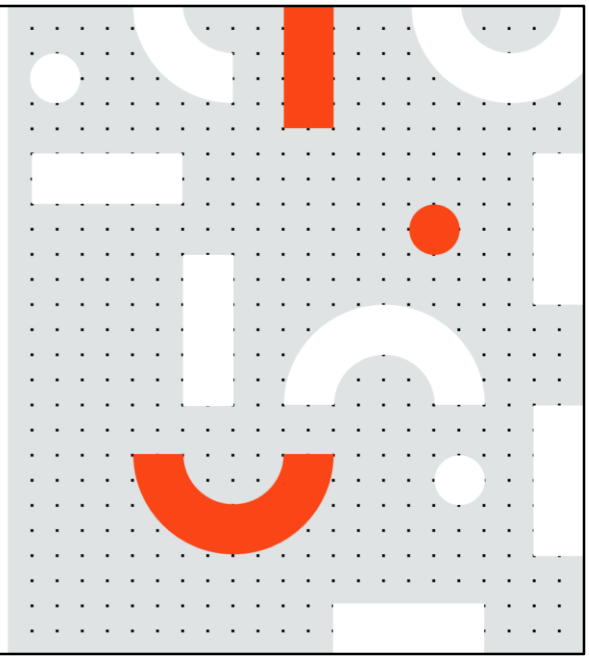
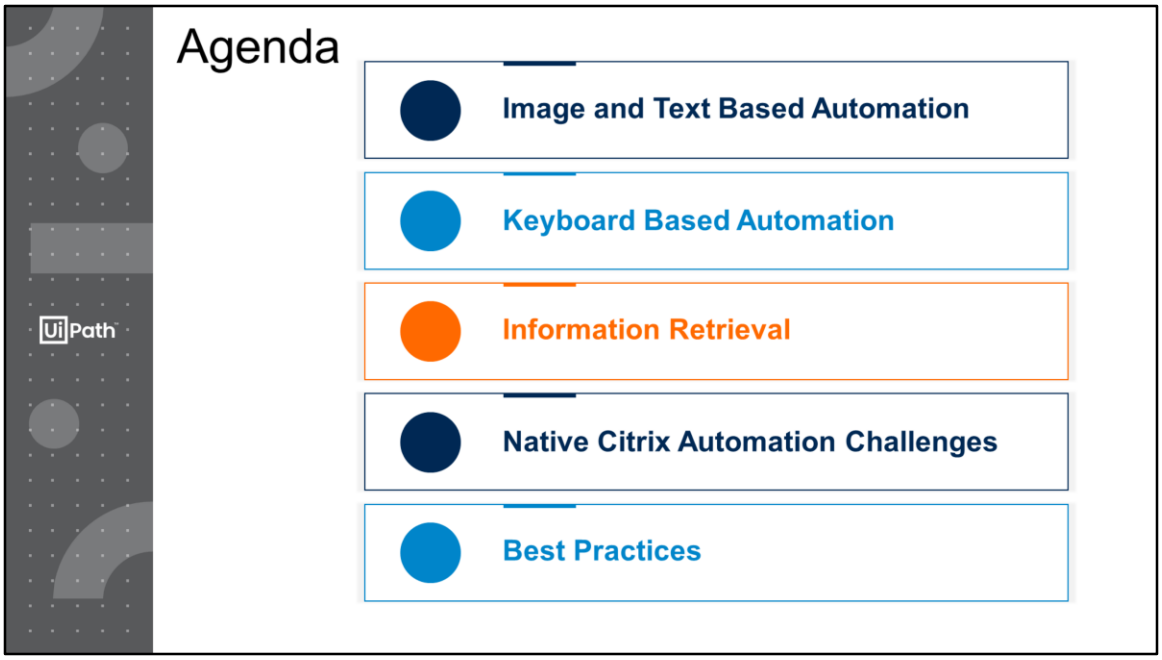
RPA Design & Development v1.1

Lesson 11
Image, Text and Advanced Citrix Automation



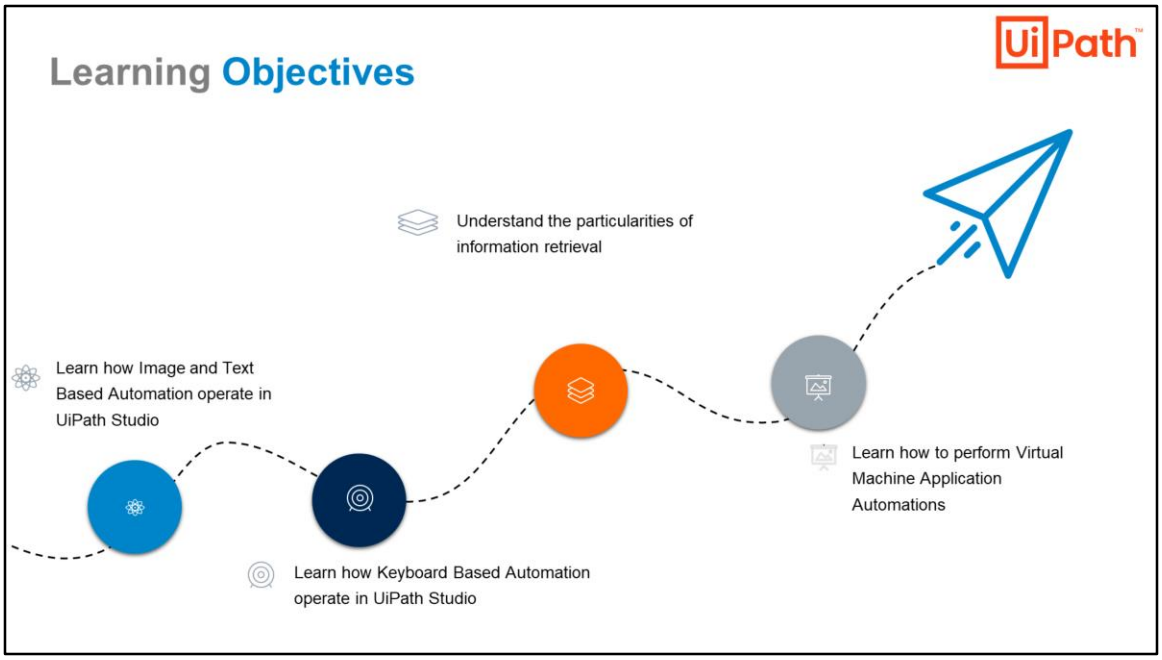
Module Objectives:
After completing this lesson, students should be able to:

- Understand Image, Text and Advanced Citrix Automation



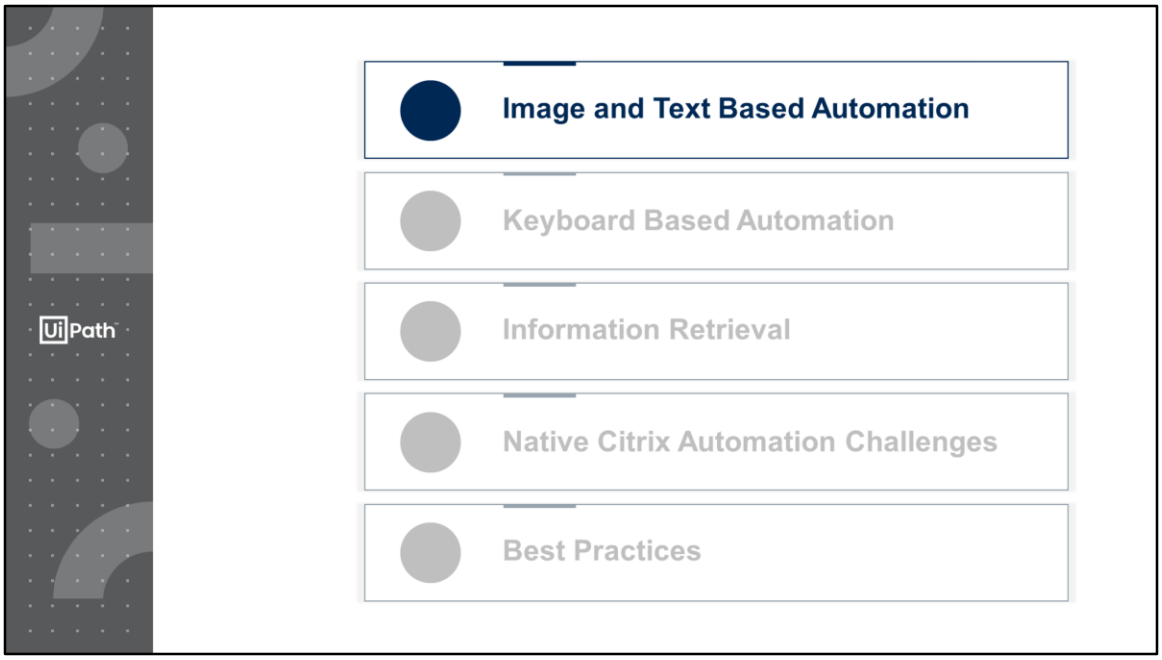
In this lesson, we will cover the following topics:

- Image and text-based automation
- Keyboard based automation
- Information Retrieval
- Native Citrix Automation challenges
- Best Practices



At the end of this lesson, you will be able to:

- Learn how Image and Text Based Automation operate in UiPath Studio
- Learn how Keyboard Based Automation operate in UiPath Studio
- Understand the particularities of information retrieval
- Learn how to perform basic Image and Text Automations as well as Virtual Machine Application Automations



In this topic, we will learn about the Image Based automation, its importance and challenges faced while using them.

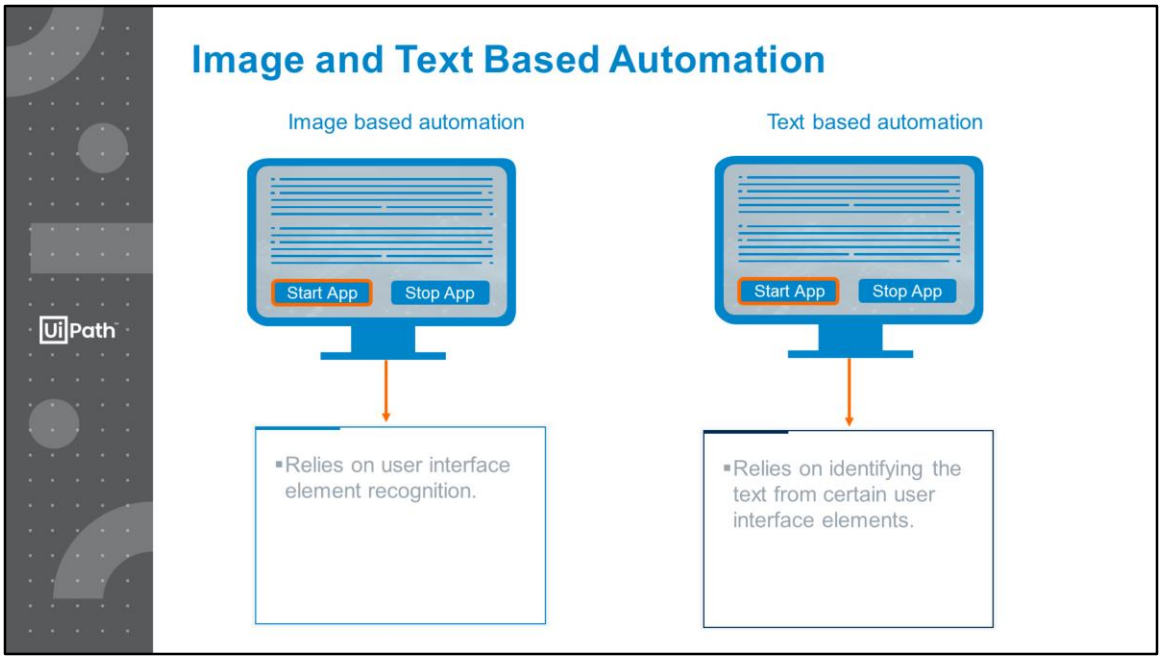
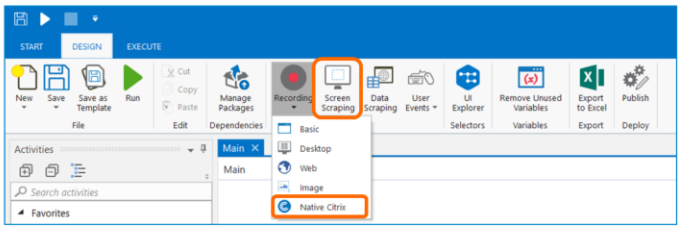


Image based automation relies on user interface element recognition. Based on the recognized elements, a robot can perform specific automation tasks.
Text based automation relies on identifying the text from certain user interface elements.
Image and Text automation is very useful in situations where UI automation fails to work.
This happens when selectors cannot be found by using regular methods. Applications hosted on Citrix, Citrix Desktop, scanned PDF, SAP which have scripting disabled examples of environments where selectors are not exposed by the application.

Image and Text Based Automation (Contd.)

The Screen Scrapping is a UiPath feature which can extract text from running apps.

- The Screen Scrapping tool is found in the UiPath Studio ribbon.
- The Native Citrix application is found by expanding the Recording drop down button.



To enable image and text-based process automation, UiPath Studio features activities that simulate keyboard and mouse input, such as clicking, hovering or typing, text recognition and OCR (optical character recognition) activities that use screen scraping to identify UI elements, as well as image recognition activities that work directly with images to identify UI elements.

Screen scrapping is a UiPath feature which is able to extract text from running apps even when they are hidden or covered by another app. The Screen Scrapping tool can be found in the UiPath Studio ribbon. This appear in the Recording Tab where there are options for Basic desktop web , image, and native citrix in the drop down.

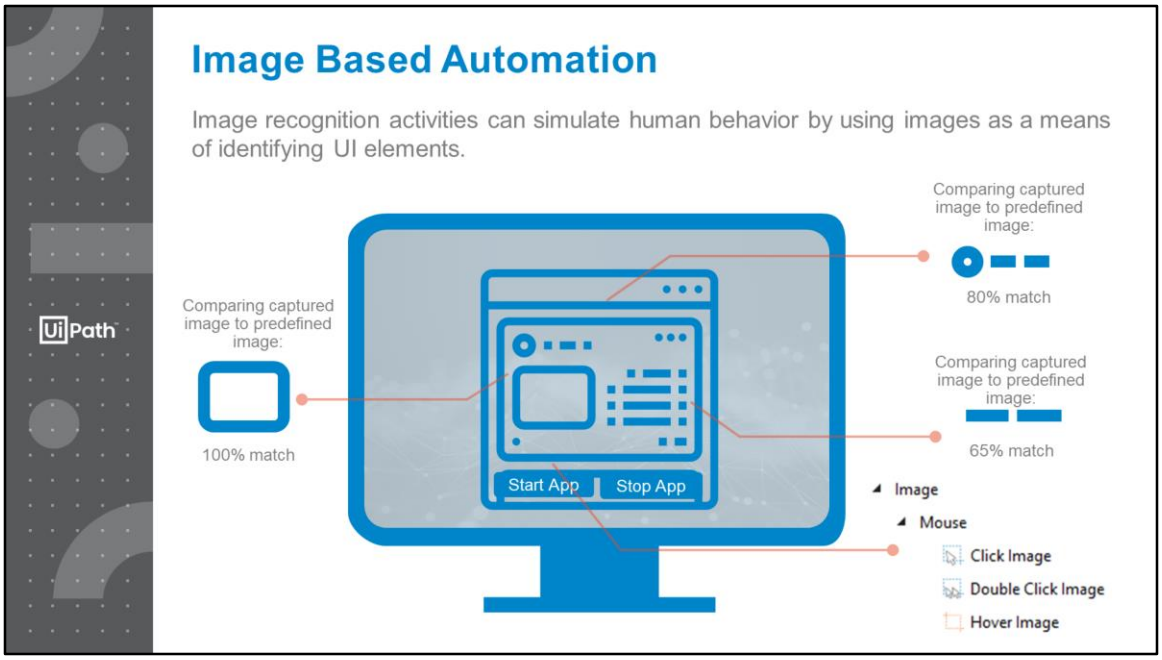
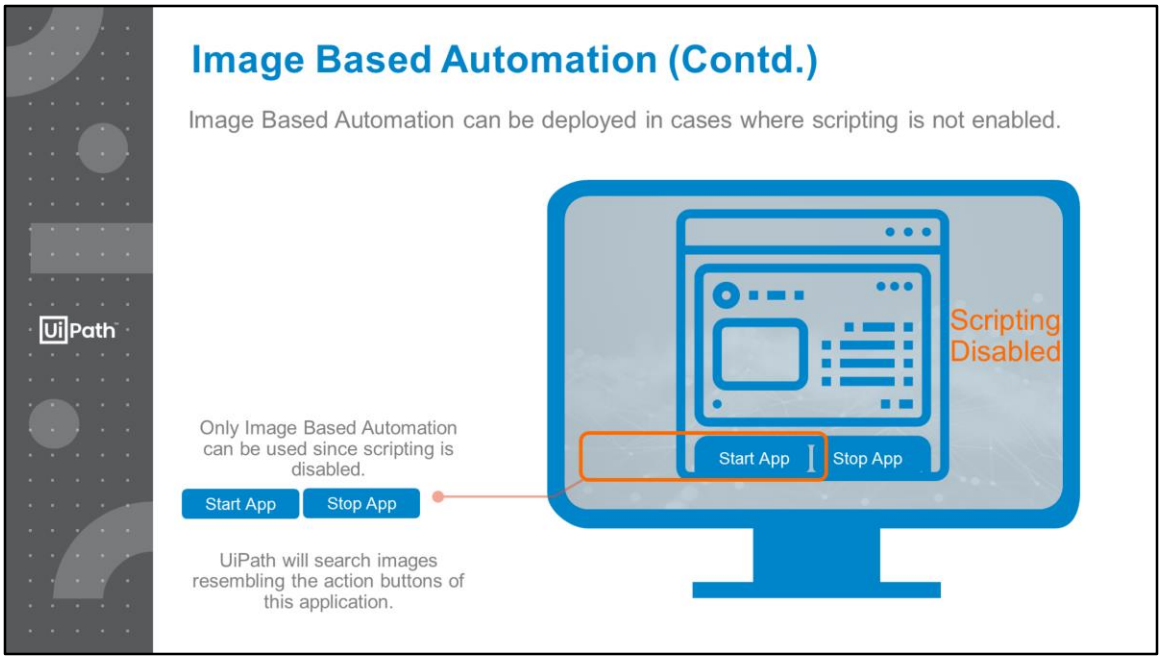


Image recognition activities can simulate human behavior by using images as a means of identifying UI elements. These activities enable the robot to make decisions.
They can also scan the screen of the machine for UI elements which appear at random positions and return UI Element variables that have the clipping region set to the found element.
Image recognition activities have an Accuracy parameter, which states whether the images must match 100% or less to register as found. This feature is useful if the graphical elements you are searching for may be slightly different.

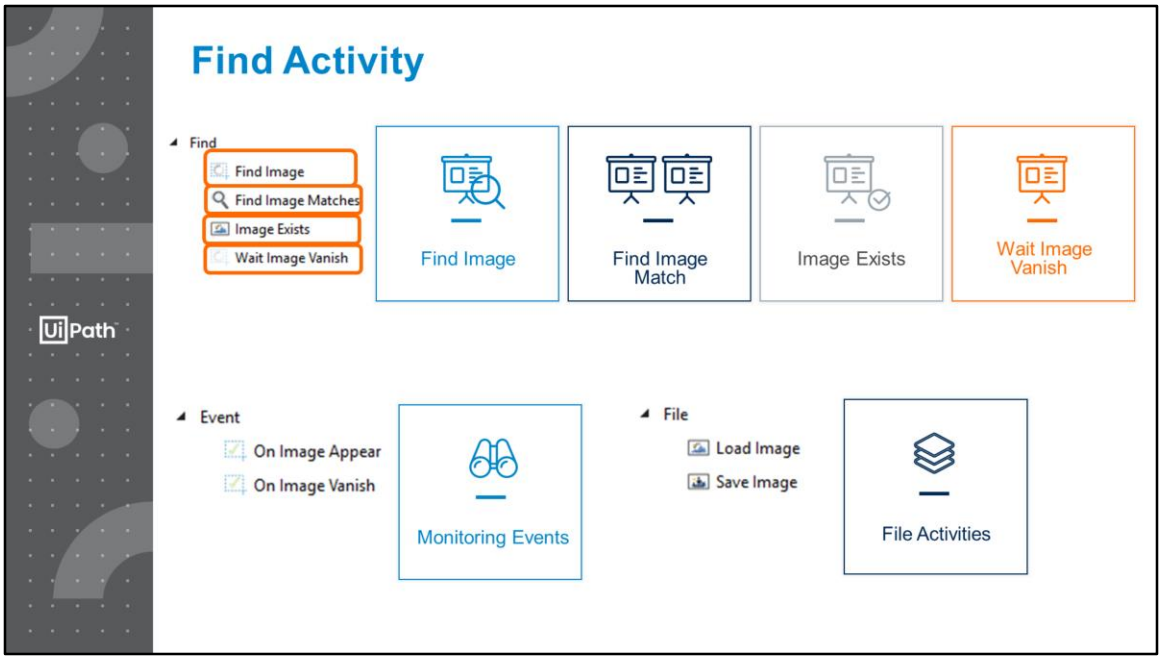
These activities use images to identify the UI Elements. After an image is specified, the activity scans the screen for a given element and either clicks or hovers over it.
Such activities are fast and reliable, but sensitive to graphical variations, as they can fail if colors or background details change.

Click, Double Click and Hover are activities that simulate the clicking or hovering of UI elements. These activities are very useful in

situations where human behavior must be mimicked.



An example where Image Based Automation can be deployed would be in the case of a customer who has disabled scripting for his SAP software processes due to security reasons. SAP stands for Systems, Applications and Products, and it's used by a wide variety of businesses.
When Scripting is not enabled, UiPath is not able to capture the selectors from SAP. In scenarios such as these, Image Based Automation can be easily deployed since it will rely on Image and text recognition and not on scripting.



Find Image is an activity in the UiPath Studio which waits for specific UI components to appear. It presents the model of UI element image by the user to be searched. Once the UI component appears, the image activity with the UI Element variable and clipping image set region. The image activity is a useful tool that identifies the UI element, variable and component in the virtual machine. It provides the best decision choosing option that image is displayed or not apart from this can it managed manifest actions. It enables to take and make decisions in image activities. It uses Retry Scope Activity to handle the condition of image activity. The image activity consists of:

- Find Image
- Find Image Matches
- Image Exist
- Wait Image Vanish

A. Find Image:

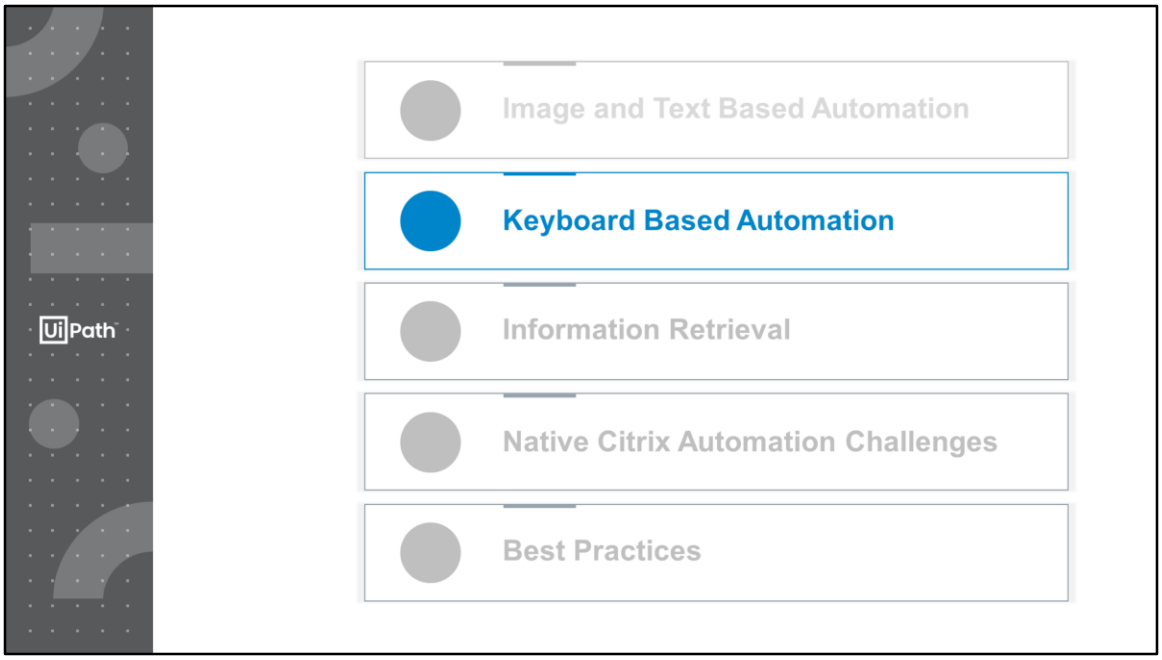
This activity is a major part of Image Activities that waits for the images to appear in UiElements. When this process is completed, it

changes the UiElement image with the clipping region set. After that, it gives an element along with UiElements.
Source code: *UiPath.Core.Activities.WaitImageAppear*

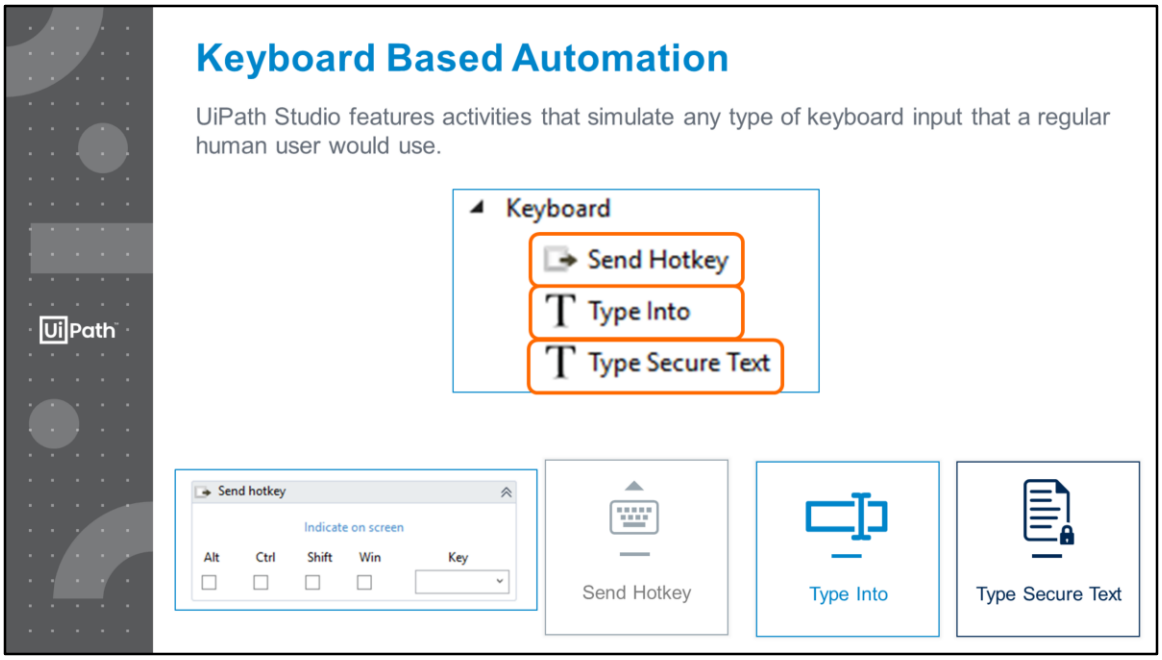
B. Find Image Matches
It searches and matches a particular image in a UiElement and delivers the collection of UiElement. In the collection, it keeps the clipping region set of matching screens.
Source Code: *UiPath.Core.Activities.FindImageMatches*

C. Images Exist
It specifies any images found with individual UiElement.
Source Code: *UiPath.Core.Activities.ImageFound.*

D. Wait Image Vanish
It waits for the image when it disappears from UiElement.
Source Code: *UiPath.Core.Activities.WaitImageVanish*



In this topic, we will learn about Keyboard based automation.



Keyboard Based Automation
The Keyboard based automation is a part of UIActivities. It processes Ui automation with the help of multiple activities. This activity passes the UIElement variable by input text. It identifies the action that we want to automate. In this process, the target can automatically create screen functionality that identifies the UIElements in a proper region and produces selectors for them. The following types of selectors make the keyboard-based automation easier:

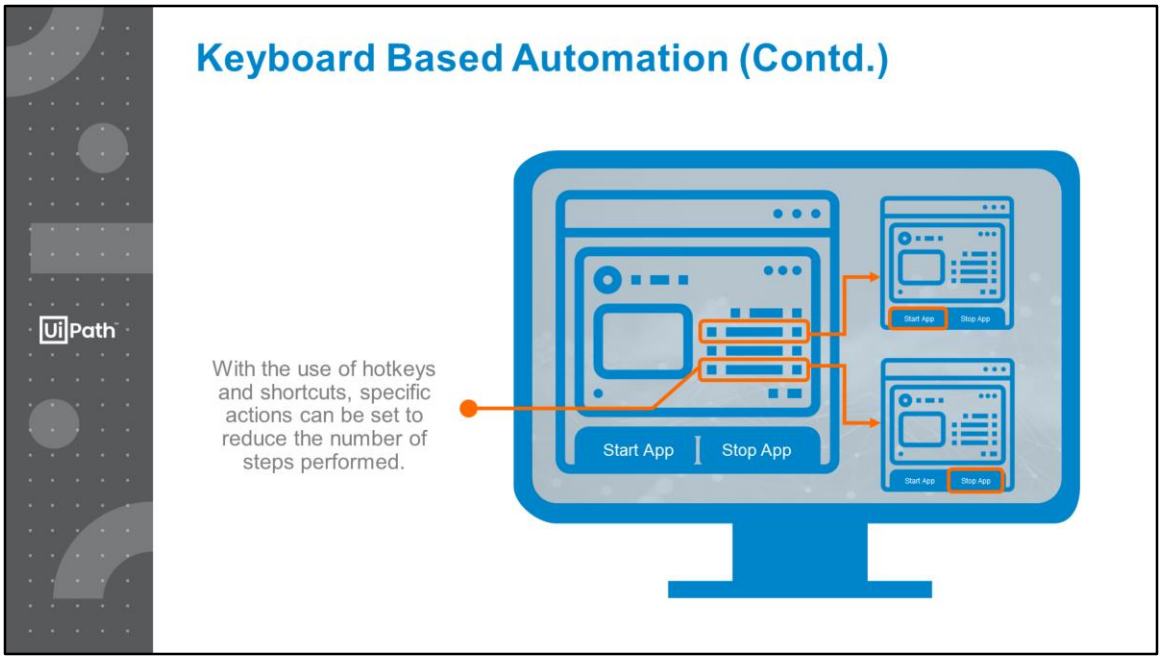
A. Send HotKey
B. Type Into
C. Type Secure Text

A. Send HotKey
This activity is used to access the UIElement application shortcut and to simplify the automation project. For example, you can replace multiple activities that perform Ui automation with the help of keyboard shortcuts. These shortcuts contain a longer path to

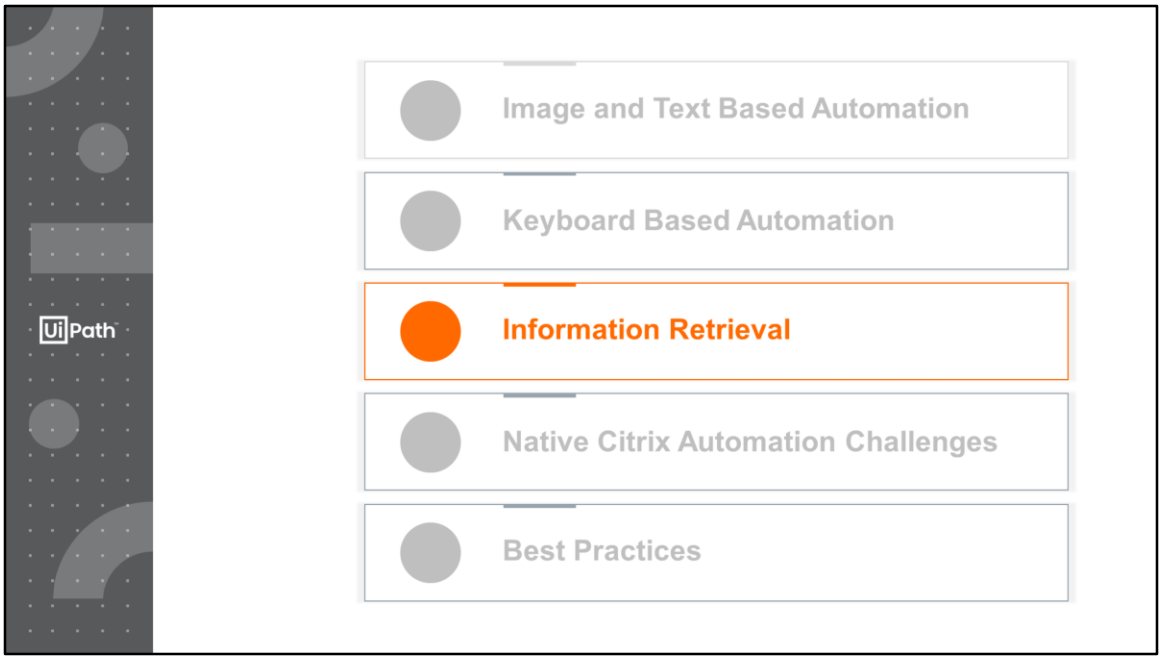
proceed with the automation process.

B. Type Into
This activity sends the keystrokes to UIElement. It supports the select and drop-down list of variables and components.

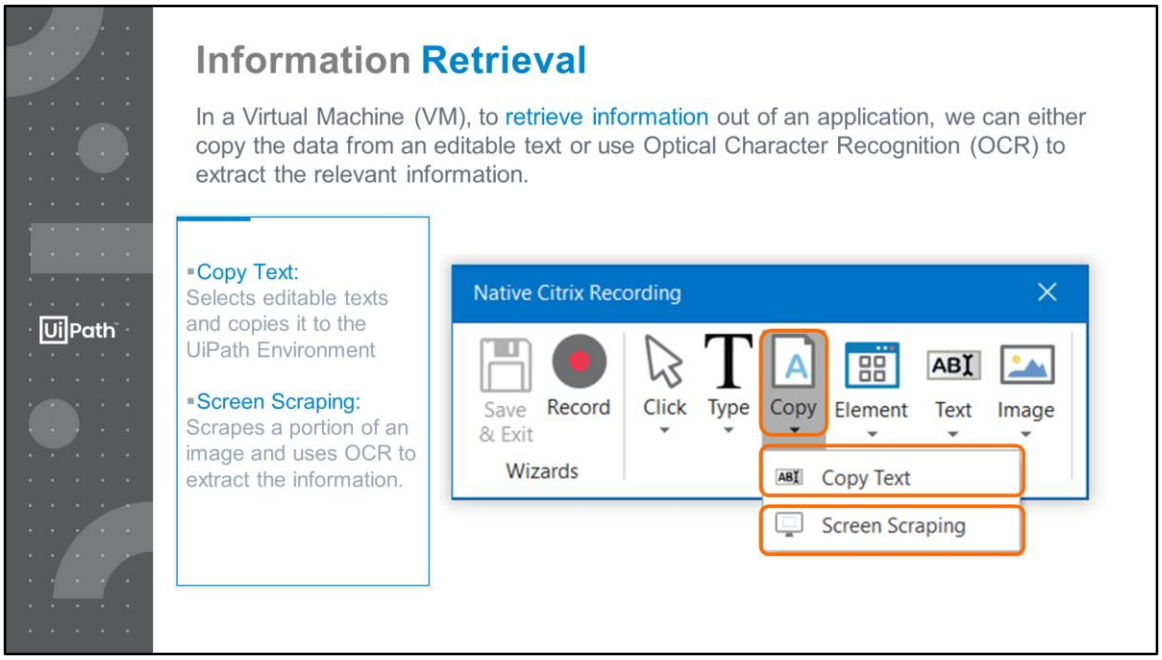
C. Type Secure Text
Type Secure Text activity sends a secure string to a Ui element which is useful for secure automation. It uses passwords that are stored in SecureString.



The Keyboard based automation is the main activity that can be deployed in a situation just like copy (Ctrl + C) or paste (Ctrl + V). Windows applications generally have shortcuts to navigate to different screens. These shortcuts reduce the number of steps to be performed in the UI leading to reduced average error handling, faster process and a bug free source on the execution time. The Keyboard based automation can be used to send shortcut keys to an application.



In this topic, we will learn about Information retrieval.



Information Retrieval

There are multiple processes involved in Retrieving Information:

A. Copy Text

Copy Text, as the name implies, selects editable texts and copies it to the UiPath environment through the clipboard. All action is performed on the active text field.

There are two activities that get generated by the wizard

Into (this is the command used to select the text and copy , this involve selecting the text activity from the target UI and copying to the UiPath (Here the copied text appear as text variable).

B. Screen Scraping

Screen Scraping allows you to scrape just a portion of an image relative to an anchor. It locates a fixed element on the screen and then using OCR extracts the information. The activities generated by the wizard are:

- Find Image (to find the anchor image)
- Set Clipping Region (to find the clipping region, offset to the

anchor image)

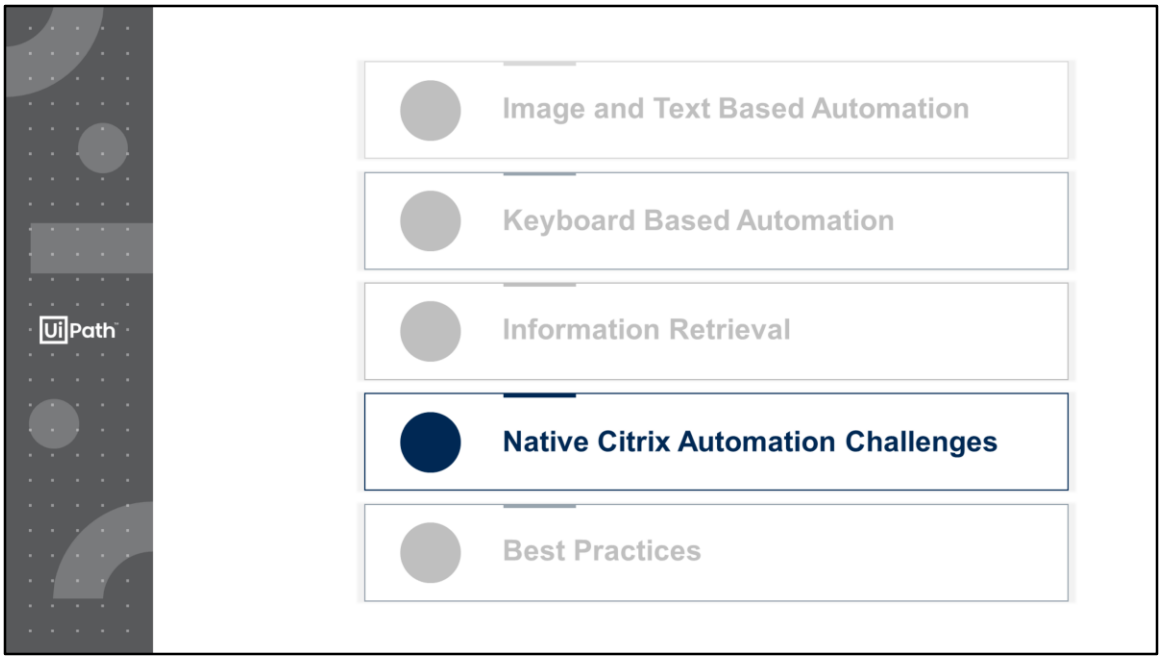
- Get OCR text (to extract the data from the clipping region).

Finally, the clipping region is reset to avoid interference with other operations.

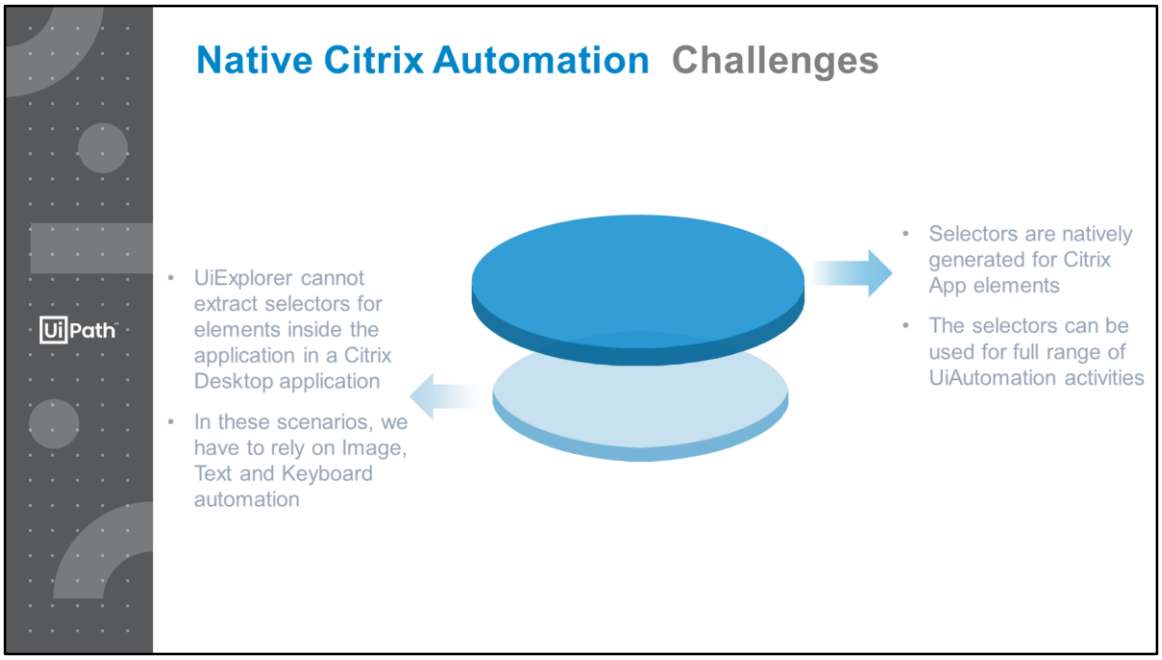
For example:

Using OCR, we can deploy information retrieval for customers that handle a large flow of documents such as Purchase Orders (PO). Or, we can manually scan them and upload them on SAP.

We can scan Each PO out of millions of documents and extract the details like Invoice Number, Date, Tax, Total, and so on. The information can then be organized in specific fields for further storage and handling.

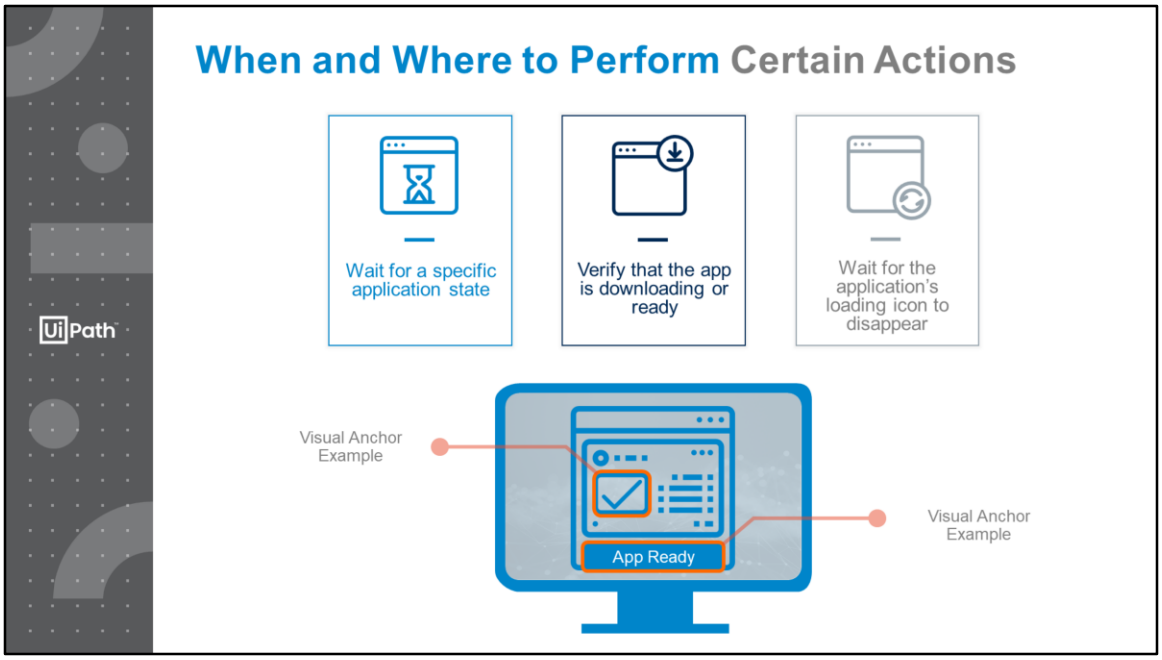


In this topic, we will learn about Native Citrix Automation Challenges.



UiPath v2018.4.1 onwards has native support for Citrix Virtual Apps (formerly known as XenApp). By installing the Citrix Extension on the client machine and the UiPath Remote Runtime component. (UiPathRemoteRuntime.msi) on the Citrix Virtual Apps application server, we can automate Citrix Apps as if they were local applications. Selectors are natively generated for Citrix App elements, and we can use the full range of UIAutomation activities, such as Click, Type Into, Get Text, Extract Data, and more.

However, this is not the case for Citrix Virtual Desktop (XenDesktop), UIExplorer is unable to extract selectors for elements inside the applications in a Citrix Desktop application. Same is the case with Citrix Virtual Apps where Citrix Extensions are not enabled. In these scenarios, we have to rely on Image, Text and Keyboard automation.



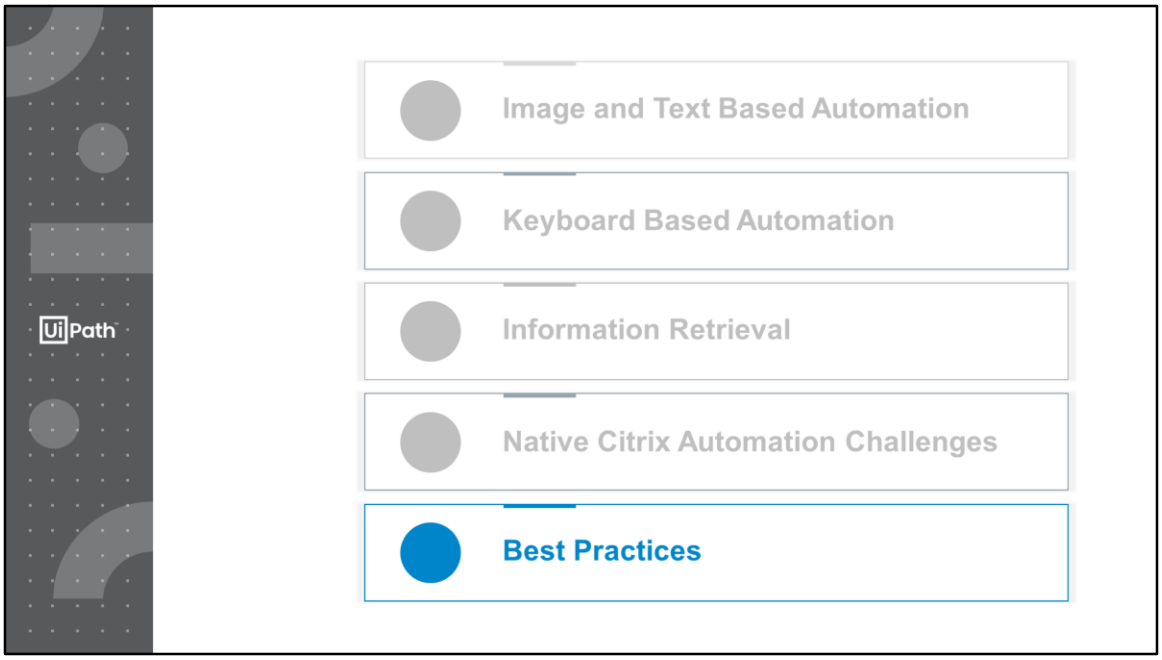
When and Where to Perform Certain Actions
The concept of automation can be fully utilized in cases where the errors in the automating project are minimal.
For example, when waiting for a particular state of an application, it is essential to create optimal automation that waits for the application or a web page to load or for a specific process to end.
At this point, we look and wait for the visual element to appear by keeping the On Image Appear on it.
A better and more general solution is to wait for the application's loading icon to disappear. An On Image Vanish activity can be used for this purpose, allowing the automation to continue only when the loading icon vanishes.

Certain activities expect control to be already present in the normal field. Sending data to the application or extracting data from the application are some of such scenarios where we can get data by using click activities like Click Image and Click Text. Apart from this, You can also click on an anchor image relative to the text. You can also achieve this by using keyboard activities like sending tab keys.

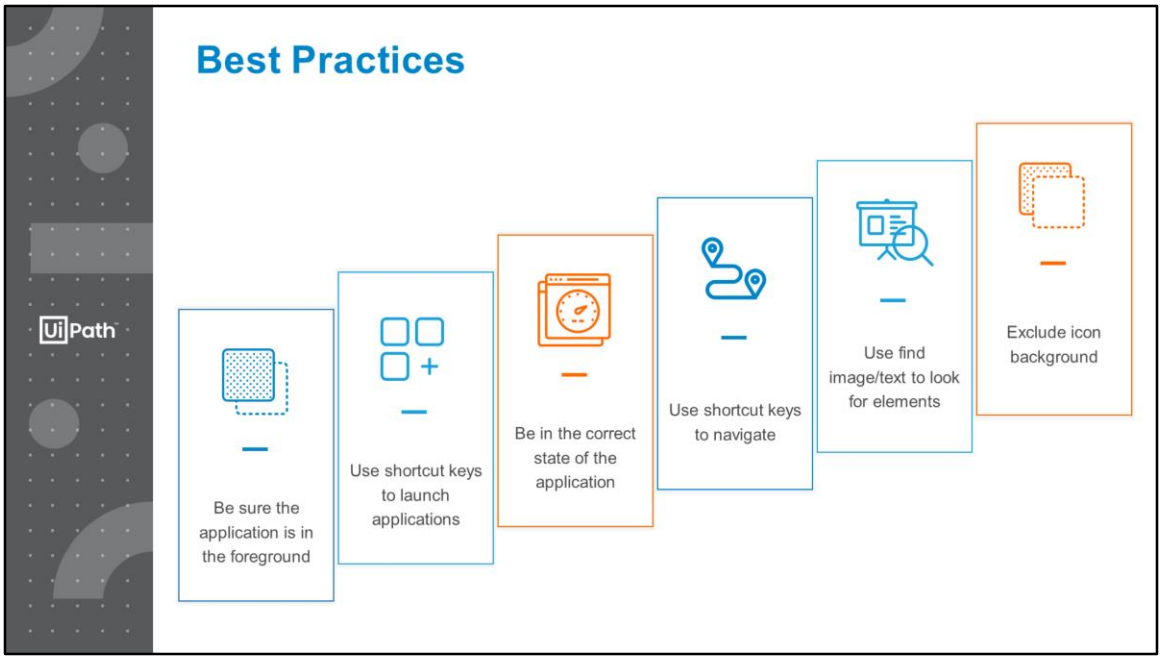
Identifying Elements

Virtual environments do not provide any way to identify Ui elements through standard means. Therefore, visual anchors are only the options. You can use the image or the extracted text to compare with the expected result.

UiPath Studio features activities that use OCR or Image Recognition technologies in such situations. Several OCR engines can be used with UiPath Studio like Google Tesseract, Microsoft MODI, and Abbyy, where Google Tesseract engine works better for scraping smaller areas, while Microsoft MODI is more suitable for larger ones.



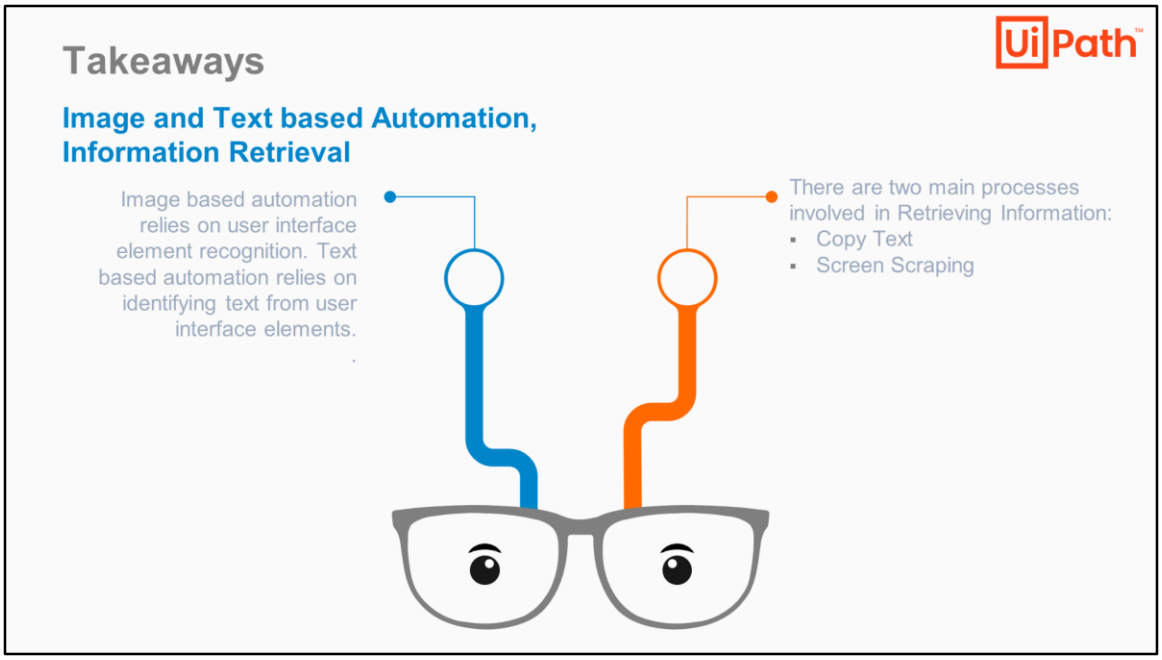
In this topic, we will learn about the importance of Best Practices.



Usually, apps are opened by clicking their shortcut or executable file. The location of these files can normally be identified by several means, such as screen coordinates or selectors. In virtualized environments the locations of shortcuts or executables are unavailable, so, Image and OCR activities must be used to identify the location of the shortcut or executable file.
These activities are based on image and text recognition, so, slight graphical differences, such as changes in resolution or highlighting the icon, can cause the identification of the shortcut to fail. A solution for this issue is to select an area of the icon that does not include any portion of the background image, such as the center area of the icon.

The Best way to open an application in the virtual environment is to create a shortcut for the respective applications. This can be further simplified by assigning the hotkey. It should be made by using a complex key for the combination. By the help of the command prompt, you can start the apps in virtual environments. For example, we can send the path of the application to the Command

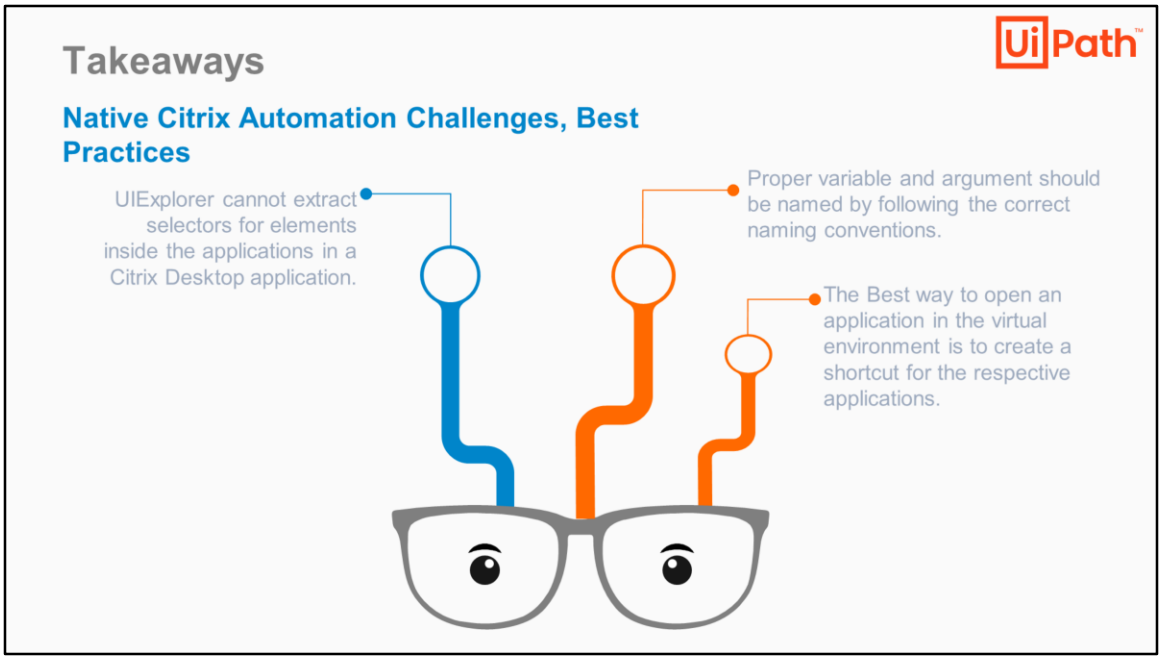
Prompt terminal with the Send Hotkey and Type Into activities. This method also enables you to input arguments for the app to be opened.



The point of the Recap & Summary section is to go through the most important points covered in the lesson, after the students had the chance to see them in practice and obtain a consolidated view.

The teacher should use facilitation questions to help the students map the key points and offer a safe space to get questions and comments from them.

Some examples of facilitation questions
1. What is screen scrapping?
2. What is Find Image activity?
3. Name the processes involved in Retrieving Information.



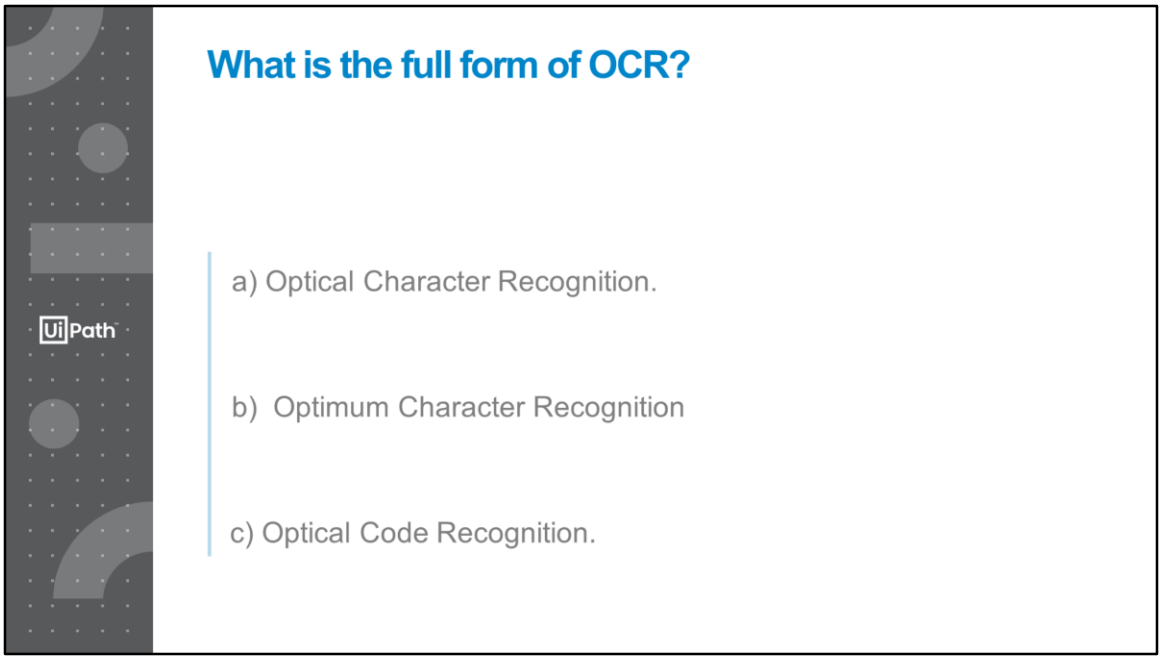
The point of the Recap & Summary section is to go through the most important points covered in the lesson, after the students had the chance to see them in practice and obtain a consolidated view.

The teacher should use facilitation questions to help the students map the key points and offer a safe space to get questions and comments from them.

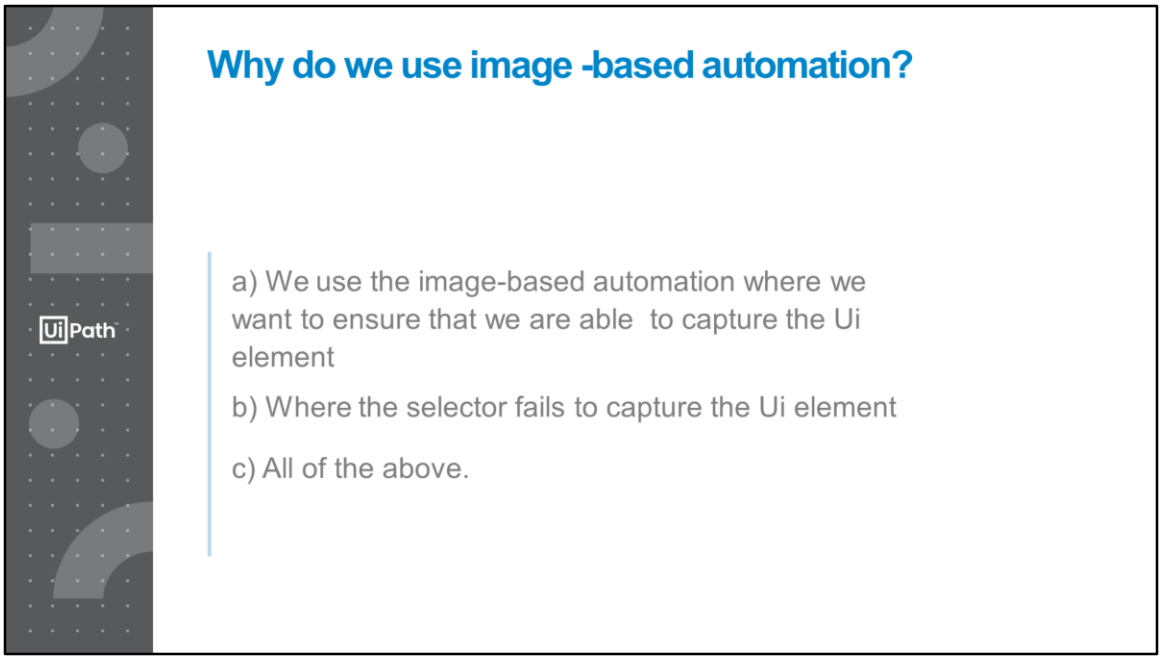
Some examples of facilitation questions

1. How do we automate Citrix Apps as if they were local applications?
2. List some best practices that should be followed in automation.

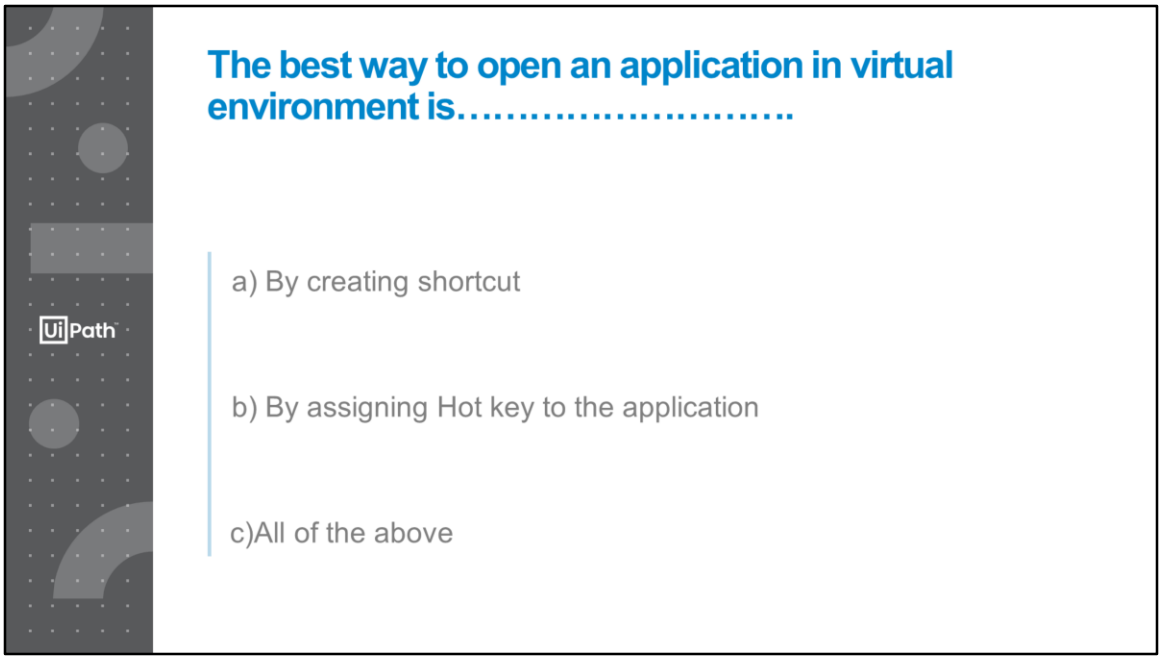
Q&A
Now it’s your turn. What’s on your mind at the end of this?



Correct answer: a) The full form of OCR is Optical Character Recognition



Correct answer: c) All of the above



Correct answer: c) All of the above

Next Steps

Module 4 Lesson 2: Excel Data Tables & PDF