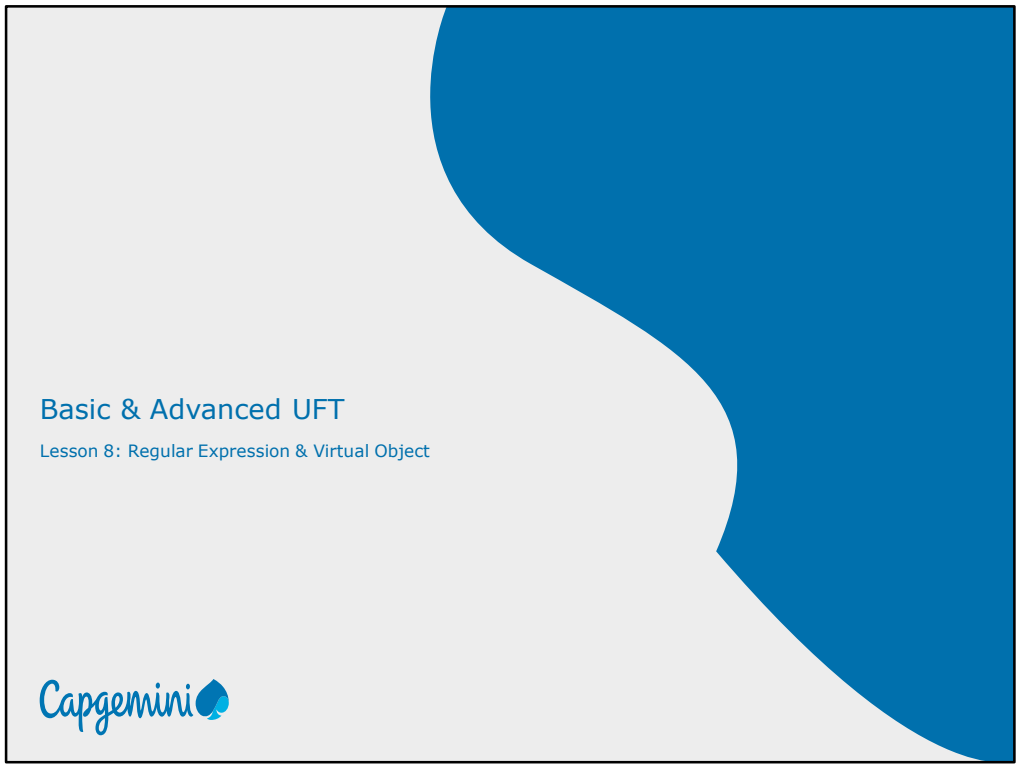

Basic & Advanced UFT
Lesson 8: Regular Expression & Virtual Object
Capgemini

Lesson Objectives

- Regular Expressions
- Use of Regular Expressions
- Regular Expressions – Object Repository
- Regular Expressions Characters
- Virtual Object
- Defining Virtual Object
- Removing Virtual Objects
- Disabling Virtual Objects
- Virtual Objects Limitations

8.1: Introduction to Regular Expression in UFT

Regular Expressions

- Regular Expressions can be used to identify the objects in the application with varying values or names or titles
- Regular expressions can be added by,
  - Defining the property values of an object in dialog boxes or in programmatic descriptions
  - Parameterize a step
  - Creating checkpoints with varying values
- A Regular Expression is a pattern of text that consists of
  - Alphabets - letters a through z
  - Special characters known as Metacharacters
  - Numbers
  - The pattern describes one or more strings to match when searching a body of text
  - The Regular Expression serves as a template for matching a character pattern to the string being searched

Regular expressions enable UFT to identify objects and text strings **with varying values**.
Regular expressions are used **when**:
- Defining the property values of an object
- Parameterizing a step
- Creating checkpoints with varying values

A regular expression is a string that specifies a complex search phrase.
Regular expressions can also be used if you **want to create a text checkpoint on a date text string**, but the displayed date changes according to the current date.

8.1: Introduction to Regular Expression in UFT

Use of Regular Expressions

- It is used to identify "objects" & "text strings" with varying values
- Use Regular Expressions only for values of type string
- Use Regular Expressions for Property Values
- Using Regular Expressions for Checkpoints

- **For example**, if you want to create a text checkpoint on a date text string, but the displayed date changes according to the current date. If we define the date as a regular expression, the checkpoint checks that the captured text string matches the expected date format, rather than checking the exact date value.

8.2: Regular Expression & Object Repository

Regular Expressions – Object Repository

- To handle windows with varying titles
- The Fax Order screen in sample Fight Application is an example for a window with varying title.
- Insert an order and playback the script.
- Follow the steps given below:

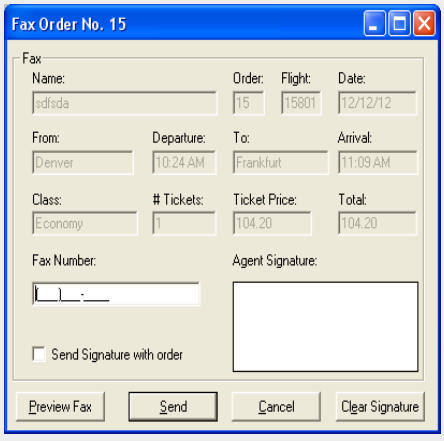
1. Start Recording
2. Insert an Order
3. Open the Fax order – File - > Fax Order
4. Close the fax order window
5. Close the AUT
6. Stop the recording
7. Run the test

8.2: Regular Expression & Object Repository

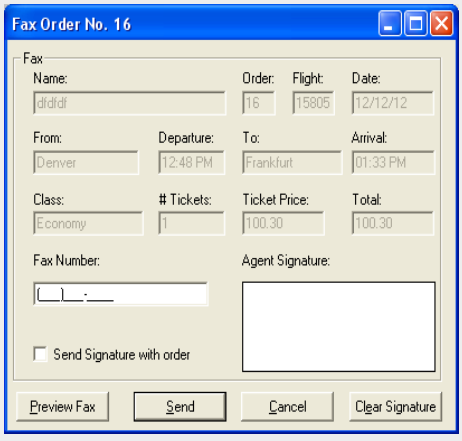
Regular Expressions – Object Repository (Cont.)

The fax order window name changes whenever a new order is inserted.

During Recording



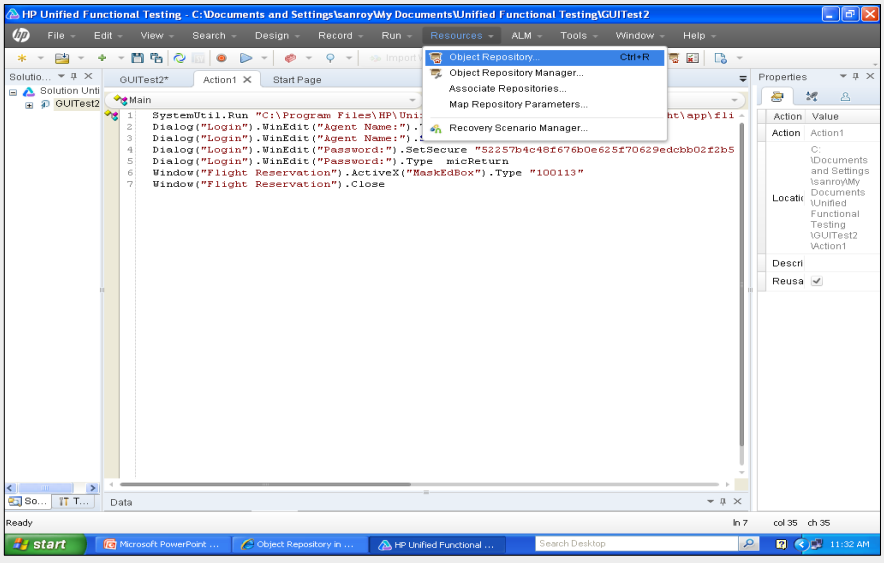
During Playback



8.3: Implementing Regular Expression

To add Regular Expression

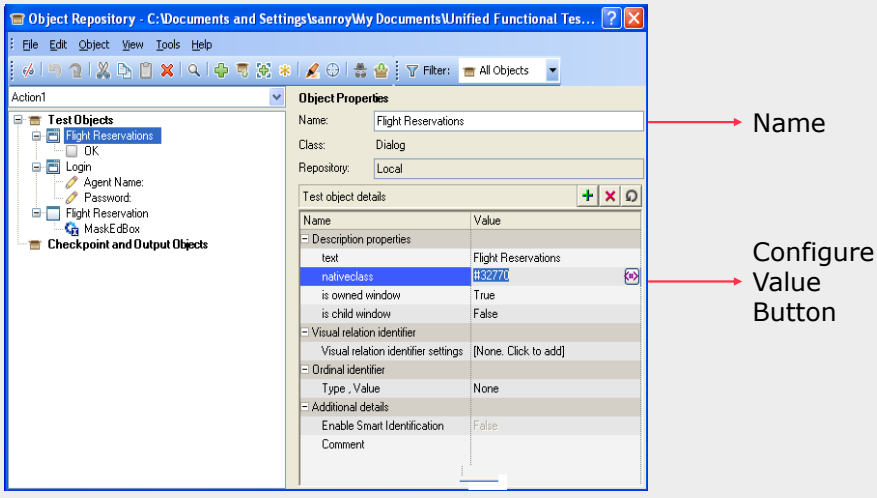
Resources – Object Repository



8.3: Implementing Regular Expression

Regular Expression and Object Repository

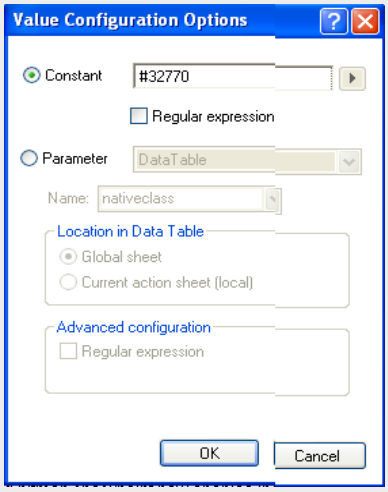
To handle this change the Flight Reservations window name in Object Repository



8.3: Implementing Regular Expression

Regular Expression and Object Repository (Cont.)

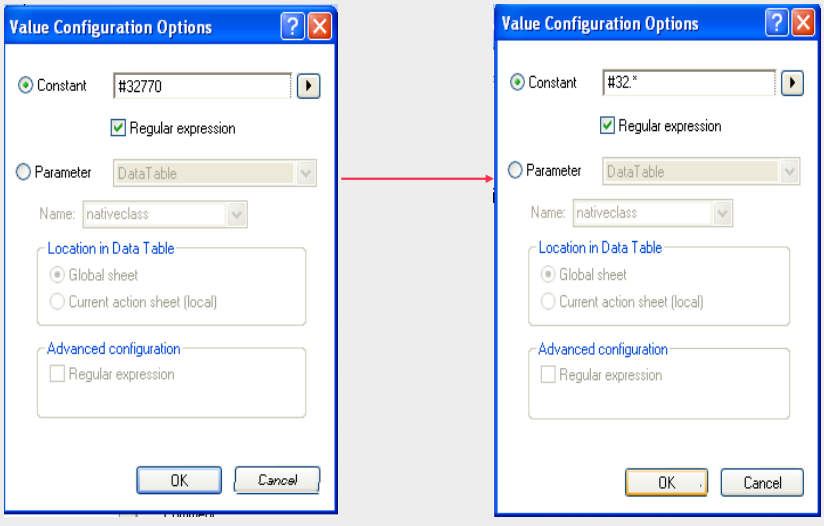
- Select Description Properties – native class
- Click the Configure value button next to the value
- The following window will be displayed



8.3: Implementing Regular Expression

Regular Expression and Object Repository (Cont.)

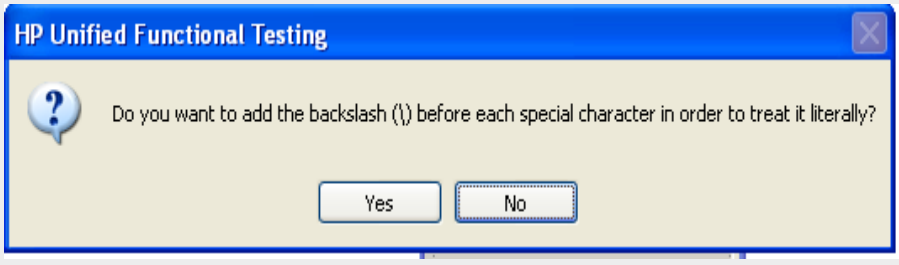
Change the name of the window as Flight Reservations
Click the Regular Expression Check box



8.3: Implementing Regular Expression

Regular Expression and Object Repository (Cont.)

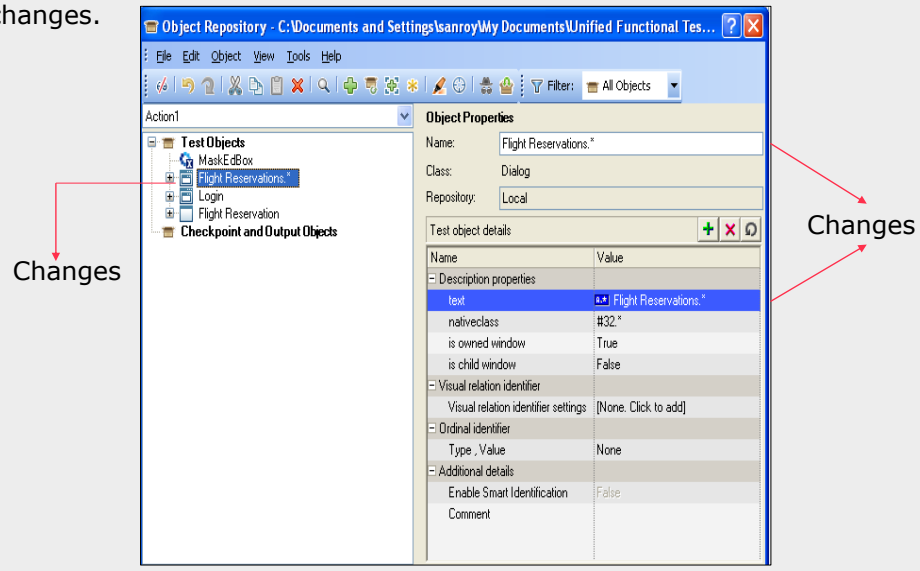
Click No in the msgbox displayed after closing the value configuration screen.



8.3: Implementing Regular Expression

Regular Expression and Object Repository (Cont.)

The Object Repository should look like this after making the necessary changes.



8.4: Regular Expression Characters

Regular Expression Characters

- Using the Backslash Character (\ ) : eg. \n is recognized as special newline character
- Matching Any Single Character (.)
- Matching Any Single Character in a List ([xy])
- Matching Any Single Character Not in a List ([^xy])
- Matching Any Single Character within a Range ([x-y])
- Matching Zero or More Specific Characters (*).For example,"zo*"
- matches either "z" or "zoo".
- Matching One or More Specific Characters (+).For example, "zo+" matches "zoo" but not "z".

8.4: Regular Expression Characters

Regular Expression Characters (Cont.)

- Matching Zero or One Specific Character (?).For ex. "a?ve?" matches the "ve" in "never".
- Grouping Regular Expressions (( ))
- Matching One of Several Regular Expressions (|)
- Matching the Beginning of a Line (^)
- Matching the End of a Line ($)
- Matching Any Alphanumeric Character Including the Underscore (\w).Equivalent to "[A-Za-z0-9_]".
- Matching Any Non-Alphanumeric Character (\W) Equivalent to "[^A-Za-z0-9_]".

8.5: Introduction to Virtual Objects

Virtual Objects

- Virtual Objects are objects that behaves like normal objects, but are not recognized by UFT
- We can define these objects as Virtual Objects and map them to standard classes, such as a button or a check box
- A Virtual Object collection is a group of virtual objects that is stored in the Virtual Object Manager under a descriptive name

8.5: Introduction to Virtual Objects

Defining Virtual Objects

- We define a Virtual Object using the Virtual Object Wizard
- Using the Virtual Object Wizard, we can map a virtual object to a standard object class, specify the boundaries and the parent of the virtual object, and assign it a name
- Only those objects can be defined as Virtual Objects on which we can click or double-click and that record a Click or DblClick step. Otherwise, the virtual object is ignored

8.5: Introduction to Virtual Objects

Steps for Creating Virtual Objects

- In UFT, choose Tools > Virtual Objects > New Virtual Object.
- Select a standard class to which you want to map your virtual object.
- Click Mark Object button. Use the crosshairs pointer to mark the area of the virtual object.
- An object in the object tree is assigned as the parent of the virtual object.
- Specify a name and a collection for the virtual object.

8.5: Introduction to Virtual Objects

Case Study of Virtual Objects

Let us consider the calculator application. Suppose we are recording the calculations that are performed in the calculator. The script will be recorded as follows.

```
Window("Calculator").WinButton("7").Click
Window("Calculator").WinButton("4").Click
Window("Calculator").WinButton("2").Click
Window("Calculator").WinButton("3").Click
Window("Calculator").WinButton("+").Click
Window("Calculator").WinButton("2").Click
Window("Calculator").WinButton("4").Click
Window("Calculator").WinButton("1").Click
Window("Calculator").WinButton("=").Click
```

The corresponding Keyword View will be as shown below.

Item	Operation	Value	Documentation
Action1			
Calculator			
7	Click		Click the "7" button.
4	Click		Click the "4" button.
2	Click		Click the "2" button.
3	Click		Click the "3" button.
+	Click		Click the "+" button.
2	Click		Click the "2" button.
4	Click		Click the "4" button.
1	Click		Click the "1" button.
=	Click		Click the "=" button.

8.5: Introduction to Virtual Objects

Case Study of Virtual Objects (Cont.)

- Suppose button 7 is made a virtual object and we give the name 7.Now, the script will be displayed as shown below. The button "7" has been assigned as an virtual object.

```
Window("Calculator_2").WinButton("7").VirtualButton("7").Click
Window("Calculator_2").Move 454,309
Window("Calculator_2").WinButton("5").Click
Window("Calculator_2").WinButton("2").Click
```

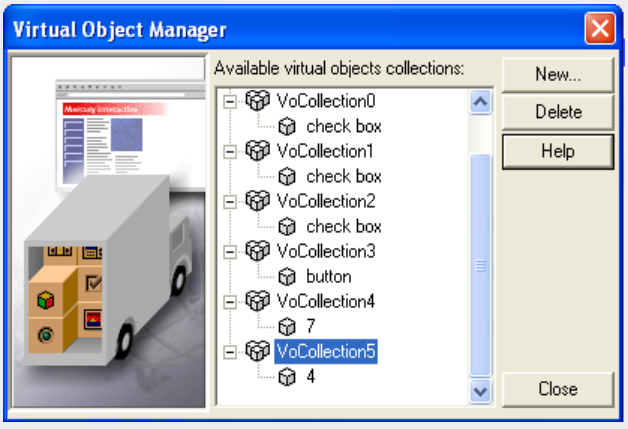
- The corresponding Keyword View will be as follows. The virtual object can be identified by the symbol 'v' attached to the object

Calculator_2	Activate		Make the "Calculator_2" window active.
7			
7	Click		Click the "7" button.
Calculator_2	Move	454,309	Move the "Calculator_2" window to the screen coordinates 454, 309.
5	Click		Click the "5" button.
2	Click		Click the "2" button.

8.5: Introduction to Virtual Objects

Removing Virtual Objects

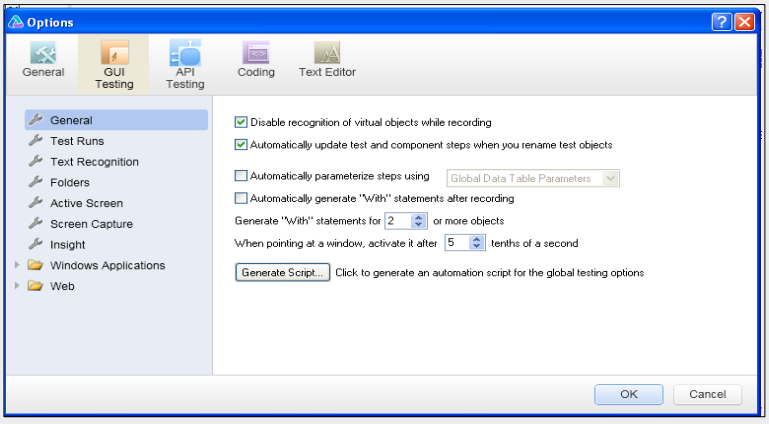
- We can remove virtual objects from were test or component by deleting them from Virtual Object Manager that can be accessed from Tools→Virtual Object→Virtual Object Manager



8.5: Introduction to Virtual Objects

Disabling Virtual Objects

- Choose Tools > Options or click the Options toolbar button. The Options dialog box opens.
- In the General tab, select the Disable recognition of Virtual Objects while recording



8.5: Introduction to Virtual Objects

Virtual Objects - Limitations

- You can define Virtual Objects only for objects on which you can click or double-click and that record a Click or DblClick step.
- You can use Virtual Objects only when recording and running a test. You cannot insert any type of checkpoint on a Virtual Object, or use the Object Spy to view its properties.
- UFT does not support Virtual Objects for analog or low-level recording.

Demo

- Demo Regular Expression - Object Repository
- Demo Virtual Object

Summary

In this lesson, you have learnt

- What is Regular Expression

Regular Expressions can be used to identify the objects in the application with varying values or names or titles

- Regular expressions can be added by,
  - Defining the property values of an object in dialog boxes or in programmatic descriptions
  - Parameterize a step
  - Creating checkpoints with varying values
- To add Regular Expression in Object Repository
- Regular Expression Character
- What is Virtual Object and when to use it

Add the notes here.