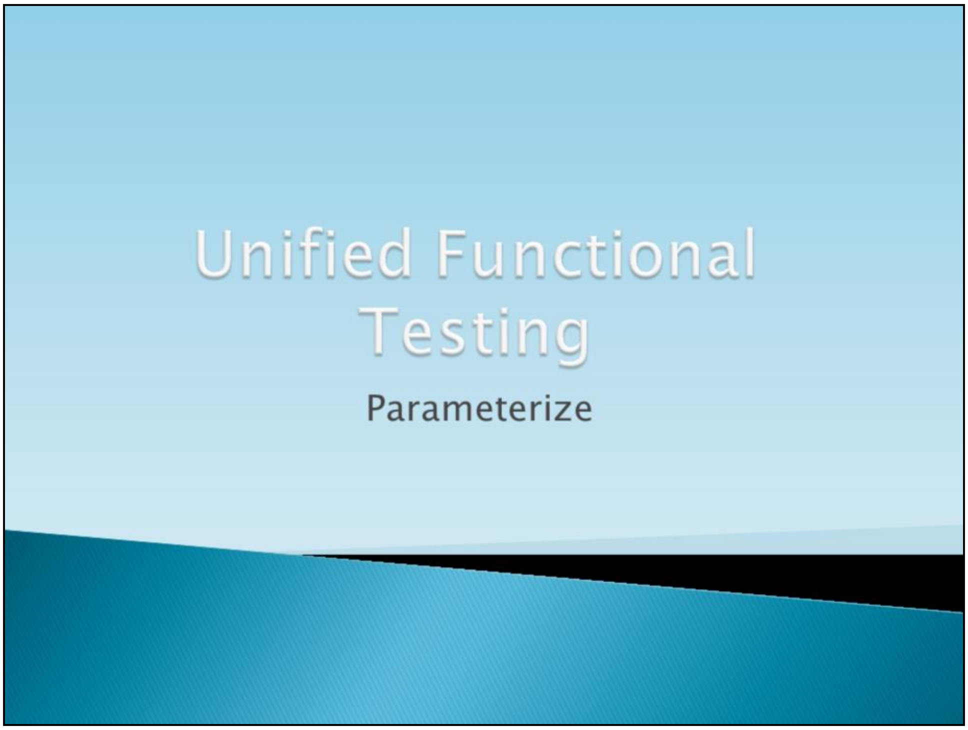
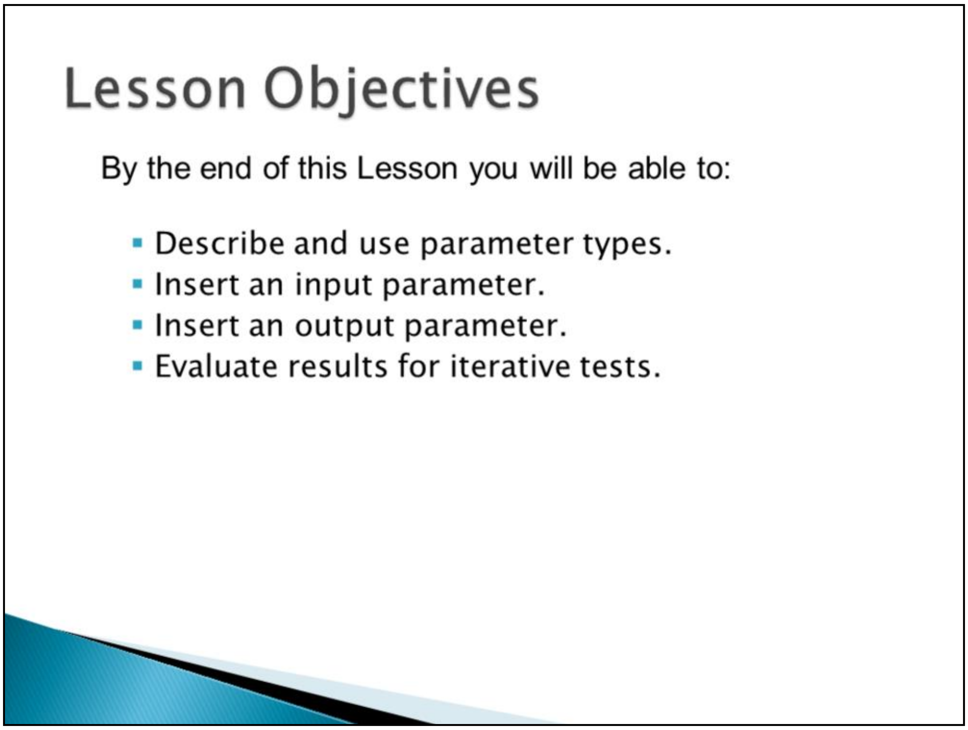


Unified Functional Testing

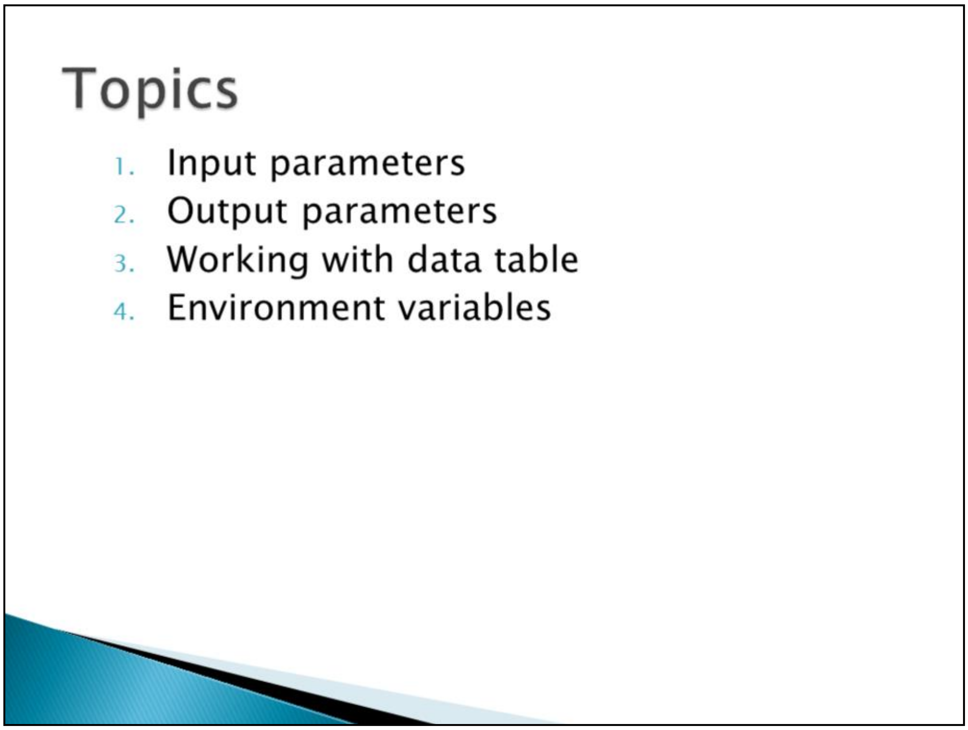
Parameterize



Lesson Objectives

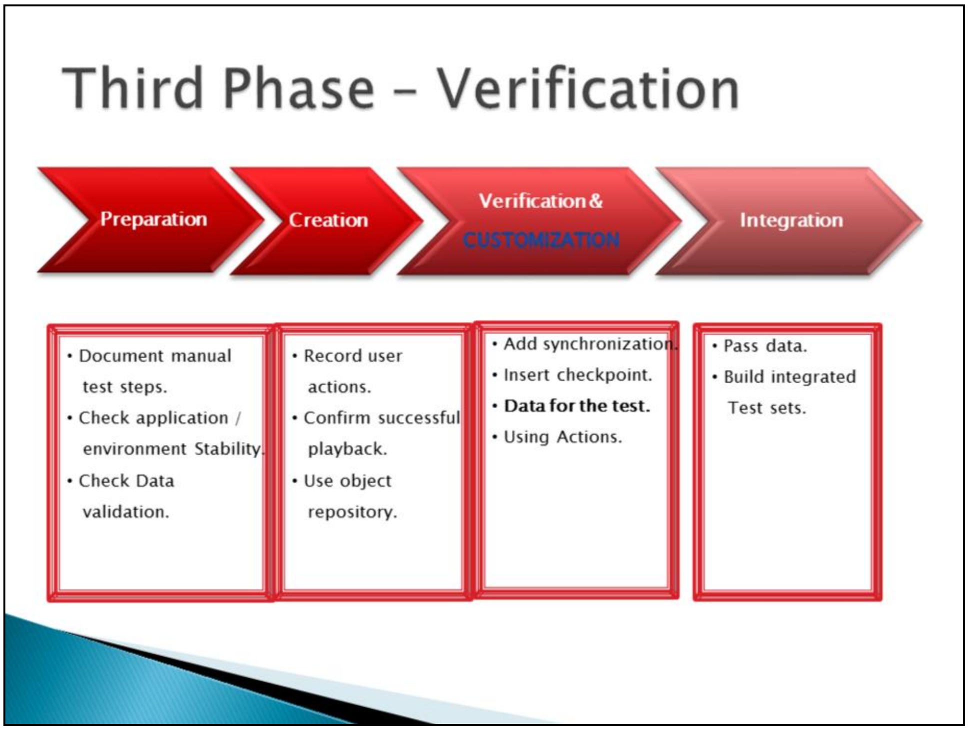
By the end of this Lesson you will be able to:

- Describe and use parameter types.
- Insert an input parameter.
- Insert an output parameter.
- Evaluate results for iterative tests.



Topics

1. Input parameters
2. Output parameters
3. Working with data table
4. Environment variables



Third Phase – Verification

Preparation

- Document manual test steps.
- Check application / environment Stability.
- Check Data validation.

Creation

- Record user actions.
- Confirm successful playback.
- Use object repository.

Verification & CUSTOMIZATION

- Add synchronization.
- Insert checkpoint.
- **Data for the test.**
- Using Actions.

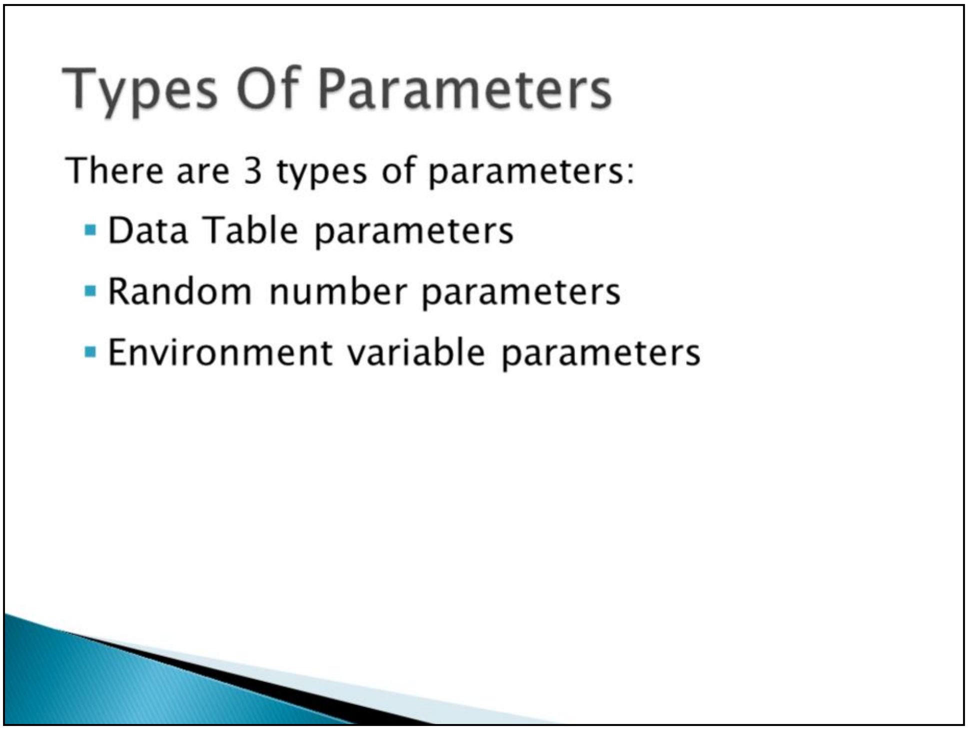
Integration

- Pass data.
- Build integrated Test sets.

Probably we will want to record test once and run it several times. For this mission we will use parameters.

A **parameter** is a variable that is assigned a value from an external data source or generator.

Parameters let us check how the application performs the same operations with multiple sets of data and make also test and data maintenance easier.



Types Of Parameters

There are 3 types of parameters:

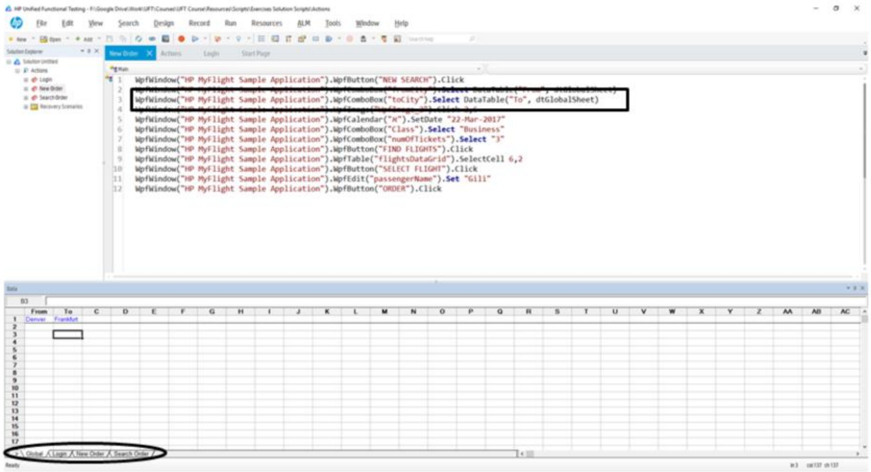
- Data Table parameters
- Random number parameters
- Environment variable parameters

Data Table parameters- Enable you to create a data-driven test that runs several times using the data you supply. In each repetition, or iteration, UFT uses a different value from the Data Table.

Environment variable parameters- Enable you to use variable values from other sources during the run session. These may be values you supply, or values that UFT generates for you based on conditions and options you choose.

Random number parameters- Enable you to insert random numbers as values in your test.

Design-Time Data Table



```
WpfWindow("HP MyFlight Sample Application").WpfButton("NEW SEARCH").Click
WpfWindow("HP MyFlight Sample Application").WpfComboBox("fromCity").Select DataTable("From", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfComboBox("toCity").Select DataTable("To", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfImage("WpfImage_27").Click 2,6
WpfWindow("HP MyFlight Sample Application").WpfCalendar("W").SetDate "22-Mar-2017"
WpfWindow("HP MyFlight Sample Application").WpfComboBox("Class").Select "Business"
WpfWindow("HP MyFlight Sample Application").WpfComboBox("numOfTickets").Select "3"
WpfWindow("HP MyFlight Sample Application").WpfButton("FIND FLIGHTS").Click
WpfWindow("HP MyFlight Sample Application").WpfTable("flightsDataGrid").SelectCell 6,2
WpfWindow("HP MyFlight Sample Application").WpfButton("SELECT FLIGHT").Click
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set "Gili"
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click
```

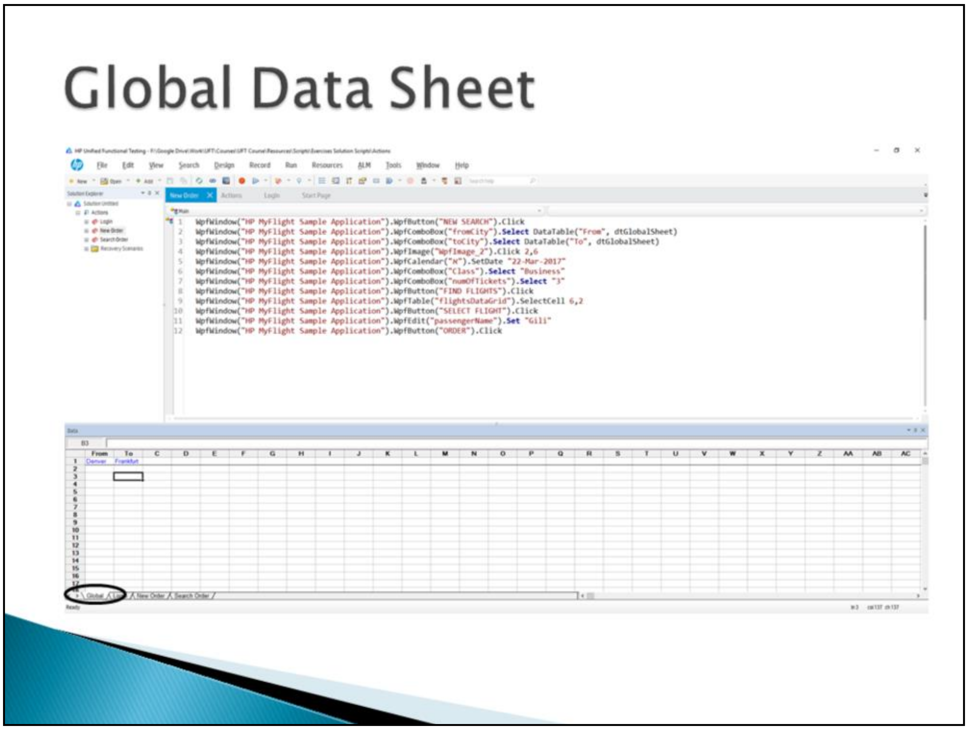
	From	To
1	Denver	Frankfurt

The data your test uses is stored in the **design-time** Data Table, which is displayed in the Data Table pane at the bottom of the screen while you insert and edit steps.

The Data Table has the characteristics of a Microsoft Excel spreadsheet, meaning that you can store and use data in its cells and you can also perform mathematical formulas within the cells.

DATA TABLE stores data either in:

- **GLOBAL** sheet: Provides a source for values that can be used by any action in the test.
- **ACTION** sheet: Provides a source for values specific to an action.



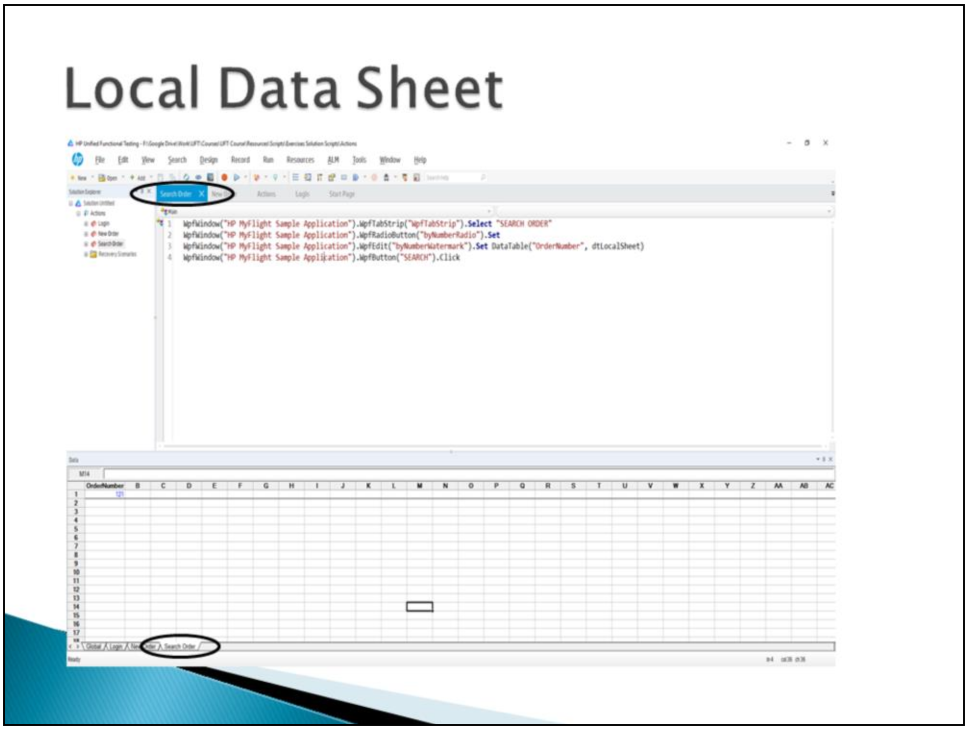
Global Data Sheet

```
WpfWindow("HP MyFlight Sample Application").WpfButton("NEW SEARCH").Click
WpfWindow("HP MyFlight Sample Application").WpfComboBox("fromCity").Select DataTable("From", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfComboBox("toCity").Select DataTable("To", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfImage("WpfImage_2").Click 2,6
WpfWindow("HP MyFlight Sample Application").WpfCalendar("N").SetDate "22-Mar-2017"
WpfWindow("HP MyFlight Sample Application").WpfComboBox("Class").Select "Business"
WpfWindow("HP MyFlight Sample Application").WpfComboBox("numOfTickets").Select "3"
WpfWindow("HP MyFlight Sample Application").WpfButton("FIND FLIGHTS").Click
WpfWindow("HP MyFlight Sample Application").WpfTable("flightsDataGrid").SelectCell 6,2
WpfWindow("HP MyFlight Sample Application").WpfButton("SELECT FLIGHT").Click
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set "Gili"
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click
```

	From	To
1	Denver	Frankfurt

The global data sheet provides a source of values that can be used by any action in the test.

When you store the input data in a global data sheet, the complete test iterates for each row of data. Default iteration settings apply to all iterations in the test.



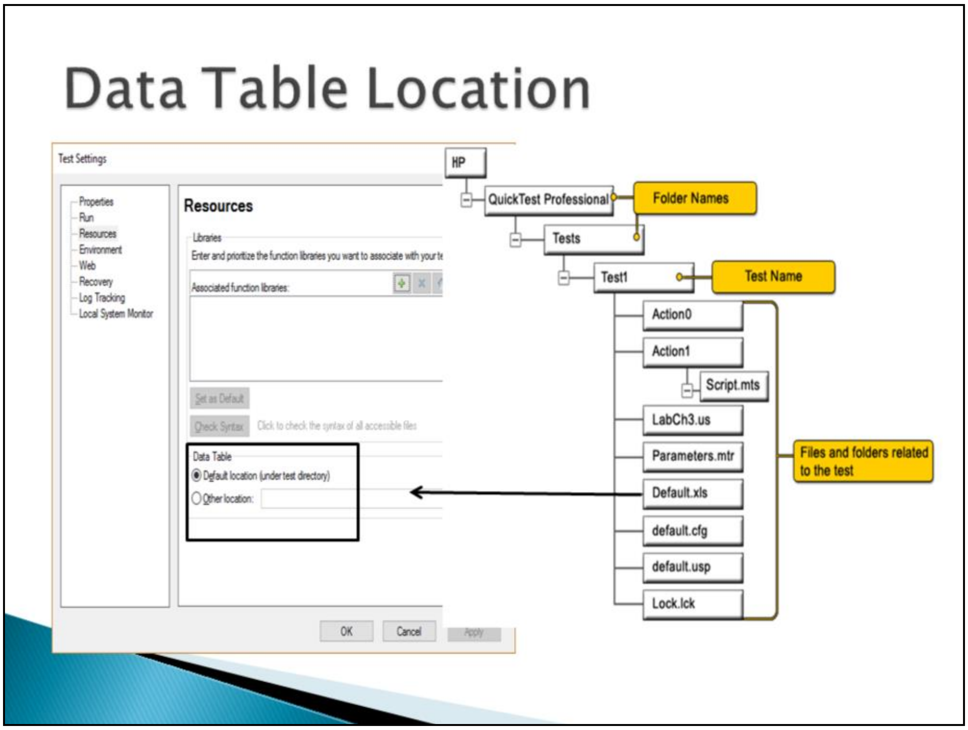
Local Data Sheet

```
WpfWindow("HP MyFlight Sample Application").WpfTabStrip("WpfTabStrip").Select "SEARCH ORDER"
WpfWindow("HP MyFlight Sample Application").WpfRadioButton("byNumberRadio").Set
WpfWindow("HP MyFlight Sample Application").WpfEdit("byNumberWatermark").Set DataTable("OrderNumber", dtLocalSheet)
WpfWindow("HP MyFlight Sample Application").WpfButton("SEARCH").Click
```

	OrderNumber	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
1	121																												

When you use a local data sheet, values specific to an action appear in the action data sheets that are available to that action only. Default iteration settings in the action data sheets are specified for one iteration.

Data Table Location



Test Settings

Resources

Libraries

Enter and prioritize the function libraries you want to associate with your te

Associated function libraries:

Data Table

● Default location (under test directory)

○ Other location:

To specify the location of DATA TABLE to be used in your test:

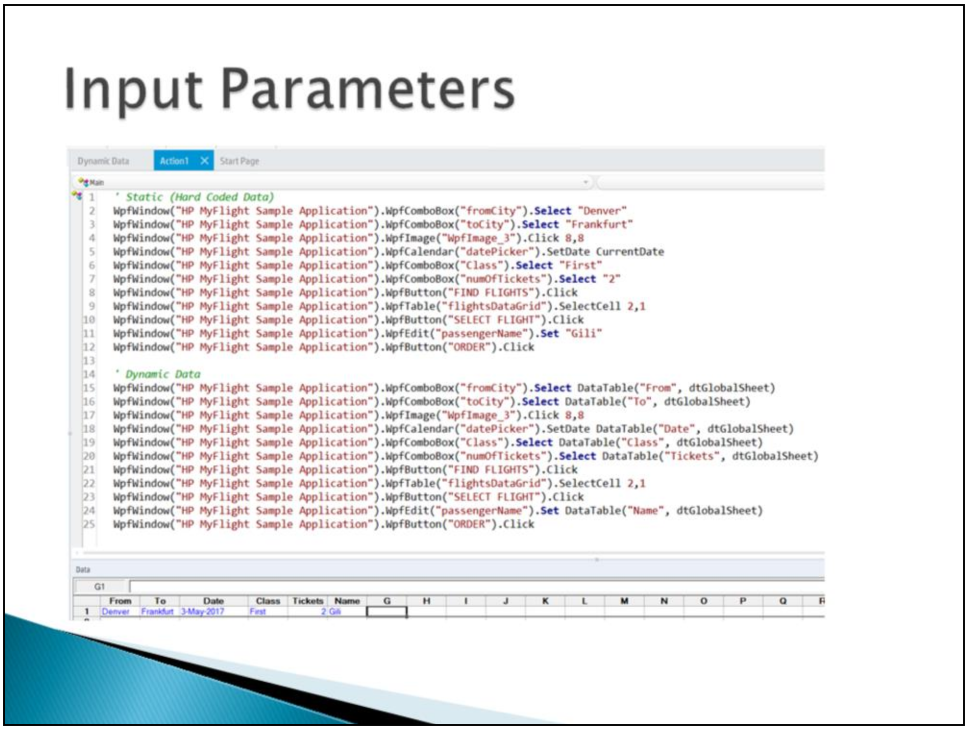
1. From the UFT menu bar, select **FILE** → **SETTINGS.** The **TEST SETTINGS** dialog box appears.

2. In the **TEST SETTINGS** dialog box, click the **RESOURCES** tab**.**

3. In the DATA TABLE section:

   •**Default location (under test directory).** Instructs UFT to use data stored in the default Data Table location under the test folder.

   •**Other location.** Instructs UFT to use data stored in the specified Data Table location. The Data Table can be any Microsoft Excel (**.xls**) file.



Input Parameters

```
' Static (Hard Coded Data)
WpfWindow("HP MyFlight Sample Application").WpfComboBox("fromCity").Select "Denver"
WpfWindow("HP MyFlight Sample Application").WpfComboBox("toCity").Select "Frankfurt"
WpfWindow("HP MyFlight Sample Application").WpfImage("WpfImage_3").Click 8,8
WpfWindow("HP MyFlight Sample Application").WpfCalendar("datePicker").SetDate CurrentDate
WpfWindow("HP MyFlight Sample Application").WpfComboBox("Class").Select "First"
WpfWindow("HP MyFlight Sample Application").WpfComboBox("numOfTickets").Select "2"
WpfWindow("HP MyFlight Sample Application").WpfButton("FIND FLIGHTS").Click
WpfWindow("HP MyFlight Sample Application").WpfTable("flightsDataGrid").SelectCell 2,1
WpfWindow("HP MyFlight Sample Application").WpfButton("SELECT FLIGHT").Click
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set "Gili"
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click

' Dynamic Data
WpfWindow("HP MyFlight Sample Application").WpfComboBox("fromCity").Select DataTable("From", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfComboBox("toCity").Select DataTable("To", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfImage("WpfImage_3").Click 8,8
WpfWindow("HP MyFlight Sample Application").WpfCalendar("datePicker").SetDate DataTable("Date", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfComboBox("Class").Select DataTable("Class", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfComboBox("numOfTickets").Select DataTable("Tickets", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfButton("FIND FLIGHTS").Click
WpfWindow("HP MyFlight Sample Application").WpfTable("flightsDataGrid").SelectCell 2,1
WpfWindow("HP MyFlight Sample Application").WpfButton("SELECT FLIGHT").Click
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set DataTable("Name", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click
```

Data

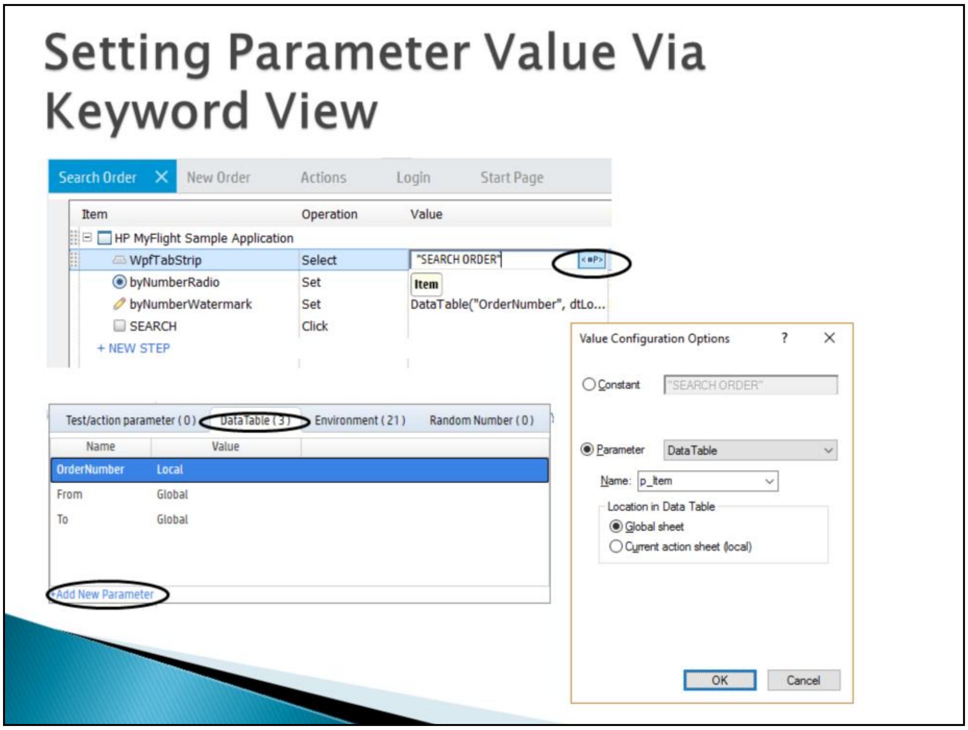
G1

	From	To	Date	Class	Tickets	Name	G	H	I	J	K	L	M	N	O	P	Q	R
1	Denver	Frankfurt	3-May-2017	First	2	Gili												

Input parameters enable you to replace a static recorded value in a step with a placeholder for a dynamic value. This placeholder is called a parameter and enables you to use multiple values during the test run.

Input parameter names and their values are located in DATA TABLE.

Setting Parameter Value Via Keyword View



Item	Operation	Value
HP MyFlight Sample Application		
WpfTabStrip	Select	"SEARCH ORDER"
byNumberRadio	Set	Item
byNumberWatermark	Set	DataTable("OrderNumber", dtLo...
SEARCH	Click	

Test/action parameter (0) DataTable (3) Environment (21) Random Number (0)

Name	Value
OrderNumber	Local
From	Global
To	Global

Value Configuration Options

○ Constant "SEARCH ORDER"

● Parameter DataTable

Name: p_Item

Location in Data Table

● Global sheet

○ Current action sheet (local)

To set the parameter value:

1. In KEYWORD VIEW, select the step that contains the recorded input value.

2. In the VALUE column, click to display the CONFIGURE THE VALUE button.

3. Select the appropriate type of parameter tab and click Add New Parameter

4. Click the CONFIGURE THE VALUE button. The VALUE CONFIGURATION OPTIONS dialog box appears.

5. In the VALUE CONFIGURATION OPTIONS dialog box, select the PARAMETER option.

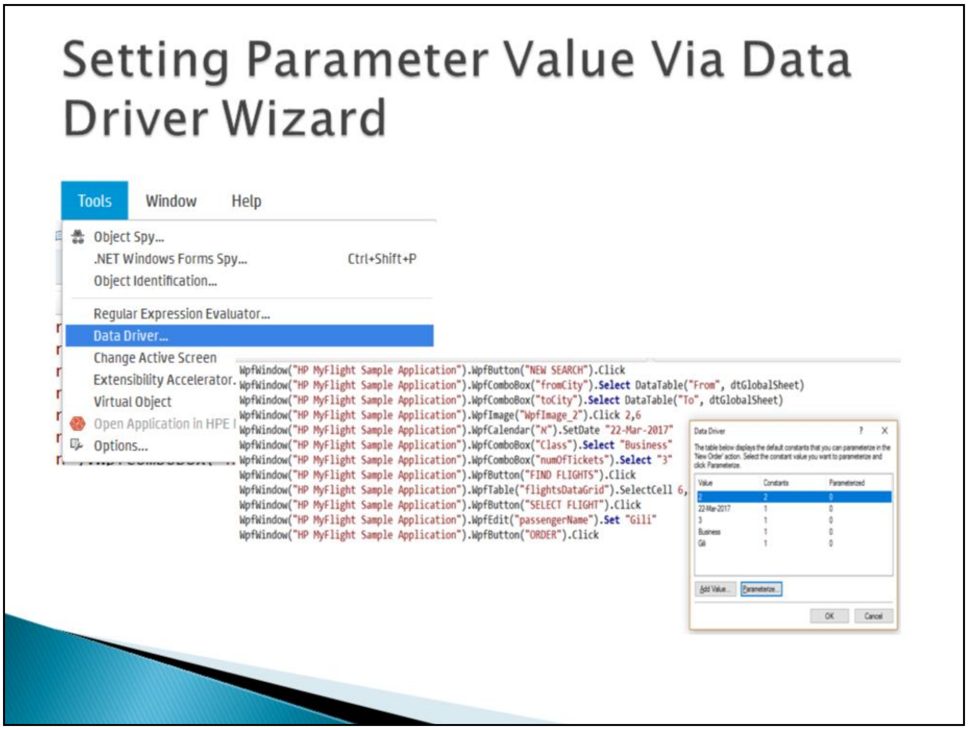
Ensure that the DATATABLE is selected from the PARAMETER list.

6. In the NAME field, type a unique column name to create a new column in your DATA TABLE or select an existing column name from the NAME list.

7. Use the default GLOBAL SHEET to store the value.

8. Click OK.

9. In the UFT window, enter the values in DATA TABLE that will be used during the test run.



Setting Parameter Value Via Data Driver Wizard

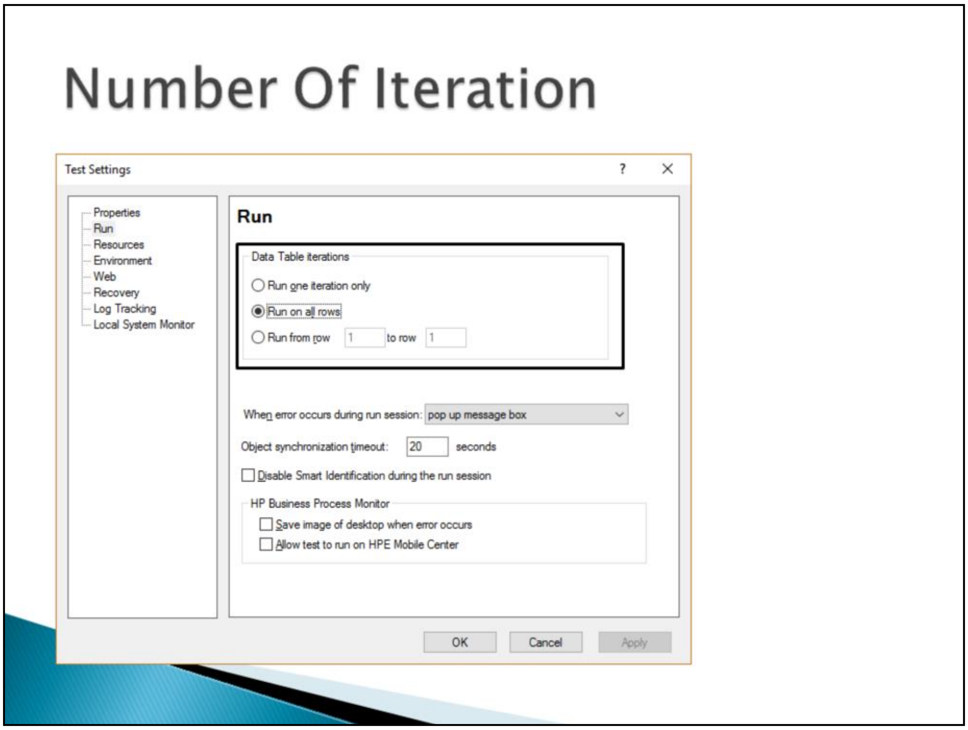
```
WpfWindow("HP MyFlight Sample Application").WpfButton("NEW SEARCH").Click
WpfWindow("HP MyFlight Sample Application").WpfComboBox("fromCity").Select DataTable("From", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfComboBox("toCity").Select DataTable("To", dtGlobalSheet)
WpfWindow("HP MyFlight Sample Application").WpfImage("WpfImage_2").Click 2,6
WpfWindow("HP MyFlight Sample Application").WpfCalendar("W").SetDate "22-Mar-2017"
WpfWindow("HP MyFlight Sample Application").WpfComboBox("Class").Select "Business"
WpfWindow("HP MyFlight Sample Application").WpfComboBox("numOfTickets").Select "3"
WpfWindow("HP MyFlight Sample Application").WpfButton("FIND FLIGHTS").Click
WpfWindow("HP MyFlight Sample Application").WpfTable("flightsDataGrid").SelectCell 6,
WpfWindow("HP MyFlight Sample Application").WpfButton("SELECT FLIGHT").Click
WpfWindow("HP MyFlight Sample Application").WpfEdit("passengerName").Set "Gili"
WpfWindow("HP MyFlight Sample Application").WpfButton("ORDER").Click
```

The table below displays the default constants that you can parameterize in the 'New Order' action. Select the constant value you want to parameterize and click Parameterize.

Value	Constants	Parameterized
2	2	0
22-Mar-2017	1	0
3	1	0
Business	1	0
Gili	1	0

To set the parameter value:

1. From the UFT menu bar, select **TOOLS→ DATA DRIVER.** The **DATA DRIVER** dialog box appears.
2. Select the row that contain the value we want to parameterize.
3. Press on parameterize…. Button. The **DATA DRIVER WIZARD – SELECT PARAMETRIZATION TYPE** dialog box appear.
4. Select **Parameterize all** radio button(Enables you to parameterize all occurrences of the selected value throughout the action).
5. In **Configure value** area select **Parameter** radio button.
6. Enter the column name in DATA TABLE sheet or press on **PARAMETERS OPTION** button. And **PARAMETERS OPTION** dialog box open.
7. Press FINISH in **DATA DRIVER WIZARD – SELECT PARAMETRIZATION TYPE** dialog box .
8. Press **OK** on **DATA DRIVER** dialog box.



Number Of Iteration

Test Settings

Run

Data Table iterations

○ Run one iteration only

● Run on all rows

○ Run from row 1 to row 1

When error occurs during run session: pop up message box

Object synchronization timeout: 20 seconds

□ Disable Smart Identification during the run session

HP Business Process Monitor

□ Save image of desktop when error occurs

□ Allow test to run on HPE Mobile Center

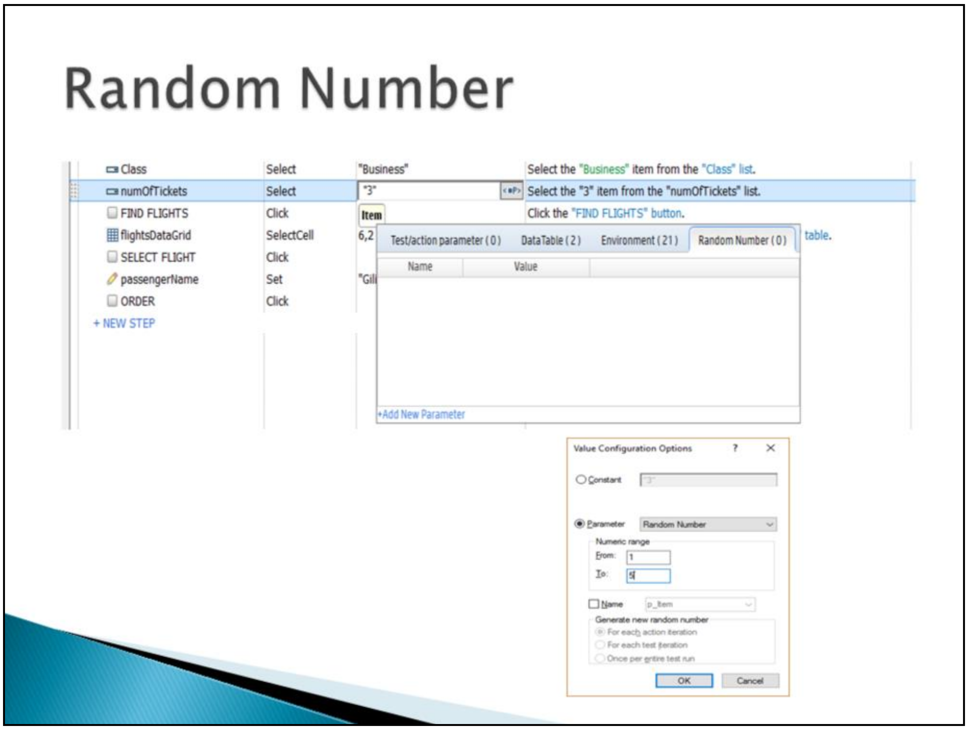
You can specify the number of rows of DATA TABLE that will be executed during a test run. To do so, from the UFT menu bar, select **FILE→SETTINGS.** The **TEST SETTINGS** dialog box appears. In the **TEST SETTINGS** dialog box, click the **RUN** tab**.**

You can select from the following options to specify DATA TABLE iterations:

- **Run one iteration only-** Runs the test only once, using only the first row in the global Data Table.
- **Run on all rows-** Runs the test with iterations using all rows in the global Data Table.
- **Run from row __ to row __-** Runs the test with iterations using the values in the global Data Table for the specified row range.

Note: Do not leave a blank row at the end of DATA TABLE. If you enter data on the last row and delete it, you create a cell that UFT considers filled with a blank value.

During the test run, UFT tries to run the test for the blank value and generates an error. To avoid this problem, in the UFT window, in DATA TABLE, select the cell or the row from where you deleted the data. From the menu bar, select **EDIT → DELETE.**



Random Number

Class	Select	"Business"	Select the "Business" item from the "Class" list.
numOfTickets	Select	"3"	Select the "3" item from the "numOfTickets" list.
FIND FLIGHTS	Click	Item	Click the "FIND FLIGHTS" button.
flightsDataGrid	SelectCell	6,2	table.
SELECT FLIGHT	Click		
passengerName	Set	"Gil	
ORDER	Click		

+ NEW STEP

Value Configuration Options

○ Constant "3"

● Parameter Random Number

Numeric range

From: 1

To: 5

□ Name p_Item

Generate new random number

● For each action iteration

○ For each test iteration

○ Once per entire test run

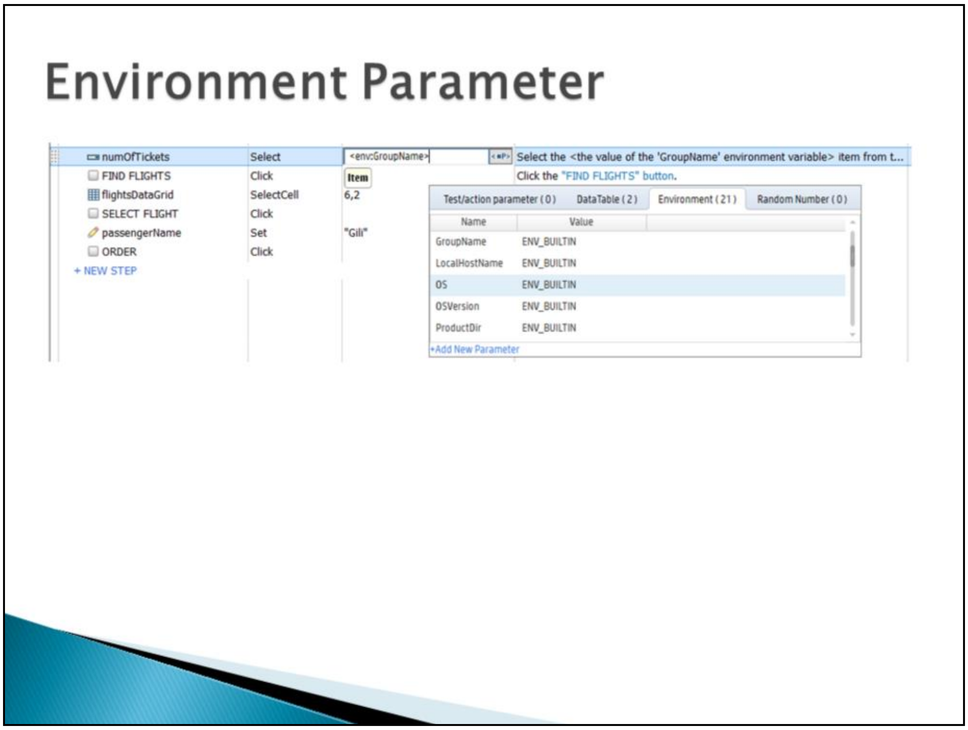
The random number input parameter generates random numbers and uses them as input value for a parameter. By default, the random number ranges between 0 and 100.

A different random number is generated every time the parameter is called for every iteration or for every test run.

You can modify these settings in the PARAMETER OPTIONS dialog box.

To display the PARAMETER OPTIONS dialog box, in the CHECKPOINT PROPERTIES dialog box, select the **PARAMETER** option. Click the **PARAMETER OPTIONS** button**.** The PARAMETER OPTIONS dialog box appears.

In the PARAMETER OPTIONS dialog box, check the **NAME** check box to reuse the random number parameter in the test.



Environment Parameter

numOfTickets	Select	<env:GroupName>	Select the <the value of the 'GroupName' environment variable> item from t...
□ FIND FLIGHTS	Click		Click the "FIND FLIGHTS" button.
flightsDataGrid	SelectCell	6,2	
□ SELECT FLIGHT	Click		
passengerName	Set	"Gili"	
□ ORDER	Click		

Name	Value
GroupName	ENV_BUILTIN
LocalHostName	ENV_BUILTIN
OS	ENV_BUILTIN
OSVersion	ENV_BUILTIN
ProductDir	ENV_BUILTIN

ENVIRONMENT parameters enable you to insert a value in your test from an environment variable list. The value of an environment parameter remains the same during the test playback, regardless of the number of iterations, unless you change the value.

Types of environment variables available in UFT are:

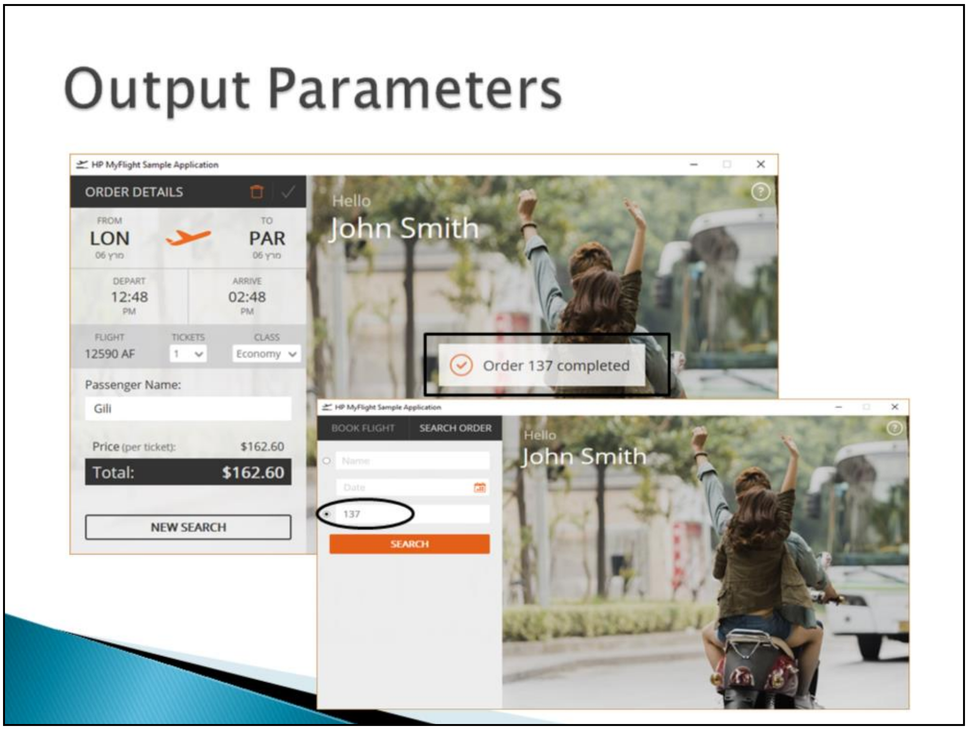
• **BUILT-IN:** Enables you to use information related to a test and the computer running the test. The information includes the test name, test path, operating system type, and version.

• **USER-DEFINED:** Enables you to create a list of variables in an external file in the .xml format. You can use the variables from the .xml file as parameters in a test.

To view the environment variable list in the UFT, from the UFT

Professional menu bar, select **FILE→SETTINGS→ENVIRONMENT**. Select the variable

type from the VARIABLE TYPE list.



Output Parameters

HP MyFlight Sample Application

ORDER DETAILS

FROM LON 06 מרץ — TO PAR 06 מרץ

DEPART 12:48 PM — ARRIVE 02:48 PM

FLIGHT 12590 AF — TICKETS 1 — CLASS Economy

Passenger Name: Gili

Price (per ticket): $162.60

Total: $162.60

Hello John Smith

Order 137 completed

HP MyFlight Sample Application

○ Name

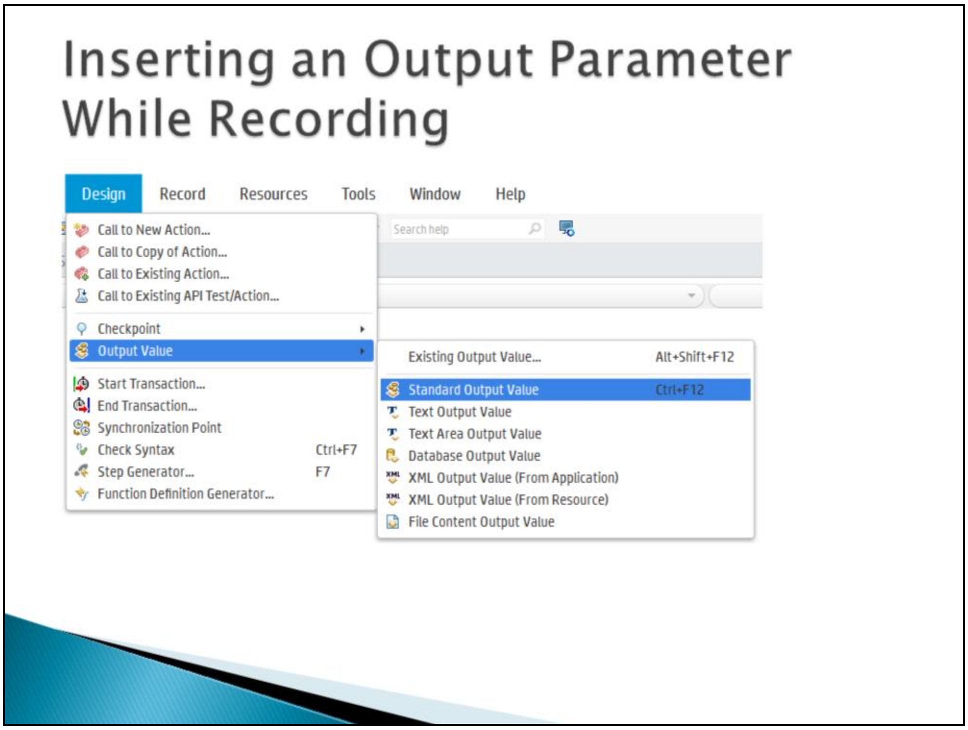
Date

● 137

Hello John Smith

An output parameter is value that captured at a specific point in your test and stored for the duration of the run session. The value can later be used as input at a different point in the run session.

Capturing a value during run-time and using it as an input variable in your test is called data correlation.



Inserting an Output Parameter While Recording

Design
Record
Resources
Tools
Window
Help
Call to New Action...
Call to Copy of Action...
Call to Existing Action...
Call to Existing API Test/Action...
Checkpoint
Output Value
Start Transaction...
End Transaction...
Synchronization Point
Check Syntax Ctrl+F7
Step Generator... F7
Function Definition Generator...
Search help
Existing Output Value... Alt+Shift+F12
Standard Output Value Ctrl+F12
Text Output Value
Text Area Output Value
Database Output Value
XML Output Value (From Application)
XML Output Value (From Resource)
File Content Output Value

To insert an output parameter during recording:

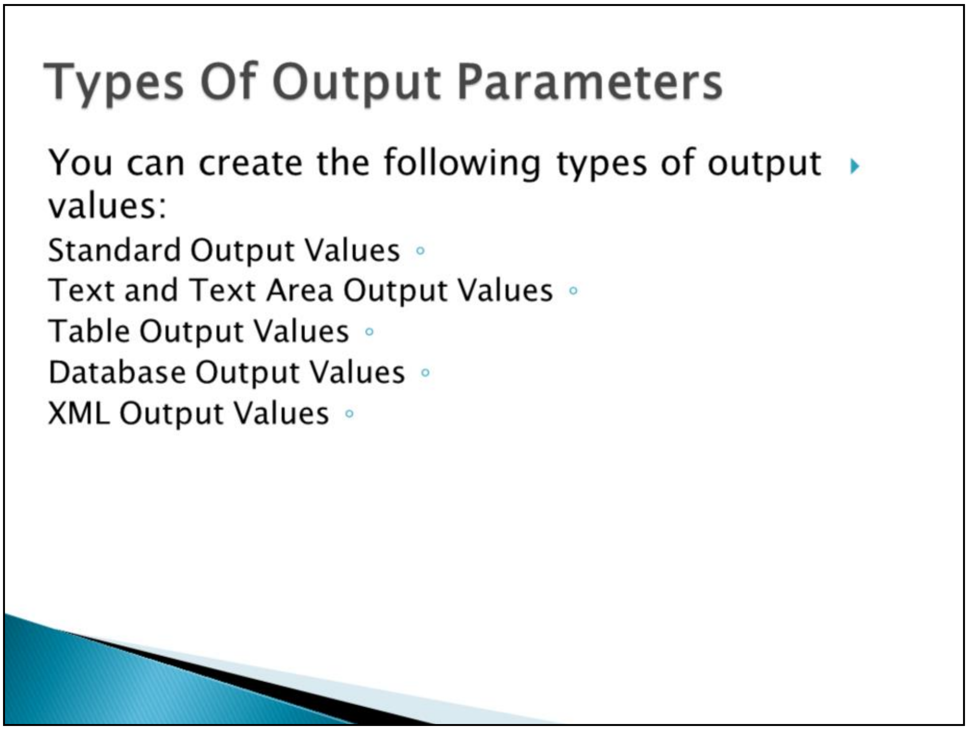
1. From the UFT menu bar, select **INSERT → OUTPUT VALUE →STANDARD OUTPUT VALUE.** The **OUTPUT VALUE PROPERTIES** dialog box appears.

2. In the OUTPUT VALUE PROPERTIES dialog box, check the properties that you want to parameterize.

3. Click the **MODIFY** button**.** The **OUTPUT OPTIONS** dialog box appears.

4. In the NAME field, select or type the column name to be used in the GLOBAL data sheet and click **OK.**

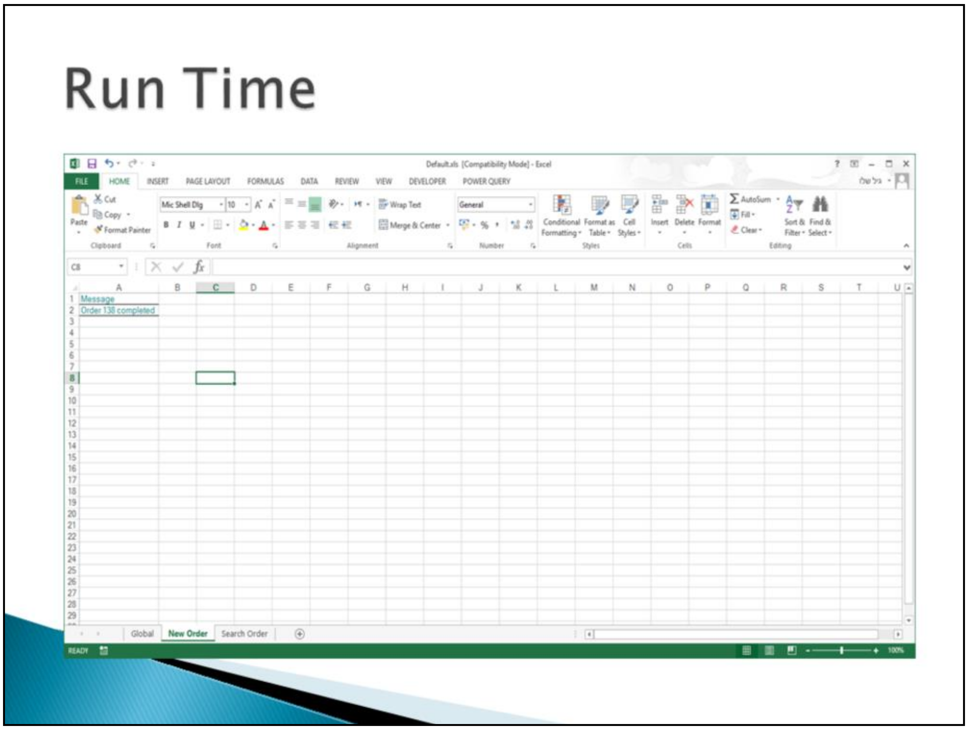
5. In the OUTPUT VALUE PROPERTIES dialog box, click **OK.**



Types Of Output Parameters

- You can create the following types of output values:
  - Standard Output Values
  - Text and Text Area Output Values
  - Table Output Values
  - Database Output Values
  - XML Output Values

1. **Standard Output Values** – you can use standard output values to output the property values of most objects.
2. **Text and Text Area Output Values** – You can output a part of the object's text. You can also specify the text before and after the output text.
3. **Table Output Values** – You can use table output values to output the contents of table cells.
4. **Database Output Values** – You can use database output values to output the value of the contents of database cells, based on the results of a query (result set) that you define on a database.
5. **XML Output Values** – You can use XML output values to output the values of XML elements and attributes in XML documents.

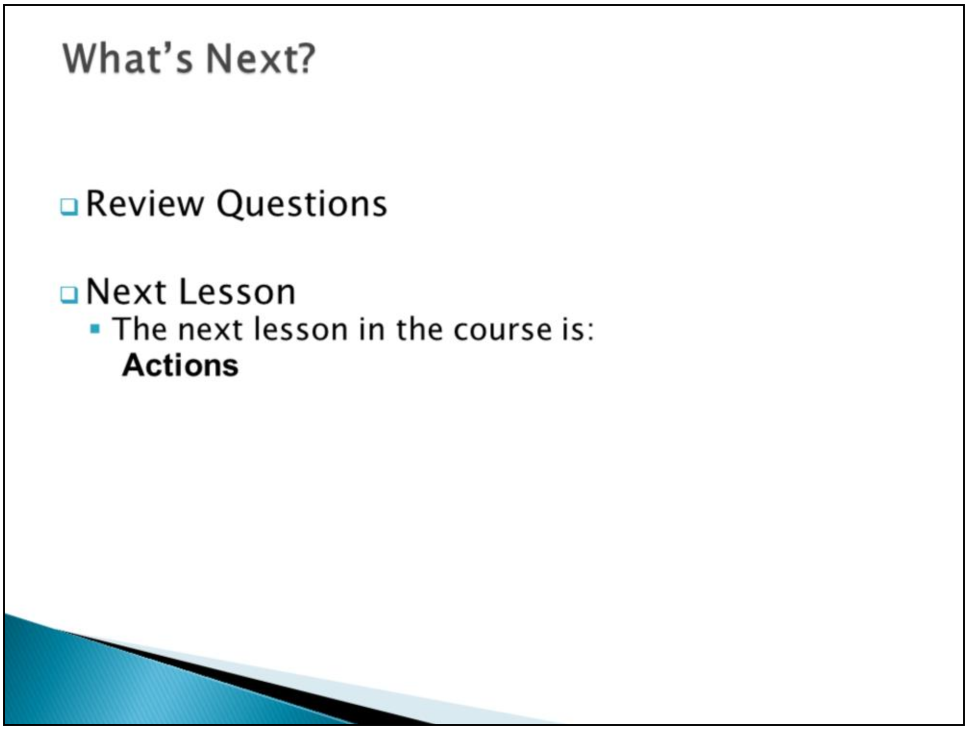


Run Time

	A
1	Message
2	Order 138 completed

The run-time DATA TABLE is created after a test is executed. This table is created and saved with the test results.

A copy of the design-time table is stored with your test. The output values are created dynamically and exist only at run time. The output values generated after each test run are located in the run-time DATA TABLE in the TEST RESULTS window.



What’s Next?

- ❑ Review Questions
- ❑ Next Lesson
  - The next lesson in the course is: **Actions**

End of Lesson