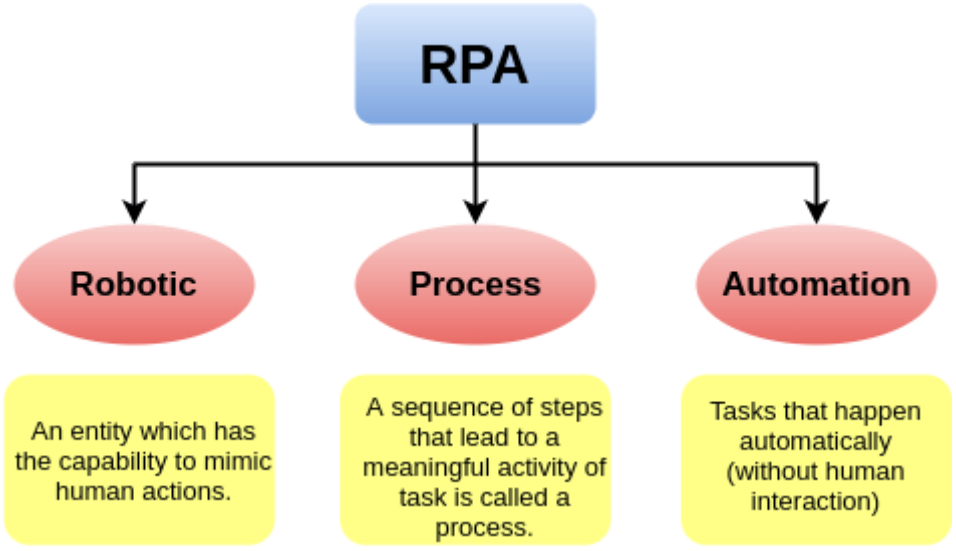
Module 1: RPA Foundations

1.1 What is RPA?

- RPA stands for **Robotic Process Automation.**
- It is the technology used for software tools that automate human tasks, which are manual, rule-based, or repetitive.
- Typically, it is like a bot that performs such tasks at a much higher rate than a human alone.
- These RPA software bots never sleep and make zero mistakes, and can interact with in-house applications, websites, user portals, etc.
- They can log into applications, enter data, open emails and attachments, calculate and complete tasks, and then log out.



- The word 'Robot' in 'RPA' is not a physical robot but a virtual system that helps in automating the repetitive manual computing or business process tasks.

Why RPA

- Robotic Process Automation is economically capable as compared to any other automation solutions.

- It is the new buzz word in the IT industry. It has shifted the traditional way of doing the business task manually into an automatic task within an organization.
- RPA technology uses bots that interact with web applications, web sites, excel worksheets, and emails to automate the tasks just like a human.

BENEFITS OF RPA

Robotic Process Automation technology provides the following benefits:

Cost Savings

RPA helps organizations to save a huge amount of cost as it is typically cheaper than hiring an employee to perform the same set of tasks.

Less Error

RPA works on standard logic and does not get bored, distracted, or tired. Hence, the probability of making errors reduces to a great extent, which means less re-work and an enhanced reputation for efficiency.

Faster Processing

RPA works faster than human employees as computer software does not need breaks, food, rest, etc., and can perform repetitive operations tirelessly. With RPA, processing time becomes predictable and consistent, which ensures high-quality customer service across the operations.

Better Regulatory Compliance

RPA software works on the logic and data fed to it and does what is only needed as per the given instructions. Hence, there are minimal chances of not complying with the standard regulations.

Better Customer Service

When RPA is implemented in a business, it frees many of its employees who can spend their time working on customer-related services. It is very beneficial for businesses that receive a lot of customer queries. It also leads to increased productivity for employees.

Auditable & Secure

RPA bots will only access the data for which they are given permission and create a detailed audit trail of all activity.

Low Technical Barrier

RPA does not require any programming skills to configure the software robot. Since it is a code-free technology, any non-technical person can set up the bot using drag and drop features. It also includes the 'Recorder' to record the steps of automation.

1.2 Flavours of RPA

On a high level, you can divide the flavors into the following:

1) Attended RPA (which may be referred to as robotic desktop automation or RDA):

- This was the first form of RPA that emerged, back in 2003 or so.
- Attended RPA means that the software provides collaboration with a person for certain tasks.
- Example: would be in the call center, where a rep can have the RPA system handle looking up information while he or she talks to a customer.

2) Unattended RPA:

- This technology was the second generation of RPA.
- With unattended RPA, you can automate a process without the need for human involvement – that is, the bot is triggered when certain events happen,
  Example: such as when a customer e-mails an invoice.
- Consider that unattended RPA is generally for back-office functions.

3) Intelligent process automation or IPA (this may also be referred to as cognitive RPA):

- This is the latest generation of RPA technology, which leverages AI to allow the system to learn over time
  Example: would be the interpretation of documents, such as invoices.
- There may be even less human intervention .

1.3 History of RPA

During the past 70 or so years that computers have been a major catalyst for this trend. Along the way, there have been different period's of automation, based on the types of technologies available. They would also provide a foundation for RPA platforms.

- **Mainframe Era**: These were huge machines developed by companies like IBM. They were expensive and mostly available to large companies (although, innovators like Ross Perot would create outsourcing services to provide affordable options). Yet they were incredibly useful in helping manage core functions for companies, such as payroll and customer accounts.

- **PC Revolution**: Intel's development of the microprocessor and Microsoft's development of its operating system revolutionized the technology industry. As a result, just about any business could automate processes; say by using word processors and spreadsheets.

    - But the automation technologies – while powerful – still had their drawbacks. They could easily result in complex IT environments, which required expensive and time-consuming integrations and custom coding.
    - From this emerged the key elements for RPA, which came about in the early 2000s.
    - A big part of this was screen scraping, which is the automation of moving data among applications, which turned out to provide a nice boost to efficiency and effectiveness.
    - But the nascent RPA market got scant attention. It was mostly perceived as low-tech and a commodity.
    - Instead, investors and entrepreneurs in Silicon Valley focused their attention on the rapidly growing cloud market that was disrupting traditional IT systems.

But around 2012 or so, the RPA market hit an inflection point. There was a convergence of trends that made this happen, such as the following:

- In the aftermath of the financial crisis, companies were looking for ways to lower their costs. Simply put, traditional technologies like ERP were reaching maturation. So companies needed to look for new drivers.

- Companies also realized they had to find ways to not be disrupted from technology companies. RPA was considered an easier and more cost-effective way to go digital.
- Some industries like banking were becoming more subject to regulation. In other words, there was a compelling need to find ways to lessen the paperwork and improve audit, security, and control.
- RPA technology was starting to get more sophisticated and easier to use, allowing for higher ROI (return on investment).
- Large companies were starting to use RPA for mission-critical applications.
- Demographics were also key. As the millennials started to enter the workforce, they wanted more engaging work. They wanted careers, not jobs.

➢ Fast forward to today, RPA is the fastest growing part of the software industry. According to Gartner, the spending on this technology jumped by 63% to $850 million in 2018 and is forecasted to reach $1.3 billion by 2019. Or consider the findings from Transparency Market Research. The firm projects that the global market for RPA will soar to $5 billion by 2020.

1.4 The Benefits of RPA

When looking at RPA, the benefits are far more than just about the impact on the bottom line. The technology can transform a company.

➢ **The Impact of Small Improvements**: On the surface, an employee who saves 10 to 20 seconds on a task – even something as simple as a series of cut-and-paste actions –may seem trivial. But it's not. When scaled across thousands of employees across a global organization, the impact can certainly be significant.

- For example, some companies will keep track of the metric of how many hours are saved by using RPA, which becomes a part of the overall ROI calculation.

➢ **Relative Ease of Implementation**: Unlike traditional business applications like a CRM or ERP, RPA generally does not involve an onerous implementation and integration. Why? Note that the software sits on top of existing IT systems.

- RPA is also relatively easy for a person to use since there is no requirement for understanding complex coding.

- **The bottom line**: The people implementing RPA will get to their objectives quicker and the IT department will have more time to devote to higher priority items. This is important as there remains a trend of less investment in IT.
- **Compliance**: Just one violation of a government regulation can have a serious adverse impact on a company. It could even be a threat to its very existence.

    - Examples like Enron or Theranos:
    - (While employees are usually diligent and trustworthy, they do make mistakes or they may not understand some of the regulations. Yet this is not an issue with RPA. You can easily configure a bot to make sure the actions are compliant with regulatory requirements. Many RPA vendors also have built in their own compliance systems, handling such laws as the Sarbanes–Oxley Act, General Data Protection Regulation (GDPR), and HIPAA (Health Insurance Portability and Accountability Act of 1996)).

    - Another compliance benefit is that there will be less intervention with the data from people, which lessens the possibility of fraud.

- **Customer Service**: Nowadays, people want quick and accurate responses from their companies. But this is difficult to provide, especially when a company is overwhelmed from incoming contacts.But this is where RPA can make a big difference. The bots are programmed to make sure that all the necessary steps are taken – at scale. The result is often an increase in customer satisfaction metrics, like the Net Promoter Score (NPS).

- **Employee Satisfaction**: Yes, your team should also enjoy the benefits of RPA. After all, it means that they do not have to spend their valuable time on tedious activities. The result may be less turnover and higher productivity.
- **Wide Application**: It's common for an enterprise application to focus on a certain part of a company's departments or functions. But RPA is wide. It can be used for virtually any part of a company, such as legal, finance, HR, marketing, sales and so on.

- **Data Quality**: It should be greatly improved as there will be less chance of human error. In fact, there will probably be much more data because of the scalability of the automation. In other words, the datasets for analytics and AI will be more robust and useful.

- **Digital Transformation**: This is a major priority for CEOs. But many companies have legacy systems that would be expensive to replace or integrate. However, RPA is an approach that can help with this process, which is often quicker and less costly.
- **Scalability:** If there is a sudden jump in demand, it can be extremely difficult to hire new employees. But RPA can be a solution. It is much cheaper and faster to ramp up new bots to meet the demand.

1.5 The Downsides of RPA

RPA is definitely not a cure-all. The software has its inherent limitations and complexities.

- **Cost of Ownership**: The business models vary. Some have a subscription or multiyear license. Other vendors may charge based on the number of bots.
- **But there is more to the costs**. There is the need for some level of training and ongoing maintenance. Depending on the circumstances, there may be requirements for buying other types of software and hardware. Oh, and it iscommon to retain third-party consultants to help with the implementation process.
- **Technical Debt**: This is an issue with RPA. As a company's processes change, the bots may not work properly. This is why RPA does require ongoing attention.
- **Enterprise Scale** :. It can be extremely difficult to manage the numerous bots and there also needs to be strong collaboration among IT.
- **Security :** This is a growing risk with RPA implementations, especially as the technology covers more mission-critical areas of a company's processes. Let's face it, if there is a breach, then highly sensitive information could easily be obtained. Actually as RPA gets more pervasive in manufacturing, there may even be risks of property damage and bodily harm. This would likely be the case with attended RPA.
- **Expectations**:. According to a survey from PEGA, the average time it takes to develop a quality bot was 18 months, with only 39% being deployed on time.
- **Preparation**: You need to do a deep dive in how your current tasks work. If not, you may be automating bad approaches.
- **Limits**: RPA technology is somewhat constrained. For the most part, it works primarily for tasks that are routine and repetitive. If there is a need for judgment – say to approve a

payment or to verify a document – then there should be human intervention. Although, as AI gets more pervasive, the issues are likely to fade away.

For example, insurance companies can use the technology to adjudicate claims for payments, based on individual claims history and firm-wide payment policies.

- **Virtualized Environments**: This is where a desktop accesses applications remotely, such as through a platform. However, some of the latest RPA offerings, such as from UiPath, are solving the problem.

1.6 RPA Compared to BPO, BPM, and BPA

Business process management (BPM)

Business process outsourcing (BPO)

Business process automation (BPA)

They can get kind of confusing but they have key distinctions.

- **BPM**:

  For example, FileNet introduced a digital workflow management system to help better handle documents (the company would eventually be purchased by IBM). Then there would come onto the scene ERP vendors, such as PeopleSoft

  - .All of this would converge into a major wave called BPM .
  - For the most part, the focus was on having a comprehensive improvement on business processes. This would encompass both optimizing systems for employees but also IT assets.
  - There were also various business process management software (BPMS) solutions to help implement BPM.
  - One was Laserfiche. Nien-Ling Wacker founded the company in 1987, when she saw the opportunity to use OCR (optical character recognition) technology to allow users to search huge volumes of text.

So then how is BPM different from RPA?

With BPM, it requires much more time and effort with the implementation because it is about changing extensive processes, not tasks. There also needs to be detailed documentation and training. Because of this

rigorous approach, BPM is often attractive to industries that are heavily regulated, such as financial services and healthcare.

However, the risk is that there may be too much structure, which can stifle innovation and agility.On the other hand, RPA can be complementary to BPM. That is, you can first undergo a BPM implementation to greatly improve core processes. Then you can look to RPA to fill in the gaps.

- **BPO** : This is when a company outsources a business service function like payroll, customer support, procurement, and HR.
    - The market is massive, with revenues forecasted to reach $343.2 billion by 2025 (according to Grand View Research). Some of the top players in the industry include ADP, Accenture, Infosys, IBM, TCS, and Cognizant.

As should be no surprise, one of the big attractions of BPO is the benefit of lower wage rates in other countries (this is often referred to as "labor arbitrage"). The employee bases will also often be educated and multilingual.bases will also often be educated and multilingual.

BPO will have three types of strategies:

- Offshore: This is where the employees are in another country, usually far away.
- Nearshore: This is when the BPO is in a neighboring country. True, there are usually higher costs but there is the benefit of being able to conveniently visit the vendor. This can greatly help with the collaboration.
- Onshore: The vendor is in the same country. For example, there can be wide differences in wages in the United States.

There are drawbacks with a BPO : Perhaps the most notable one is the quality issue (you know the situation when you call a company and get an agent you can barely understand!).

Yet here are some others to consider:

- Security: If a BPO company is developing an app with your company's data, are there enough precautions in place so there is not a breach? Even if so, it can still be difficult to enforce and manage.
- Costs: Over the years, countries like China and India have seen rising labor costs. This has resulted in companies moving to other locations, which can be disruptive and expensive.
- Politics: This can be a wildcard. Instability can easily mean having to abandon a BPO operator in a particular country.

- **BPA** : This is the use of technology to automate a complete process. One common use case is onboarding.

For example, bringing on a new employee involves many steps, which are repeatable and entail lots of paperwork. For a large organization, the process can be time-consuming and expensive. But BPA can streamline everything, allowing for the onboarding at scale.

OK, this kind of sounds like RPA, right? Yes, this is true. But there is a difference in degree. RPA is really about automating a part of the process, whereas BPA will take on all the steps.

1.7 Consumer Willingness for Automation

The automation of consumer-facing activities, such as with chatbots on a smartphone or web site, are becoming more ubiquitous.

Consider a report from Helpshift, an AI-based digital customer service platform automating 80% of customer support issues for huge D2C (direct-to-consumer) brands including companies like Flipboard, Microsoft, Tradesy, and 60 others. Its report is based on the analysis of 75 million customer service tickets and 71 million bot-sent messages.

Here are some of the findings:

- A total of 55% of the respondents – and 65% of millennials – prefer chatbots with customer service so long as it is more efficient and reduces phone time to resolve an issue and explain a problem.
- A total of 49% say they appreciate the 24/7 availability of chatbots.- Granted, there is much progress to be made. Chatbot technology is still in the early phases and can be glitchy, if not downright annoying in certain circumstances. But in theyears to come, this form of automation will likely become more important – and also a part of the RPA roadmap.
- According to the CEO of Helpshift, Linda Crawford: “Seeing as the vast majority of Americans dread contacting customer support, there’s a huge opportunity here for chatbots to fill the void and improve the customer support experience for consumers—and agents

1.8 The Workforce of the Future

- The interesting thing is that the fundamentals of work have not changed much since then. True, there has been the trend of the gig economy, in which people get paid for offering services through Uber and Lyft. Yet when it comes to office work, the structure has remained quite durable.
- . According to research from the McKinsey Global Institute, white collar workers still spend 60% of their time on manual tasks, such as with answering e-mails, using spreadsheets, writing notes, and making calls.

- In light of all this, RPA is likely to have a significant impact on the workplace because more and more of the repetitive processes will be automated away. One potential consequence is that there may be growing job losses.
- A survey from Forrester predicts that – as of 2025 – software automation will mean the loss of 9% of the world's jobs or 230 million. Then again, the new technologies and approaches will open up many new opportunities.
- Its analysis shows that technologies like RPA could automate a whopping 45% of the activities of a company's workforce. Now when a company engages in an automation project, the CEO will usually not talk about job loss. It's something that will frighten the workforce and generate awful headlines. Instead the messaging will be vague, focusing on the overall benefits of the transformation.
- This may make it sound like not much is happening. But it does seem like a good bet that the reverberations will grow and grow, as RPA systems get increasingly robust. As we've seen in prior periods where technology resulted in job loss – such as in the Industrial Revolution – there are serious changes in politics and regulations.
- Companies really do try to avoid layoffs, since they are expensive and take a toll on the organization. But in the years ahead, managers will probably need to find ways to navigate the changes from automation, such as finding new roles or reskilling the workforce.
- All in all, the rise of automation has the potential for leading for a much better society. Again, workers can focus on more interesting and engaging activities – not repetitive and mundane tasks. There will also be ongoing renewing of knowledge and understanding. But there must be proactive efforts, say from companies and governments, to provide for a smoother transition

---

The Technologies You Need to Know

While RPA does not require programming skills, there is still a need to understand high-level concepts about technology. Unfortunately, the concepts can get extremely complex and confusing. It seems like there is an endless number of acronyms like ACL, API, OCR, CPU, HTTP, IP, JSON, NOC, PCI, RAM, and SaaS.

Even tech veterans do not know many of the terms – or have just a vague understanding of their meanings. For example, here's how Kubernetes is defined:

Kubernetes (K8s) is an open-source system for

automating deployment, scaling, and management of containerized applications.[1]

Huh? To get a sense of this, you really need to have a deep understanding of computer and software architecture.

But the good news is that – to use RPA effectively – there are only a handful of terms and concepts you need to know. So this is what we'll cover in this chapter.

On-Premise Vs. the Cloud

The traditional IT system approach is the use of on-premise technology. This means that a company purchases and sets up its own hardware and software in its own data center.

Some of the benefits include:

- A company has complete control over everything. This is particularly important for regulated industries that require high levels of security and privacy.
- With on-premise software, you may have a better ability to customize the solution to your company's unique needs and IT policies.

However, the on-premise technology model has serious issues as well. One of the biggest is the cost, which often involves large up-front capital expenses. Then there is the ongoing need for maintenance, upgrades, and monitoring. And finally, the use of point applications like Excel can lead to a fragmented environment, in which it becomes difficult to centralize data because there are so many files spread across the organization.

- But as the Internet became more robust, there was a move to so-called cloud computing.
- One of the first business applications in this industry was developed by Salesforce.com, which made it possible for users to use the software through a browser.

- Companies could pay per-user, per-month fees for the services they used, and those services would be delivered to them immediately via the Internet, in the cloud.

The downsides with cloud software. Here are just some to consider:

- With less control of the platform, there are more vulnerability to security and privacy lapses.
- Outages do happen and can be extremely disruptive and costly for enterprises that need reliability.
- Cloud computing is not necessarily cheap. In fact, one of the biggest complaints against Salesforce.com is the cost.
- Regardless, the fact remains that the technology continues to gain traction.

Besides the impact of Salesforce.com and other cloud applications companies, another critical development was Amazon.com's AWS platform.

AWS essentially handles the complex administrative and infrastructure requirements like storage, security, compute, database access, content delivery, developer tools, deployment, IoT (Internet of Things), and analytics (there are currently more than 165 services).

This means the development of applications can be much quicker. The costs are generally lower and the fees are based on a per-use basis.

The cloud also has different approaches, such as the following:

- **Public Cloud**: The cloud is accessed from remote servers, such as from AWS, Salesforce.com, and Microsoft. The servers have an architecture known as multitenant that allows the users to share a large IT infrastructure in a secure manner.

  This greatly helps to achieve economies of scale, which would not be possible if a company created its own cloud.
- **Private Cloud**: This is when a company owns the data center. True, there are not the benefits of the economies of scale from a public cloud. But this may not be a key consideration. Some companies might want a private cloud because of control and security.
- **Hybrid Cloud**: This is a blend of the public and private clouds. For example, the public cloud may handle less mission-critical functions.

As for RPA, the **cloud** has different implications and impacts. One is that a platform needs to deal with complex distributed applications, which can be difficult if a company develops custom programs on a cloud service.

.In some cases, an **on-premise** RPA system may be loaded onto a cloud service like AWS. While there are benefits with this, it is not cloud native. This is because you will still need to upgrade and maintain the software.

Web Technology

The mastermind of the development of the World Wide Web – which involved the use of hyperlinks to navigate web pages – was a British scientist, Tim Berners-Lee.

At the core of this was HTML or hypertext markup language, which was a set of commands and tags to display text, show colors, and present graphics. A key was that the system was fairly easy to learn and use, which helped to accelerate the number of web sites.

For example, many of the commands in HTML involve surrounding content with tags, such as the following:

This is a Title

HTML would ultimately be too simple. So there emerged other systems to provide even richer experiences, such as with CSS (Cascading Style Sheets, which provides for borders, shadows, and animations) and JavaScript (this makes it possible to have sophisticated interactivity, say, with the use of forms or calculations).

RPA must deal with such systems to work effectively. This means it will have to take actions like identify the commands and tags so as to automate tasks.

Programming Languages and Low Code

- A programming language allows you to instruct a computer to take actions.
- The commands generally use ordinary words like IF, Do, While, and Then. But there can still be lots of complexity, especially with languages that use advanced concepts like object-oriented programming.
- Some of the most popular languages today include Python, Java, C++, C#, and Ruby.
- To use an RPA system, you have to use some code – but it's not particularly difficult. It's actually known **as low code**. As the name implies, it is about using minimal manual input.

  For example, an RPA system has tools like drag-and-drop and visualizations to create a bot.

This is not to imply that low code does not need some training. To do low code correctly, you will need to understand certain types of workflows and approaches.

OCR (Optical Character Recognition)

- A key feature for an RPA platform is OCR (Optical Character Recognition), a technology that has actually been around for decades.
- It has two parts:

  ->Document scanner (which could even be something like your smartphone)

  -> software that recognizes text.

  In other words, with OCR, you can scan an image, PDF, or even handwritten documents – and the text will be recognized. This makes it possible to manipulate the text, such as by transferring it onto a form or updating a database.

There are definitely many challenges with effective OCR scanning, such as:

- The size of a font
- The shape of the text
- The skewness (is the text rotated or slanted?)
- Blurred or degraded text
- Background noise
- Understanding different languages

Then how does this technology help with RPA?

- One way is with recoding a person's actions while working on an application. The OCR can better capture the workflows by recognizing words and other visuals on the screen. So, even if there is a change of the location of these items, the RPA system can still identify them.
- Something else: Automation involves large numbers of documents.
  Thus, OCR will greatly improve the processing. An example of this would be processing a loan. With OCR, a document will use OCR to extract information about a person's financial background, the information about the property, and any other financial details. After this, the RPA system will apply the workflows and tasks to process the loan, say, with applying various rules and sending documents to different departments and regulatory agencies.

But there are OCR systems that can help out, such as HyperScience. The software leverages sophisticated machine learning (ML) technology to quickly and accurately extract the information (understanding cursive writing, for example).

Databases

- At the heart of most applications is a database, which stores data that can be searched and updated. This is usually done by putting the information in tables (i.e., rows and columns of information).
- To interact with this, there is a scripting language called SQL (Structured Query Language), which was relatively easy to learn.
- It was not until the late 1970s that relational databases were commercialized, led by the pioneering efforts of Oracle.
- While relational databases proved to be quite effective, there were still some nagging issues. Perhaps the biggest was data sprawl. Another problem was that relational databases were not cheap. And as

new technologies came on the scene, such as cloud computing and real-time mobile applications, it became more difficult to process the data.

- In the meantime, there have been new approaches that have gone against the model for relational databases. They include offerings like MySQL (which is now owned by Oracle) and PostgreSQL. Yet these systems did not get enough traction in the enterprise.
- But there is one next-generation database technology that has done so: NoSQL. It also began as an open source project and saw tremendous growth. As of now, MongoDB has 14,200 customers across 100 countries and there have been over 70 million downloads.
- Where relational databases are highly structured, a NoSQL system is quite flexible. It's based on a document model that can handle huge amounts of data at petabyte scale.
- And going forward, there is likely to be much innovation with database technology. Yet relational databases will remain the majority of what companies use – which also means that this will also be what RPA interacts with as well.

APIs (Application Programming Interfaces)

- An API – which is the acronym for "application programming interface" – is software that connects two applications.

  For example: let's say you want to create a weather app. To get access to the data, you can setup an API, which often is fairly straightforward, such as by putting together a few lines of code to make data requests (say, for the city). By doing this, you will increase the speed of the development.
- APIs are pervasive in enterprise environments since they are so effective. They also have different structures. Although, the most common is a REST (representational state transfer) API.
- It's true that APIs can be used as a form of automation.
- The technology requires having people with technical backgrounds. The development of an API can take time and require complex integration.
- There is also the need for ongoing testing. However, there are third-party services that can help out. There must be a focus on maintaining an API (it's not uncommon for an API to break if there is a change in the structure).
- APIs can still have bugs and glitches, especially when in complex IT environments.

Because of the difficulties, RPA has proven to be a very attractive alternative. Again, the development is much easier and there is less of a need for integration. But, interestingly enough, an RPA platform can be a vehicle for delivering advanced APIs within the enterprise.

AI (Artificial Intelligence)

- A typical RPA system does not have much AI (Artificial Intelligence). The main reason is that there is a literal carrying out of tasks, which does not require any smart system. But as AI gets more powerful and accessible, RPA will increasingly start to use this powerful technology – which should greatly enhance the outcomes.

AI: It's software that ingests large amounts of data that is processed with sophisticated algorithms that help answer questions, detect patterns, or learn. Interestingly enough, AI is actually made up of a variety of subcategories

Machine Learning : This is where a computer can learn and improve by processing data without having to be explicitly programmed. Machine learning is actually one of the oldest forms of AI and uses traditional statistical methods like k-nearest neighbor (k-NN) and the naive Bayes classifier.

Deep Learning: Deep learning became a major force in AI. Some of the important factors for this included the enormous growth in data, the use of GPUs (graphics processing units) that provided for ultrafast parallel processing, and innovation in techniques like backpropagation.

Deep learning is about using so-called neural networks – such as recurrent neural networks (RNNs), convolutional neural networks (CNNs), and generative adversarial networks (GANs) – to find patterns that humans often cannot detect.

NLP (natural language processing): This is AI that helps understand conversations. The most notable examples of this include Siri, Cortana, and Alexa.

But there are also many chatbots that focus on specific uses cases (say, with providing medical advice).

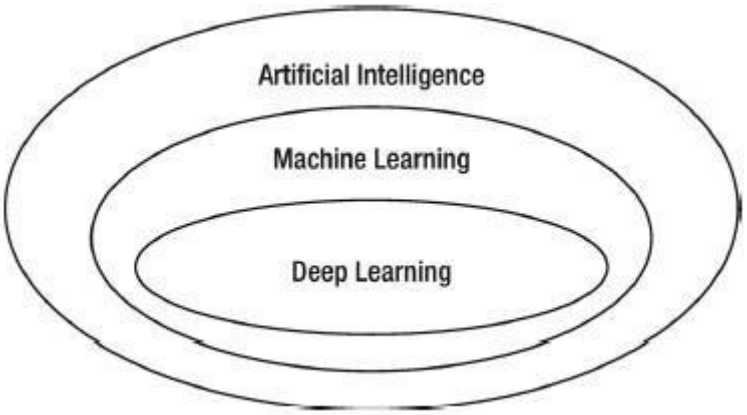


Figure 2-1 This is a high-level look at the key components of the AI world

Besides, AI has some major issues, such as the following:

Bias : According to IBM: "Bad data can contain implicit racial, gender, or ideological biases. Many AI systems will continue to be trained using bad data, making this an ongoing problem."

Causation: Humans have a strong grasp of this. We know what will happen if we use a hammer to hit a glass. It's pretty much instinctive. But AI is another matter. This technology is really about finding correlations in data not causation – and this is a major limiting factor.

Common Sense: A human does not have to process many cases to understand certain rules of thumb. We just naturally understand them. But with AI, common sense has been extremely difficult to code because of the ambiguity and the lack of useful data for the seemingly infinite use cases.

Black Box : Deep learning can have an enormous number of layers and parameters. This means it can be nearly impossible for a person to understand why the model is generating certain results. Now there is more innovation in trying to find ways to understand deep learning outcomes – which is something called "explainability" – but the efforts are still in the nascent stages.

Comprehension : An AI system cannot truly understand what it is reading or observing.

For example, if it read *War and Peace*, it would not be able to provide thoughts on the character development, themes, and so on.

Static: It is also possible to conduct millions of simulations to learn. But of course, the real world is much more dynamic, open-ended, and chaotic.

Conceptual Thinking: AI cannot understand abstract ideas like justice, misery, or happiness. There is also a lack of imagination and creativity.

Brain: It's really a miracle of evolution. A typical brain has 86 billion neurons and

.

Structured Data: This is data that is formatted (social security numbers, addresses, point of sale information, etc.) that can be stored in a relational database or spreadsheet.

Unstructured Data: This is data that is unformatted (images, videos, voicemails, PDFs, emails, and audio files).

- For the most part, RPA uses structured data. However, this represents about 30% of what's available in a typical organization. But with AI, an RPA system will likely be much more effective since it will be better able to process unstructured data.
- Furthermore, there are other potential benefits of the technology: judgement, the use of reasoning, and the detection of highly complex patterns. With these, the automation will be greatly enhanced, helping with things like detecting fraud.

Cognitive Automation

.Consider cognitive automation to be an application of AI, actually.

- First of all, it is automation uses a combination of technologies like speech recognition and NLP. By doing this, the goal is to replicate human actions as best as possible, such as by analysing patterns of workers and then finding patterns and correlations.
- Something else: Unlike other forms of AI, cognitive automation is usually effective with the use of much less data. There may also be not as much reliance on highly technical talent, such as data scientists.

Agile, Scrum, Kanban, and Waterfall

- In today's world, software development has become even more difficult because of the emergence of new platforms like the cloud and the hybrid cloud. This is why it's important to look at software management approaches.

- **Agile**

-One is called Agile, which was created back in the 1990s (a big part of this was the publication of the Manifesto for Agile Software Development).

-The focus of this was to allow for incremental and iterative development, which begins with a detailed plan. This also requires much communication across the teams and should involve people from the-business side of the organization.

-Nowadays, Agile has gotten easier because of the emergence of sophisticated technologies like Slack and Zoom that help with collaboration. "

- **Scrum:**

-This is actually a subset of Agile. But the iterations are done as quick sprints, which may last a week or two. This can help with the momentum of a project but also make a larger project more manageable (just as a side note: Scrum was first used for manufacturing but it was later found to work quite well with software development).

Kanban:

-This comes from the Japanese word for visual sign or card (the roots of the system go back to Toyota's high-quality manufacturing processes).

So yes, with Kanban, there is the use of visuals to help streamline the process. What's more, the general approach is similar to Agile as there is iterative development.

Waterfall

-This is the traditional code development model, which goes back to the 1970s.

-The waterfall model is a structured plan that goes over each step in much detail. To help this along, there may be the use of a project management tool, say, a Gantt chart.

-While the waterfall approach has its advantages, it has generally fallen out of favor. Some of the reasons are as follows: It can be tough to make changes, the process can be tedious, and there is often a risk of a project being late.

DevOps

- DevOps has emerged as a critical part of a company's digital transformation.
- The "Dev" part of the word is actually more than just about coding software.
- It also refers to the complete application process (such as with project management and quality assurance or QA). As for "Ops," it is another broad term, which encompasses system engineers and administrators as well as database administrators, network engineers, security experts, and operations staff.
- For the most part, DevOps has come about because of some major trends in IT. One is the use of agile development approaches . Next is the realization that organizations need to combine technical and operational staff when introducing new technologies and innovations.
- And finally, DevOps has proven effective in working with cloud computing environments.

Flowcharts

- An essential part of RPA is understanding workflows and systems, the use of flowcharts is common.
- It's usually at the core of the software application.
- With a flowchart, you can both sketch out the existing workflows of a department. And then from here, you can brainstorm ways of improving them. Then you can use the flowchart to design a bot for the automation.
- The flowchart is relatively simple to use and it also provides a quick visual way to understand what you are dealing with. As the old saying goes, a picture is worth a thousand words.
- some of the basics:

Terminator: This is a rectangle with rounded corners and is used to start and end the process, as seen in Figure 2-2

Figure 2-2 This is a terminator, which starts and ends a flowchart

Process: This is represented by a rectangle. With this, there is only one next step in the process. Figure 2-3 shows an example:

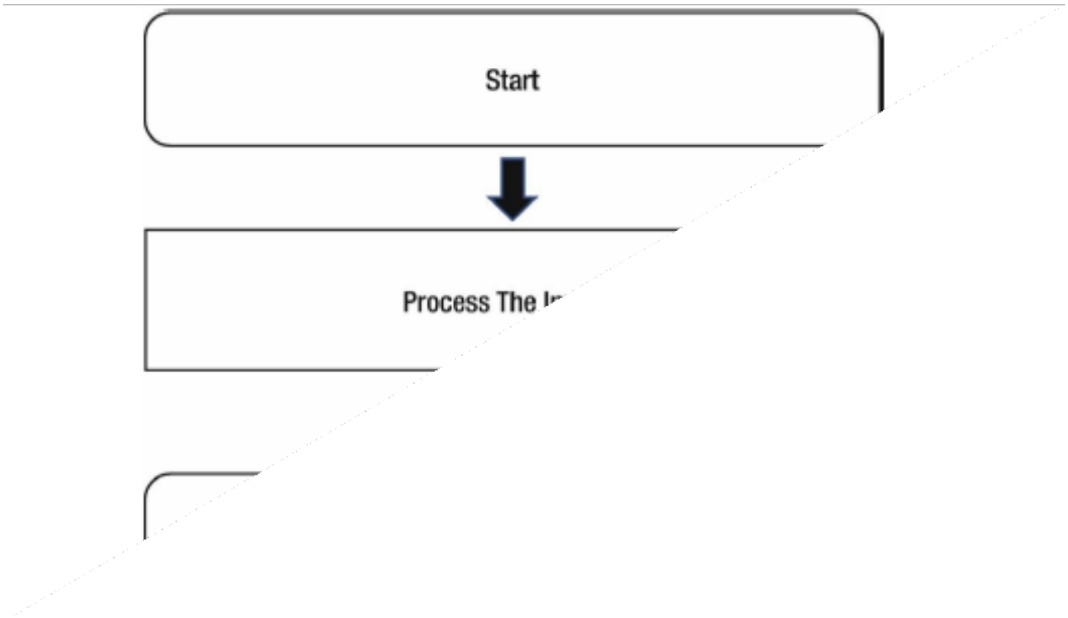


Figure 2-3 This shows a process in a flowchart

Decision: This is a square symbol that is at an angle. There will be at least two possible paths. Figure 2-4 is an example:

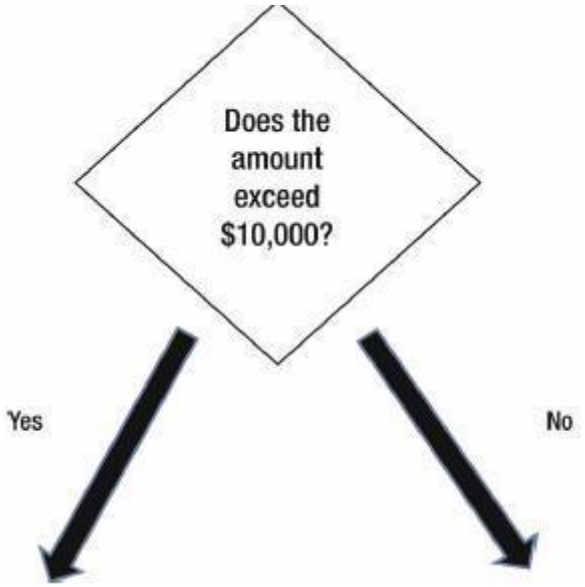


Figure 2-4 This shows a decision process in a flowchart

Module-2

RPA Platforms

2.1 COMPONENTS OF RPA

Any Robotics process automation platform provides some basic components, which together build the platform.

The following are the basic or core components of RPA:

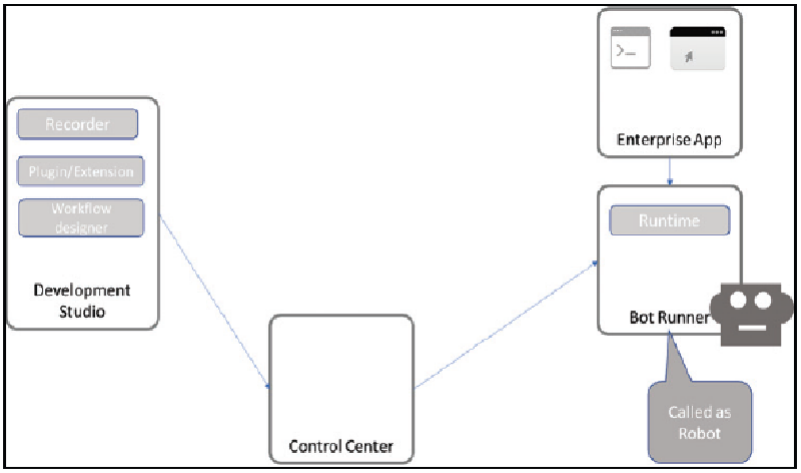
Recorder

Development Studio

Plugin/Extension

Bot Runner

Control Center:



Recorder

- The recorder is the part of the development studio that developers use to configure the Robots.

- It is like the macro recorder in Excel, the bot recorder in any platform, records steps.
- It records mouse and keyboard movements on the UI and this recording can be replayed to do the same steps again and again.
- This enables rapid automation. This component has played a very big role in the popularity of RPA.

Development studio

- The development studio is used by developers to create Robot configuration or train the Robots.
- Using the development studio, a set of instructions and decision-making logic is coded for Robots to execute.
- Some platforms provide flow-charting capabilities such as Visio, so it becomes very easy to plot steps in a process, whereas some other platforms require coding.
- In most studios, in order to do commercial development, developers need to have a fair amount of knowledge of programming,
  Example, loops, if else, variable assignment, and so on.

Extensions and plugins

- Most platforms offer many plugins and extensions to ease the development and running of bots.
- In many applications, such as Java SAP, it is not easy to individually identify controls of the UI through traditional techniques.
- RPA vendors have developed plugins and extensions to help with these issues.

Bot runner

- This is also referred to as the Robot, other components make it run.

Control center

- The objective of the control room is to provide Robot management capabilities.
- It monitors and controls a Robot's operation in a network.

- It can be used to start/stop Robots, make schedules for them, maintain and publish code, redeploy Robots to different tasks, and manage licenses and credentials.

2.2 RPA Platforms

- RPA vendor market has been showing continual and steady growth.
- While the largest market is the US, followed by the UK, the market in **Asia Pacific Countries**(**APAC**) is also showing considerable progress. Successful pilot projects and increased customer satisfaction among the early adopters of RPA will encourage new players to adopt this technology.
- There is growing demand for RPA, especially in industries that need large scale deployments.
- The major markets for RPA are banking and finance, healthcare and pharmaceuticals, telecom and media, and retail.

1. Automation Anywhere

Automation Anywhere helps to automate business processes for companies. They focus on RPA, cognitive data (machine learning and natural language processing), and business analytics. Their bots are capable of handling both structured as well as unstructured data.

The system has three basic components:

1. *A development client* for the creation of a bot
2. *A runtime environment* for the deployment of a bot
3. *A centralized command system* for handling multiple bots, analyzing their performance:

2. UiPath

- UiPath is an RPA technology vendor who designs and delivers software that helps automate businesses.
- The RPA platform consists of three parts:

->*UiPath Studio* to design the processes

->*UiPath Robot* to automate tasks designed in UiPath Studio

->*UiPath Orchestrator* to run and manage the processes

3. Blue Prism

- Blue Prism aims to provide automation that enterprises can use according to their needs.

- Blue Prism aims to do this by providing automation that is scalable, configurable, and centrally managed.
- It sells its software through its partners, some of which are Accenture, Capgemini, Deloitte, Digital Workforce Nordic, HPE, HCL, IBM, TCS, Tech Mahindra, Thoughtonomy, and Wipro

4. WorkFusion

- Work Fusion offers automation that is based on RPA and machine learning.
- It delivers software as a solution for automating high volume data.
- Work Fusion enables man and machine to work in tandem while managing, optimizing, or automating tasks.

5. Thoughtonomy

- Thoughtonomy delivers software that helps automate business and IT processes.
- It uses Blue Prism and other automation software and customizes it

6. KOFAX

- Kofax's RPA platform is capable of automating and delivering processes that are repetitive and rule-based.
- It uses Robots for extracting and consolidating information.
- The software platform consists of a management console to deploy and manage bots, Robot performance, and a monitoring system.
- This software can also group together high priority tasks that should be completed first by the Robot during times of high workload. Kofax's software, however, doesn't have machine learning.

	UiPath	Blue Prism	WorkFusion	Thoughtonomy	KOFAX
HQ	Bucharest, Romania	United Kingdom	New York, USA	London, UK	Irvine, California
Est		2001	2011	2013	
CEO	Daniel Dines	Alastair Bathgate	Max Yangkelivich, Andrew Volkov	Terry Walby	Paul Rooke
Key Clients	Atos, AXA, BBC, Capgemini, CenturyLink, Cognizant, Middlesea, OpusCapita, and SAP	BNY Mellon, RWE npower, and Telefonica O2	Thomson Reuters, Infogroup, Citi, and Standard Bank	Atos, Fujitsu, CGI, Unite BT, and Business Systems	Arrow Electronics, Delta Dental of Colorado, Pitt Ohio, Audi
Source of revenue by region	North America, Continental Europe, the UK, and APAC	More than half of its revenue source comes from the UK, followed by North America, Continental Europe, and APAC	North America provides more than 80% of Work Fusion's revenue,followed by Europe, APAC, and MEA	Around 70% of revenue comes solely from the UK. The rest comes from Continental Europe, North America, APAC,	North America accounts for almost half of its revenue, followed by Continental Europe, APAC, and LATAM (Latin America)
Source of revenue by industry	BFSI, healthcare, telecom and media, and retail	BFSI, health, and pharmaceuticals, retail and consumer, telecom and media, manufacturing, public sector, travel, and transportation	Around 90% of its revenue comes from the BFSI sector, followed by the retail and consumer sectors	A major part of its revenue comes from third party clients, followed by BFSI, public sector, telecom, healthcare, retail, and consumer sectors	BFSI,retail, consumer, travel, transportation, public sector, manufacturing, and healthcare

2.3 ABOUT Ui PATH

- UiPath is an RPA vendor that provides software to help organizations automate their business processes. The company aims to do away with repetitive and tedious tasks, allowing humans to engage in more creative and inspiring activities.
- UiPath was founded by Daniel Dines, who is the CEO. It has offices in London, Bucharest, Tokyo, Paris, Singapore, Melbourne, Hong Kong, and Bangalore. With clients spread across the world, from North America to the United Kingdom, Continental Europe to Asia Pacific countries, the company has shown remarkable growth in the last year, both in terms of revenue and its workforce.
- Today, its software is being widely used to automate business processes. However, the IT sector is also gradually embracing UiPath's software. Major clients of UiPath in the industry include BFSI, Telecom and media, healthcare, retail and consumer, and manufacturing.
- With UiPath automation software, one can configure software Robots to mimic human action on the user interface of computer systems.

The basic components of the UiPath RPA

. The components of the UiPath platform are UiPath Studio, UiPath Robot, and UiPath Orchestrator, see the following sections.

1. UiPath Studio

- UiPath Studio helps users with no coding skills to design Robotic processes in a visual interface.
- It is a flowchart-based modeling tool. Thus, automation is faster and more convenient.
- Multiple people can contribute to the same workflow.
- The presence of a visual signal that points out errors in the model, and a recorder that performs what the user executes, make modeling much easier.

2. UiPath Robot

- UiPath Robot runs the processes designed in UiPath Studio.

- It works in both attended (working only on human trigger) and unattended environments (self-trigger and work on their own).

The following are types of Robots:

Attended: It operates on the same workstation as a human to help the user accomplish daily tasks. It is usually triggered by user events. *You cannot start a process from Orchestrator on these type of Robots, and they cannot run under a locked screen*.

Unattended: It can run unattended in virtual environments and can automate any number of processes. In addition to the Attended Robot's capabilities, this Robot is responsible for remote execution, monitoring, scheduling, and providing support for work queues.

Free: It is similar to Unattended Robots, but can be used *only* for development and testing purposes, not in a production environment.

3. UiPath Orchestrator

- UiPath Orchestrator is a web-based platform that runs and manages Robots.
- It is capable of deploying multiple Robots, and monitoring and inspecting their activities.
- Orchestrator's main capabilities:
  -It helps in creating and maintaining the connection between Robots
  -It ensures the correct delivery of the packages to Robots
  -It helps in managing the queues
  -It helps in keeping track of the Robot identification
- It stores and indexes the logs to SQL or Elasticsearch
- Behind the scenes, Orchestrator Server uses:

-IIS Server

-SQL Server

-Elasticsearch

-Kibana

2.4 THE FUTURE OF AUTOMATION

- The buzzword today is the **Fourth Industrial Revolution**-the current age where technology is embedded within societies and even the human body-be it Robotics, 3D printing, nanotechnology, Internet of Things, or autonomous vehicles. This will fundamentally change the way we live, work, and interact with one another.

- Technological innovation has reached a stage where machines have now entered the realm of what was once considered exclusively human. For these reasons, there is a wide section of people who fear this age of Robots.

- There are various advantages of automation today; there are also fears surrounding its advancement, which are not completely unfounded. This time automation is capable of impacting a wide range of disciplines. Thus, unlike in the past where only blue collar jobs were at risk of being replaced by machines, this time even white collar jobs are believed to be at risk.
- While this is not untrue, reports suggest that only around 5% of the total jobs may be *totally* replaced by automation. For other jobs, automation will only replace a part of the job and not completely take over.
- There are, of course, those jobs in the 5% category that run the risk of being completely automated. These are the jobs that are routine, repetitive, and predictable. A few Examples: telemarketing, data entry operation, clerical work, retail sales, cashiers, toll booth operators, and fast food jobs.
- However, like in the past, people should be able to find a way to adapt to the changes. With each generation, humans become smarter, more adaptable to change, and also progressive.
- Also, with automation mostly taking over routine and tedious tasks, humans are provided the opportunity to make better use of their capabilities-be it reasoning, emotional intelligence, or their creativity.
- What we can do is not fret over the inevitable rather prepare for it. One way of doing so is to start changing the pattern of education. The next generation should be taught how to recognize and adapt to changes quickly. An important aspect of their education should be to *learn how to learn*.

2.5 RECORD AND PLAY

The facility of recording user steps on a computer and playing them back has made **Robotic Process Automation** (**RPA**) highly successful.

UiPath stack

There are three basic components in UiPath:

1. UiPath Studio
2. UiPath Robot
3. UiPath Orchestrator

The UiPath platform is available in two variations:

1. **Enterprise Edition**: This edition is suitable for large companies starting their RPA projects and looking to scale their Robot deployments in the future. It is integrated with UiPath Orchestrator (we will discuss UiPath Orchestrator later). This version can be updated by visiting the UiPath website and by downloading the newest version of the UiPath platform installer. Running the installer automatically replaces all the old files without modifying any of your settings.
2. **Community Edition**: This is suitable for individual developers and small organizations with fewer employees. The Community Edition is always up-to date, and it automatically updates itself as soon as a new version is available

- The Community Edition can be used to learn UiPath free of cost.

UiPath Studio

- UiPath Studio is the development environment of UiPath. It is the primary tool to develop UiPath Robots.
- It can be used to configure steps of a task or launch a full recorder to record a sequence of steps. The recording facility in the Studio is a game-changing feature for RPA tools.
- Its simplicity lets even nontechnical business users design/record steps of a process. This studio lets the user configure Robots, that is, develop steps to perform tasks visually.

- Most of the configuration and coding in UiPath is visual. By using the drag-drop facility from the toolbox, you may write a whole sequence of workflows to perform a set of tasks by Robots.
- These steps look like a data flow diagram and are very easy to understand. It is one of the simplest visual flow diagramming. The studio gives the same look and feel as a workflow. The designer.
- An activity or action includes clicking a button, writing and reading a file, and so on.

2.6 DOWNLOADING AND INSTALLING UIPATH STUDIO

UiPath Community Edition is free to use in academia, nonprofits, and small businesses

The UiPath Community Edition has the following features:

Auto update

No server integration

Community forum for support

Online self-learning

No complex installation required

Online activation is mandatory

To get your Community Edition of UiPath Studio, type the following link in your Browser: https://www.uipath.com/Community.

1. A **Community Edition** page opens. Click on **Get Community Edition**

2. On the next page, you must register yourself in order to download the Community Edition. So, use the correct details and remember them because the same email will be used to activate the software.

   ---Fill in the following details: **First Name***, **Last Name***, and **Email***. Filling in the **Twitter User** field is not mandatory, but it is good to provide it:

   ---Click on **REQUEST COMMUNITY EDITION**.

3. You will be directed to a page that requests you to check your email for downloading the link. Click on the link to download UiPath Studio.

   You may also directly download UiPath Studio. Just click on the word **here** in **download it**.

4. Once the download is complete, open the downloaded file, UiPathtsudioSetup.Exe
5. The installation will then begin. Once the installation is complete, a welcome message will be displayed. Click on the **Start Free** option.
6. Then, as requested, enter your **Email Address** once again and click on **Activate**. Please remember to use the same email ID that you used to download the software. This email ID will be bound to the computer. The activation will happen online. An offline activation option is not available for the Community Edition.
7. A message will then be displayed on the screen informing you of the successful installation. Close this window.

For more convenient use, you can pin it to your taskbar immediately; otherwise, you may have to unnecessarily search for UiPath.Exe in your computer every time you wish to use it.

Your UiPath Studio is now ready for use!

2.7 LEARNING Ui PATH STUDIO

- The **UiPath Studio** platform helps to design Robotic processes with a visual interface.
- Automation in UiPath Studio requires no or very little prior programming knowledge. It is a Flowchart-based modeling tool. Thus, automation is faster and more convenient.

Projects

The main types of project supported by UiPath Studio are as follows:

Sequence: This is suitable for simple actions or tasks. It enables you to go from one activity to another, without interfering with your project.

-It consists of various activities.

-Creating sequences is also useful for debugging purposes.

-One activity from a particular sequence can easily be tracked.

-The Basic type of project can be started using the **Blank** option in the start tab and then adding the sequence in the diagram from the toolbox.

Flowchart: This is suitable for dealing with more complex projects.

-It enables you to integrate decisions and connect activities.

To start this kind of project, choose the **Flowchart - Simple Process** option from the new project menu.

Assistant: This is suitable for developing attended or Front Office Robots: sometimes these Robots are called assistants. To start this kind of project, choose the **Assistant - Agent Process Improvement** option from the new project menu.

State machine: This is suitable for very large projects that use a finite number of states in their execution, triggered by a condition.

-To start this kind of project, choose the **Process - Transaction Business Process** option from the new project menu

- However, if you click on the **New** option in the DESIGN tab, you only get three options:

Sequence

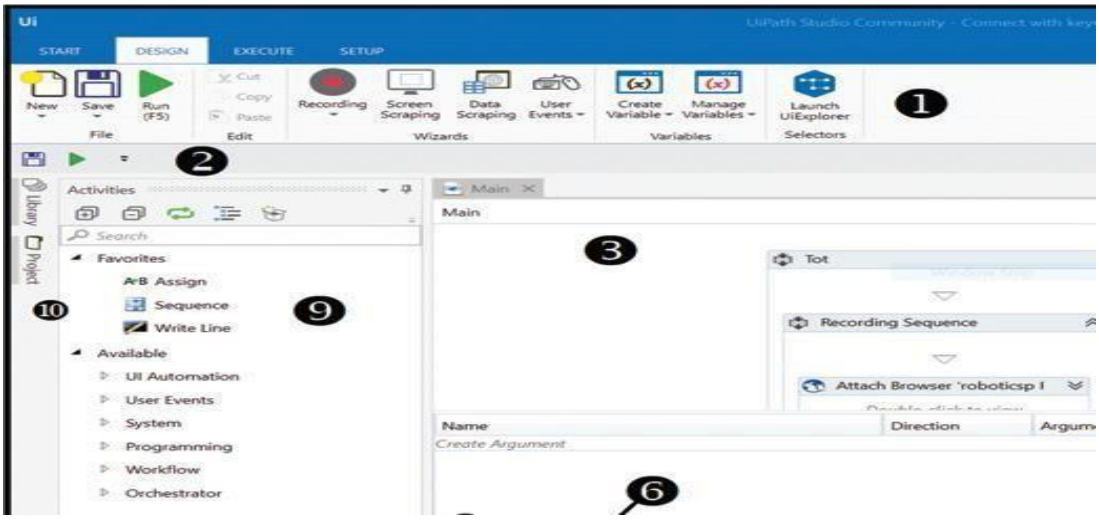
Flowchart

State Machine

The preceding options selected from the DESIGN tab's **New** menu become part of an existing project and are referred to as a diagram.

THE USER INTERFACE

- When you first open UiPath Studio, you are directed to the page.
- You can either open an old project or create a new one. Let us say we are making a new project.
- We click on **Blank** and name it. We will then be directed to a screen, which will display the following:



1. The Ribbon
2. Quick Access Toolbar
3. Designer panel
4. Properties panel
5. Outline panel
6. Arguments panel
7. Variable panel
8. Import panel
9. Activity panel
10. Library panel
11. Project panel
12. Output panel

1. The Ribbon

- This panel located at the top of the user interface and consists of four tabs:

1. **START**: This is used to start new projects or to open projects previously made.

2. **DESIGN**: This is to create new sequences, Flowcharts, or state machines, or to manage variables:

3. **EXECUTE**: This is used to run projects or to stop them, and also to debug Projects

4. **SETUP**: This panel is for deployment and configuration options; it has three tools available:

-Publish: This is used to publish a project or create a shortcut for it and schedule tasks

-Setup Extensions: This is used to install extensions for Chrome,
Firefox, Java, and Silver light

-Reset Settings: This is used to reset all settings to defaults:

2. The Quick Access Toolbar

- This panel gives the user a shortcut to the most used commands.
- One can also add new commands to this panel. This is located above the Ribbon on the user interface.
- The Quick Access Toolbar has been circled in the following screenshot and is indicated by the arrow:can be moved above or below the Ribbon. By default, there are two buttons available,**Save** and **Run**, which are also available in the **DESIGN** tab of the Ribbon.

3. Designer panel

- This is the panel where one defines the steps and activities of the projects.
- It is where a developer does most of the things to record activities or manually drop activities on the canvas.
- In UiPath, this is equivalent to the code windows of Microsoft Visual Studio. When we develop a Robot, this is the window where we will be organizing various activities in a flow or chain to accomplish a task.
- The project a user makes is clearly displayed on the Designer panel and the user has the option of making any changes to it.

4. Properties panel

- The panel located on the right-hand side of the user interface is for viewing the properties of the activities and for making any changes, if required.
- You need to select an activity first and then go to the **Properties** panel to view or change any of its properties

5. Activities panel

- Located on the left-hand side of the user interface, this panel contains all the activities that can be used in building the project.
- The activities can easily be used in making a project by simply dragging and dropping the required activity into the required location in the Designer panel.

6. Project panel

- With the **Project** panel, you can view the details of your current project and open it in a **Windows Explorer** window.
- It is located on the extreme left-hand side of the design panel,below the **Library** panel

7. Outline panel

- As the name suggests, this panel gives a basic outline of the project.
- The activities that make up the workflow are visible in this panel.
- Using this, you may see a high-level outline of the project and you can drill down to see deeper.
- This panel is especially helpful of large automation projects, where one may otherwise have a tough time going through it

8. Output panel

- This panel displays the output of the *log message* or *writes line* activities. It also displays the output during the debugging process.
- This panel also shows errors, warnings, information, and traces of the executed project. It is very helpful during debugging. The desired level of detail can be changed in **Execute** | **Options** | **Log activities**

9. **Library panel**
   - With this panel, you can reuse automation snippets. It is located on the extreme left-hand side of the Designer panel:
10. **Variable panel**
   - This allows the user to create variables and make changes to them. This is located below the Designer panel.
   - In UiPath Studio, variables are used to store multiple types of data ranging from words, numbers, arrays, dates, times, and timetables.
   - As the name suggests, the value of the variable can be changed.
   - An important point to note is that variables can only be created if there is an activity in the Designer panel.
   - To create new variables, you can go to the **DESIGN** tab on the Ribbon and click on **create variable**, and then choose the type of variable. Otherwise, one can simply go to the Variable panel located below the Designer panel and create a variable.

Argument

- While variables pass data from one activity to another in a project, arguments are used for passing data from one project to another.
- Like variables, they can be of various types-String, Integer, Boolean, Array, Generic, and so on.
- Since arguments are used to transfer data between different workflows, they also have an added property of *direction*. There are four types of direction:

In

Out

In/Out

Property

These depend on whether we are giving or receiving data to or from another workflow

2.8 TASK RECORDER

- The task recorder is the main reason for RPA's success.

- With the task recorder, we can create a basic framework for automation. The user's actions on the screen are recorded by the recorder and turned into a recording sequence in the current project.
- That's how Robots are able to mimic human actions.
- The recording is collection of execution steps that has to be taken, on the applications in the scope, in order to accomplish a task.
- These steps can be recorded one by one (manually) by pointing it on the screen or many steps in a go that is, automatically.

There are four types of recording in UiPath Studio:

Basic

Desktop

Web

Citrix

Basic recorder: Basic recorder is used to record activities on the desktop. This ype of recorder is used for single activities and simple workflows. The action here is self-contained and not contained in separate windows.

Desktop recorder: The desktop recorder, like the basic recorder, is used to record activities on the desktop. However, it is used to record and automate multiple actions and complex workflows. Each activity here is contained in an **Attach Window** component

-The **Attach Window** component is especially important to ensure that other windows of the same application do not interfere in the workflow.

Web recorder: The web recorder, as the name suggests, is used to record actions on web applications and browsers.

Citrix recorder: Citrix is used to record virtual machines, VNC, and Citrix environments. This recording allows only keyboard, text, and image automation.

- Some actions are recordable while others are not:

Recordable actions: Left-click on buttons, check boxes, drop-down lists, and other GUI elements. Text typing is also recordable.

Actions that cannot be recorded: Keyboard shortcuts, mouse hover, right-click.
Modifier keys such as *Ctrl* and *Alt* cannot be recorded.

- There are two types of recording:
- **Automatic recording**: This is for recording multiple actions in one go. This is a very good feature for preparing a solid foundation for automating a task. It can be invoked with the **Record** icon available in basic, desktop, and web recorders.
  Example, hotkeys, rightclick,double-click, and a few more.

- **Manual recording**: This type of recording is used to record each step one at a time and hence offers more control over the recording.
- Also, it can record all actions that cannot be recorded using automatic recording such as keyboard shortcuts, mouse hover, right-click, modifier keys, such as *Ctrl* and *Alt*, finding text from apps, and many other activities.
  - Citrix recorder can only record a single action (manual recording).

Shortcut keys:
***F2* key**: This pauses the recording for 3 seconds. The countdown menu is also shown on the screen.
Right-click: Exits the recording.
***Esc* key**: Exits the recording. If one presses the *Esc* key again, then the recording will be saved.

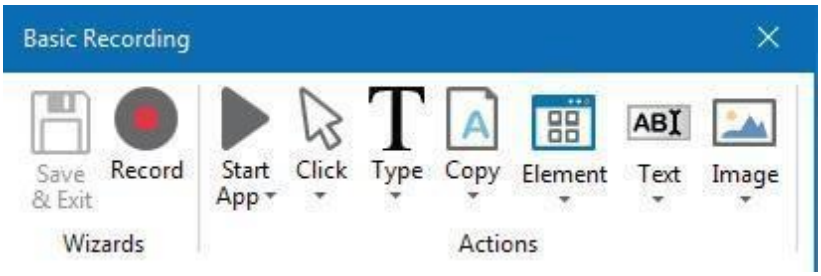
Recordings

The functions of these recordings. The operations that can be completed with the help of recording are as follows:

- Click (clicking a UI element: button, image, or icon)
- Type (typing any value into the available text field)
- Copy and paste

Basic Recorder

Recording. For example; **Start App**, **Click**, **Type**, **Copy**, and so on.



- **Start App**: This is used to start an application. When we left-click on this option, we are asked to point to an application that we want to open. When we are done, we can click on the **Save & Exit** option.
- **Click**: Another option is **Click**, which is used to click on a UI element. This feature is used as a mouse input. That is, it is used for clicking, checking, or selecting an item. When we click on this option, we are asked to indicate the location of the UI element we want to click. We can change the type of click to right-click or double-click in the **Click Type** property from the **Properties** panel.

- **Type**: Another option shown in the recording panel is Type. As the name suggests, it is used for typing something inside the indicated element. All you need to do is to indicate the area where you want to type. Then, you need to type your input in the popup that appears for typing
- After you are done typing, do not forget to press the *Enter* key. When the ***Enter* key** is pressed, the step is recorded.
- You can then click on **Save & Exit** to view the recording sequence.
- The recording sequence is shown in the following screenshot. You can change the text you have written (by changing the value of the **Type** in the block). You can write the desired text in double quotes (" "), or you can simply use a variable to store the data
- There are UI three more options in the recording panel:

Element

Text

Image

Advanced UI interactions

Advanced UI interactions are input and output interactions. In other words, it refers to the types of input methods and output techniques that are used while automating.

Input methods

The input that we give in the form of text can be of three types:

1. Default
2. Simulate
3. Window message

-Default is the generated method, while the other two are available in the **Properties** panel..

-The other two methods work in the background. Out of these three methods, the simulate type is the fastest method and is mostly preferred because in the window message input type, it types only the lowercase characters.

Output methods

These are the methods we use for getting our output, which can be in the form of text or images. The available methods are:

Native

Full text

OCR

Native is, by default, the generated method to extract data from the window. When you indicate to any element, the scraping window appears, and here all of the options can be found.

In OCR, there are two types of **OCR engine**: One is Google OCR and the other is Microsoft OCR. We can choose whichever displays better results. Also, we can adjust the scale mentioned in the properties of the OCR.

Step-by-step examples using the recorder

We will illustrate two examples of using the UiPath recorder:

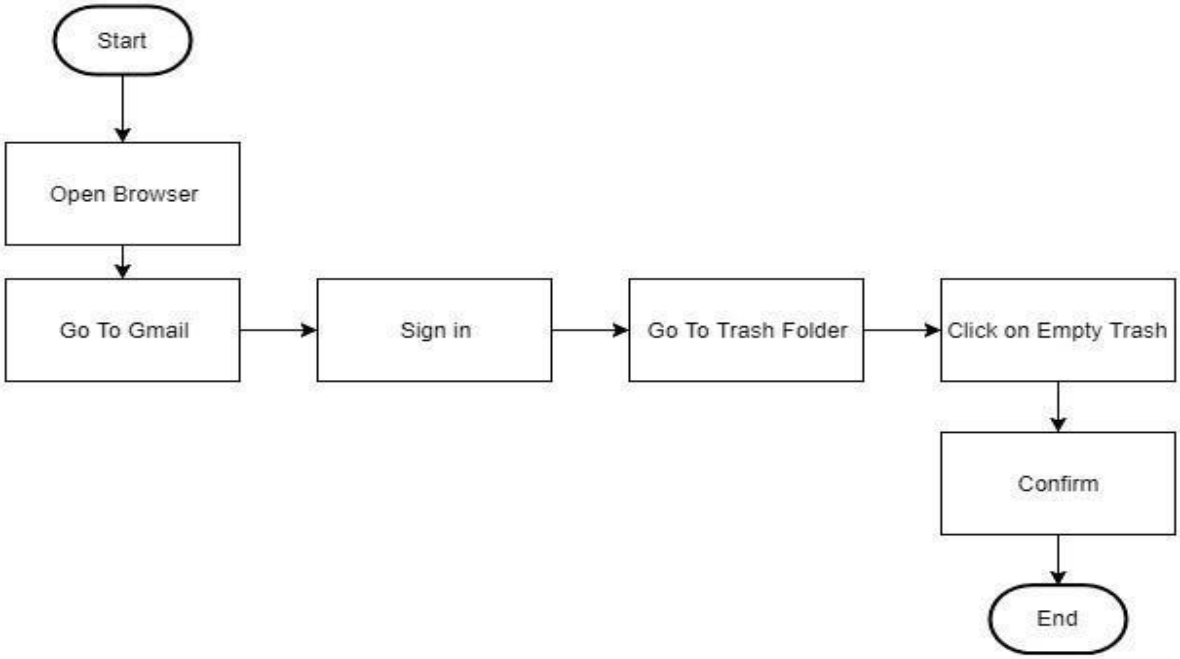
1. Emptying the trash folder in Gmail (**web-based application)**

2. Emptying Recycle Bin (**Windows based aplication**)

The first one is to show a recording of a web-based application, and the second is Windows based.

1. Emptying trash in Gmail

- This is an example of how we can empty a folder in Gmail with the help of a UiPath Robot, solely on the basis of recording.
- To do this, we are going to record all the actions that have to be performed to empty this Trash folder so that our Robot understands the sequence to be performed.
- The process flow of this simple activity in the following diagram:



Process flow for emptying Gmail trash

.

- First and foremost, we begin with a blank project in UiPath Studio and then choose **Web** recorder from the **Recording** drop-down list:

- To click on the **Recording** option and select the type of recording. We will use **Web** recording for this process since we are working on a website.
- Just click on the **Recording** icon at the top of the page. From the four types of recording that appear, choose **Web** recording. A **Web Recording** panel will appear, as shown in the above screenshot.
- Notice **Open Browser** between **Record** and **Click**; this is available with web recorder to record steps in browser-based applications.
- **Preparation**: Open your favourite browser, navigate to https://gmail.com, and keep this browser open.

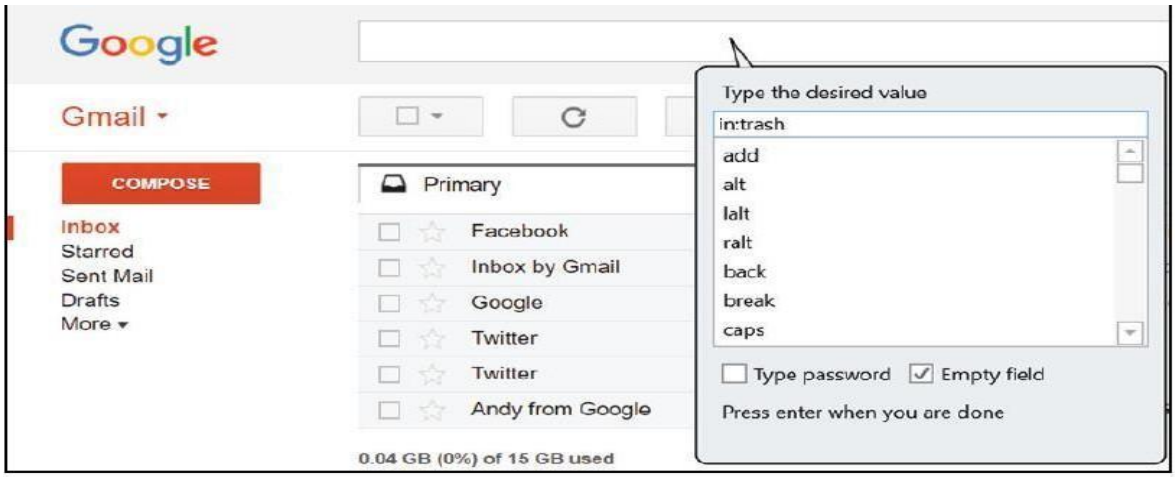
The following are the six steps in our process flow:

1. **Open Browser**: Although we have already opened Gmail in the browser, we did not record that step. Here, we will note that step in the recorder using the **Open Browser** button in the recorder. A drop-down menu will appear. Again, choose **Open Browser** from the drop-down menu. It will ask to highlight the browser, highlight the already opened browser and click on the top of the browser.
2. **Go to gmail.com**: You will be prompted to enter the **URL** of the website to navigate to. Type https://gmail.comor gmail.com and press **OK**:

3. **Sign In**: Start recording by clicking on the **Record** icon of the recording panel. Go to the already open Gmail and click on the Email or Phone field. UiPath will pop up a prompt for typing the email:

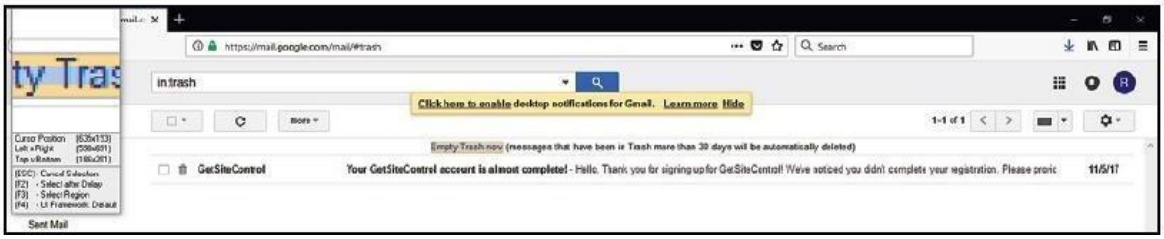
- Type Email in the box provided by the UiPath recorder and press *Enter*. The Gmail textbox will automatically fill up with your typed content. Click on the **NEXT** button of the Gmail interface; it will also get recorded.
- Now, you have recorded an entry in the password field. For simplicity, you may the password in the prompt provided by UiPath. Type your password in the text field of the popup that appears.
- Then, click **NEXT** to log in to your account. Clicking on the **NEXT** button will also get recorded.

4. **Locate Trash Folder**: In this step, we have to click on the search box of Gmail and type in:trash in the UiPath prompt and hit *Enter*:



- Now, click on the Search button beside the search box. It will also get recorded automatically and the Trash folder will appear.

5 **Click on Empty Trash now**: Once you are done with clicking on the Trash action, You can see a link showing **Empty Trash now**. Hover mouse on this link and it will get highlighted, click on it to delete all the messages in the Trash folder:

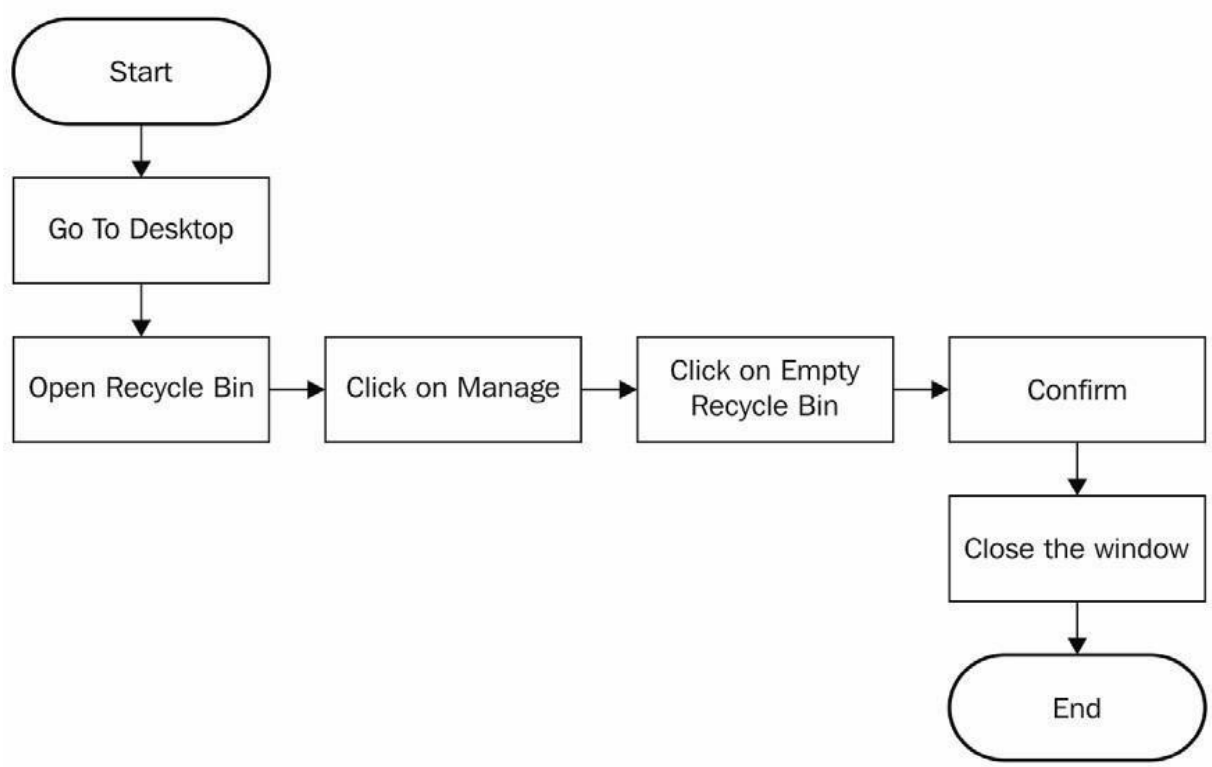


6 **Confirm**: When you click on **Empty Trash now**, a confirmation dialog will appear asking your permission for the action. Just confirm your action by clicking on the **OK** button.

- In the indicate anchor wizard, we have to indicate the adjacent button, that is, the **Cancel** button, so that the recorder will identify that the button is adjacent to **Cancel**.
- Now recording is complete, press *Esc* to get to the recording dialog. Click on the **Save & Exit** button.
- Then, in UiPath Studio, you can see a recording sequence in the Designer panel.
- Now run it by pressing the; *F5* key; it should perform the same task again. You have created your first Robot, which empties trash from your Gmail!

2. Emptying Recycle Bin

We are going to automate emptying the Recycle Bin. There are various steps that are involved. Let's map the process of how to empty the Recycle Bin:



This diagram is simpler and more detailed than in the Emptying trash in Gmail example; we need to do exactly the same steps in order to perform this task.

- Open UiPath Studio and choose a blank project.
- Since we are working in the recorder, and since we are working on the desktop and not a web application, we are required to choose the **desktop** recorder:

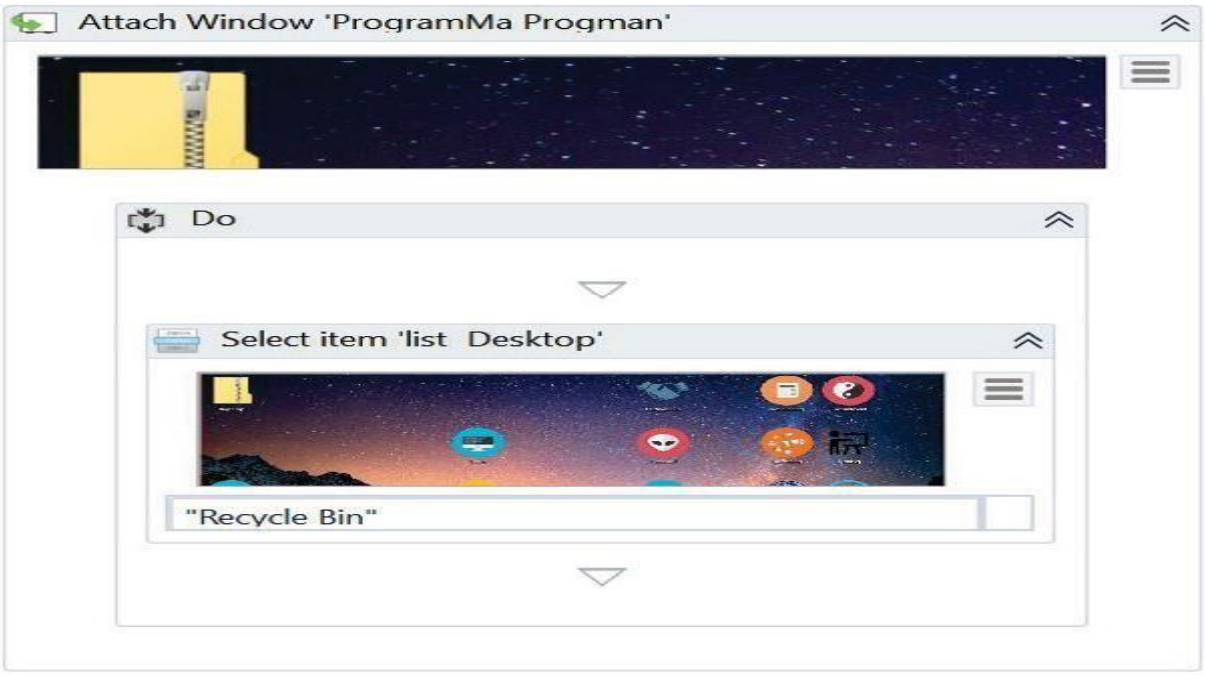
Start the recorder and simply perform the following steps:

1. Go to the desktop by pressing the *Windows + D* keys.
2. Open Recycle Bin by clicking on Recycle Bin and then pressing *Enter* key.
3. Click on the **Manage** tab of the Recycle Bin folder.
4. Click on the **Empty Recycle Bin** button.
5. Confirm by clicking on the **Yes** button in the dialog box.
6. Close the Recycle Bin folder by pressing the **cross** button.
7. Press the *Esc* key and **Save & Exit** the recorder.

Now your recording is ready to view, let's examine each step recorded:

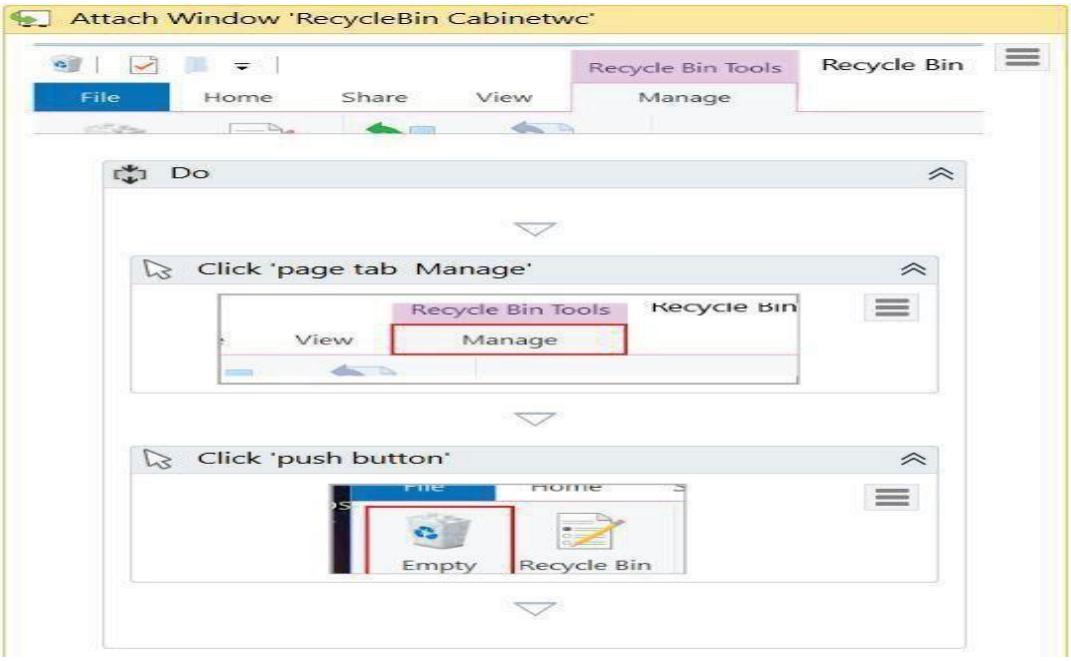
1. Go to the desktop by pressing *Windows + D* keys: This step is not recorded! Never mind, it is not needed. Please note that the recorded steps attach themselves to an application, and execute commands for that application, so the next step (Open Recycle Bin) will be executed on the desktop whether you are there or not.

2. Open Recycle Bin by clicking on Recycle Bin and then pressing the *Enter* key-We can see the recorded step in the following screenshot:

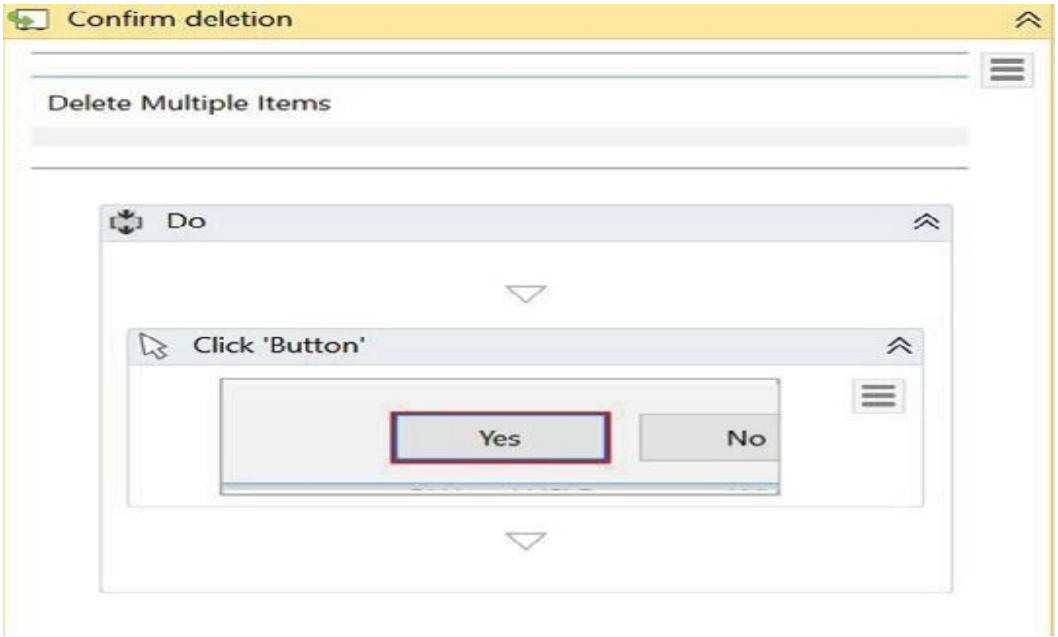


- selecting the Recycle Bin is recorded, not the *Enter* key. We should manually add that step. Search for Send hot key in the **Activities** window and insert it into the workflow just below the **Select item 'list Desktop'** step, as shown in the following screenshot:

- Click on the **Manage** tab of the Recycle Bin folder: This is recorded as it is and so is the fourth step, click on the **Empty Recycle Bin** button:



4. confirming by clicking on the **Yes** button on the dialog box is also recorded Smoothly:



In the last step, closing the Recycle Bin folder by pressing the **cross** button, you may have to indicate an anchor. Save it and press *F5* to run it. You see how easy it is to record steps taken on a computer and automate them.

Module-3

Sequence, Flowchart and Control Flow

3. Sequencing the workflow

Ui Path provides four types of projects:

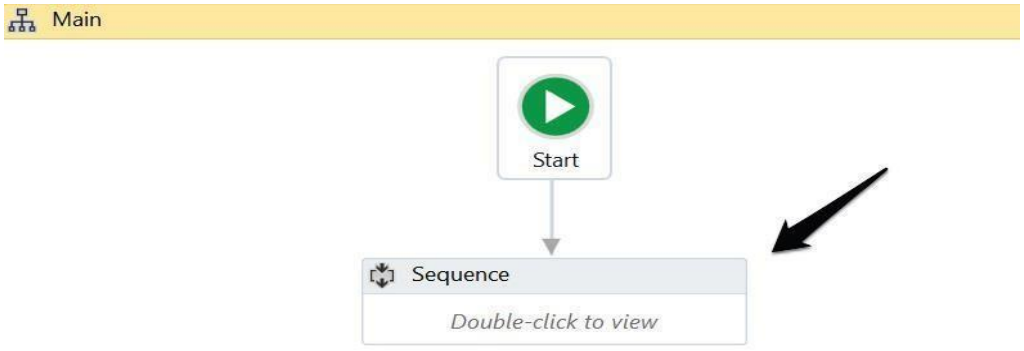
- Sequences
- Flowcharts
- User Events
- State Machines
  - A Flowchart and Sequence are mainly used for simple automation.
  - User Events are beneficial for implementing front office robots.
  - While State Machines are used for dealing with complex business processes.

3.1 What is a Sequence?

- A Sequence is a group of logical steps. Each step represents an action or a piece of work. A Sequence is used for processes that happen in linear succession, that is, one after the other.
- Among the three types of projects in UiPath, Sequences are the smallest.
- In the following example, we will make a simple project that asks for the name of the user and then displays his or her response:

1. Open Ui Path Studio and click on **Blank** to start a fresh project. Give it a meaningful name. On the Designer panel, drag and drop a **Flowchart** activity from the **Activities** panel.

2. Search for Sequence in the **Activities** panel and then drag and drop it into the **Flowchart**, as shown in the following screenshot:

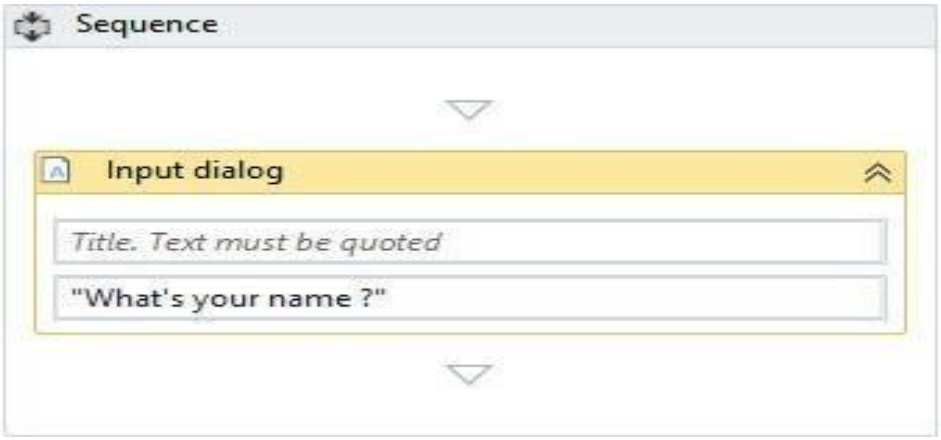


3. Double-click on the **Sequence**. We now have to add the steps that we want to perform. Consider each step as an action.

We can add many steps inside a **Sequence**. For the sake of simplicity, we will add two steps:

- Ask for the username in an **Input dialog**
- Display the username in a **Message box**

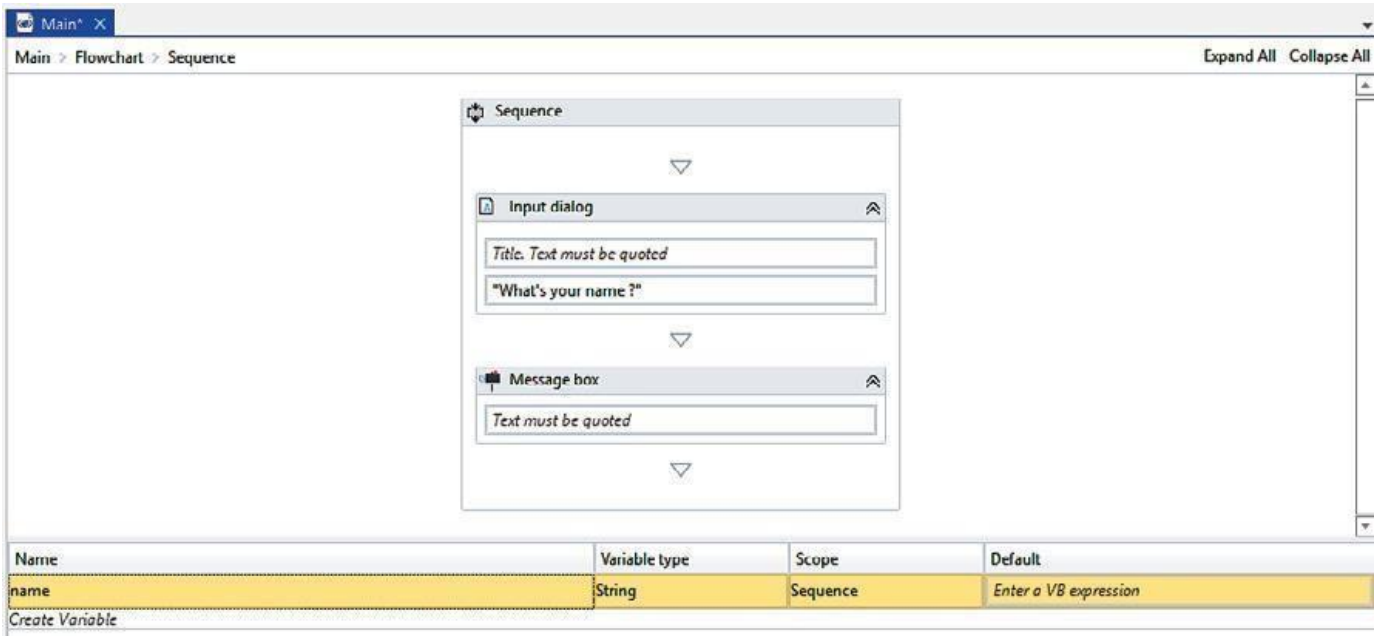
4. Search for Input dialog in the Search panel of the Activities panel. Drag and drop the Input dialog activity inside the Sequence (the Input dialog activity is a dialog box that appears with a message or a question; in response to which the user is required to put in his or her reply):



Write the appropriate message on the **Label** of this **Input dialog** to ask for the user's name. In our case, we have put in “What’s your name?”.

5. Drag and drop a **Message box** activity into the **Sequence**. (A **Message box**, as the name suggests, displays a given text. In this case, we will use it to display the text/reply that the user has given in the **Input dialog** box on being asked his/her name.)

6. Next, create a variable and give it the desired name. This variable will receive the text that the user has entered in the **Input dialog** box in response to our question, that is, the user's name:

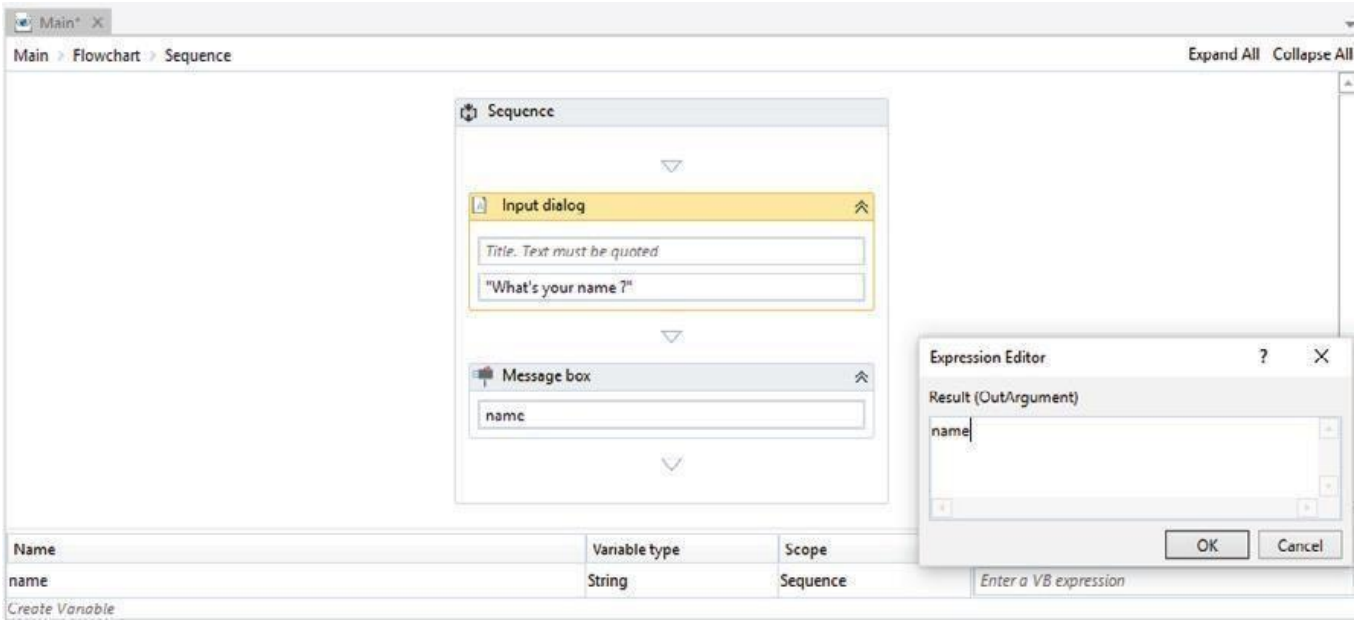


7. We now have to specify the **Result** property (in the **Properties** panel) of the **Input dialog** box.

On specifying the variable name there, it will receive the text that the user entered.

Click on the dotted icon that appears on the right side of the **Result** property.

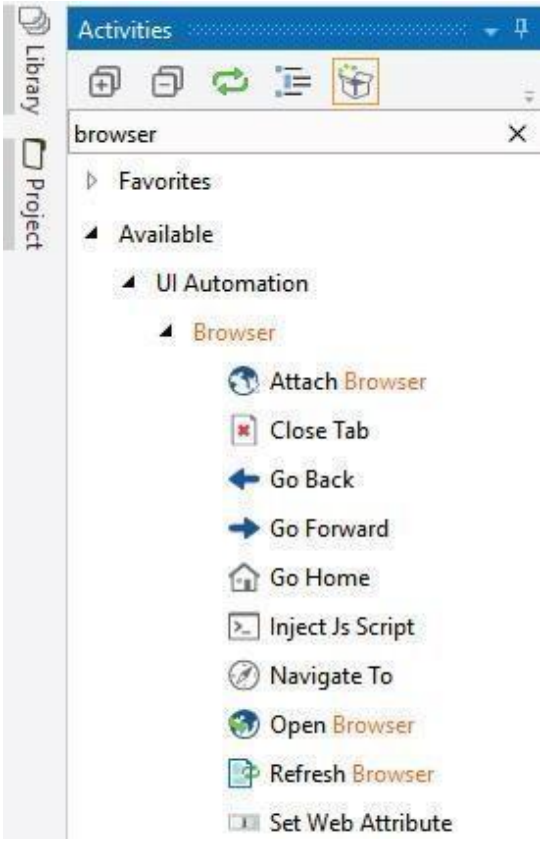
Now, specify the variable name:



8. Specify the variable name that we have created in the Text area of the **Message box** (the Text area of the **Message box** is used to input text that will be displayed in the **Message box**). We just need to connect the **Sequence** to the **Start** icon.
This can be done by right-clicking on the **Sequence** activity and choosing the **Set as Start node** option.
9. Hit the **Run** button and see the result.

3.2 Activities

- **Activity** represents the unit of an action. Each activity performs some action. When these activities combine together, it becomes a process.
- Every activity resides on the **Activities** panel of the main Designer panel. You can search for a particular activity and use it in your project.
- For example, when we search for **browser**, all the browser activities will appear in the **Activities** panel, as shown in the following screenshot:



3.2.1 **Using activities with workflows**

We have seen how we can easily search for a particular activity. Now, let us see how to use them in a workflow:
1. Search for **Flowchart** in the same way that we have searched for the browser activities in the **Activities** panel search bar. Drag and drop the **Flowchart** activity inside the Designer panel.

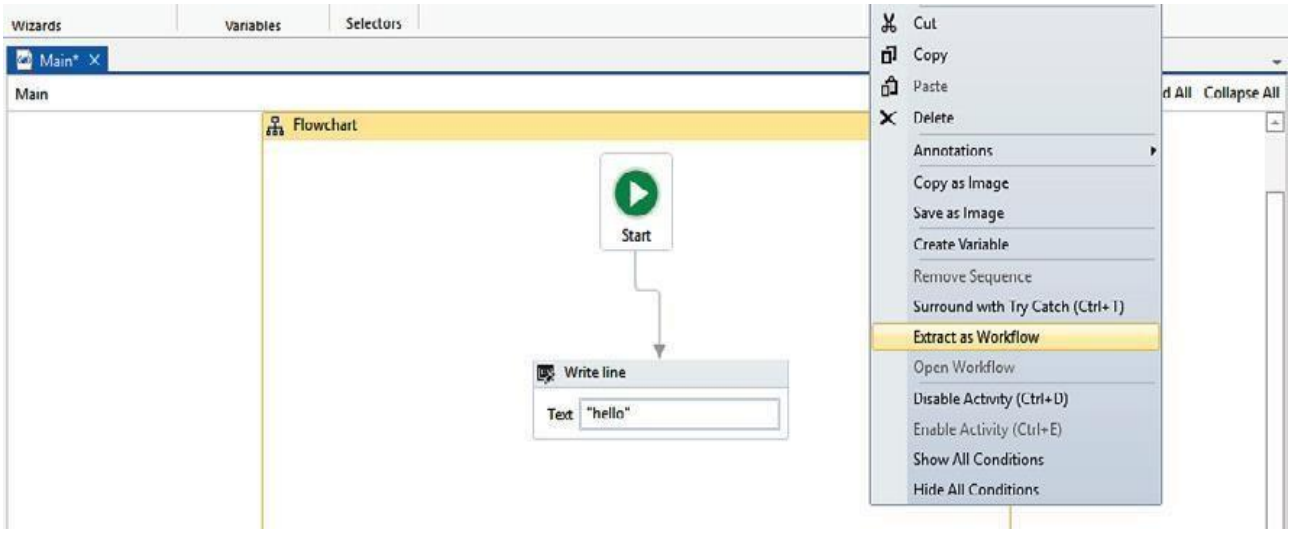
2. The **Flowchart** appears in the Designer panel and we have a given **Start** node. The **Start** node specifies where the execution begins.

3. We are ready to use different activities in our **Flowchart**. You can use any activity/activities inside the **Flowchart**. For the sake of simplicity, let us just use a **Write line** activity.

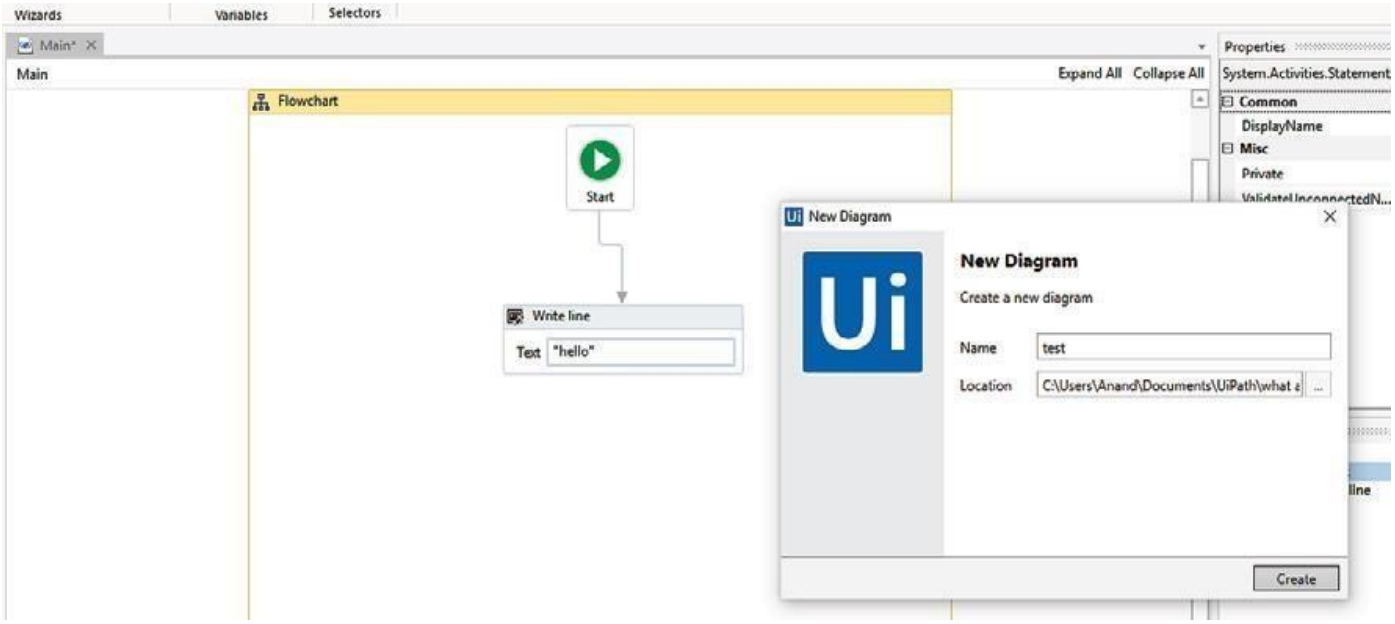
4. Drag and drop the **Write line** activity inside the **Flowchart**. Set its text property by providing a string value. Connect this **Write line** activity with the **Start** node by right-clicking on the **Write line** activity and selecting **Set as Start Node**.

Creating different workflows and combining them into a logical **Sequence** will enhance our code quality, maintainability, reliability, and readability.

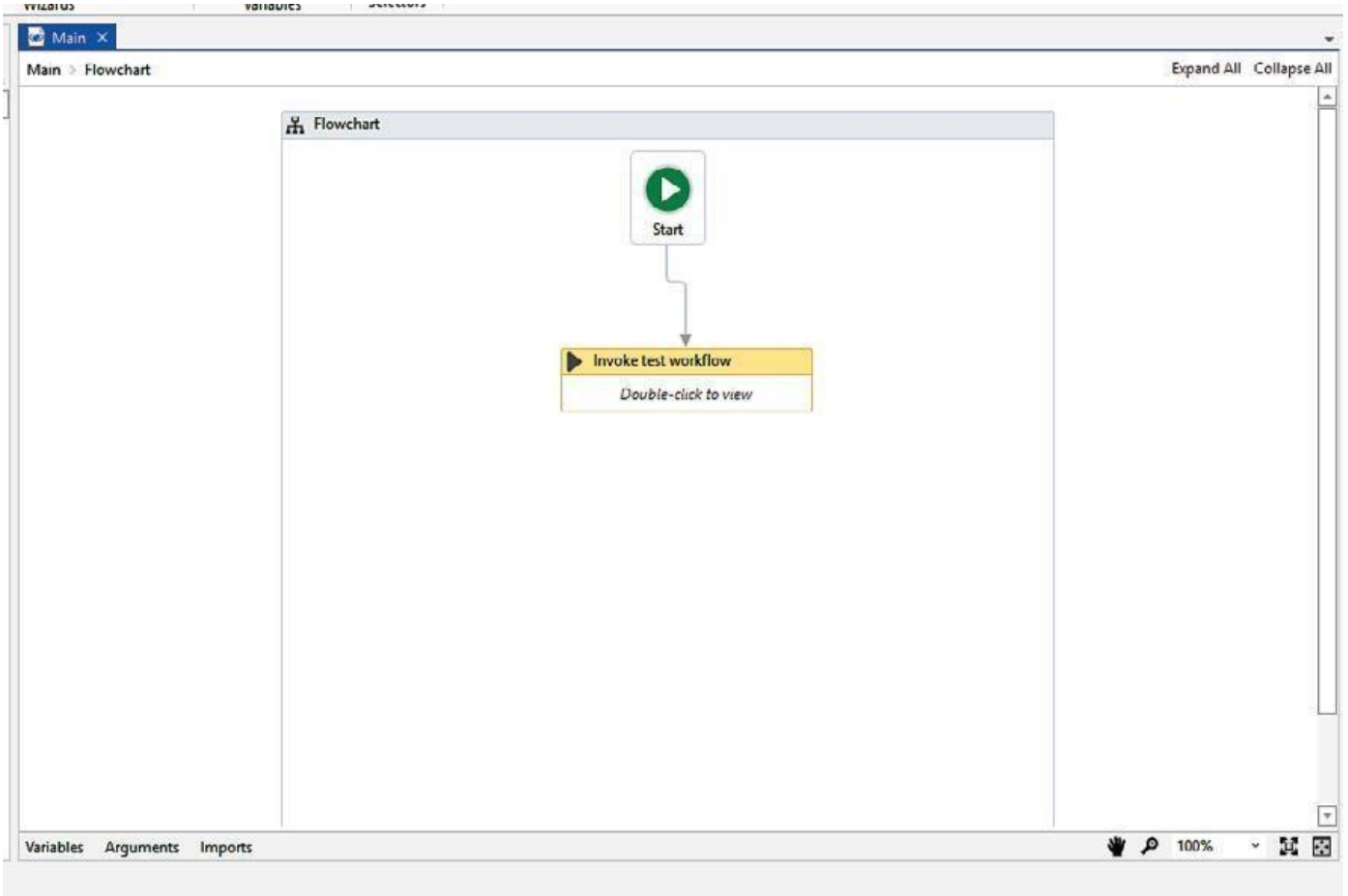
Right-click on the main Designer panel and choose **Extract as Workflow**:



A window will pop up asking for the name. Give it a meaningful name and click on **Create**. This will be the name of your workflow:



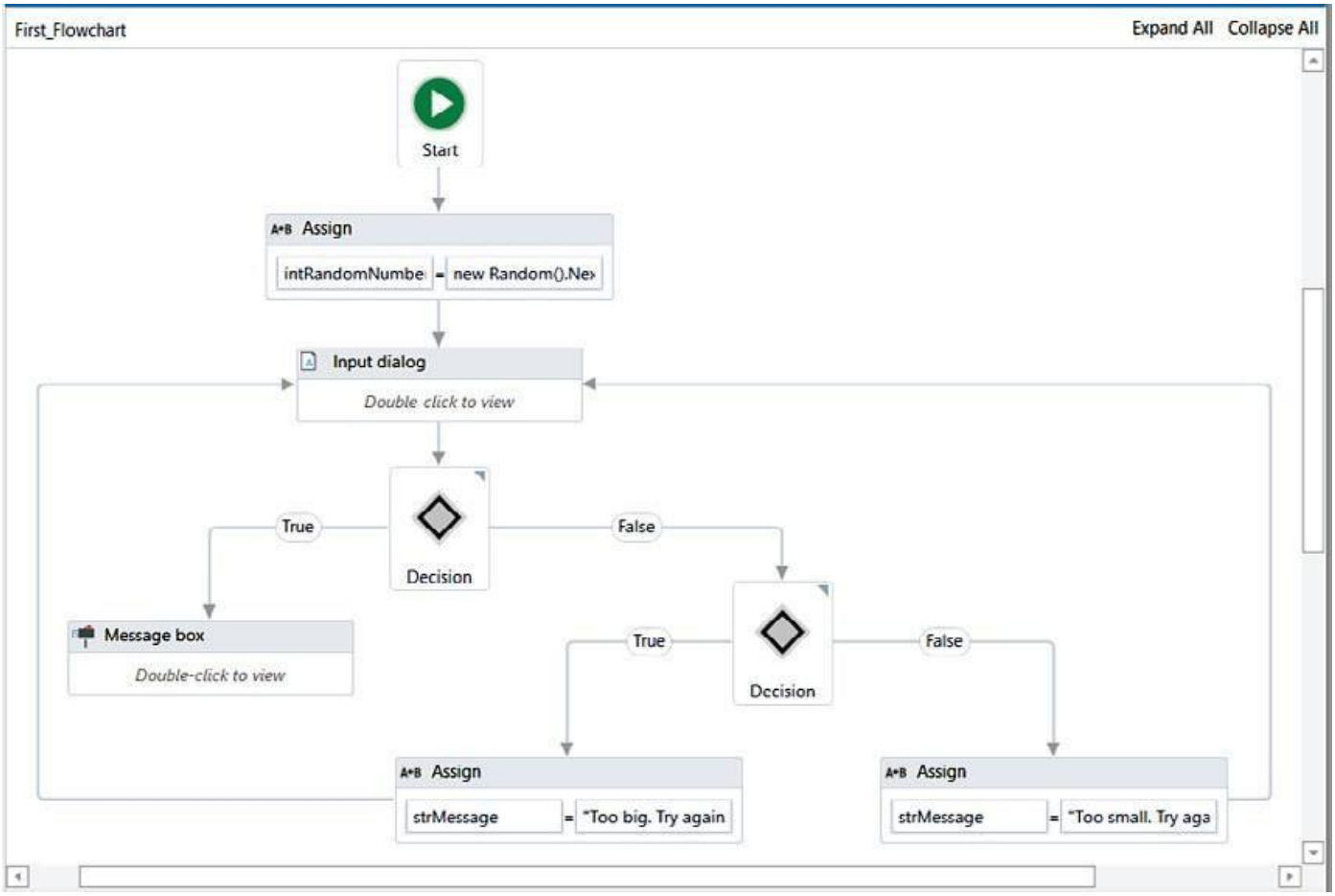
We have just used activities and extracted them in a workflow. If you check the main Designer panel, it looks like the following screenshot:



It automatically generates the **Invoke test Workflow** activity. Now, when we run the program, it will invoke the workflow that we have extracted (double-click on the **Invoke test workflow** activity to see which workflow it is going to invoke and where it is generated).

3.1.2 What Flowcharts are and when to use them

A Flowchart is generally used for complex business processes. It provides decision-making Facilities and can be used for both small and large projects. Here, we can add activities in Different ways:



A Flowchart provides multiple branching logical operators to make decisions. A Flowchart is able to run in reverse. Also, it can be used inside Sequences. A Flowchart facilitates reusability for distinct projects. Once we create it to use in a project, it can be used for a different but similar project.

A Flowchart's branches are set to true/false by default. However, its names can be manually changed from the **Properties** panel.

For example, enter two numbers and check whether their sum is less than 20.

Perform the following steps:

1. First, add a **Flowchart** from the **Activities** panel into the Designer panel.
2. Add a **Sequence** activity within the **Flowchart**.
3. Take two **Input dialog** activities (for entering the numbers to be added) inside the **Sequence** activity.
4. Create the variables *x* and *y* to save the values.
5. Next, add a **Message box** activity to perform a mathematical operation. In our case, the sum of the two numbers is less than 20:
x + y < 20
6. Now, add a **Flow Decision** activity to check the mathematical operation.
7. If true, the **Flow Decision** will flow toward the true branch. Otherwise, it will flow towards the false branch.

3.3 Control flow, various types of loops, and decision making

Control flow refers to the order or the particular manner in which actions are performed in an automation. UiPath provides numerous activities for performing the decision-making process.
These activities, present in the **Activities** panel, are put into the workflow either using the double-click method or the drag and drop method.

Different types of control flow activities are as follows:

- The Assign activity
- The Delay activity
- The Break activity
- The While activity
- The Do While activity
- The For each activity
- The If activity
- The Switch activity

3.3.1 The Assign activity

- The **Assign** activity is used to designate a value to the variable.
- The Assign activity can be used for different purposes, such as incrementing the value of a variable in a loop, or using the results of a sum, difference, multiplication, or division of variables and assigning it to another variable.

3.3.2 The Delay activity

- The **Delay** activity, as the name suggests, is used to delay or slow down an automation by pausing it for a defined period of time.
- The workflow continues after the specified period of time. It is in the hh:mm:ss format.
- This activity plays a significant role when we need a waiting period during automation, perhaps say, a waiting period required for a particular application to open.

Example: To better understand how the **Delay** activity works; let us see an example of an automation that writes two messages to the **Output** panel with a delay of 50 seconds. Perform the following steps:

1. First, create a new **Flowchart**.

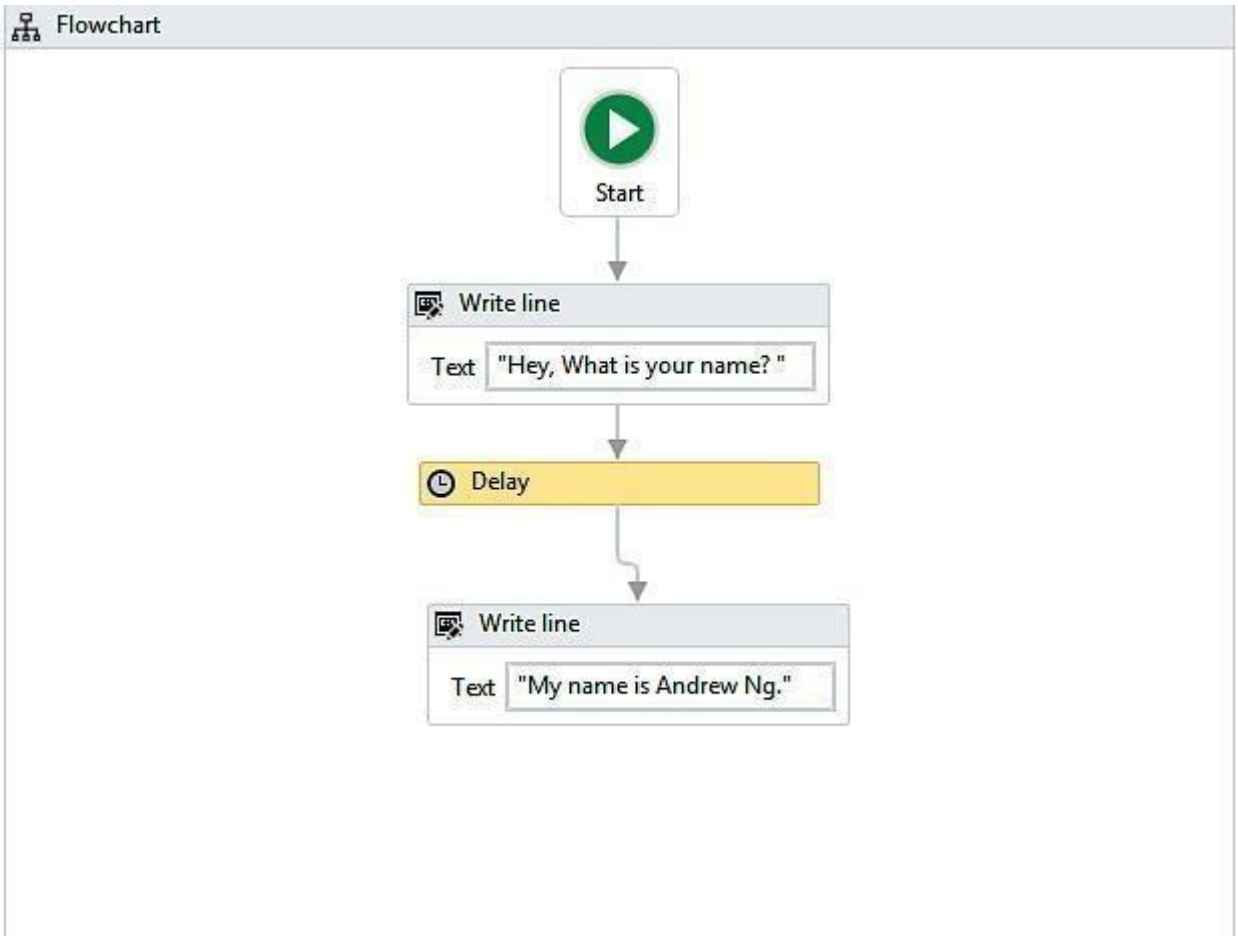
2. Add a **Write line** activity from the **Activities** panel and connect it to the **Start** node.

3. Select the **Write line** activity. Now, type the following text into the **Text** box: "Hey, what is your name".

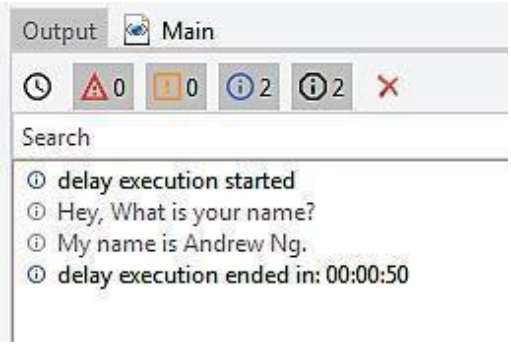
4. Next, add a **Delay** activity and connect it to the **Write line** activity.

5. Select the **Delay** activity and go to the **Properties** panel. In the **Duration** field, set 00:00:50. This is a 50-second delay between the two logged messages.

6. Take another **Write line** activity and connect it to the **Delay** activity. In the **Text** field, write "My name is Andrew Ng,":



7. After clicking on the **Run** button, the **Output** panel shows the message that delays it by 50 seconds:



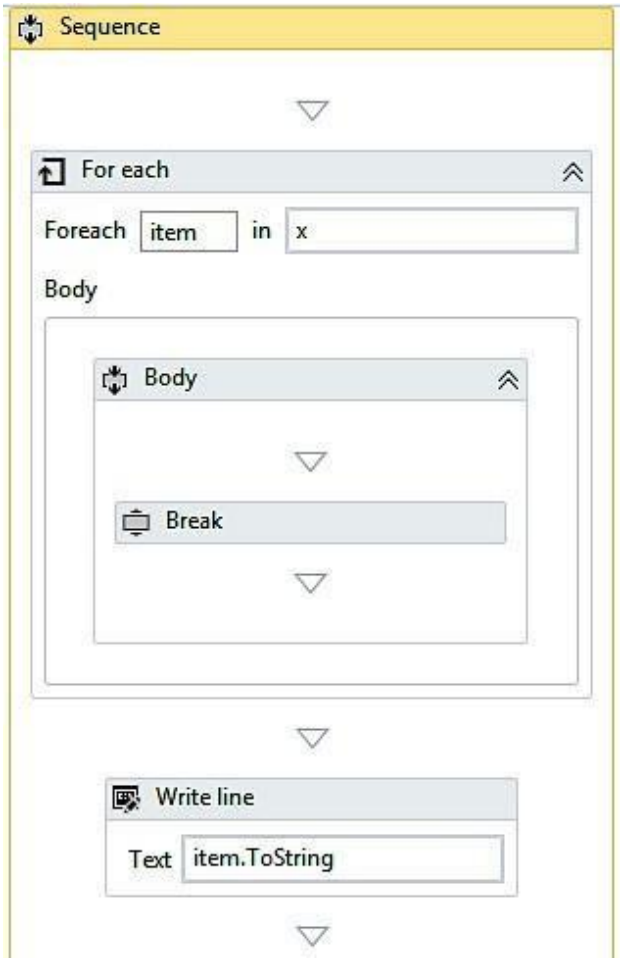
3.3.3 The Break activity

- The **Break** activity is used to break/stop the loop at a particular point, and then continue to the next activity according to the requirement.
- It cannot be used for any other activity apart from the For each activity.
- It is useful when we want to break the loop to continue to the next activity in the For each activity.

In this example, we will use the Break activity to execute only one iteration. Perform the following steps:

1. Add a **Sequence** activity to the Designer panel.

2. Next, add a **For each** activity inside the **Sequence** (as mentioned in the preceding section, to use the **Break** activity, we need the **For each** activity):



3. Create two variables; an integer variable named Item, and an array integer variable named X. Then, set them to the text field.

4. Now, assign a default value to the integer variable X.
5. Add a **Break** activity inside the body of the loop.

6. Under the **For Each** activity, add a **Write line** activity.

7. In the **Write line** activity, type Item to string in the text field.

8. When we click the **Run** button, it will display one element, as shown in the following screenshot. This is due to the **Break** activity, which has stopped execution after the first iteration:

A loop can simply be created by connecting the end of the workflow to the point where we want the workflow to resume. The While, Do while, and For each activities mentioned among the various control flow activities are examples of loops.

3.3.4 The While activity

- The While activity is used in automation to execute a statement or process based on a certain condition. If found true, the loop is executed; that is, the process is executed repeatedly.
- The project only exits from the loop when the condition does not hold true. This Activity is useful while iterating through an array of elements.

In the example, we will see how an integer variable will increase from 5 to 50 in increments of 5.Perform the following steps:

1. On a **Blank** project, add a **Sequence** activity.

2. Now, create an integer type variable X. Set its default value to 5.

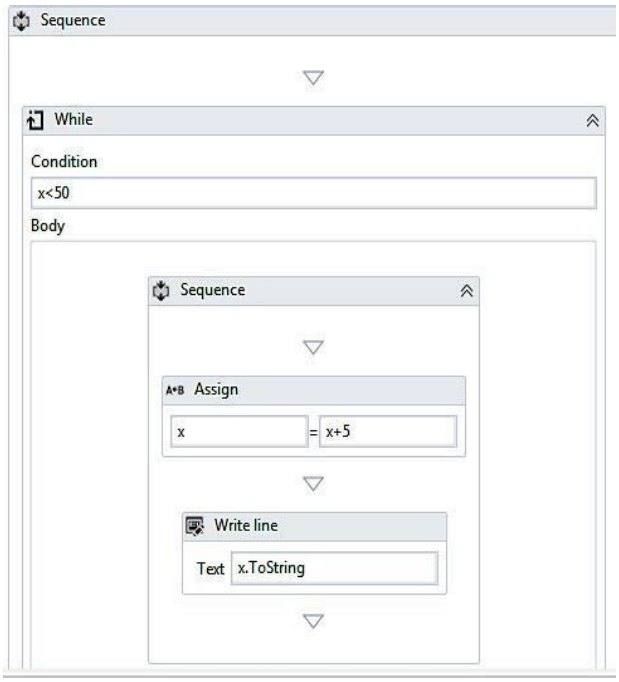
3. Next, add a **While** activity to the **Sequence**.

4. In the condition field, set X<5.

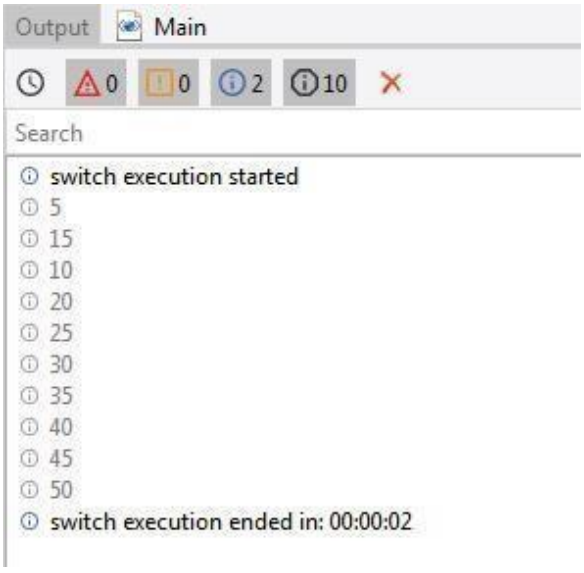
5. Add an **Assign** activity to the body section of the **While** loop.

6. Now, go to the **Properties** panel of the **Assign** activity and type in the text field Integer variable for value field integer X+5.

7. Drag and drop a **Write line** activity and specify the variable name X and apply Tostring method on this variable:



8. Now, click the Run button. The output will display in the Output panel, as shown in the following screenshot:



3.3.5 The Do while activity

- The Do while activity is used in automation when it is required to execute a statement based on the fulfillment of a certain condition.
- While activity executes a statement, then checks whether the condition is fulfilled.
- If the condition is not fulfilled, it exits the loop.

example to understand how the Do while activity works in automation. Take an integer variable. Starting with this variable, we shall generate all multiples of 2, less than 20.

Perform the following steps:

1. Add a Sequence to the Designer panel.

2. Add a Do while activity from the Activities panel.

3. In the body section of the Do while activity, add an Assign activity.

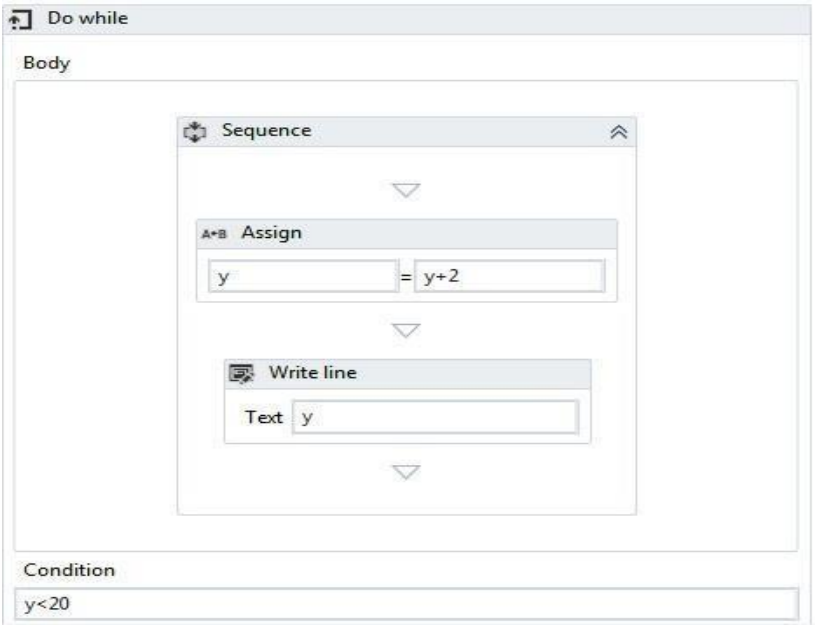
4. Now, select the Assign activity. Go to the Properties panel and create an integer variable y. Set its default value to 2.

5. Set y+2 in the value section of the Assign activity to increment the result each time by 2 until the loop is executed.

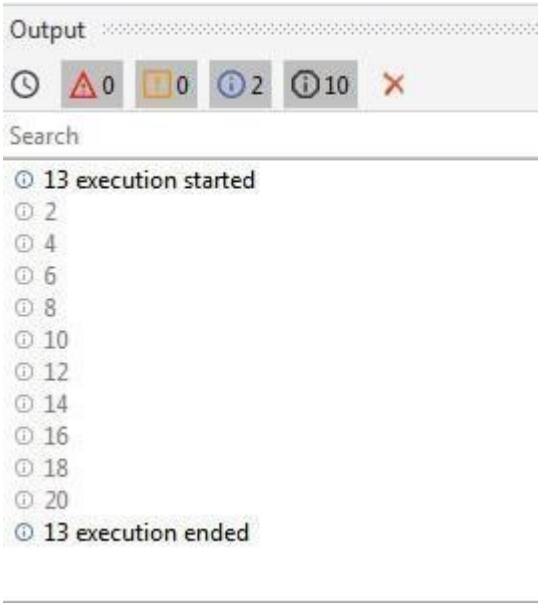
6. Add a Write line activity inside the Assign activity.

7. In the text field of the Write line activity, type y.

8. In the condition section, set the condition y<20. The loop will continue until the condition holds true:



9. On clicking the Run button, the output displayed will be as follows:



3.3.6 The For each activity

- The For each activity works by iterating each element from the collection of items or list of elements, one at a time.
- In the process, it will execute all the actions that are available inside the body. Thus, it iterates through the data and processes each piece of information separately.

Example, we shall use the For each activity to go through a collection of even numbers and display each element one at a time.

Perform the following steps:

1. Start with a **Blank** project in UiPath.

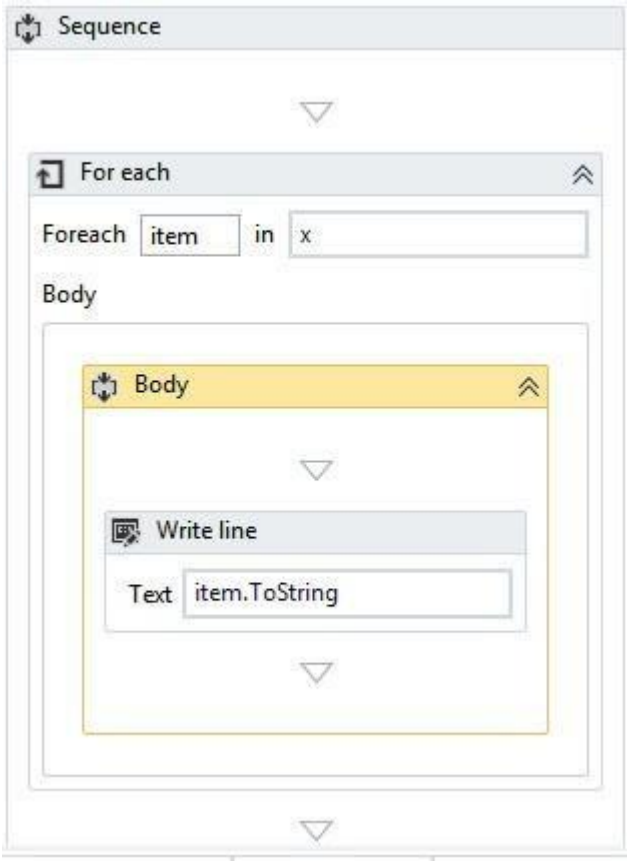
2. Add a **Sequence** activity to the Designer panel.

3. Next, add a **For each** activity within the **Sequence** and create an integer type array variable, X.

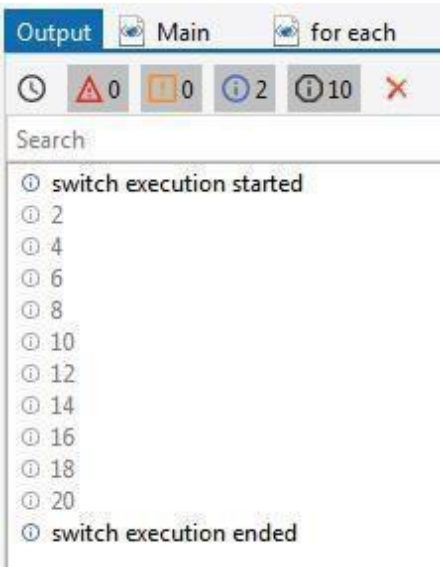
3. In the default value of the variable, put in ({2,4,6,8,10,12,14,16,18,20}).

4. Add a **Write line** activity to the Designer Panel (this activity is used to display the results).

6. In the **Text** field of the **Write line** activity, type item. To string to display the output:



7. Now, run the program. You will see that each number of the array is displayed one by one because of the use of the For each activity:



The If activity and the Switch activity are the Control flow's decision-making activities.

3.3.7 The If activity

- The If activity consists of a statement with two conditions: true or false.
- If the statement is true, then the first condition is executed; if not, the second condition is executed.
- This is useful when we have to take decisions on the basis of statements.

example that checks whether the sum of any two numbers is less than 6.
Perform the following steps:

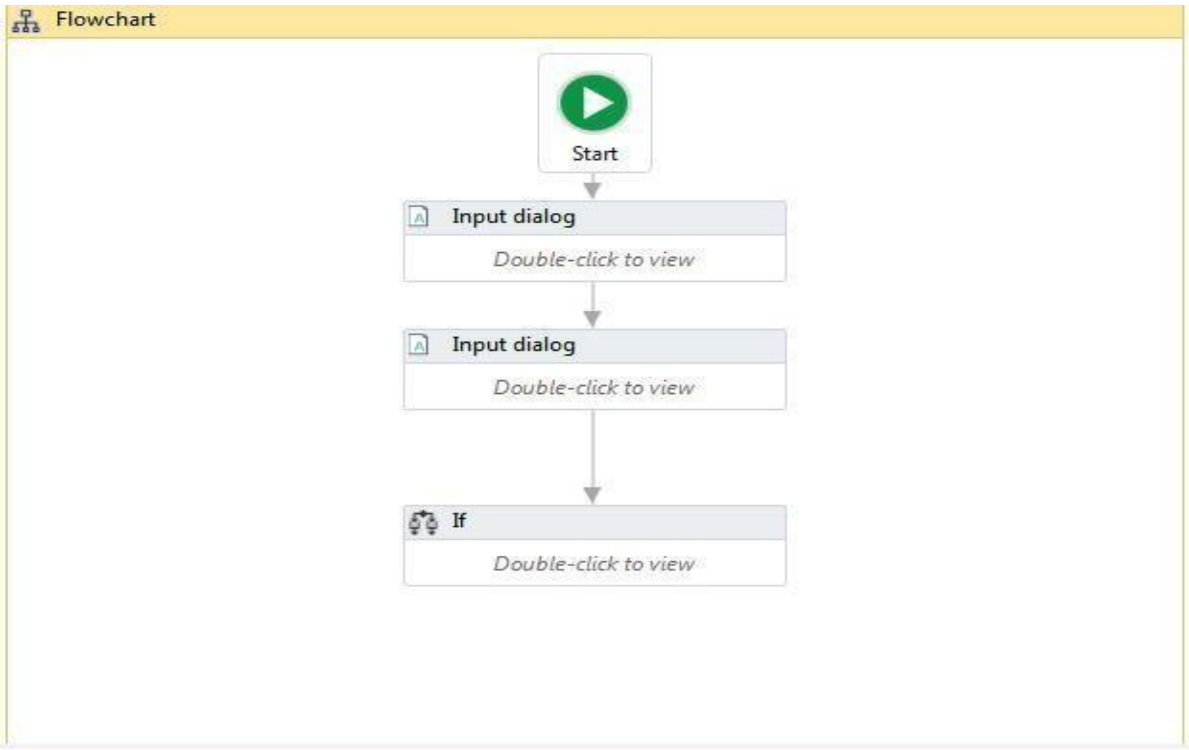
1. Add a **Flowchart** from the **Activities** panel.

2. Add two **Input dialog** activities. Create two integer variables, *x* and *y*.

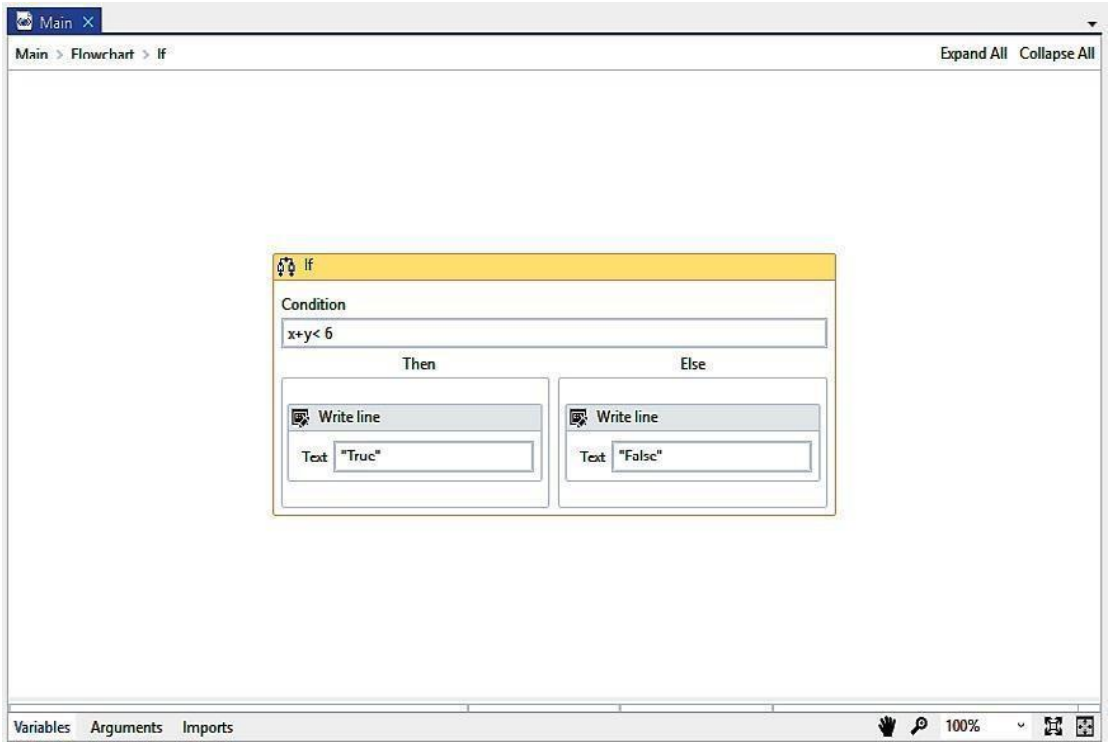
3. In the **Properties** panel, change the label name and title name of both the **Input dialog** activities.

4. Now, specify these name of these two variables in the **Result** property of both the **Input dialog** activities.

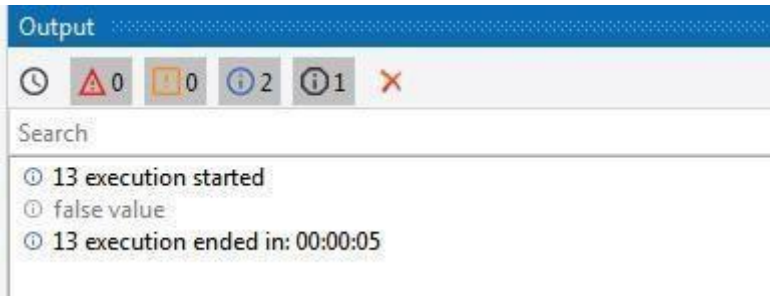
5. Now add the **If** activity to the Designer panel.



6. In the condition part, X+Y<6 check whether it is true or false. Add two Write line activities and type “True” in one and “False” in the other:



7. Click the **Run** button to check the output. If the condition holds true then it will show the true value; otherwise, it will show the false value, as shown in the second screenshot (in our case, we put in the values of *x* and *y* as x and y, respectively, thus getting a sum of 13, which is not less than 6; hence, the output shows it as false value):



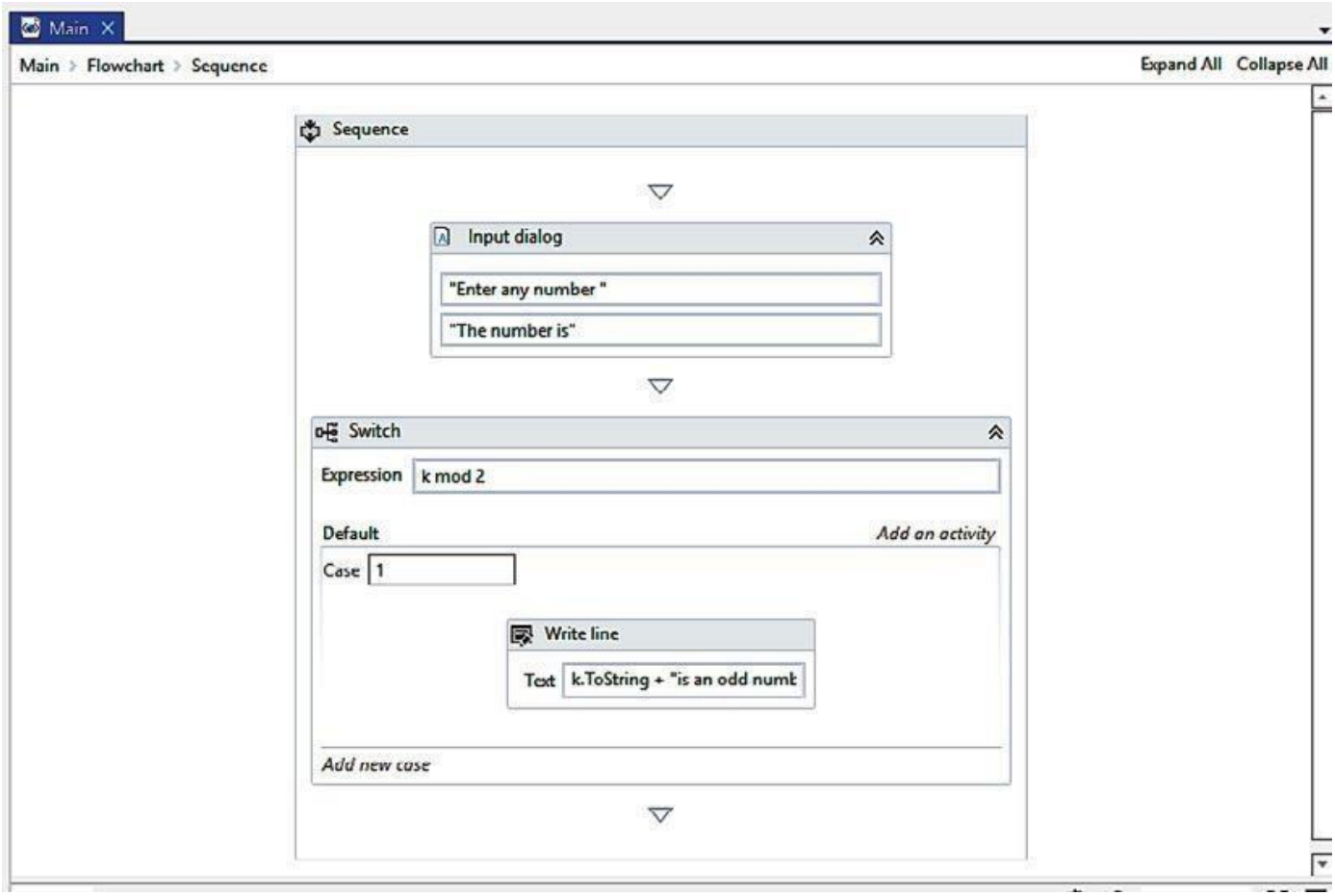
3.3.8 The Switch activity

- The **Switch** activity can be used to make a choice.
- When we have various options available and want to execute one option, we frequently usethe Switch activity.
- By default, the Switch activity takes an integer argument. If we want to take a desired argument, then we can change it from the Properties panel, from the Type Argument list.
- The Switch activity is very useful in the categorization of data according to one's ownChoice.

- Example where we have to check whether a given number is odd or even.
- We know that all odd numbers, when divided by 2, leave a remainder of 1.
- On the other hand, even numbers, on being divided by 2, leave a remainder of 0.
- Hence, we will have two cases getting a remainder of 1 or 0.

Perform the following steps:

1. Add a **Sequence** activity.
2. Add an **Input dialog** activity inside the **Sequence**.
3. Now, create an integer type variable k.
4. Specify the newly created variable's name in the **Result** property inside the **Properties** panel.
5. Add the **Switch** activity under the **Input dialog** activity.
6. In the **Expression** field, set k mode 2 to check whether the number is divisible by 2 or not.
7. Add a **Write line** activity to the **Default** section and type the k. To string + "is an even number" in the text field.



3.4 Step-by-step example using Sequence and Flowchart

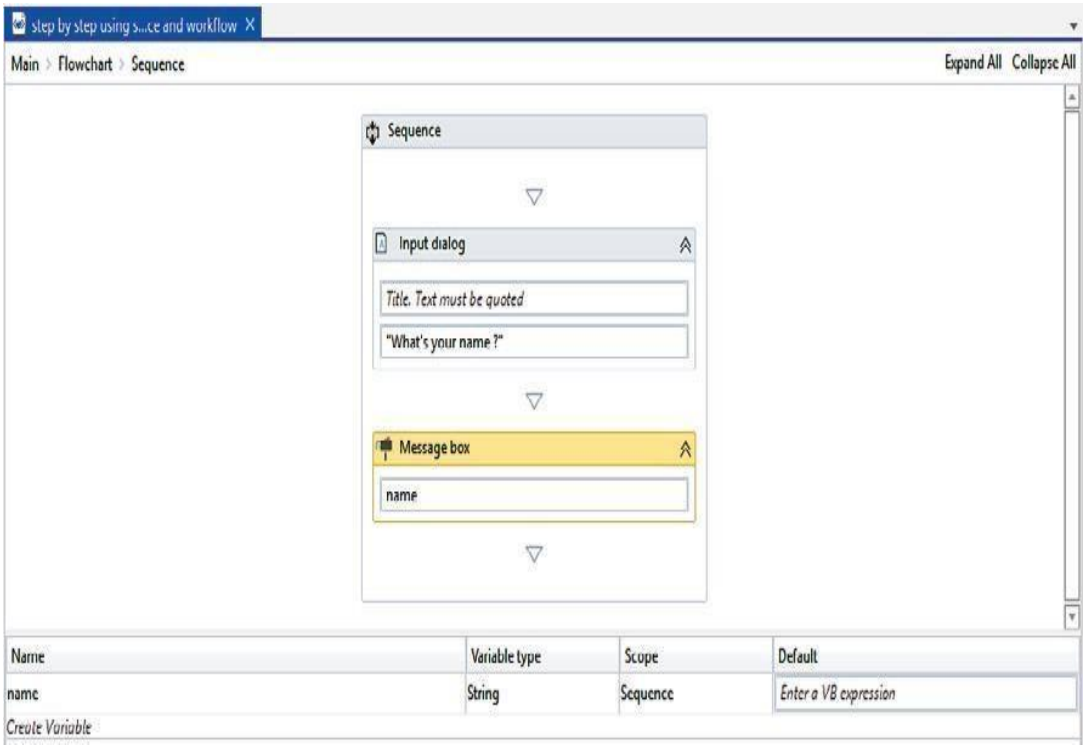
- A Sequence and a Flowchart are similar concepts.
- They are both used to contain logical steps or actions.

3.4.1 How to use a Sequence

- There may be different Sequences doing their jobs. We can easily put similar Sequences into a workflow; each workflow represents a task.
- It is very easy to test a separate workflow alone.
- Perform the following steps:

1. Drag and drop a **Flowchart** onto the Designer panel. Drag and drop a **Sequence** activity. Connect the **Sequence** activity with the **Start** node.
2. Double click on the **Sequence** activity. Drag and drop an **Input dialog** activity and a **Message box** activity. Specify a message in the **label** property of the **Input dialog** activity.
3. Create a variable of type **String**. Give it a name. Also, specify this newly created

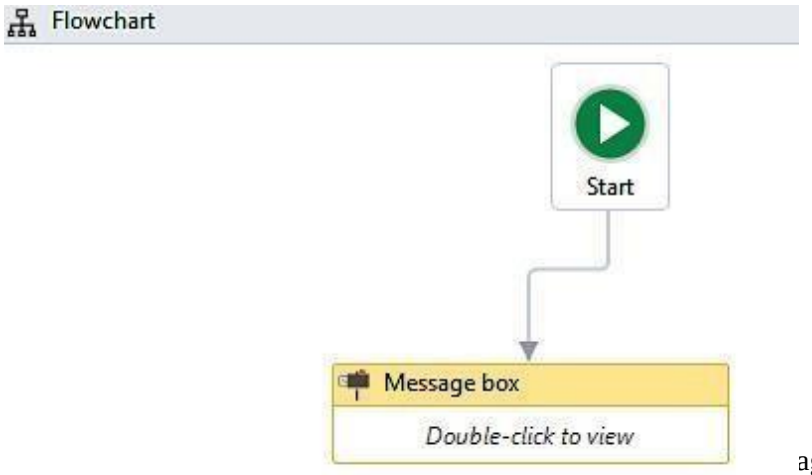
variable's name in the content property of the **Message box** activity:
Hit the **Run** button or press *F5* to see the result.



- We can see clearly that we have used two activities inside the **Sequence** that are logically related (one for inputting the name and the other for popping it up). Here, the **Sequence** contains two activities.

3.4.2 How to use a Flowchart

- A **Flowchart** is a container. It can contain activities inside it.
- Let us drag and drop a **Message box** activity inside the **Flowchart**. Double click on the
- **Message box** and type "Hello World!" in the area where the text is to be quoted. Press *F5* to see the result):



- So, when the program has only a few steps, we can use activities directly inside the **Flowchart**.
- However, it becomes more complex when we have a large number of steps. That is why it is necessary to arrange the related activities into Sequences and then group the Sequences into a **Flowchart**.
- Example to see how to use Sequences in the **Flowchart**.
  Perform the following steps:

1. Drag and drop two **Flowchart** activities on the main **Flowchart**. Rename them as Send mail and Message.

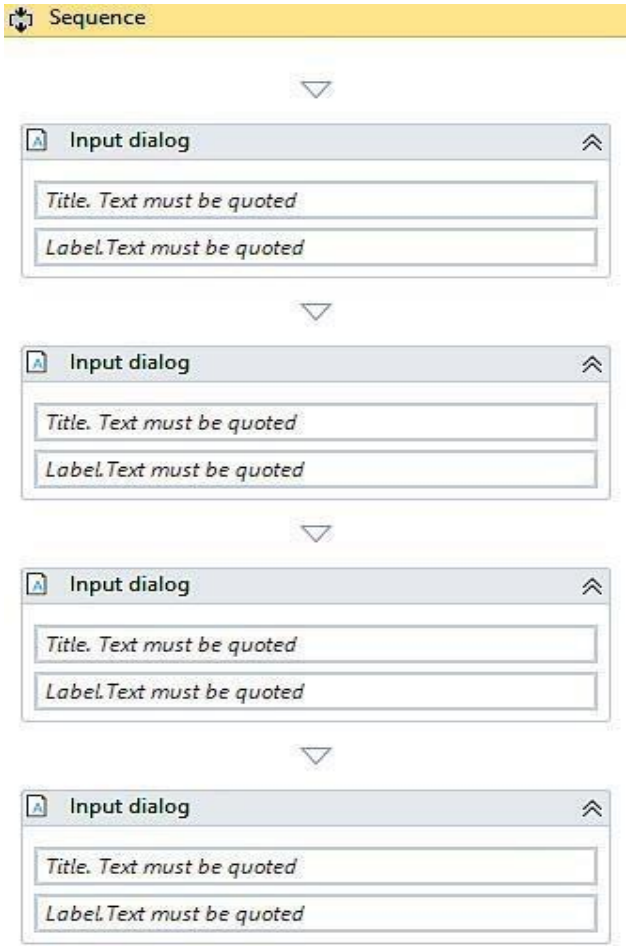
- We have two different workflows.
  - The **Send Mail** workflow will send the mail to an email address.
  - The **Message** workflow has the message body of that email and will ask the user for a name, message, sender, and receiver.

2. We have to implement the desired steps in both workflows. For that, we are using a **Sequence** inside the **Flowchart**.

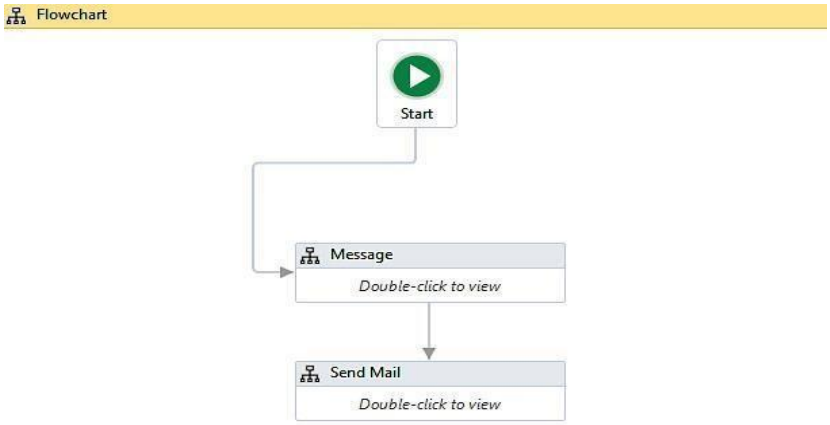
- Double click on the **Flowchart**. Drag and drop a **Sequence** activity inside both Flowcharts.
- Connect the **Sequence** to the **Start** node by right-clicking on the **Sequence** and selecting the **Set as Start node** option.

3. Double click on the **Sequence** in the **Message** Flowchart.

- Drag and drop four **Input dialog** activities for the name, message, sender, and receiver.



4. Double click on the **Send Mail Flowchart**. Double click on the **Sequence**. You can drag and drop the email activities here.
5. That's it. Now, go to the main Flowchart. Connect the Message Flowchart to the Start node. Also, connect the Send Mail activity to the Message Flowchart:



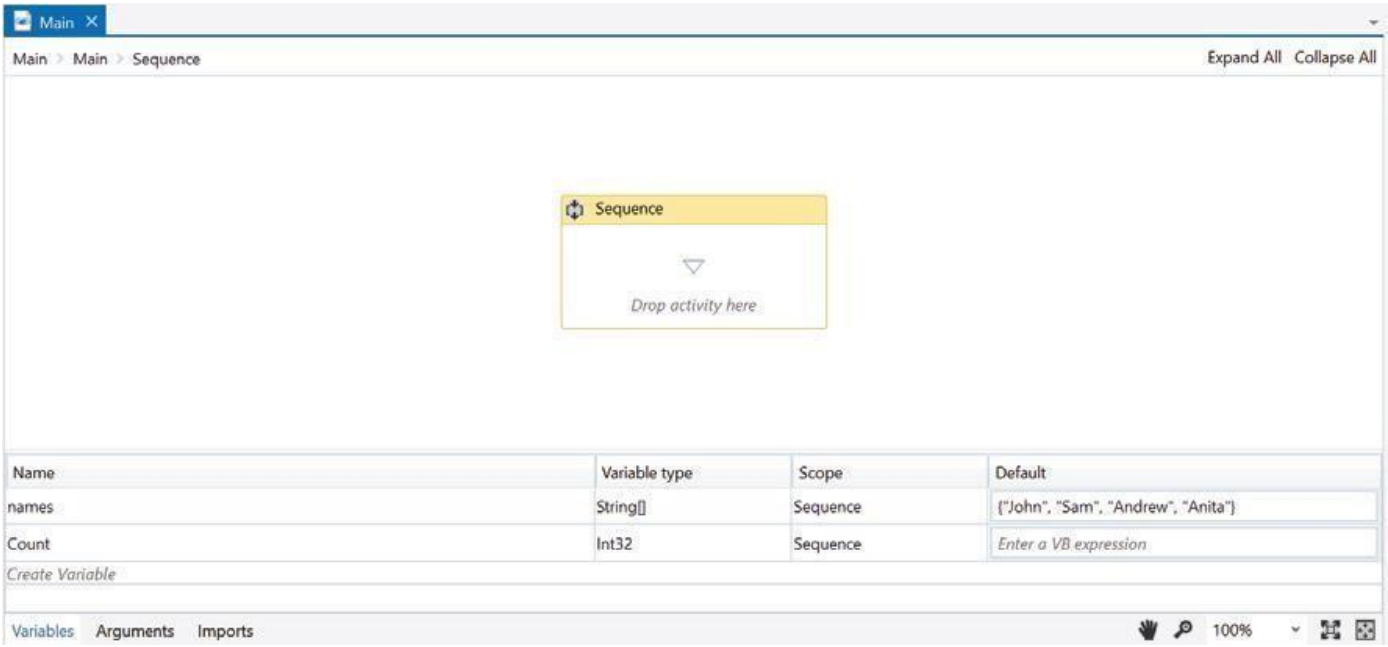
6. Run the program and visualize it.

3.4.3 Step-by-step example using Sequence and Control flow

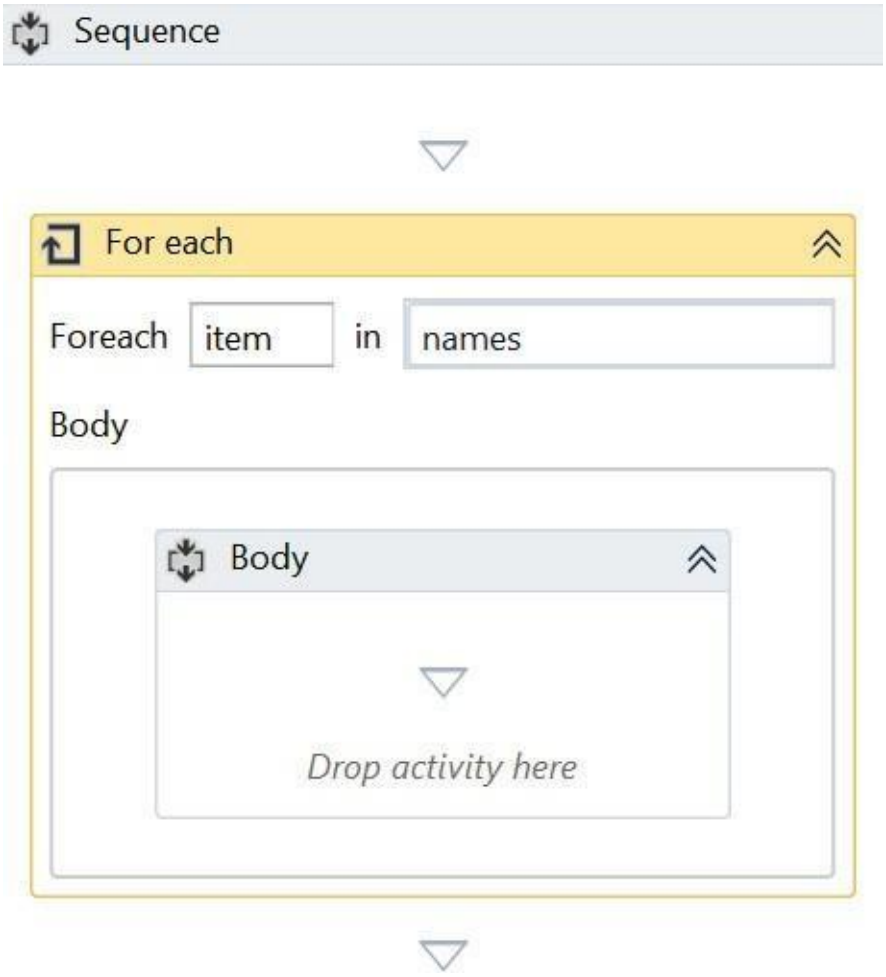
- Consider an array of names. Say we have to find out how many of them start with the letter *a*.
- We will then create an automation where the number of names starting with *a* is counted and the result is displayed.

- Perform the following steps:

1. Drag and drop a **Flowchart** activity from the **Activities** panel.
2. Drag and drop a **Sequence** activity inside the **Flowchart**. Connect the **Sequence** to the **Start** node by right-clicking on the **Sequence** activity and selecting the **Set as Start node** option.
3. Double click on the **Sequence** activity. Create a variable. Give it a name (in our case, we will create an array of type string and name the variable as *name*). Set the variable type to **Array of [T]**. When asked for the type of array, select **String**. Also, initialize the array in the **Default** section of the variable by giving it a default values. For example, *{"john", "sam", "Andrew", "Anitha"}*.
4. Create a variable of type integer **Count** for storing the result. Set the variable type to **Int32**:

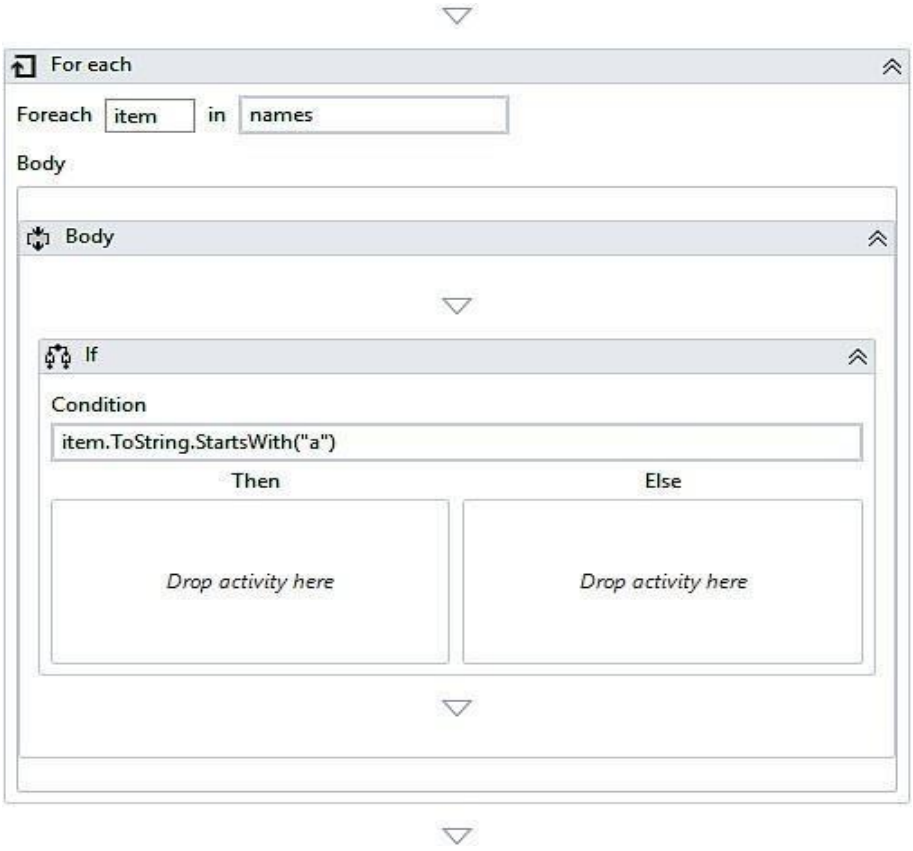


5. Drag and drop a **For each** activity inside the **Sequence**. Also, specify the array name in the expression box of the **For each** activity. The **For each** activity is used to iterate over the array. It will pick up one name from the array each time until it reaches the end:


Sequence
For each
Foreach
item
in
names
Body
Body
Drop activity here

6. Drag and drop the **If** activity from the **Activities** panel and place it inside the **For each** activity at the location where ***Drop activity here*** is mentioned.

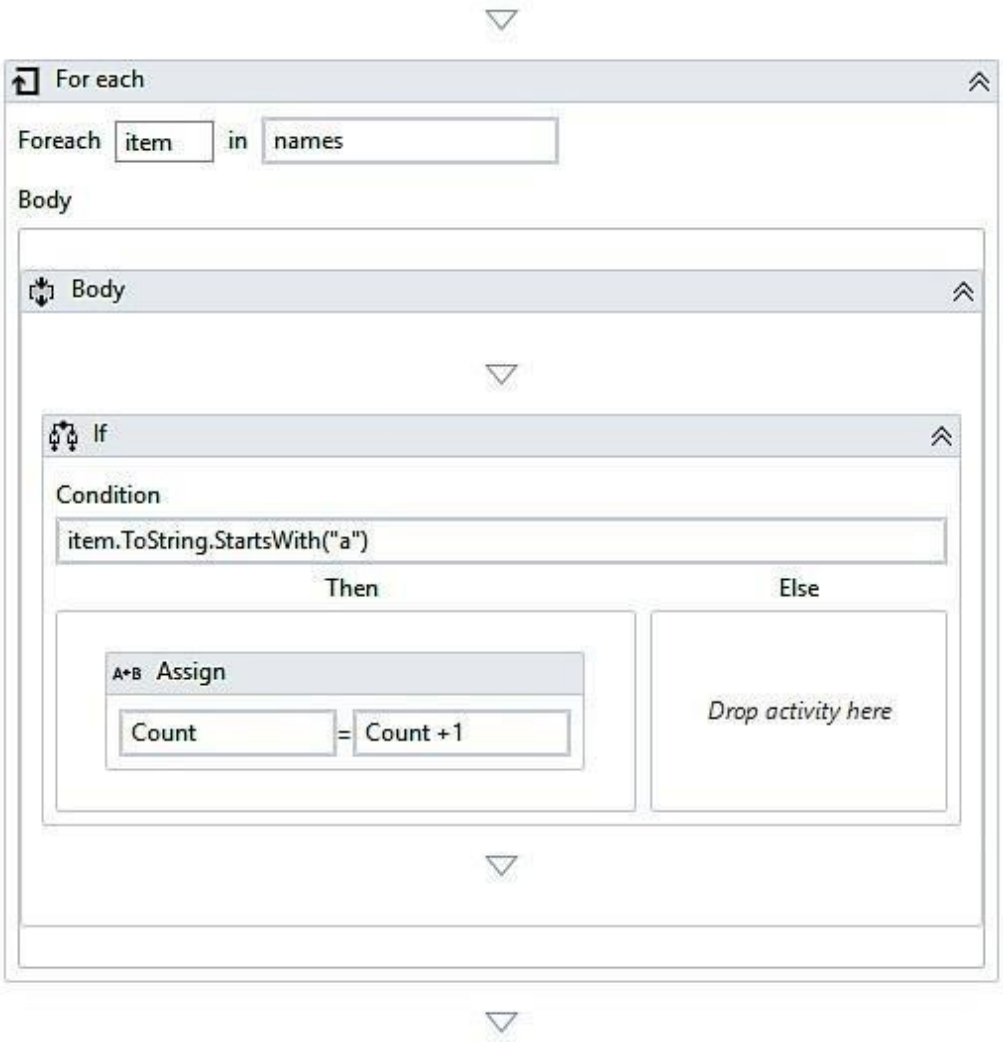
- Specify the condition in the expression box of the **If** activity. The **If** activity is used to check for a particular condition/expression.
- If that expression is satisfied, the **Then** block will be executed. Otherwise, the **Else** block will be executed.
- We have specified the expression as *Item.Tostring.Startswith('a').* This expression specifies the name present in the item variable starts with the letter 'a'.
- The **For each** activity iterates over the array, picks up one name at a time,
- and stores it as a variable, *item*:



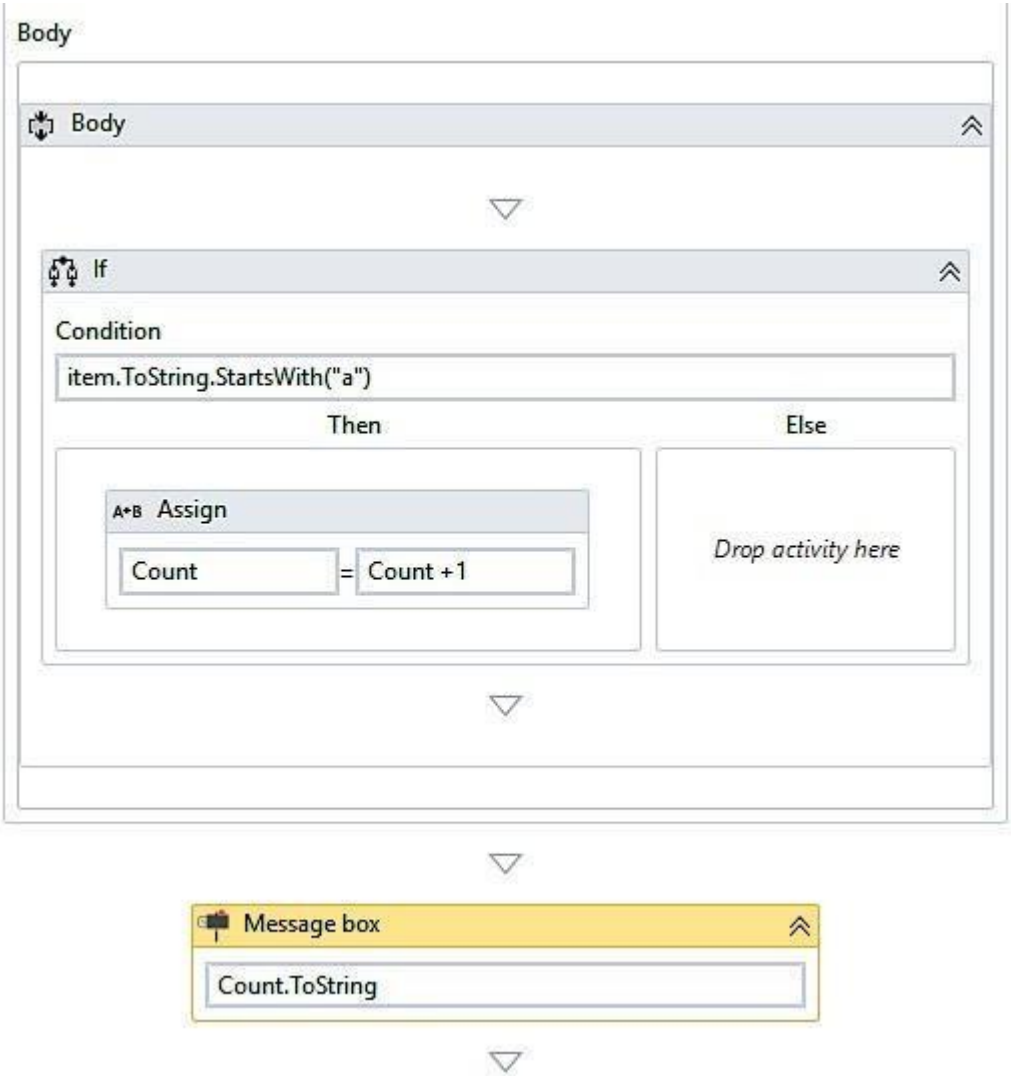
7. Now, we are going to use the *count* variable and increment it each time a name from an array starts with the letter *a*.

- we have to use the **A+B Assign** activity. Drag and drop the **A+B Assign** activity inside the **If** activity.

- Set the **To** property to *count*(variable name) and the **Value** property to *Count+1*(to increment its value) of the **A+B Assign** activity:



8. Just drag and drop a **Message box** activity inside the **Sequence** activity. Specify the count variable in the expression box of the **Message box** activity. But remember, the variable that we have created is of type **Int32**, so, it cannot be used with the **Message box** activity without converting it to a string. To convert it to a string, we have the *'.tostring'* method available in UiPath Studio. Just apply it to the variable and select *'.tostring*:



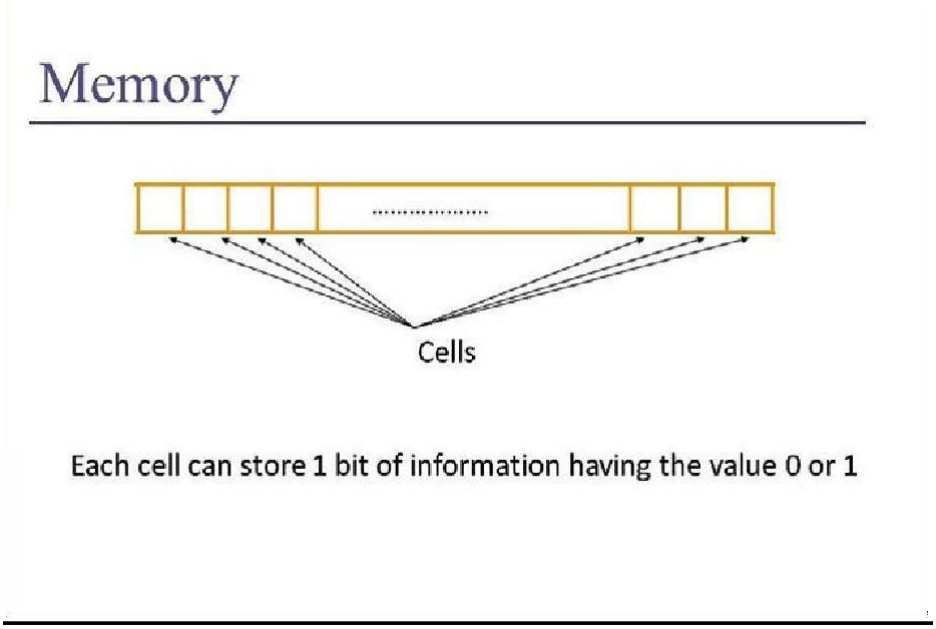
Hit the **Run** button or press *F5* and see the result.

4. Data Manipulation

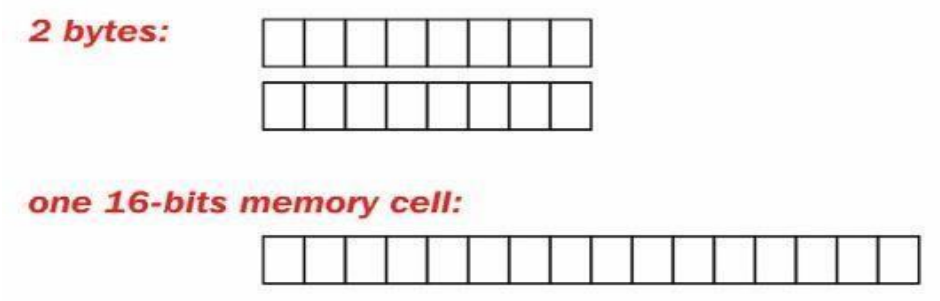
Data manipulation is the process of changing data whether it is adding, removing, or updating it.

4.1 Variables and scope

- Before discussing variables, let us take a look at Memory and its structure:



- Memory consists of millions of memory Cells and each memory cell stores data in the form of 0s and 1s (binary digits).
- Each cell has a unique address, and by using this address, the cell can be accessed.

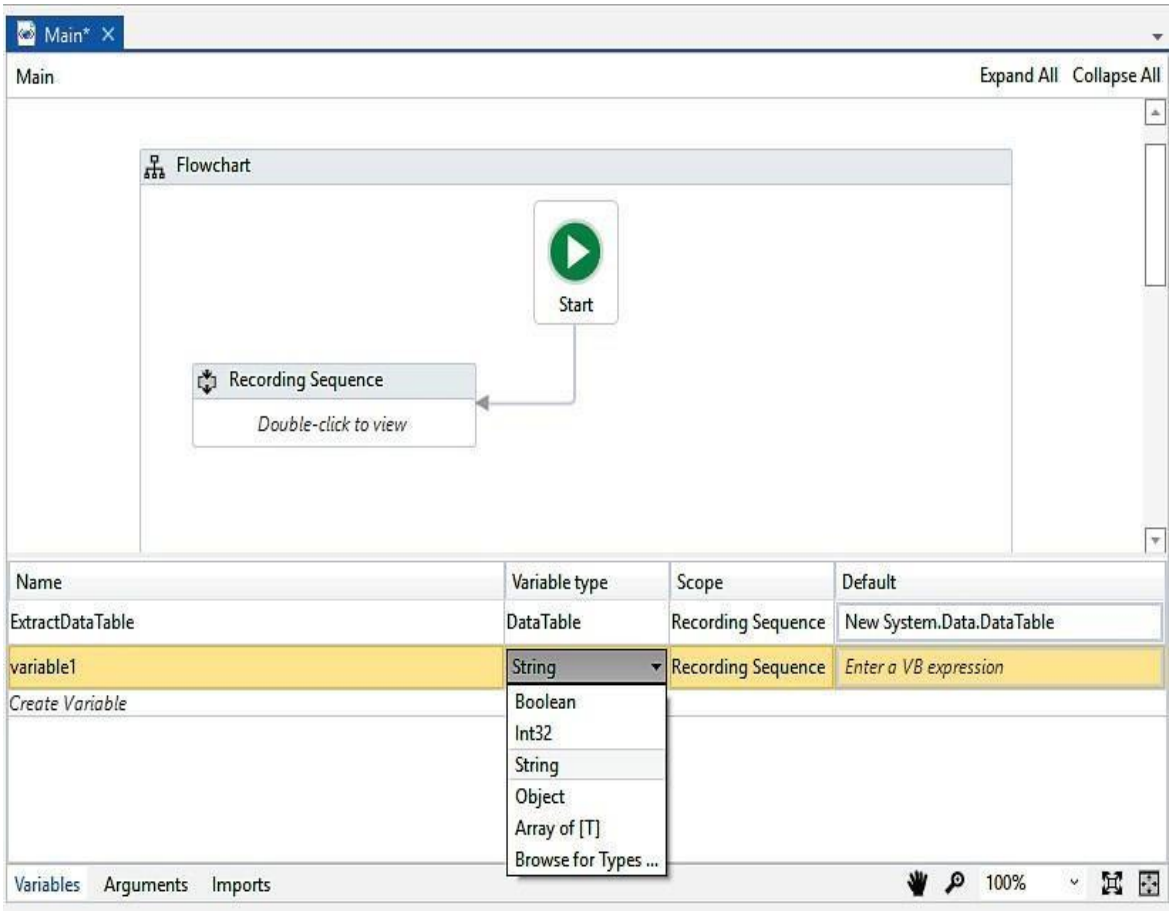


- When data is stored in memory, its content gets split into further smaller forms (binary digits). As shown in the preceding diagram, **2 bytes** of data consists of several memory cells.
- A **variable** is the name that is given to a particular chunk of memory cells or simply a block of memory and is used to hold data.
- A variable is used to store data. Data is present around us in different Types-it can be an mp3 file, text file, string, numbers, and so on.
- A particular type of variable can hold only that type of data.
- If there is a mismatch between the data and the variable type, then an error occurs.

- The following table shows the type a of variable available with UiPath:

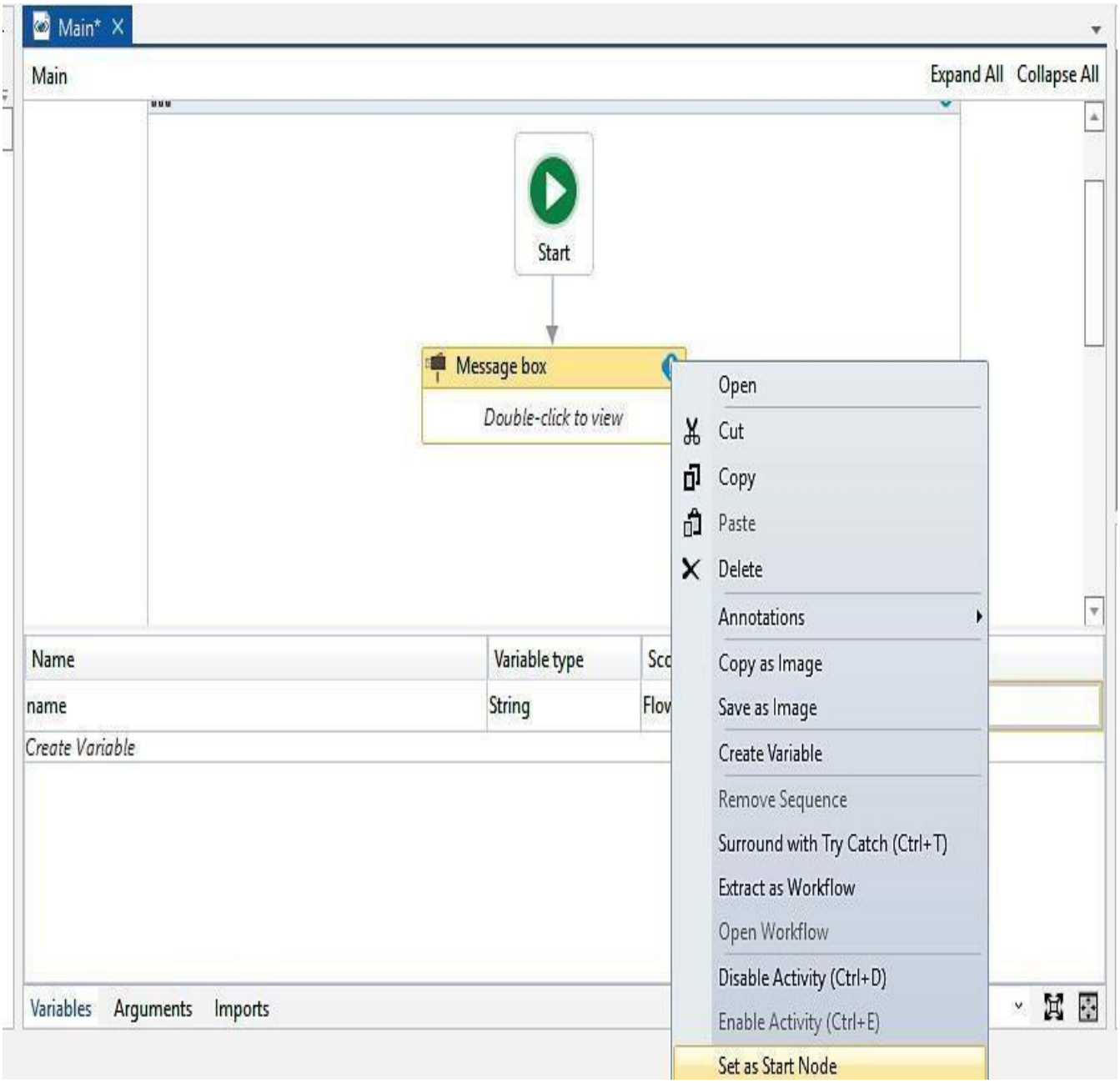
Type	Content
Integer	Whole numbers
String	Text of any kind: "The Quick Fox @4598"
Boolean	True or false
Generic	Anything

- In UiPath, we can declare a variable in the **Variables** section. Just give it a meaningful name and select the appropriate type from the drop-down list.
- We can also specify the scope of a variable. The **Scope** is the region under which the data has its effect or availability.
- You can choose the **Scope** of the variable according to your requirements; try to limit it as far as possible.

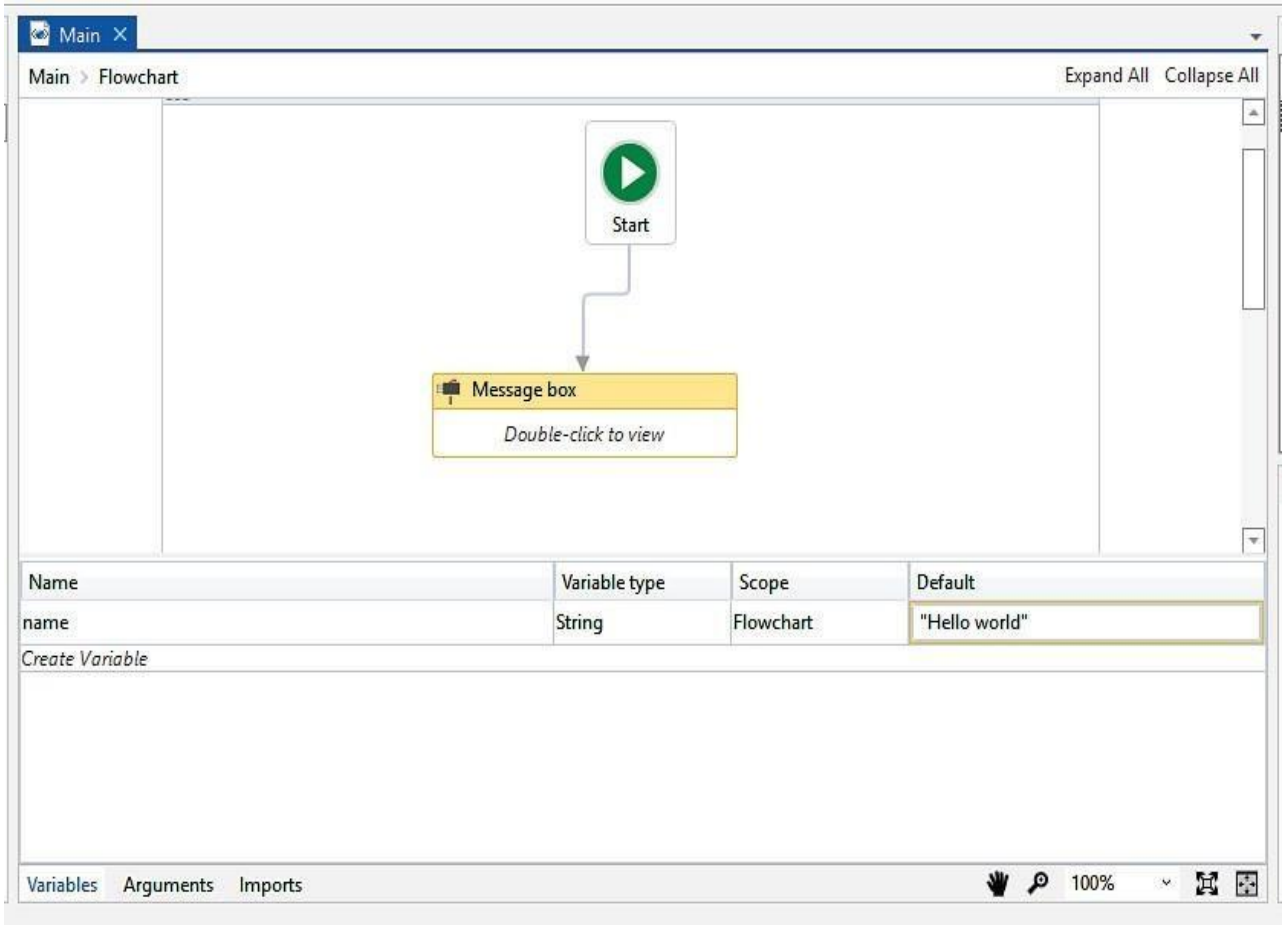


- Let us take an example of creating a variable and then displaying a **Message box** using that variable:

1. We have declared a variable as *name* in the **Variables** section and set its **Default** value to *"Hello World"*. By default, the type of the variable is **String** (we can change its type according to our needs).
2. Search *for Message box* in the **Activities** panel. Drag and drop that **Message box** template into a **Flowchart**.
3. Right-click on the message template and select **Set as Start node**:



4. Double-click on the **Message box** template and specify the variable name that we Created earlier. At this stage, we are ready to run our application by simply clicking on the **Run** button:



A dialogue box will pop up with the "Hello World" text displayed on it.

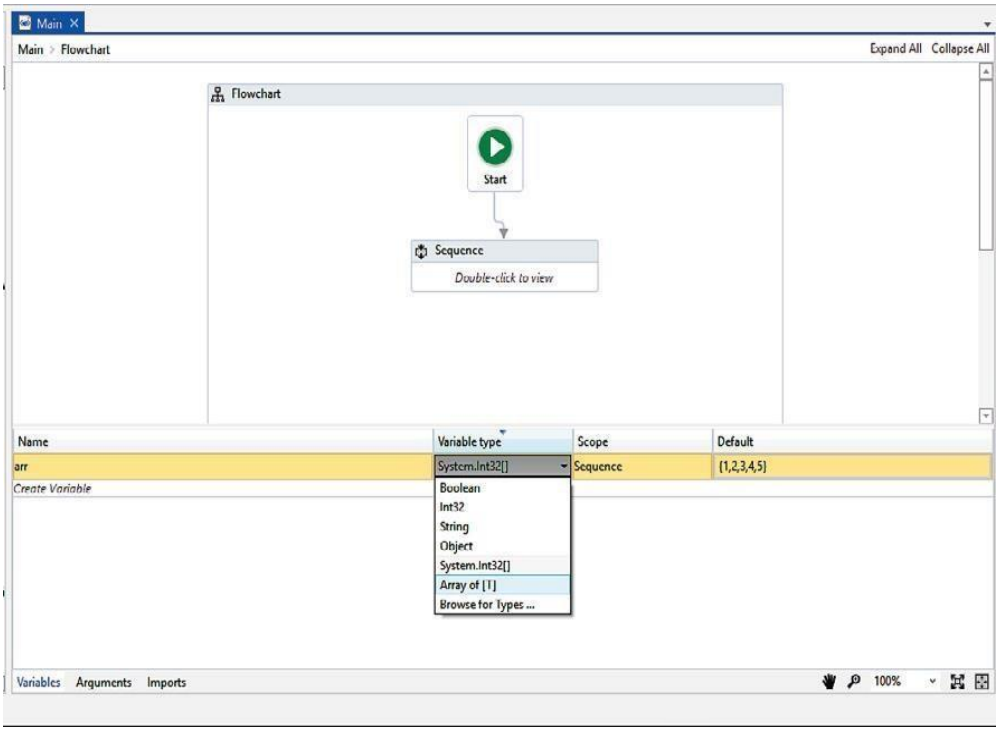
4.2 Collections

- There are different types of variables. Variables can be classified into three categories:
  - **Scalar:** These are variables that can only hold a single data point of a particular data type, for example; Character, Integer, Double, and so on.
  - **Collections:** These are variables that can hold one or more data point of a particular data type. For example; array, list, dictionary, and so on.
  - **Tables:** These are a tabular form of the data structure which consists of rows and columns.
- In a collection, we can store one or more data points, but all the data must be the same.
- In this example, we are going to take an array of integers, initialize it, and then iterate through all the elements of the array:

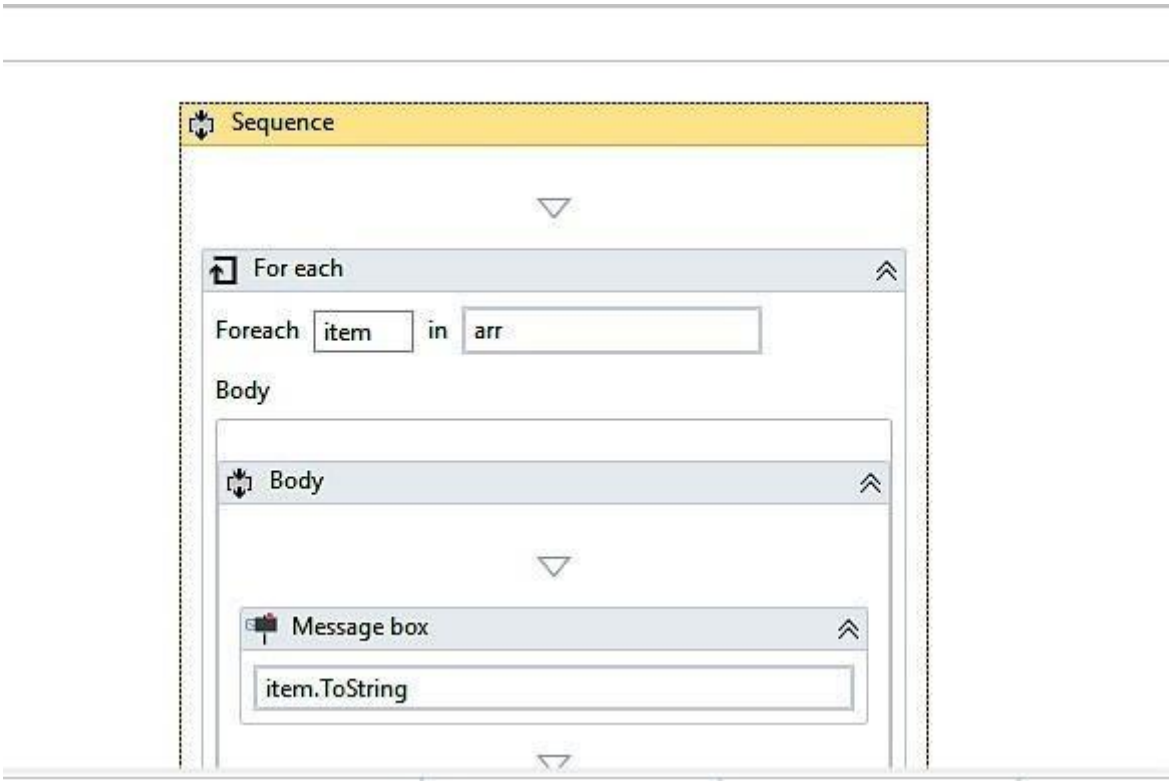
1. Drag and drop a **Flowchart** activity onto the main Designer panel, and drag and drop a **Sequence** activity inside the **Flowchart**. Set the sequence as **Start** node.

2. Create a variable in the **Variables** panel and give it a meaningful name (in this example, we have created a variable named ***arr***, which is an array of integers). Choose the data type as an array of integers.

3. We have initialized the array as {1, 2, 3, 4, 5} in the **Default** section. You can initialize it with the **int32** data type:



4. Drag and drop a **For each** activity from the **Activities** panel inside the **Sequence**, and drag and drop a **Message box** activity inside the **For each** activity.
5. Specify the array name in the expression text box of the **For each** activity.
6. Specify the *item* variable that is auto-generated by the **For each** activity, inside the **Message box** activity. But hold on, we have to convert the *item* variable into the **String** type because the **Message box** activity is expecting the string data type in the text box. Just press the dot (.) along with the *item* variable and choose the **ToString** method:



Hit the **Run** button to see the result. All the values will pop up at once.

4.3 Arguments - Purpose and use

- An Argument is simply a variable that can store a value.
- You can create an argument in the Argument section of the main Designer panel. variable and is used to pass values between different workflows.
- Suppose we have a big project to build; we break down the project into different workflows because smaller workflows can be easily tested separately.
- It is very easy to build smaller workflows and combine them, thus turning them into the real solution of the project.
- These Arguments are used for interacting with different workflows by exchanging data between them.
- That is why the direction property is associated with Arguments.
- We can choose the direction on the basis of our requirement either giving the value to some workflow or receiving the value from another workflow.
- We can easily create arguments in the **Arguments** panel. We can also specify the direction:

  - **In**: When we have to receive the value from another workflow.
  - **Out**: This is the current value if we have to send the value to a workflow.
  - **In/Out**: This specifies both; it can take or receive the value.
  - **Property**: This specifies that it is not being used currently: