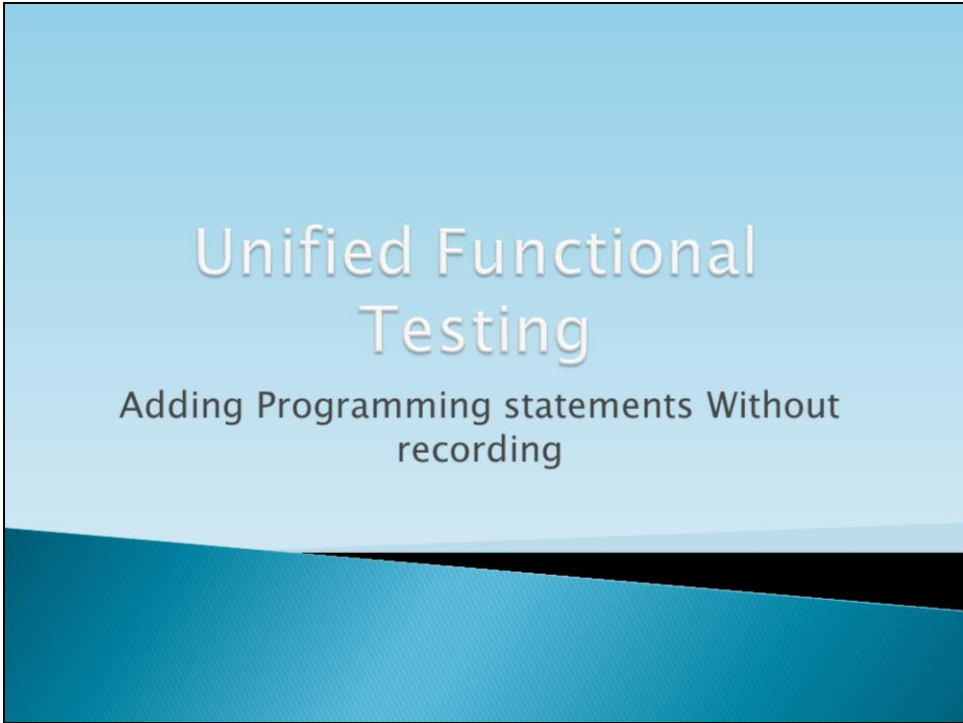# Unified Functional Testing

Adding Programming statements Without recording

Lesson Objectives

By the end of this Lesson you will be able to:

- List the types of programming statement that can be added without using Record feature.
- Use Step Generator.
- Use Report Object.
- Use Conditional Statement.
- Understand Common Methods

Topics

1. Adding statements from menu
2. Adding statements from keyword view
3. Step generator
4. Adding statements manually

Fourth Phase – Integration

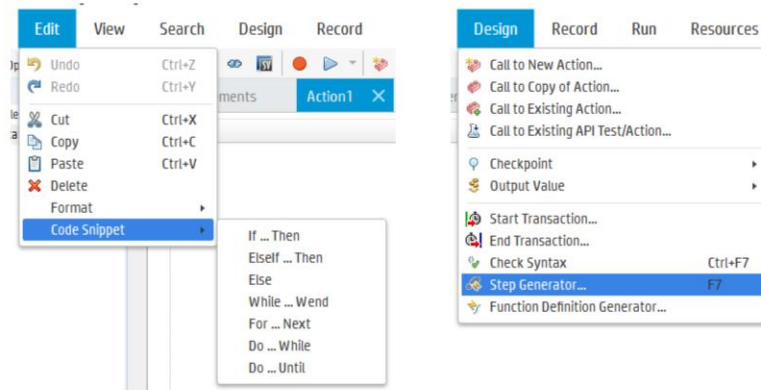| Preparation	Creation	Verification & Customization	INTEGRATION
• Document manual test steps. • Check application / environment Stability. • Check Data validation.	• Record user actions. • Confirm successful playback. • Use object repository.	• Add synchronization. • Insert checkpoint. • Data for the test. • Using Actions	• Pass data. • Build integrated Test sets.

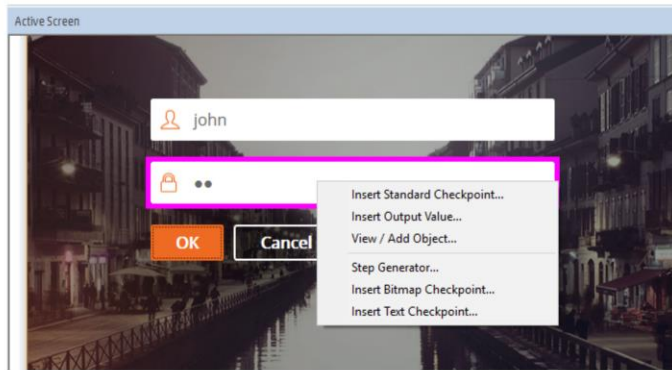## Types Of Steps That Can Be Added After Recording

- Recordable Test Object Methods
- Non Recordable Test Object Methods
- Run Time Methods
- Functions of Utility Object
- Custom Function
- VBScript Programming
- Supplemental statements
- Manual Steps

Insert Statements From Menu

You can also open the Step Generator from the Active Screen while editing.

1. Confirm that the Active Screen is displayed. If it is not, choose **View > Active Screen** or toggle the **Active Screen** toolbar button .

2. In the Keyword View or Expert View, click the step which you want the new step to follow. (When you finish defining the new step, UFT will insert it after this step.) The Active Screen displays the captured bitmap or HTML source corresponding to the selected step.

3. In the Active Screen, right-click the object for which you want to insert a step, and choose **Step Generator**. If the location you clicked is associated with more than one object, the **Object Selection - Step Generator** dialog box opens.

4. Select an object and click **OK**. The Step Generator dialog box opens and displays the object from the selected step in the **Object** box.

7

The Step Generator dialog box helps you add steps that use test object methods, Utility object methods, and function calls, so that you do not need to memorize syntax or to be proficient in high-level VBScript. You can use the Step Generator from the Keyword View and also from the Expert View.

In the **Category** list box, you can choose one of the following options:

- TEST OBJECTS: Displays contents of Object Repository.

- UTILITY OBJECTS: Displays predefined list of utility objects likes DataTable, SystemUtil, QCUtil .

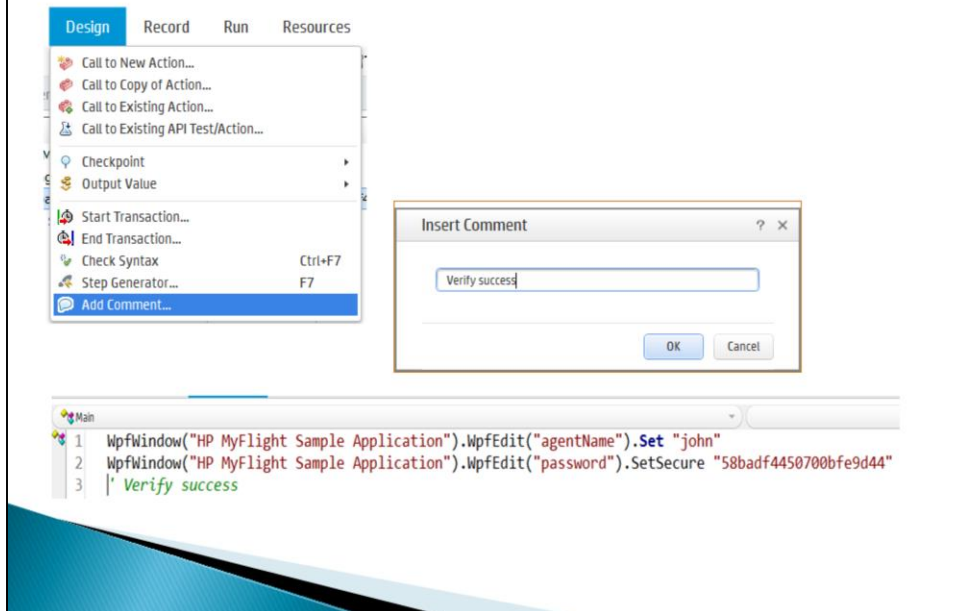- FUNCTIONS: Displays all the functions including built-in, library and local script functions.

Operation : Specifies available options based on the category selected.

Arguments : Specifies a value set for an argument. The arguments may be optional depending on the object and operation being used.

Return Value : Where to restore the return value

Generate Step : Displays the defined statement for the step.
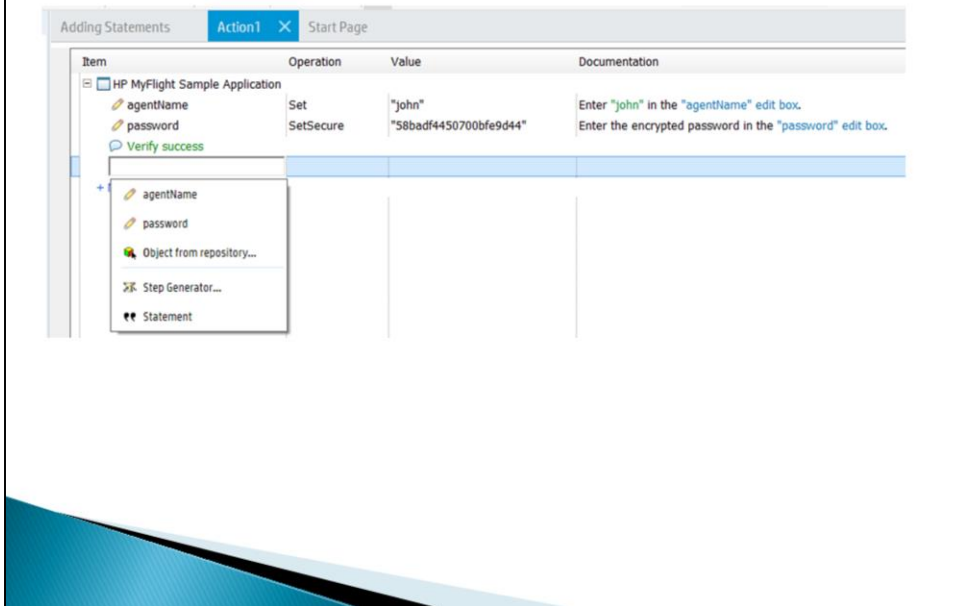
You can use comments to explain sections of your tests to improve readability and to make them easier to update.

While editing your test, you can add comments in the Keyword View ,These are also displayed as inline comments in the Expert View.
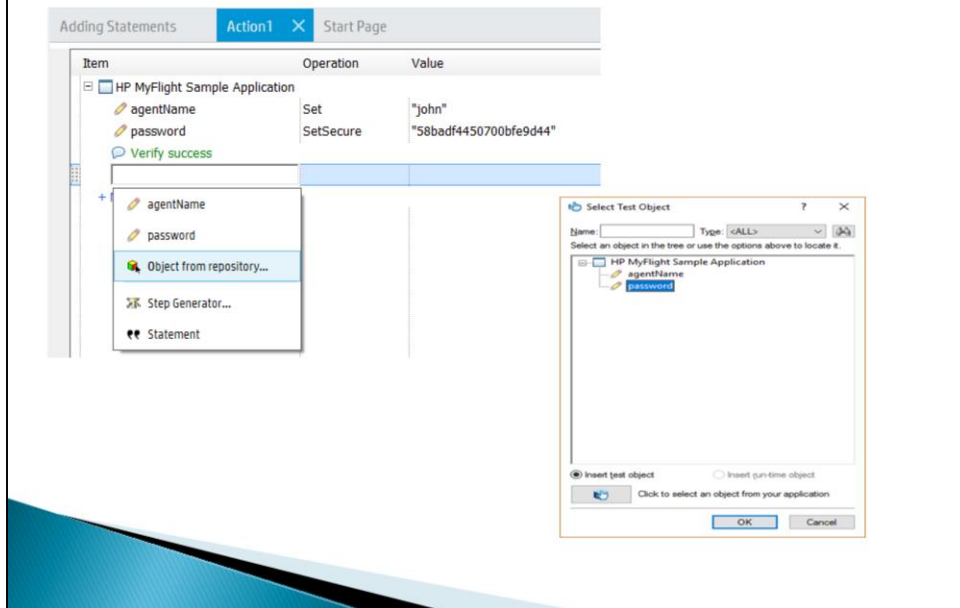
Adding Statement in Keyword View

To add a conditional step to a test in KEYWORD VIEW:

1. From the UFT menu bar, select **EDIT→CODE SNIPPET→IF ... THEN. The STATEMENT step is added to KEYWORD VIEW.**

2. Click the STATEMENT in the IF step. The step changes to a list.

3. Select one of the following statement types from the list:

– **OBJECT FROM REPOSITORY**

– **STEP GENERATOR**

The Select Object for Step dialog box displays the object repository tree and enables you to select an object for your step from the shared object repository or from your application

Common Methods

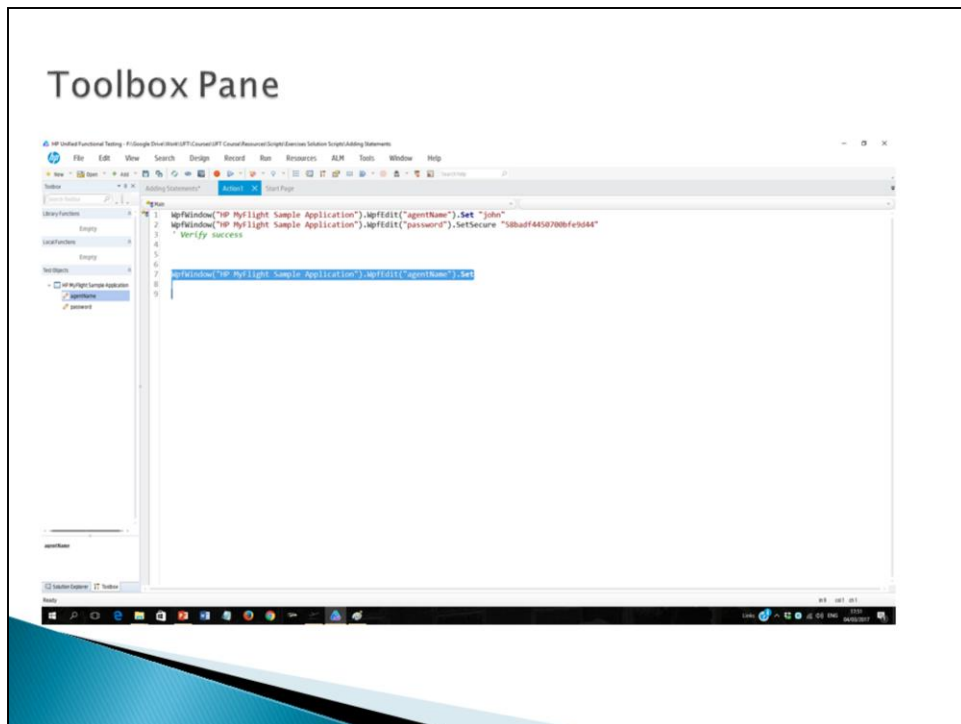| Method	Description
SET	Sets the value provided to the edit box.
SETSECURE	Decrypts a provided encrypted value.
SELECT	Selects an item from a list box or combo box.
TYPE	Enters the supplied keyboard characters into an object.
CLICK	Executes a mouse-click on an object.
CHECK and OUTPUT	Run when checkpoints or output values are inserted into a test or component. The CHECK and OUTPUT methods check or return the values of the properties defined in the checkpoint object that is passed into the method as input.
WAITPROPERTY	Pauses the test while an application process completes, before moving on to the next step.

Inserting a Step Manually

❑ You can add steps directly to tests within EXPERT VIEW.

❑ Use the following features in EXPERT VIEW to assist you:
- Statement completion
- Auto–expand VBScript syntax
- Context–sensitive help
- Available Keywords Pane

Statement Completion

Context	Example
When you type a class followed by a **parenthesis,** UFT uses an auto complete list to display the test objects in the OBJECT REPOSITORY for that class and methods for the object**.**	Window("Flight Reservation").WinButton("Button" "FLIGHT" "Insert Order"
When you type an object followed by a **dot,** UFT uses an auto complete list to display the child objects and methods for that object.	Window("Flight Reservation").Dialog("Flights Table").WinButton("OK"). CaptureBitmap CheckProperty ChildObjects Click DblClick Drag Drop Exist GetROProperty GetTextLocation
When you type a method followed by a **space,** UFT uses an auto complete list to display the arguments for that method.	Window("Flight Reservation").WinComboBox("Fly From:").Select Select (**Item**)

If you press CTRL+SPACE, UFT displays a list of the relevant methods, properties, VBScript functions, user-defined functions, VBScript constants, and utility objects that you can add.

The Toolbox pane displays the keywords available to your component. It enables you to view and drag and drop objects or calls to functions into your component. When you drag and drop an object into your component, UFT inserts a step with the default operation for that object. When you drag and drop a function into your component, UFT inserts a call to that function.

What's Next?

❑ Review Questions

❑ Exercise

❑ Next Lesson
  ▪ The next lesson in the course is:
    Parameterizing

End of Lesson