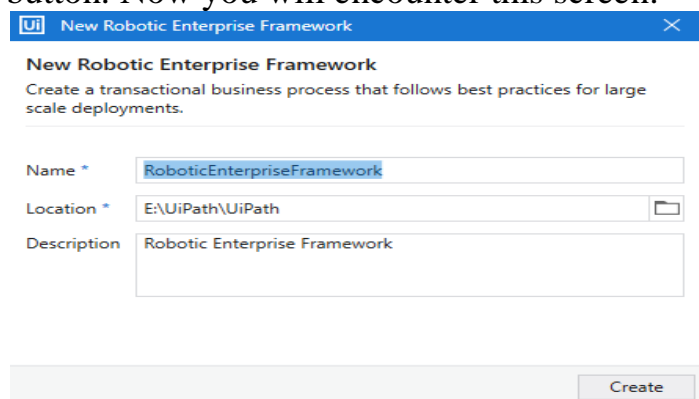
Reframe work Architecture:

Templates contain several pre-artificial country containers to initialize the application, retrieve input data, process and end the transaction. All of these states are connected through various transitions that cover almost every need in standard automation scenarios. There are also several workflows called, each handle certain aspects of the project.
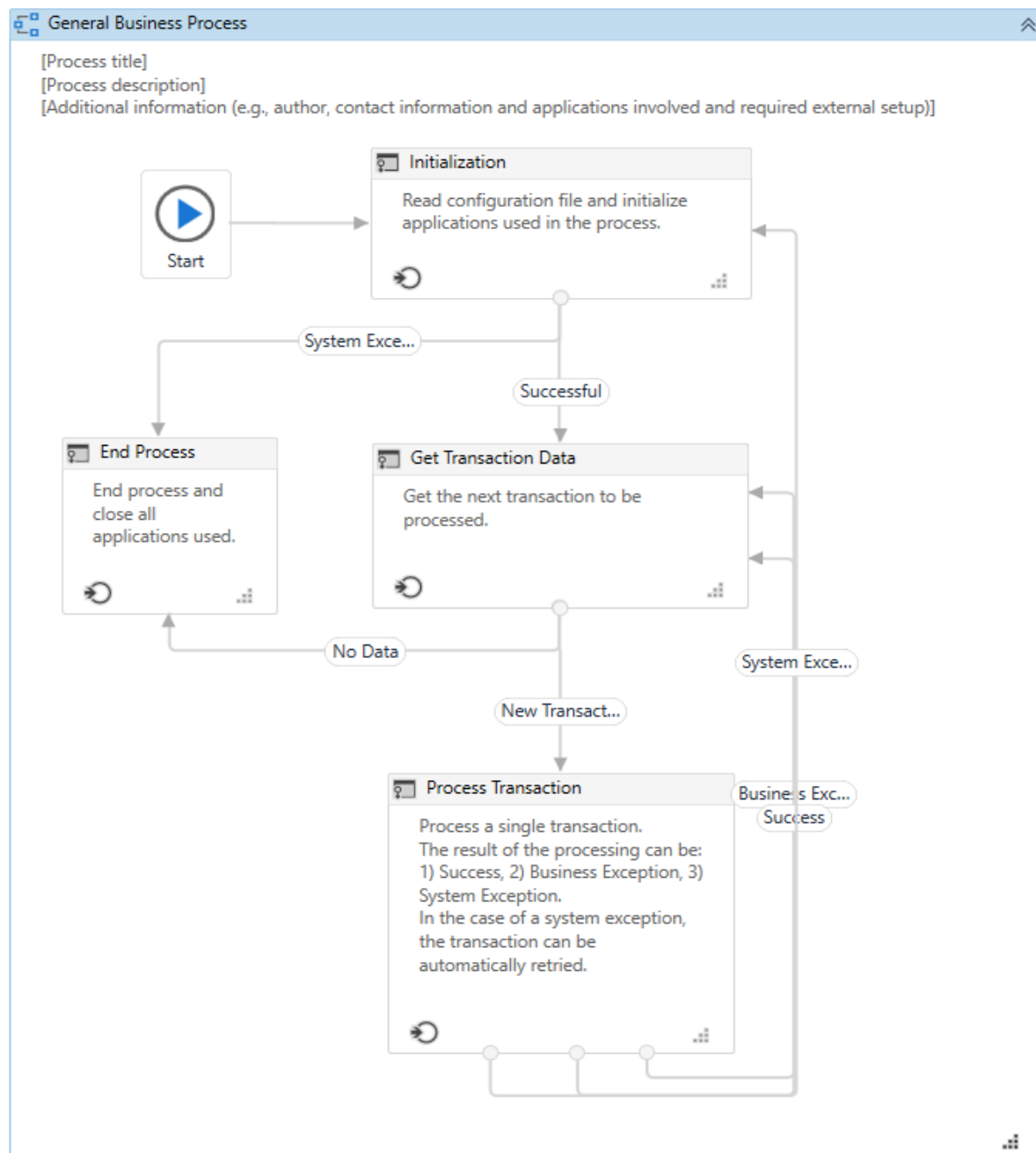
We have four stages in the Reframe work architecture

1. **Init**
2. **Get Transaction**
3. **Process Transaction**
4. **End Transaction**

As you can see, It has a main.xml file which uses the state machine workflow and each stage is connected with each other for the work processes.

Since the system can only be in one state at a time, at least one transition condition from a given state must be true for it to move to another state. If it gets failed in any of these states, Reframe work is built in a way to either retry it or end the process and send an exception log with the respective reason.

Overall, Reframe work architecture comprises 3 state activities,1final state activities and 7 transitions where each state is connected with each other with transitions.

Now, let's understand each stages of the framework in a better way:

Init state:–
- This is the initial state which is used to read and store the config data in the dictionary, close all the unnecessary applications and open the required one. For example, if in the Process we need to upload an attachment in SAP, we first require the Bot to fetch the credentials from Orchestrator, open the SAP application and then login to SAP using the credentials.
- The best part of UiPath Reframe work is that it is built in such a way that you can initialize all the applications necessary here in the Init state and if the conditions you have built are met, the Bot will then move forward with processing the transaction.
- If there is an error while initializing the applications or the credentials provided are invalid, an exception notification will be sent from within Init State and end the process.

Get Transaction-
- It is the data retrieval state, which is used to get the transaction from queue, data table, folder, database and other sources. Yes, once the Bot successfully initialized all the applications it requires for a process. Hence, it looks for any transaction items in the Orchestrator Queue to process.

- Reframe work prebuilt functionality looks for a Stop Request and will stop the process even though there are items still in the Queue to process.
- This becomes helpful in situations where you notice the Bot is not doing what it is expected to do and want to stop the process before it works on any other transaction items.

Process Transaction-

- In this state, the transactions are processed that are fetched from the previous state. And if there is no record to process, the robot automatically moves to the end process state. If we look again at our attaching a document in SAP example, The Bot, which has now got the Transaction Data from the Queue should now upload the attachment to SAP.
- And in Process State, all the activities required to attach the document will be invoked. For example, a workflow to check if the document exists or not on the Shared Path, a workflow with activities like clicking on appropriate GUI Elements for the Bot to navigate through SAP, type in the file path and look for the confirmation that the document was successfully attached.
- Reframe work has prebuilt SetTransactionStatus workflow that can be used to set the Transaction Status of an item to either Success, Business Exception or Application Exception. And among them, Business Exceptions are prebuilt exceptions that are built into the script and looked by a Bot.
- After the successful transaction, bot will now move forward with the next item in the Queue.

End Transaction-

This is the final state of Reframe work that ends the robot successfully and closes all the applications. After the Bot has finished processing all the transaction items from the Queue, it will log out of applications, irrespective of if it is a failed or successful transaction.

Here comes the actual *role of the UiPath Reframe work.* Because once you create the new framework, It stores all the dependencies and work flows in the one folder, you can go and check one by one.

To initialize the process you will see a screen similar to this:
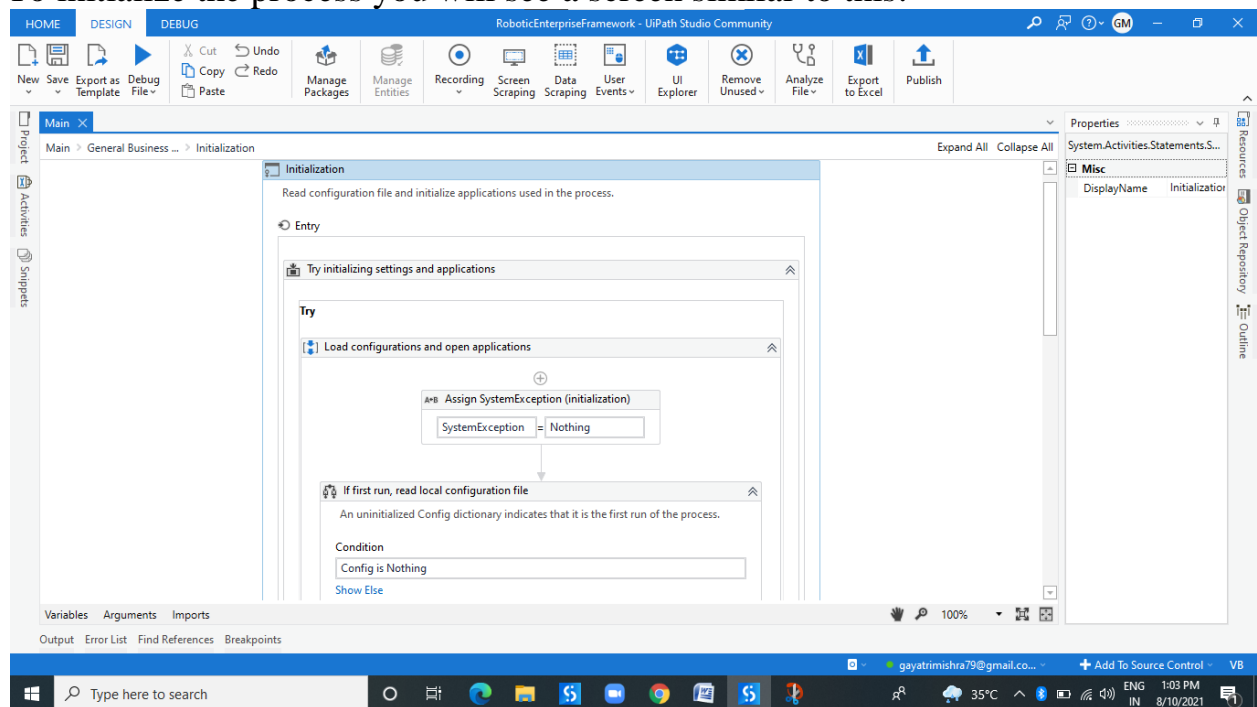
UiPath Reframe work becomes the best choice for several cases. For example:
You want to scrap 100 pages on a site that requires a login, then reframe work will ease the process in unmatchable speed. All you need to do is

1. Include the credentials required for the site in your Config file
2. Store the url of your site in the config file (as it is easier to change once in Excel than a lot of times in the workflows)
3. Now you can control the number of pages that need to be scraped
4. Also can retry the scraping on pages that failed.
5. That's it!!

Let's see some of the most commonly use cases where UiPath Reframe work is making things easier for the enterprises

Web Scrapping – Whenever you need to scrap data from various locations and want to feed them in a particular excel sheet. Like in the case of comparison sites, stock markets, e-commerce and other for a specific purpose in a large scale, then the framework can become very useful in making the process fast and errorless.
Inventory/supply chain management– As retailers rely on legacy systems for stock keeping. The Framework bots can perform constant checks on the systems providing data on key metrics like items with low stock levels or the one changing rapidly.

Customer/User onboarding– In many B2C businesses customer on-boarding is critical. And using this automation framework, customer on-boarding actions can be completed instantaneously even in companies that rely on legacy systems, that hugely improve the customer experience.

Generating mass emails– Mass emails relying on data from multiple systems are laborious to produce manually. Hence for enterprises who need to send them frequently, this automation framework is easing the process.

Validating and processing online loan applications – You can use this framework to built intermediary bots between the online system and the main frame that used business logic to ask the user to fix incorrect entries as well as make the loan decisions and generate confirmation letters.
These are just to name a few, their implementations can be seen on large scale at manufacturing, healthcare industries, banking, E-commerce and many other industries.
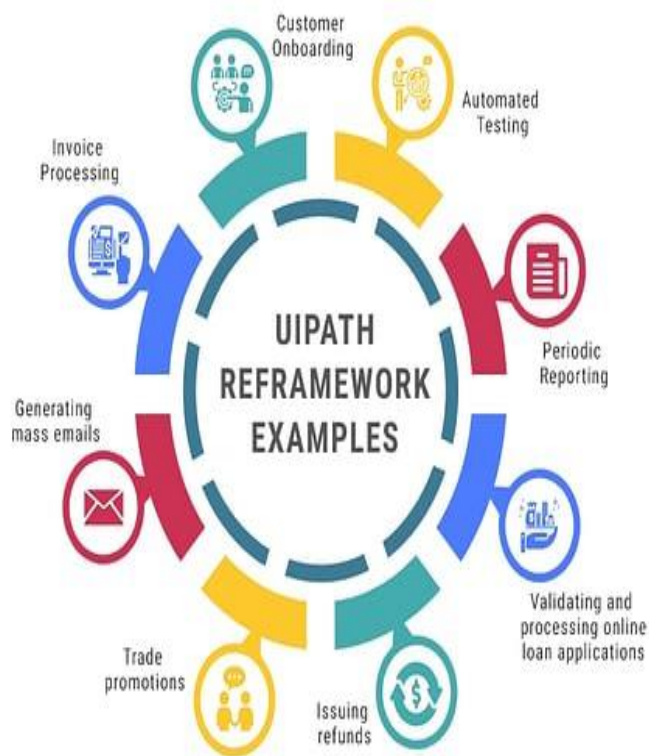
REAL TIME SCENARIO

Task: We need to pick data from the Excel sheet one by one for each row. Fill out the information on the screen of the HR On-Boarding website and establish an Employee ID.
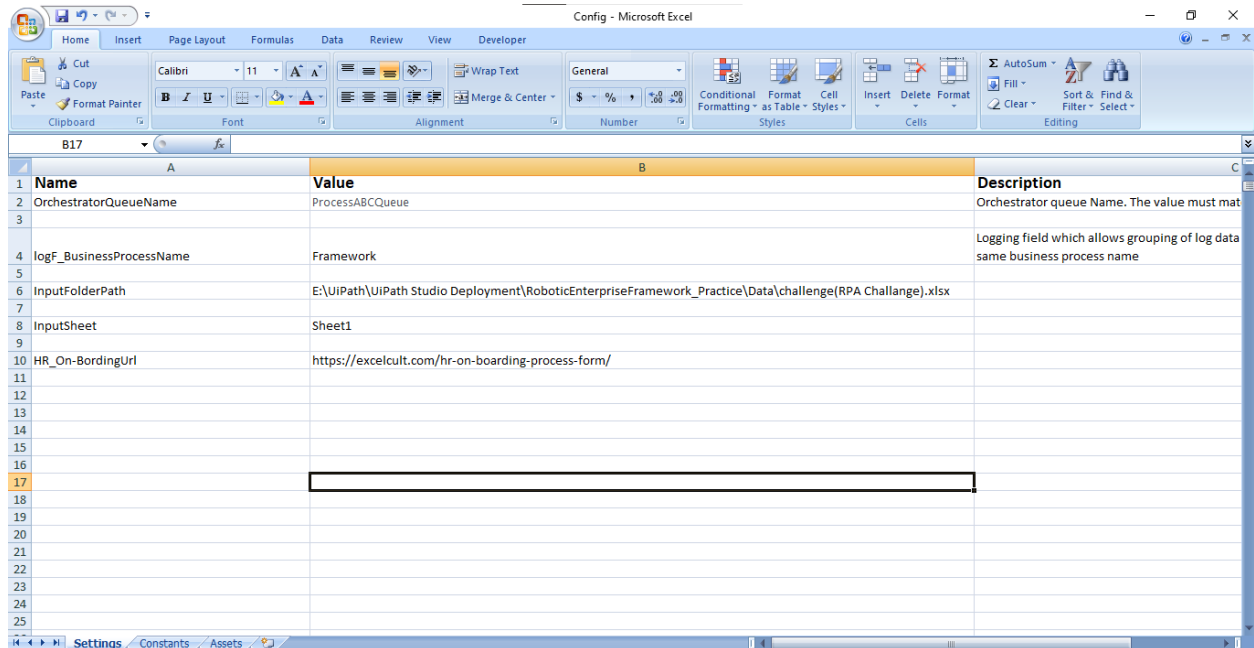
Solution:

Step 1.Create an Input Data Excel Sheet.

	A	B	C	D	E	F	G	H
	A2		fx	Jane				
1	First Name	Last Name	DOB	Role in Company	Address	Email	Phone Number	Gender
2	Jane	Dorsey	28/01/1988	Medical Engineer	11 Crown Street	jdorsey@mc.com	40791345621	Male
3	Albert	Kipling	29/01/1989	Accountant	22 Guild Street	kipling@waterfront.con	40735416854	Male
4	Michael	Robertson	25/04/1990	IT Specialist	17 Farburn Terrace	mrobertson@mc.com	40733652145	Male
5	Doug	Derrick	23/07/1991	Analyst	99 Shire Oak Road	dderrick@timepath.co.(	40799885412	Male
6	Suresh	Marlowe	24/01/1992	Scientist	27 Cheshire Street	jmarlowe@aperture.us	40733154268	Male
7	Stan	Hamm	28/05/1993	Advisor	10 Dam Road	shamm@sugarwell.org	40712462257	Male
8	Michelle	Norton	26/08/1994	Scientist	13 White Rabbit Stree	mnorton@aperture.us	40731254562	Male
9	Stacy	Shelby	22/01/1995	HR Manager	19 Pineapple Bouleva	sshelby@techdev.com	40741785214	Male
10	Ramesh	Palmer	22/01/1996	Programmer	87 Orange Street	lpalmer@timepath.co.u	40731653845	Male
11								

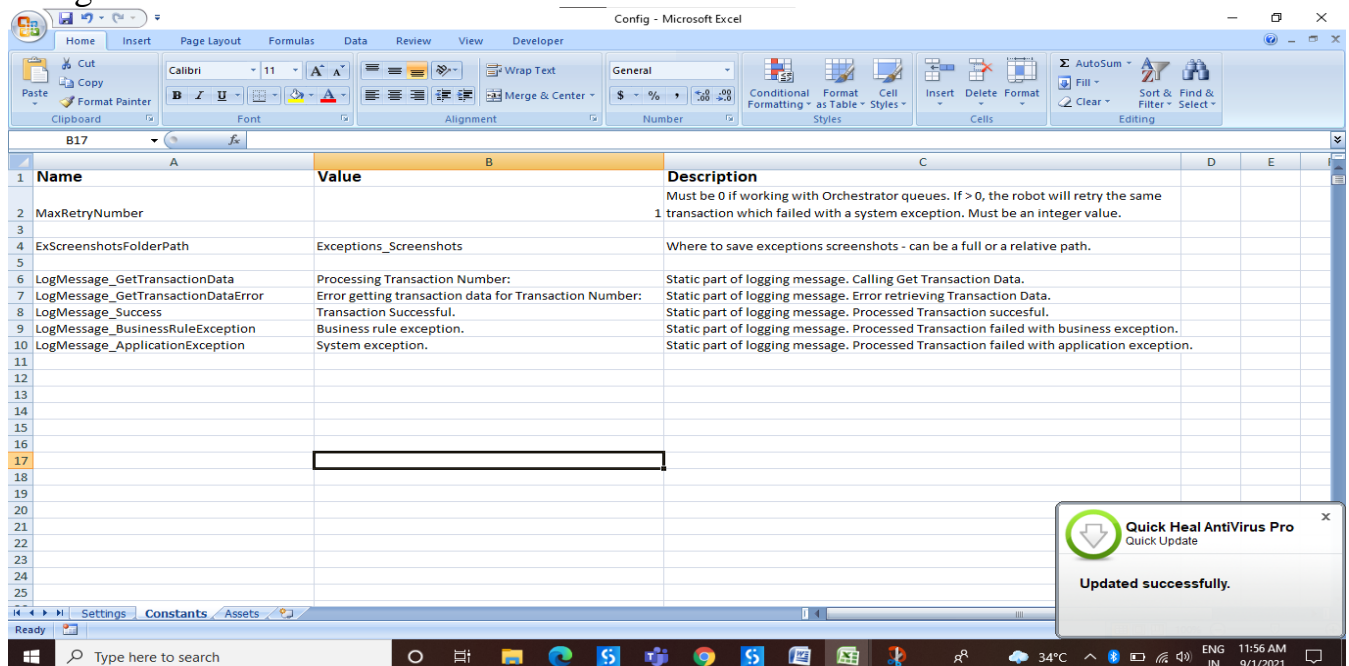Step 2. Edit The Config File in setting Sheet, by default Config file is there inside the data,
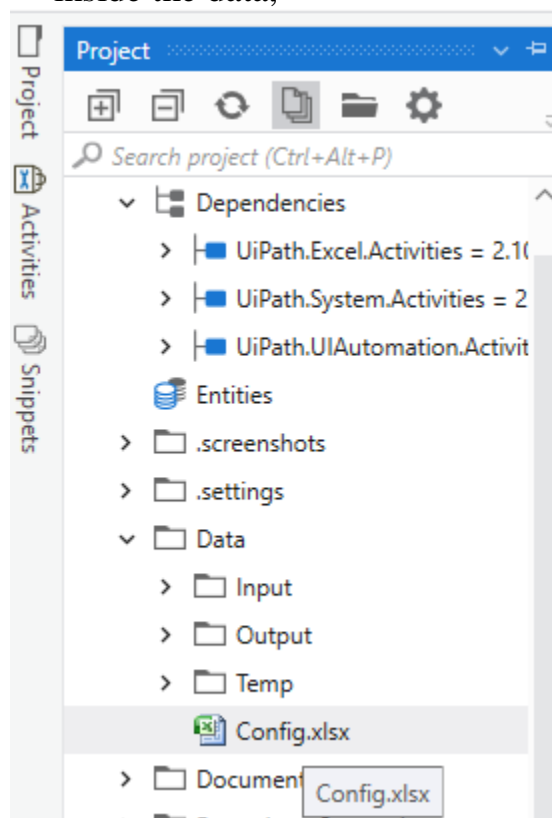
Step 3.Open the config file and modify it; there are three sheets, but we can only alter the **setting sheet**,Put the path to the input folder and the path to the input sheet, as well as the URL of the website.

Constant Sheet:If system Exception occurs MaximumRetryNumber we can change 0-1.
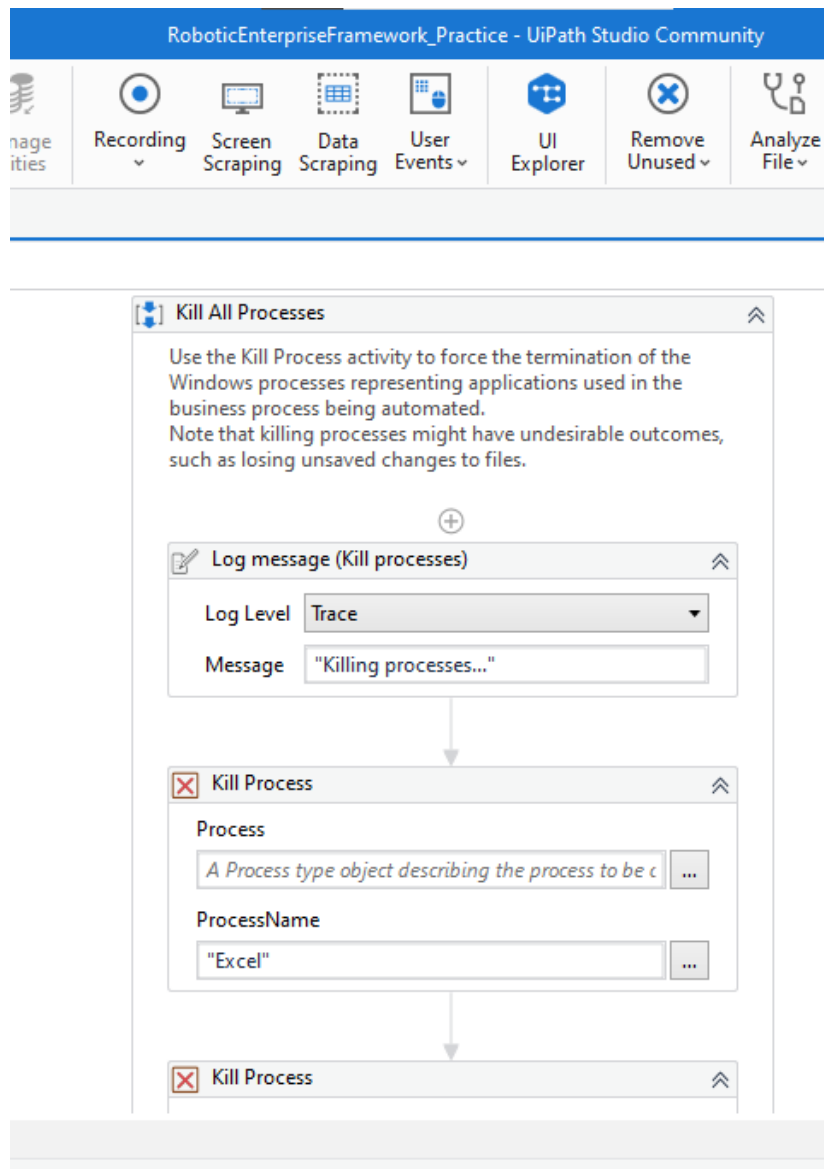
Step 4..
Go to UiPath Studio:

I. **Init state:** Read the Excel sheet &stored inside the data table.
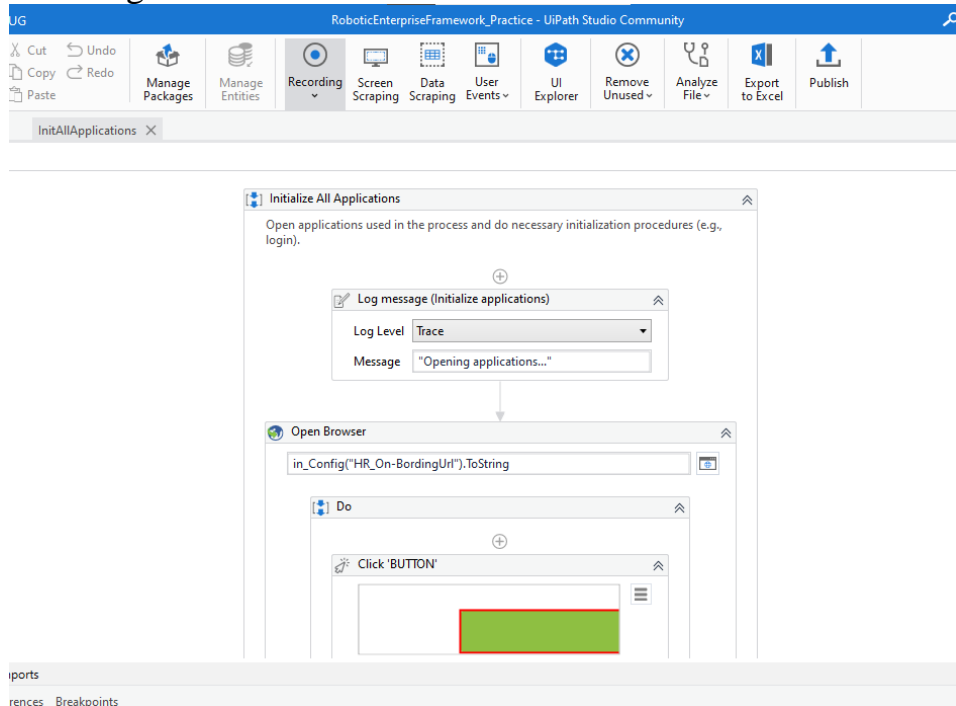Go to Invoke KillAllProcesses workflow (first run)⟶ Open Workflow
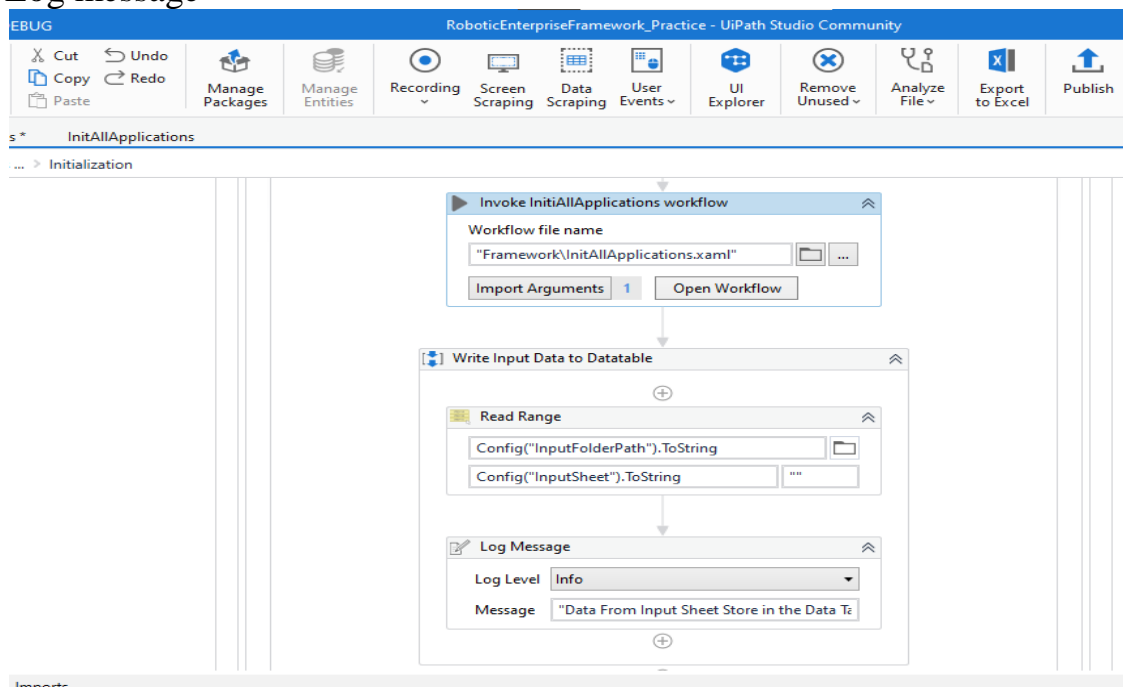
Kill All Process
Process Name: "Excel"
Process Name: "Chrome"

II. Go to InitAllApplications: Open workflow :Drag and drop the open browser activity and put the Url path and drop the Click Activity for Click in the New Registration bottom.

III. Create a sequence(Write input data to data table)
Read range activity
Log message
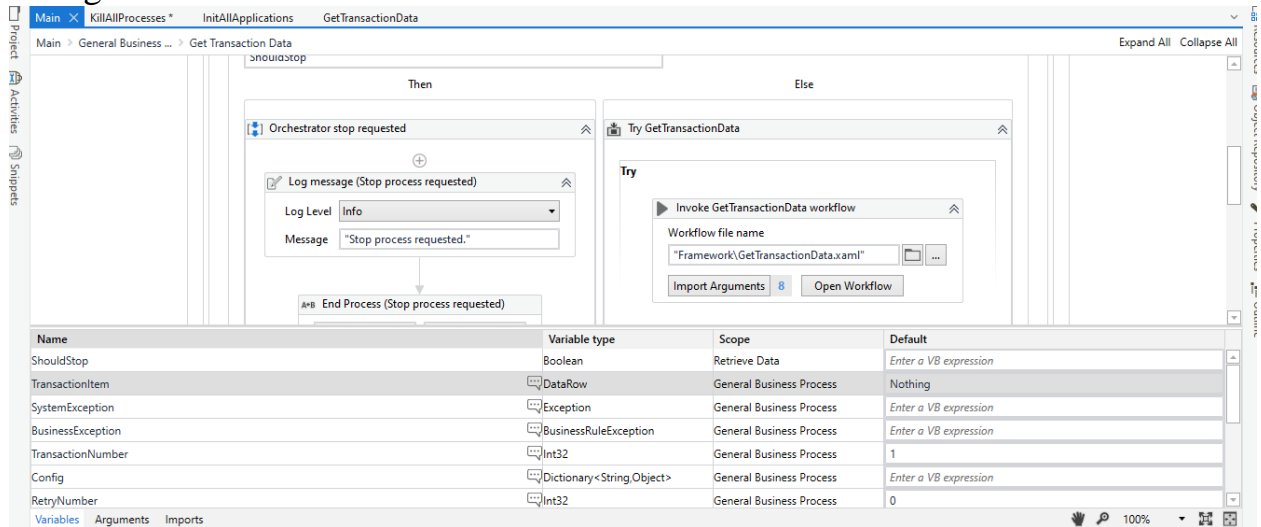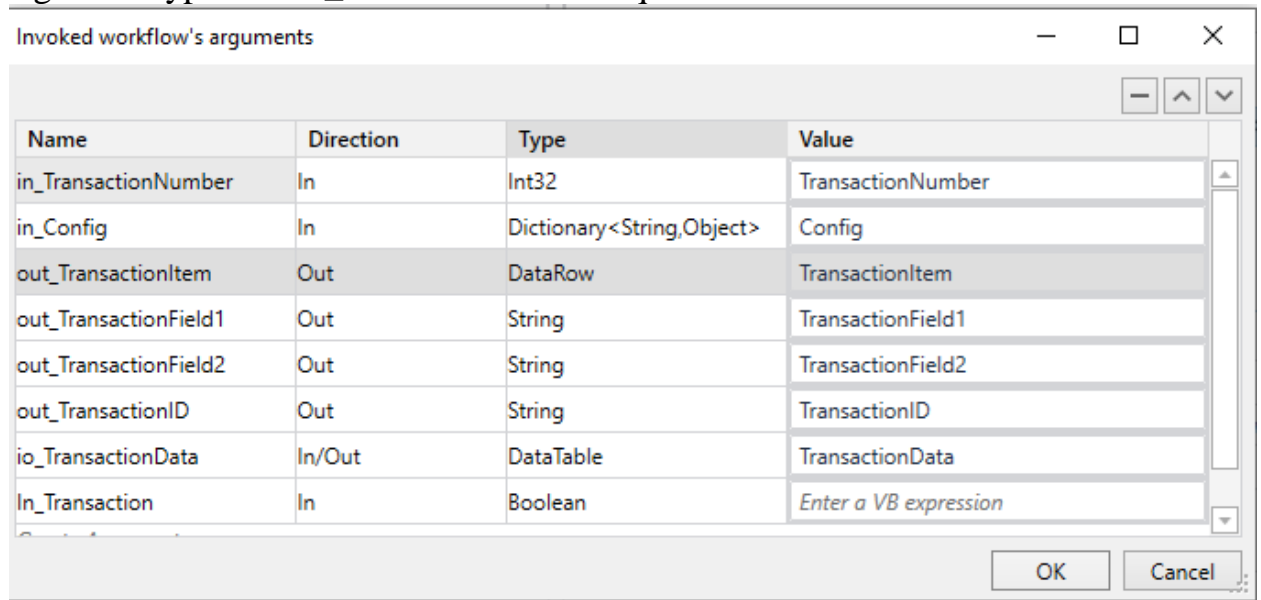
Step 5:
Get Transaction Data:

Invariable panel:
- ♦ Change the Variable type of Transaction item Queue item to Data row,
- ♦ Change the Transaction Number is 0.

- ♦ Also change in invoke get transaction data workflow and change the argument type of Out_Transactionitem is queue item to data row.

- ♦ Get transaction item we can disable because we are not using orchestrator queue.

In IF Activity We can change the
   Condition: In_TransactionData(io_Transactiondata.rows.count)
Then just modify the assign activity.
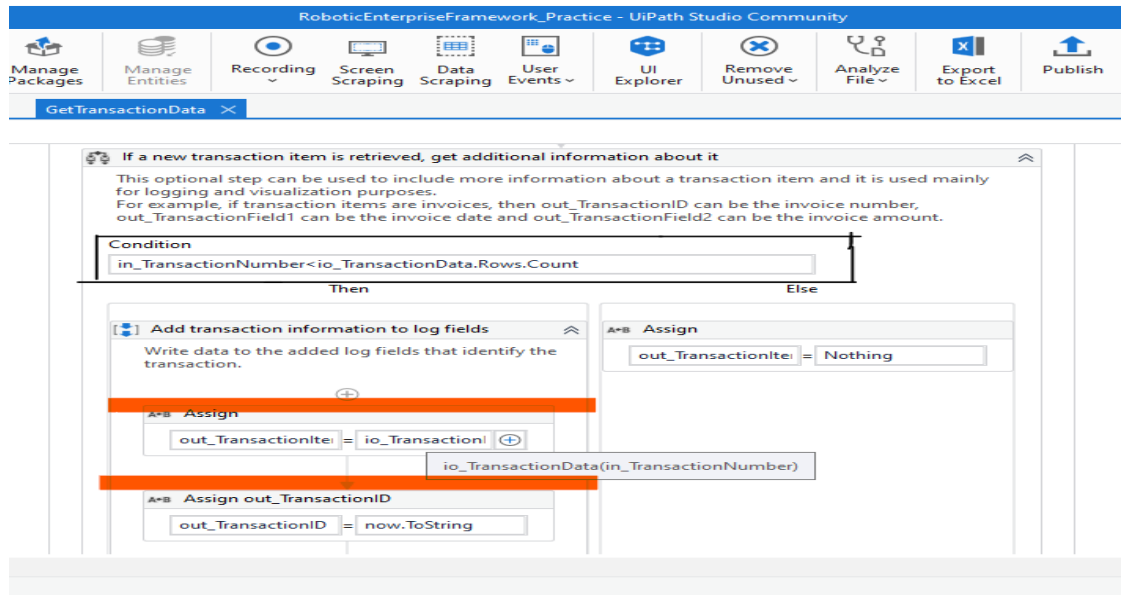
Step 6 :
Process Transaction:

- ♦ Go to invoke process workflow:
- ♦ Click the open workflow
- ♦ Drag &drop the sequence rename it fill the form
- ♦ Drag and drop the attach browser and in the DO block drag the for specific data we should use for Anchor base such as first name,last name,Dob,...etc.
- ♦ Then, underneath the anchor base, we should be utilised for finding elements and type into activities.
- ♦ Click on find element activity then open the selector modify the selector uncheck the **id.**
- ♦ and at last we use click activity for submit the data.

Check 'INPUT'

☑ I accept the Pr

Click 'INPUT'

Get Text 'INPUT Employeeld'

Employee Id* 412

Log Message

Log Level   Info

Message   "Employee ID"+GetTxt_Ver.ToString
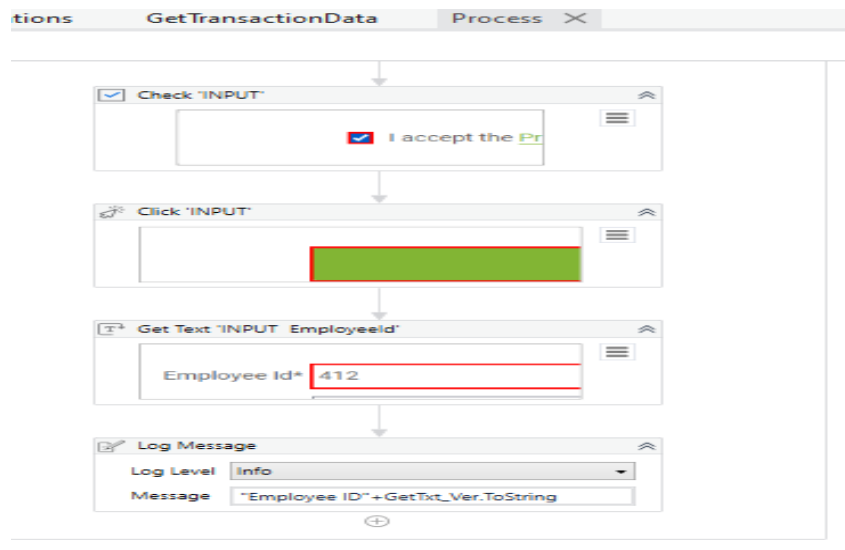
For Gender click the Radio bottom

itAllApplications   GetTransactionData   Process ✕

If

Condition

Gender_Ver.Contains("Female")

Show Else

Then

Click 'INPUT male'

Gender* ☐ Male ○ Female

If

Condition

Gender_Ver. Contains("Male")

Show Else

Then

Click 'INPUT female'

ender* ○ Male ☐ Female
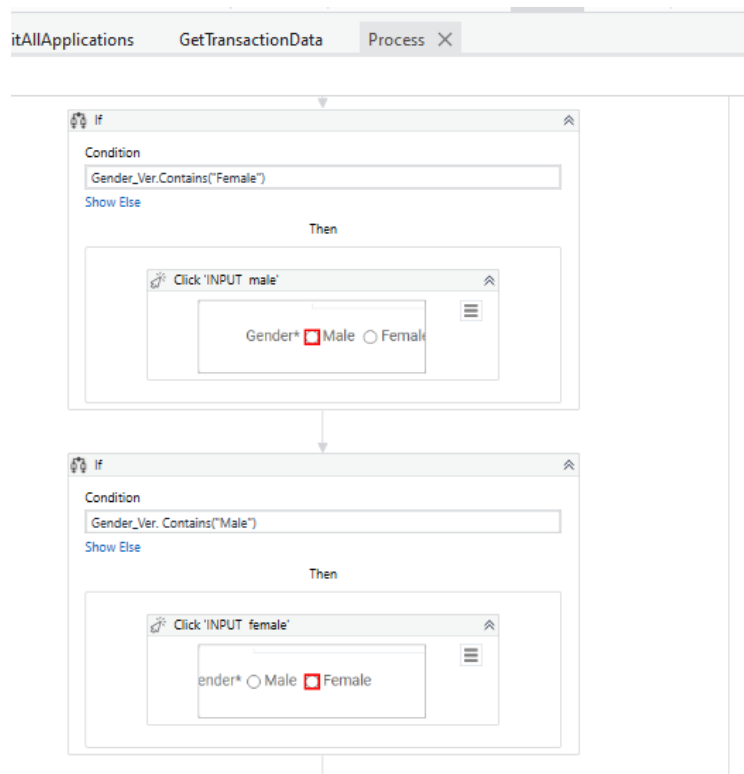
We want automate to check the accept bottom and click the submit bottom and get the employee ID.

So I use the activity Check activity, click activity and get text activity.

Output:

1.Automatically fill up the data on the screen

HR On-Boarding Process Form

NEW REGISTRATION

Personal Details

Employee Id* 258   Official Mail Id* dderrick@timepath.co.uk

First Name* Doug   Last Name* Derrick

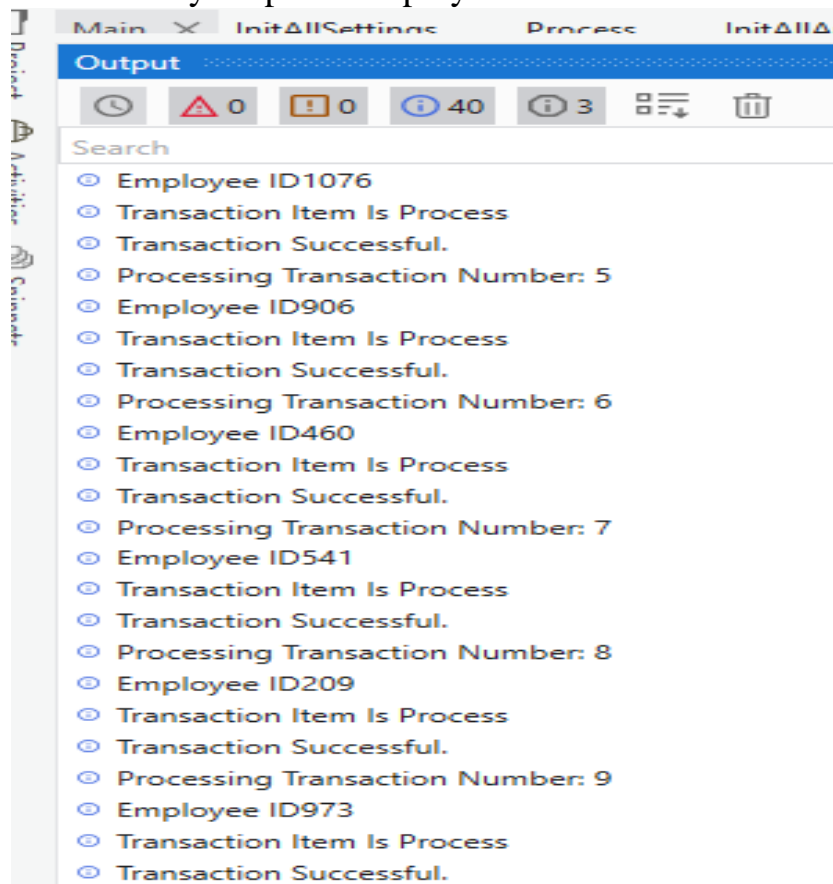DOB* 23/07/1991   Designation* Analyst

Mobile Number* 40799885412   Gender* ● Male ○ Female

Terms and Mailing

☑ I accept the Privacy Policy for W3Docs.

SUBMIT

2.Here is my output &Employee ID

Main ╳   InitAllSettings   Process   InitAllA

Output

🕐   ⚠ 0   ❗ 0   ⓘ 40   ⓘ 3   ▦   🗑

Search

⊚ Employee ID1076
⊚ Transaction Item Is Process
⊚ Transaction Successful.
⊚ Processing Transaction Number: 5
⊚ Employee ID906
⊚ Transaction Item Is Process
⊚ Transaction Successful.
⊚ Processing Transaction Number: 6
⊚ Employee ID460
⊚ Transaction Item Is Process
⊚ Transaction Successful.
⊚ Processing Transaction Number: 7
⊚ Employee ID541
⊚ Transaction Item Is Process
⊚ Transaction Successful.
⊚ Processing Transaction Number: 8
⊚ Employee ID209
⊚ Transaction Item Is Process
⊚ Transaction Successful.
⊚ Processing Transaction Number: 9
⊚ Employee ID973
⊚ Transaction Item Is Process
⊚ Transaction Successful.

<u>Reference:</u>

- **https://www.youtube.com/watch?v=jk7idLC9bHQ&t=370s**
- **https://excelcult.com/hr-on-boarding-process-form/**

Thank You

For any implementation in your organization please reach out to rpa@gxplabs.com