Mercury QuickTest Professional™

Tutorial

MERCURY™

Mercury QuickTest Professional

Tutorial

Version 8.0

MERCURY™

Mercury QuickTest Professional Tutorial, Version 8.0

Mercury Interactive Corporation
379 North Whisman Road
Mountain View, CA 94043
Tel: (650) 603-5200
Toll Free: (800) TEST-911
Customer Support: (877) TEST-HLP
Fax: (650) 603-5300

If you have any comments or suggestions regarding this document, please send them via e-mail to documentation@mercury.com.

QTPTUT8.0/06

Table of Contents

Welcome to the QuickTest Tutorial

Welcome to the QuickTest tutorial. The tutorial is a self-paced guide that teaches you the basics of testing your application with QuickTest, Mercury Interactive's powerful functional testing tool.

QuickTest enables you to test standard Web objects, ActiveX controls, and Visual Basic controls.

This tutorial will familiarize you with the process of creating and running automated tests and analyzing the test results.

Note: In addition to tests, QuickTest enables you to create business components for use in business process tests, if you have Quality Center with Business Process Testing support installed. The procedures described in this tutorial are designed for creating tests, but you can also apply the majority of these procedures to creating business components. For more information on business components and Business Process Testing, refer to the *QuickTest Professional User's Guide* and the *Business Process Testing User's Guide*.

This introduction describes:

➤ Using This Tutorial

➤ Typographical Conventions

Using This Tutorial

The tutorial is divided into nine short lessons. In each lesson you will create and run tests on the sample Mercury Tours Web site.

After completing the tutorial, you can apply the skills you have learned to your own Web site.

Lesson 1, Introducing QuickTest compares automated and manual testing methods. It introduces the QuickTest testing process and familiarizes you with the QuickTest user interface and the sample Mercury Tours Web site.

Lesson 2, Recording Tests teaches you how to record a test and provides an overview of the resulting Keyword View display.

Lesson 3, Running and Analyzing Tests shows you how to run a test and view the test results.

Lesson 4, Creating Checkpoints explains how to add checkpoints to your test to verify that information in your Web site is displayed as expected.

Lesson 5, Parameterizing Tests shows you how to parameterize your test so that you can run it on multiple sets of data.

Lesson 6, Creating Output Values teaches you how to use output parameters to retrieve data from your Web site.

Lesson 7, Using Regular Expressions teaches you how to create and run a test using regular expressions.

Lesson 8, Dividing Tests into Multiple Actions explains how to divide your test into actions so that you can design more efficient and modular tests.

Lesson 9, Where Do You Go from Here? shows you how to get started testing your own Web site and tells you where you can find more information about QuickTest.

Typographical Conventions

This book uses the following typographical conventions:

1, 2, 3
Bold numbers indicate steps in a procedure.

➤
Bullets indicate options and features.

>
The greater than sign separates menu levels (for example, **File** > **Open**).

Stone Sans
The **Stone Sans** font indicates names of interface elements (for example, the **Run** button) and other items that require emphasis.

Bold
Bold text indicates method or function names.

Italics
Italic text indicates method or function arguments, file names in syntax descriptions, and book titles.

<>
Angle brackets enclose a part of a file path or URL address that may vary from user to user (for example, <**MyProduct installation folder**>\bin).

Arial
The Arial font is used for examples and text that is to be typed literally.

Arial bold
The **Arial bold** font is used in syntax descriptions for text that should be typed literally.

...
In a line of syntax, an ellipsis indicates that more items of the same format may be included. In a programming example, an ellipsis is used to indicate lines of a program that were intentionally omitted.

[ ]
Square brackets enclose optional arguments.

|
A vertical bar indicates that one of the options separated by the bar should be selected.

1

Introducing QuickTest

In this lesson you will learn about the:

- Benefits of Automated Testing
- Testing Process
- QuickTest Window
- Mercury Tours Sample Web Site

Benefits of Automated Testing

If you have ever tested applications or Web sites manually, you are aware of the drawbacks. Manual testing is time-consuming and tedious, requiring a heavy investment in human resources. Worst of all, time constraints often make it impossible to manually test every feature thoroughly before the application is released. This leaves you wondering whether serious bugs have gone undetected.

Automated testing with QuickTest addresses these problems by dramatically speeding up the testing process. You can create tests that check all aspects of your application or Web site, and then run these tests every time your site or application changes.

As QuickTest runs tests, it simulates a human user by moving the mouse cursor in a Web page or application window, clicking Graphical User Interface (GUI) objects, and entering keyboard input; however, QuickTest does this faster than any human user.

Benefits of Automated Testing	
Fast	QuickTest runs tests significantly faster than human users.
Reliable	Tests perform precisely the same operations each time they are run, thereby eliminating human error.
Repeatable	You can test how the Web site or application reacts after repeated execution of the same operations.
Programmable	You can program sophisticated tests that bring out hidden information.
Comprehensive	You can build a suite of tests that covers every feature in your Web site or application.
Reusable	You can reuse tests on different versions of a Web site or application, even if the user interface changes.

Testing Process

The QuickTest testing process consists of 7 main phases:

1 Preparing to record

Before you record a test, confirm that your application and QuickTest are set to match the needs of your test.

Make sure your application displays elements on which you want to record, such as a toolbar or a special window pane, for example, and that your application options are set as you expect for the purposes of your test.

You should also view the settings in the Test Settings dialog box (**Test** > **Settings**) and the Options dialog box (**Tools** > **Options**) to ensure that QuickTest will record and store information appropriately. For example, you should confirm that the test is set to use the appropriate object repository mode.

2 Recording a session on your application

As you navigate through your application or Web site, QuickTest graphically displays each step you perform as a row in the Keyword View. A step is any user action that causes or makes a change in your application, such as clicking a link or image, or entering data in a form.

3 Enhancing your test

- Inserting checkpoints into your test lets you search for a specific value of a page, object, or text string, which helps you determine whether your application or site is functioning correctly.
- Broadening the scope of your test, by replacing fixed values with parameters, lets you check how your application performs the same operations with multiple sets of data.
- Adding logic and conditional or loop statements enables you to add sophisticated checks to your test.

4 Debugging your test

You debug a test to ensure that it operates smoothly and without interruption.

5 Running your test

You run a test to check the behavior of your application or Web site. While running, QuickTest opens the application, or connects to the Web site, and performs each step in your test.

6 Analyzing the test results

You examine the test results to pinpoint defects in your application.

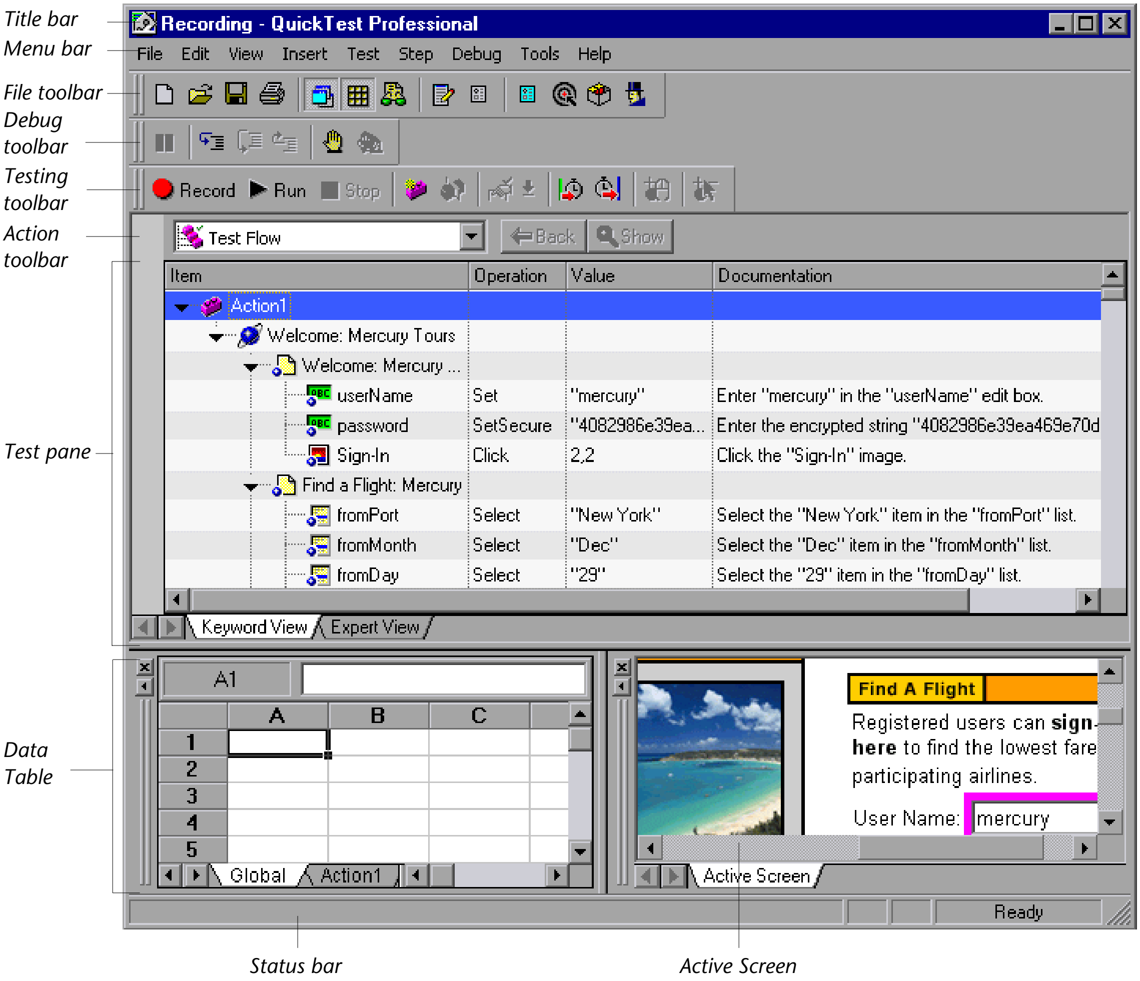
7 Reporting defects

If you have Quality Center installed, you can report the defects you discover to a database. Quality Center is Mercury Interactive's software test management tool.

QuickTest Window

Before you begin creating tests, you should familiarize yourself with the main QuickTest window.

The image below shows a QuickTest window as it would appear after you record a test, with all toolbars and panes (except the Debug Viewer pane) displayed:



Item	Operation	Value	Documentation
Action1			
Welcome: Mercury Tours			
Welcome: Mercury ...			
userName	Set	"mercury"	Enter "mercury" in the "userName" edit box.
password	SetSecure	"4082986e39ea...	Enter the encrypted string "4082986e39ea469e70d
Sign-In	Click	2,2	Click the "Sign-In" image.
Find a Flight: Mercury			
fromPort	Select	"New York"	Select the "New York" item in the "fromPort" list.
fromMonth	Select	"Dec"	Select the "Dec" item in the "fromMonth" list.
fromDay	Select	"29"	Select the "29" item in the "fromDay" list.

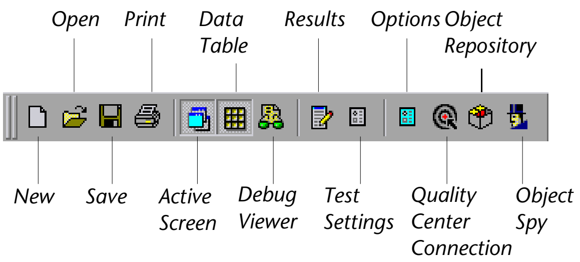
Find A Flight

Registered users can **sign-here** to find the lowest fare participating airlines.

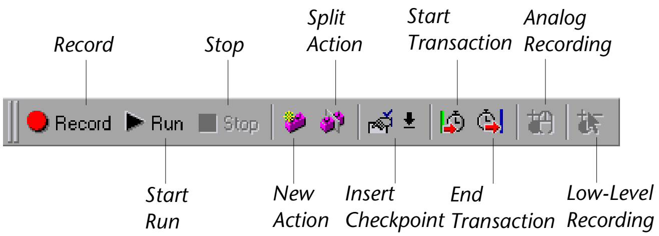
User Name: mercury

The QuickTest window contains the following key elements:

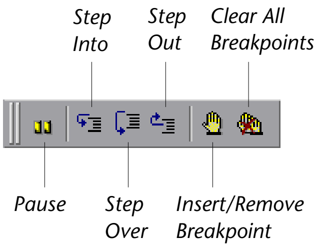
- **Title bar**—Displays the name of the currently open test.
- **Menu bar**—Displays menus of QuickTest commands.
- **File toolbar**—Contains buttons to assist you in managing your test.



- **Testing toolbar**—Contains buttons to assist you in the testing process.

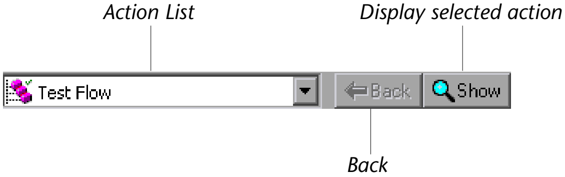


- **Debug toolbar**—Contains buttons to assist you in debugging tests.



Note: The Debug toolbar is not displayed when you open QuickTest for the first time. You can display the Debug toolbar by choosing **View** > **Toolbars** > **Debug**. Note that this tutorial does not describe how to debug a test. For additional information, refer to the *QuickTest Professional User's Guide*.

- **Action toolbar**—Contains buttons and a list of actions, enabling you to view the details of an individual action or the entire test flow.



Note: The Action toolbar is only displayed in the Keyword View. It is not displayed when you open QuickTest for the first time. You can display the Action toolbar by choosing **View** > **Toolbars** > **Action**. If you insert a reusable or external action in a test, the Action toolbar is displayed automatically in the Keyword View. For additional information, refer to the *QuickTest Professional User's Guide*.

- **Test pane**—Contains the Keyword View and Expert View tabs.
- **Active Screen**—Provides a snapshot of your application as it appeared when you performed a certain step during the recording session.
- **Data Table**—Assists you in parameterizing your test.
- **Debug Viewer pane**—Assists you in debugging your test. The Debug Viewer pane contains the **Watch Expressions**, **Variables**, and **Command** tabs. (The Debug Viewer pane is not displayed when you open QuickTest for the first time. You can display the Debug Viewer by choosing **View** > **Debug Viewer**.)
- **Status bar**—Displays the status of the QuickTest application.

The buttons used in the File and Testing toolbars are described in detail in subsequent lessons. The buttons used in the Debug toolbar are described in detail in the “Debugging Tests” chapter in the *QuickTest Professional User's Guide*. The Action toolbar is described in Lesson 8, “Dividing Tests into Multiple Actions.”

Note that you can also execute many menu commands using shortcut keys. For additional information, see the “QuickTest at a Glance” chapter in the *QuickTest Professional User's Guide*.

Now that you are familiar with the main QuickTest window, take a few minutes to explore its components.

Mercury Tours Sample Web Site

Mercury Tours is the sample Web application used throughout this tutorial. It simulates a Web-based flight information and reservation service. You should familiarize yourself with this application before starting the tutorial.

Optimizing Browser Settings for Your Test

If you are using Internet Explorer as your browser, you should clear the option to use AutoComplete for user names and passwords for the purposes of this tutorial. This will ensure that all of your operations are accurately recorded while creating your tests.

To clear the AutoComplete option:

1. In your Internet Explorer's menu bar, choose **Tools** > **Internet Options** > **Content** tab.
2. Click **AutoComplete** in the **Personal information** area. The AutoComplete Settings dialog box opens.
3. In the **Use AutoComplete for** area, clear the **User names and passwords on forms** option.
4. Click **OK** to save your changes and close the AutoComplete Settings dialog box, then click **OK** again to close the Internet Options dialog box.

Using the Mercury Tours Web Site for the First Time

Before you begin recording your tests on the Mercury Tours Web site, you must register as a user.

To run Mercury Tours:

1. **Launch the Mercury Tours application.**

   In your Web browser, type the following URL: http://newtours.mercuryinteractive.com

   The Mercury Tours home page opens.

2 Register with Mercury Tours.

To log-in and use the Mercury Tours application, you must be a registered user.

From the home page, click the **REGISTER** navigation button. The Register page opens.

In the User Information section at the bottom of the page, enter any user name and password, and confirm your password. (Other information is not required.)

Click **SUBMIT**. Mercury Tours confirms your registration. From this confirmation page, click the **SIGN-ON** navigation button to enter the application. The Sign-on page opens. Enter your registered user name and password, and click **SUBMIT**.

The Flight Finder page opens.

3 Explore the Mercury Tours site.

Beginning with the Flight Finder page, follow the on-screen instructions to get flight information and reserve a flight.

Note that on the Book a Flight page, you do not have to fill out actual personal or credit card information; just enter the required information (indicated by red) in the Passengers and Credit Card sections. (You may enter fictitious information.)

4 End your Mercury Tours session.

When you are done exploring the Mercury Tours application, click the **LOG OUT** button on the Flight Confirmation page or the **SIGN-OFF** link at the top of any application page.

To sign-in again, simply enter your registered user name and password in the Sign-on page or in the Mercury Tours home page.

5 Close the Web browser.

You are now ready to begin using this tutorial to create tests on the Mercury Tours application using QuickTest!

2

Recording Tests

As you navigate through your Web site or application, QuickTest records your steps. These operations are the basis of your test. When you stop recording, you can see the steps of your newly-created test in a graphical format in the Keyword View.

In this lesson you will learn about:

- Preparing to Record a Test
- Recording a Test
- Analyzing the Test in the Keyword View

Preparing to Record a Test

Before you begin recording a test, ensure that both your application or Web site and QuickTest are set to match the needs of your test.

For the purposes of this tutorial, ensure that:

- You have registered as a user in the Mercury Tours Web site. For more information, see "Using the Mercury Tours Web Site for the First Time" on page 7.
- If you are using Internet Explorer as your browser, the **AutoComplete** option is cleared for user names and passwords. For instructions, see "Optimizing Browser Settings for Your Test" on page 7.
- All browsers are closed before you begin recording.

Recording a Test

In this section, you will record the process of making a reservation for a flight from New York to San Francisco on the Mercury Tours Web site.

1 Start QuickTest and open a new test.

➤ If QuickTest is not currently open, choose **Start** > **Programs** > **QuickTest Professional** > **QuickTest Professional**.

In the Add-in Manager, confirm that the **Web** Add-in is selected, and clear all other add-ins. Click **OK** to close the Add-in Manager and open QuickTest.

Note: While QuickTest loads your selected add-ins, the QuickTest splash screen is displayed. This may take a few seconds.

If the Welcome window opens, click **Blank Test**.

Otherwise, choose **File** > **New**, or click the **New** button.

A blank test opens.

➤ If QuickTest is already open, check which add-ins are loaded by selecting **Help** > **About QuickTest Professional**. If the **Web** Add-in is not loaded, you must exit and restart QuickTest. When the Add-in Manager opens, select the **Web** Add-in, and clear all other add-ins.

Choose **File** > **New**, or click the **New** button.

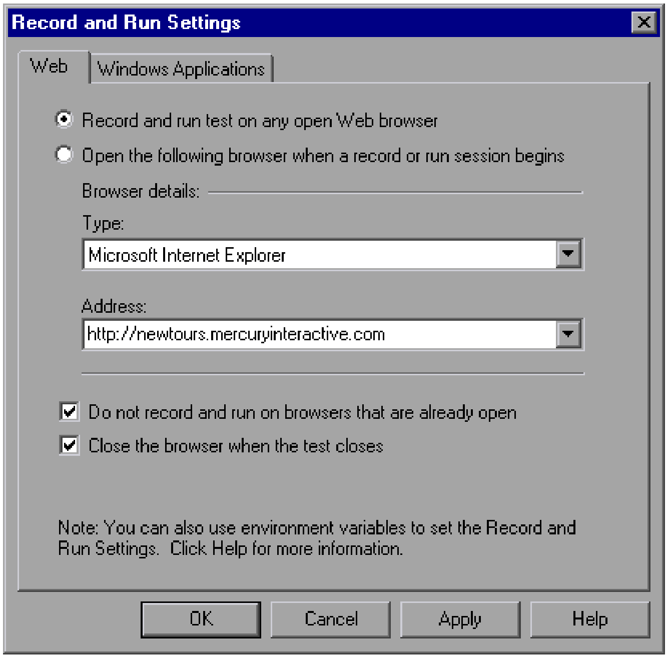
A blank test opens.

Note: If the Add-in Manager does not open when starting QuickTest, choose **Tools** > **Options**. In the General tab, select **Display Add-in Manager on startup**. When you exit and restart QuickTest, the Add-in Manager opens.

---

2 Start recording on the Mercury Tours Web site.

Choose **Test** > **Record** or click the **Record** button. The Record and Run Settings dialog box opens.



Record and Run Settings

● Record and run test on any open Web browser

○ Open the following browser when a record or run session begins

Browser details:

Type: Microsoft Internet Explorer

Address: http://newtours.mercuryinteractive.com

☑ Do not record and run on browsers that are already open

☑ Close the browser when the test closes

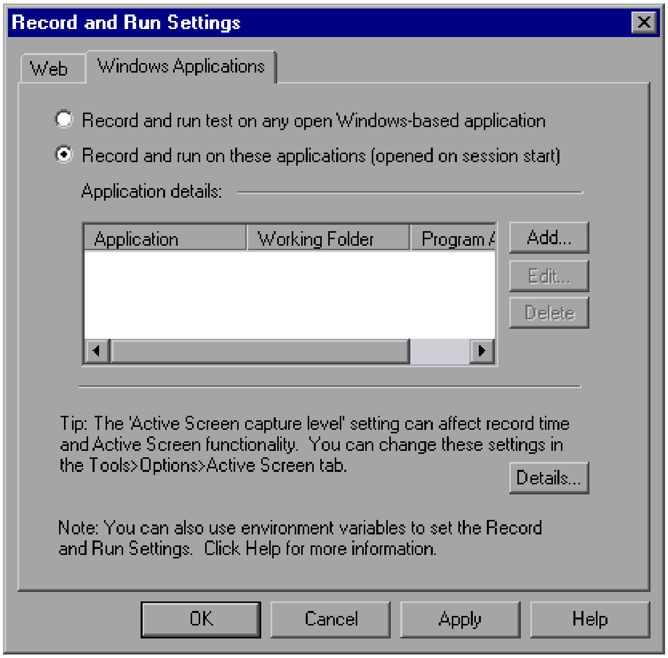
Note: You can also use environment variables to set the Record and Run Settings. Click Help for more information.

➤ In the Web tab, select **Open the following browser when a record or run session begins**.

Choose a browser from the **Type** list and confirm that the URL in the **Address** box is **http://newtours.mercuryinteractive.com**.

Confirm that **Close the browser when the test closes** and **Do not record and run on browsers that are already open** are selected.

➤ In the Windows Applications tab, confirm that **Record and run on these applications (opened on session start)** is selected, and that there are no applications listed.



Record and Run Settings

Web | Windows Applications

○ Record and run test on any open Windows-based application

● Record and run on these applications (opened on session start)

Application details:

Application	Working Folder	Program A

Tip: The 'Active Screen capture level' setting can affect record time and Active Screen functionality. You can change these settings in the Tools>Options>Active Screen tab.

Note: You can also use environment variables to set the Record and Run Settings. Click Help for more information.

This setting prevents you from inadvertently recording operations performed on various Windows applications (such as e-mail) during a recording session.

Click **OK**.

QuickTest begins recording, and your browser opens to the Mercury Tours Web site.

3 Log in to the Mercury Tours Web site.

In the **User Name** and **Password** boxes, type the name and password you registered with Mercury Tours.

Click **Sign-In**.

The Flight Finder page opens.

4 Enter flight details.

Change the following selections:

- Departing From: **New York**
- On: **Dec 29**
- Arriving In: **San Francisco**
- Returning: **Dec 31**
- Service Class: **Business class**

Click **CONTINUE** to accept the other default selections. The Select Flight page opens.

Notes:

When selecting the date, you must click on the drop-down list, scroll to any entry that is not showing, and select it. This is because QuickTest only records a step if the value in the list changes. The tutorial will not work correctly if you accept the currently displayed month or day.

When entering dates while recording this test, do not click the **View Calendar** button, which opens a Java-based calendar. Your test will not record the date selected using this calendar because the Java Add-in is not loaded for this tutorial (The Java Add-in is an external add-in that can be purchased separately.)

To check which add-ins are loaded, click **Help** > **About QuickTest Professional**. To change the available add-ins for your tests, you must close and reopen QuickTest Professional.

---

5 Select a flight.

Click **CONTINUE** to accept the default flight selections. The Book a Flight page opens.

6 Enter required passenger and purchase information.

Enter the required information (fields with red text labels) in the Passengers and Credit Card sections. (You may enter fictitious information.)

In the Billing Address section, select **Ticketless Travel**.

At the bottom of the page, click **SECURE PURCHASE**. The Flight Confirmation page opens.

7 Review and complete your booking.

Click **BACK TO HOME**. The Mercury Tours home page opens.

8 Stop recording.

In QuickTest, click **Stop** on the Testing toolbar to stop the recording process.

You have now reserved an imaginary business class ticket from New York to San Francisco. QuickTest recorded your Web browser operations from the time you clicked the **Record** button until you clicked the **Stop** button.

9 Save your test.

Select **File** > **Save** or click the **Save** button. The Save dialog box opens to the Tests folder.

Create a folder named Tutorial, select it, and click **Open**.

Type Recording in the **File name** field.

Confirm that **Save Active Screen files** is selected.

Click **Save**. The test name (Recording) is displayed in the title bar of the main QuickTest window.

Analyzing the Test in the Keyword View

As you recorded your test, QuickTest generated steps in the Keyword View representing each operation you performed in the Web browser.

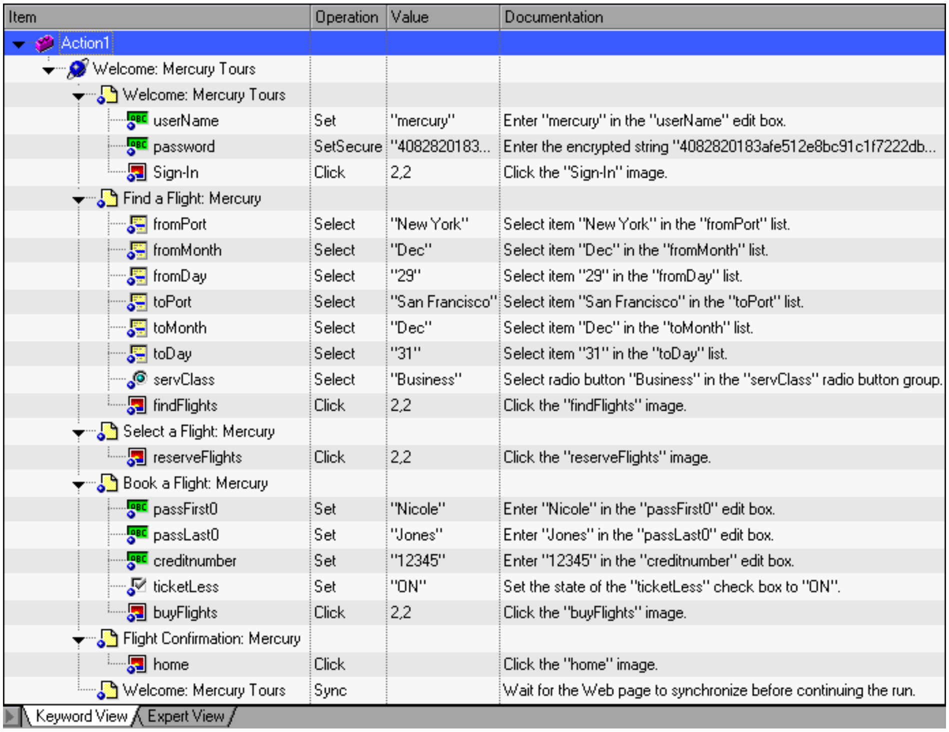
The columns in the Keyword View show different information for each step, as follows:

- **Item**—Displays the item for the step (test object, utility object, function call, or statement) in a hierarchical icon-based tree.
- **Operation**—The operation to be performed on the item, for example, **Click** or **Select**.
- **Value**—The argument values for the selected operation, for example, the mouse button to use when clicking the image.
- **Assignment**—The assignment of a value to or from a variable so you can use the value later in the test.
- **Comment**—Any textual information you want to add regarding the step, for example, Return to page used in first step of the test.
- **Documentation**—Auto-documentation of what the step does, in an easy-to-understand sentence, for example, Click the "findFlights" image.

Note: You can choose to hide or display individual columns as required, by right-clicking the column heading in the Keyword View and selecting a column name from the list.

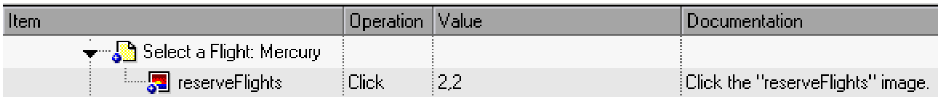
In the **Item** column of the Keyword View, you can click the branch arrows to expand or collapse the steps under each Web page. You can expand the entire test by choosing **View** > **Expand All**.

When you expand your test, it should look similar to this:



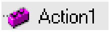
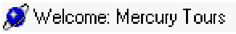
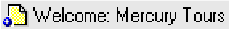
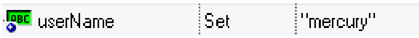
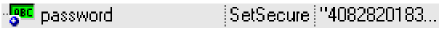
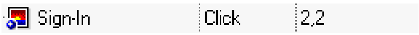
Item	Operation	Value	Documentation
Action1			
Welcome: Mercury Tours			
Welcome: Mercury Tours			
userName	Set	"mercury"	Enter "mercury" in the "userName" edit box.
password	SetSecure	"4082820183...	Enter the encrypted string "4082820183afe512e8bc91c1f7222db...
Sign-In	Click	2,2	Click the "Sign-In" image.
Find a Flight: Mercury			
fromPort	Select	"New York"	Select item "New York" in the "fromPort" list.
fromMonth	Select	"Dec"	Select item "Dec" in the "fromMonth" list.
fromDay	Select	"29"	Select item "29" in the "fromDay" list.
toPort	Select	"San Francisco"	Select item "San Francisco" in the "toPort" list.
toMonth	Select	"Dec"	Select item "Dec" in the "toMonth" list.
toDay	Select	"31"	Select item "31" in the "toDay" list.
servClass	Select	"Business"	Select radio button "Business" in the "servClass" radio button group.
findFlights	Click	2,2	Click the "findFlights" image.
Select a Flight: Mercury			
reserveFlights	Click	2,2	Click the "reserveFlights" image.
Book a Flight: Mercury			
passFirst0	Set	"Nicole"	Enter "Nicole" in the "passFirst0" edit box.
passLast0	Set	"Jones"	Enter "Jones" in the "passLast0" edit box.
creditnumber	Set	"12345"	Enter "12345" in the "creditnumber" edit box.
ticketLess	Set	"ON"	Set the state of the "ticketLess" check box to "ON".
buyFlights	Click	2,2	Click the "buyFlights" image.
Flight Confirmation: Mercury			
home	Click		Click the "home" image.
Welcome: Mercury Tours	Sync		Wait for the Web page to synchronize before continuing the run.

For every step performed on an object, QuickTest displays a row in the Keyword View with an icon and details of the step. For example, the step below instructs QuickTest to click the **reserveFlights** image at a specific location in the **Select a Flight: Mercury** page of the Mercury Tours Web site.



Item	Operation	Value	Documentation
Select a Flight: Mercury			
reserveFlights	Click	2,2	Click the "reserveFlights" image.

The table below provides an explanation of the first few steps shown in the test in the Keyword View above.

Step	Description
 Action1	**Action1** is the action name.
 Welcome: Mercury Tours	The browser invokes the **Welcome: Mercury Tours** Web site.
 Welcome: Mercury Tours	**Welcome: Mercury Tours** is the name of the Web page.
 userName Set "mercury"	**userName** is the name of the edit box. **Set** is the method performed on the edit box. **mercury** is the value of the edit box.
 password SetSecure "4082820183...	**password** is the name of the edit box. **SetSecure** is an encryption method performed on the edit box. **4082820183afe512e8bc91c1f7222dbd** is the encrypted value of the password.
 Sign-In Click 2,2	**Sign-In** is the name of the image link. **Click** is the method performed on the image. **2, 2** are the x- and y-coordinates where the image was clicked.

Take a few minutes to examine the Keyword View in detail, tracing the steps you performed while recording your test.

For more information on the Keyword View, refer to the *QuickTest Professional User's Guide*.

3

Running and Analyzing Tests

When you run your test, QuickTest opens the appropriate application or Web site and performs each step as it was originally recorded in the test. When QuickTest finishes running the test, it displays the results of the run.

In this lesson you will learn about:

- Running a Test
- Analyzing Test Results

Running a Test

In this lesson, you will run the test you recorded in the previous lesson.

1 Start QuickTest and open the Recording test.

If QuickTest is not already open, choose **Start** > **Programs** > **QuickTest Professional** > **QuickTest Professional**.

- If the Welcome window opens, click **Open Existing**.
- If QuickTest opens without displaying the Welcome window, choose **File** > **Open** or click the **Open** button.

In the Open Test dialog box, locate and select the **Recording** test, then click **Open**.

2 Confirm that all images are saved to the test results.

QuickTest allows you to determine when to save images to the test results. In this lesson, all images should be saved to the test results.

Choose **Tools** > **Options** and select the **Run** tab. In the **Save step screen capture to test results** option, select **Always**.

Click **OK** to close the Options dialog box.

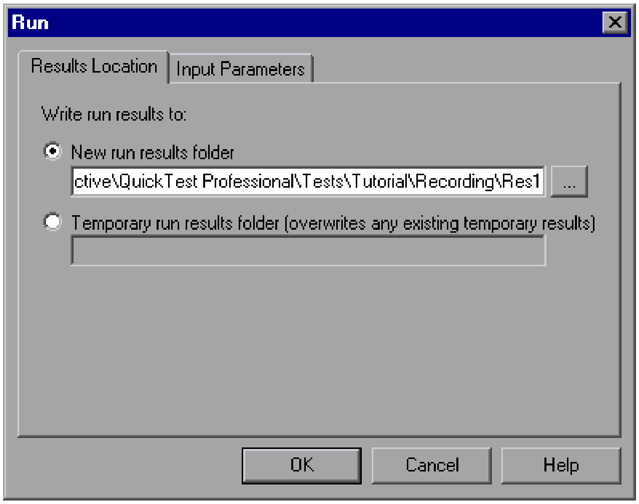
3 Start running your test.

▶ Run

Click **Run** or choose **Test** > **Run**. The Run dialog box opens.

Select **New run results folder**. Accept the default results folder name.

Click **OK** to close the Run dialog box.



Run

Results Location | Input Parameters

Write run results to:

● New run results folder

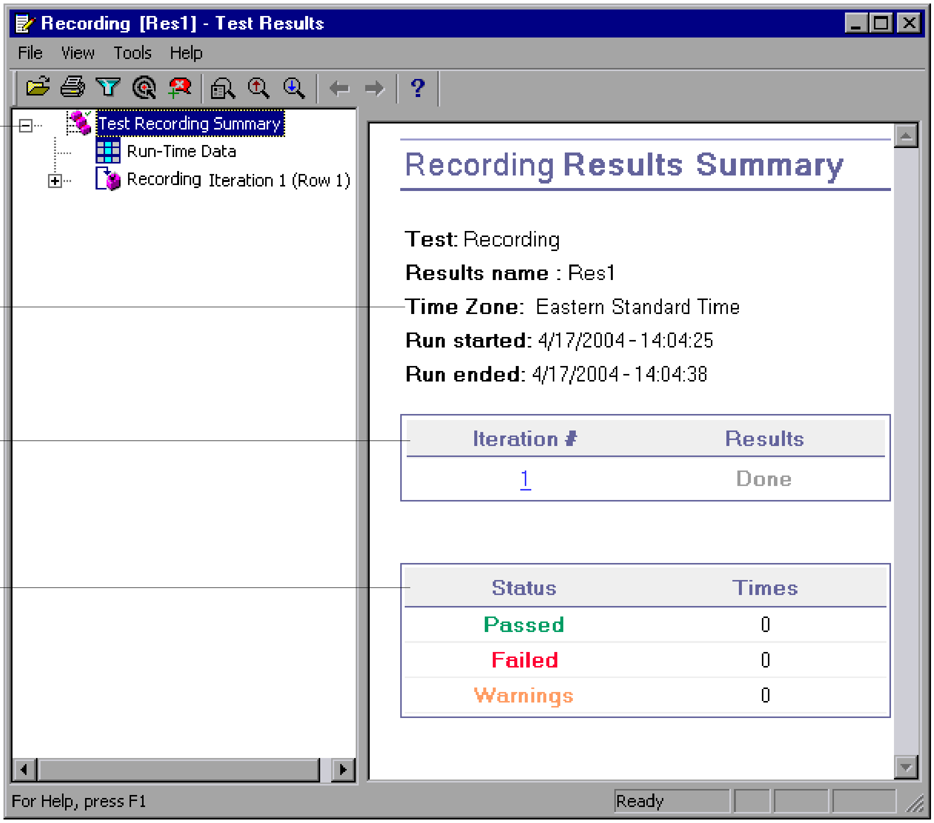
ctive\QuickTest Professional\Tests\Tutorial\Recording\Res1 ...

○ Temporary run results folder (overwrites any existing temporary results)

Watch carefully as QuickTest opens your browser and starts running the test. In the browser, you can see QuickTest perform each step you recorded; a yellow arrow in the left margin of the Keyword View indicates the step that QuickTest is running.

Analyzing Test Results

When QuickTest finishes running the test, the Test Results window opens.



Results tree

Test results details

Iteration summary

Status summary

Test Recording Summary
Run-Time Data
Recording Iteration 1 (Row 1)

Recording Results Summary

Test: Recording
Results name : Res1
Time Zone: Eastern Standard Time
Run started: 4/17/2004 - 14:04:25
Run ended: 4/17/2004 - 14:04:38

Iteration #	Results
1	Done

Status	Times
Passed	0
Failed	0
Warnings	0

Initially, the Test Results window contains two panes for displaying the key elements of your test run.

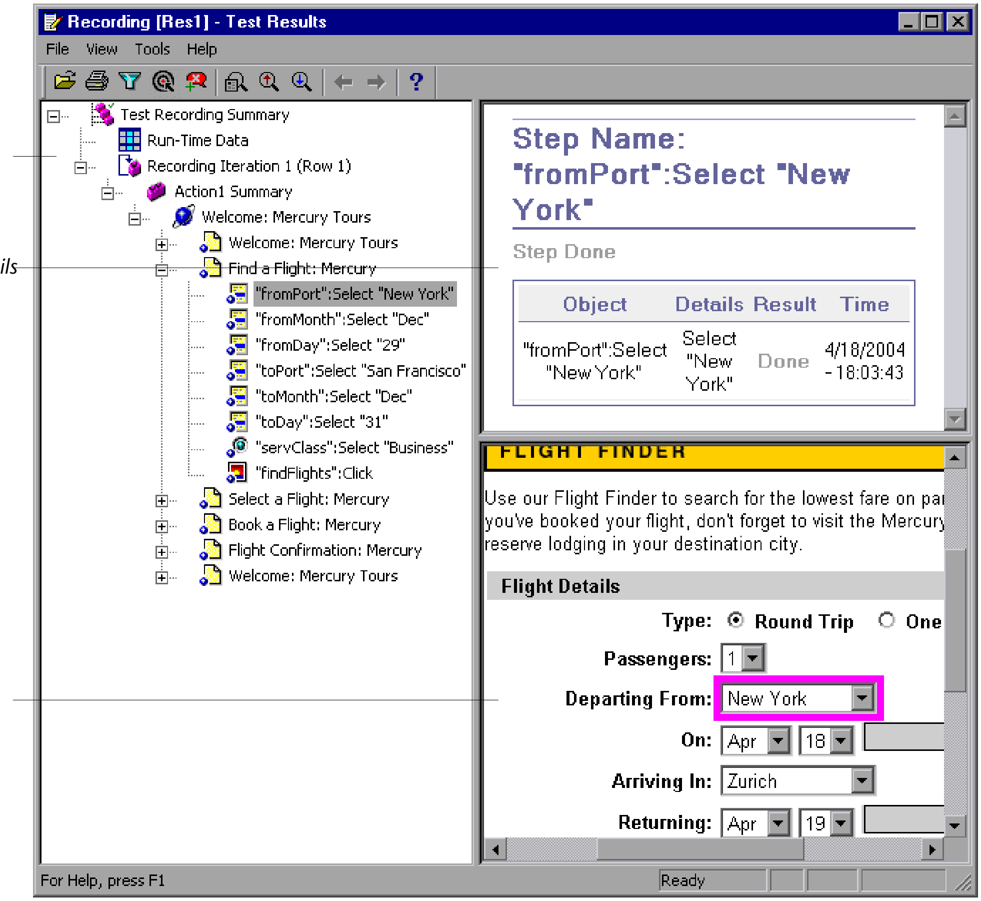
- The left pane displays the results tree, an icon-based view of the steps that were performed while the test was running. The results tree is organized according to the Web pages visited during the test run and can be expanded (+) to view each step. The steps performed during the test run are represented by icons in the tree. You can instruct QuickTest to run a test or action more than once using different sets of data in each run. Each test run is called an iteration, and each iteration is numbered. (The test you ran had only one iteration.)
- The right pane displays the test results details. The iteration summary table indicates which iterations passed and which failed. The status summary table indicates the number of checkpoints or reports that passed, failed, and raised warnings during the test.

Your test run succeeded because QuickTest was able to navigate the Mercury Tours site just as the test was originally recorded. In this section, you will inspect the steps QuickTest performed when running your test, and how the application window appeared when a step was performed.

1 View the test results for a specific step.

In the results tree, expand **(+) Test Recording Summary** > **Recording Iteration 1 (Row 1)** > **Action1 Summary** > **Welcome Mercury Tours** > **Find a Flight**.

Highlight "**fromPort":Select "New York**" in the results tree.



Results tree

Test results details

Active Screen image

- Test Recording Summary
  - Run-Time Data
  - Recording Iteration 1 (Row 1)
    - Action1 Summary
      - Welcome: Mercury Tours
        - Welcome: Mercury Tours
        - Find a Flight: Mercury
          - "fromPort":Select "New York"
          - "fromMonth":Select "Dec"
          - "fromDay":Select "29"
          - "toPort":Select "San Francisco"
          - "toMonth":Select "Dec"
          - "toDay":Select "31"
          - "servClass":Select "Business"
          - "findFlights":Click
        - Select a Flight: Mercury
        - Book a Flight: Mercury
        - Flight Confirmation: Mercury
        - Welcome: Mercury Tours

Step Name: "fromPort":Select "New York"

Step Done

Object	Details	Result	Time
"fromPort":Select "New York"	Select "New York"	Done	4/18/2004 - 18:03:43

FLIGHT FINDER

Use our Flight Finder to search for the lowest fare on pa
you've booked your flight, don't forget to visit the Mercury
reserve lodging in your destination city.

Flight Details

Type: ● Round Trip ○ One

Passengers: 1

Departing From: New York

On: Apr 18

Arriving In: Zurich

Returning: Apr 19

The Test Results window now contains three panes, displaying:

- the results tree, with one step highlighted
- the test results details of the highlighted step
- the Active Screen, showing a screen capture of the Web page on which the step was performed.

When you click a page in the results tree, QuickTest displays the corresponding page in the application view. When you click a step (an operation performed on an object) in the results tree, the corresponding object is highlighted in the application view. In this case, the **Departing From** text box is highlighted.

2 Close the Test Results window.

Choose **File** > **Exit**.

You have now successfully created and run a test that reserves a flight from New York to San Francisco on the Mercury Tours Web site.

4

Creating Checkpoints

In the previous lessons, you created and ran a test checking that a series of steps performed on the Mercury Tours Web site runs smoothly. A checkpoint verifies that expected information is displayed in your application while the test is running.

In this lesson you will learn about:

- Understanding Checkpoint Types
- Checking Objects
- Checking Pages
- Checking Text
- Checking Tables
- Running and Analyzing a Test with Checkpoints

Understanding Checkpoint Types

QuickTest Professional offers the following types of checkpoints:

Checkpoint Type	Description	Example of Use
Standard Checkpoint	Checks values of an object's properties.	Check that a radio button is selected.
Image Checkpoint	Checks the property values of an image.	Check that the image source file is correct.
Table Checkpoint	Checks information in a table.	Check that the value in a table cell is correct.
Page Checkpoint	Checks the characteristics of a Web page.	Check how long a Web page takes to load or if a Web page contains broken links.
Text / Text Area Checkpoint	Checks that a text string is displayed in the appropriate place in a Web page or application window.	Check whether the expected text string is displayed in the expected location on a Web page or dialog box.
Bitmap Checkpoint	Checks an area of a Web page or application after capturing it as a bitmap	Check that a Web page (or any portion of it) is displayed as expected.
Database Checkpoint	Checks the contents of databases accessed by an application or Web site	Check that the value in a database query is correct.
Accessibility Checkpoint	Identifies areas of a Web site to check for Section 508 compliancy.	Check if the images on a Web page include ALT properties, required by the W3C Web Content Accessibility Guidelines.
XML Checkpoint	Checks the data content of XML documents.	**Note:** XML file checkpoints are used to check a specified XML file; XML application checkpoints are used to check an XML document within a Web page.

You can add most checkpoints to your test either while recording or afterward. The following sections explain how to create some of the checkpoints described above on the test you created in "Recording Tests" on page 9.

Note: When QuickTest creates a checkpoint, it assigns a name based on information inside the checkpoint—the checked value, for example. The checkpoint name remains unchanged, even if you subsequently modify the information on which it was based. Keep this in mind when looking for checkpoints displayed in the Keyword View. However, note that QuickTest may shorten the name displayed in the Keyword View.

For more information on how to create checkpoints, refer to the *QuickTest Professional User's Guide*.

Checking Objects

In this section, you will add a standard checkpoint in the Book a Flight page. This checkpoint will verify the value in the box containing the first name of the passenger.

1 Start QuickTest and open the Recording test.

If QuickTest is not already open, choose **Start** > **Programs** > **QuickTest Professional** > **QuickTest Professional**.

➤ If the Welcome window opens, click **Open Existing**.

➤ If QuickTest opens without displaying the Welcome window, choose **File** > **Open** or click the **Open** button.

In the Open Test dialog box, locate and select the **Recording** test, then click **Open**.

2 Save the test as Checkpoint.

Select **File** > **Save As**. Save the test as Checkpoint.

3 Confirm that the Active Screen option is enabled.

If you do not see the Active Screen at the bottom of the QuickTest window, click the **Active Screen** button, or choose **View** > **Active Screen**.

4 Locate the page where you want to add a standard checkpoint.

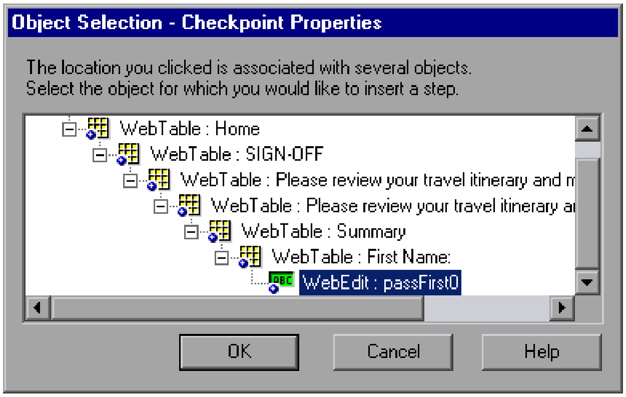
You want to add a checkpoint that checks the property values of the **First Name** edit box after the passenger's name has been entered in the box.

In the **Item** column of the Keyword View, expand **(+) Action1** > **Welcome: Mercury Tours** > **Book a Flight: Mercury**.

The Book a Flight page is displayed in the Active Screen and the object corresponding to the selected step is highlighted. You can see that the passenger's first name is displayed in the **First Name** box.

5 Create a standard checkpoint.

In the Active Screen, right-click the **First Name** box and choose **Insert Standard Checkpoint**. The Object Selection – Checkpoint Properties dialog box opens.



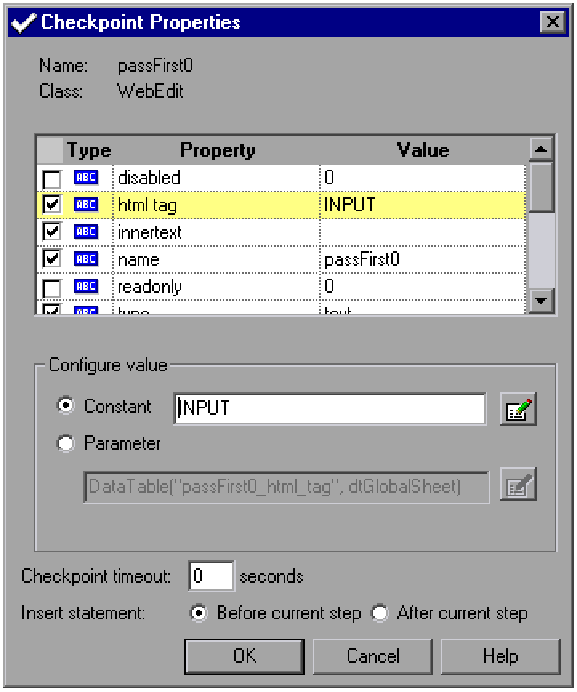
Object Selection - Checkpoint Properties

The location you clicked is associated with several objects.
Select the object for which you would like to insert a step.

WebTable : Home
WebTable : SIGN-OFF
WebTable : Please review your travel itinerary and m
WebTable : Please review your travel itinerary ar
WebTable : Summary
WebTable : First Name:
WebEdit : passFirst0

Confirm that **WebEdit: passFirst0** is highlighted, and click **OK**.

The Checkpoint Properties dialog box opens.



Name: passFirst0
Class: WebEdit

	Type	Property	Value
□	ABC	disabled	0
☑	ABC	html tag	INPUT
☑	ABC	innertext	
☑	ABC	name	passFirst0
□	ABC	readonly	0
☑	ABC	type	text

Configure value

● Constant INPUT

○ Parameter

DataTable("passFirst0_html_tag", dtGlobalSheet)

Checkpoint timeout: 0 seconds

Insert statement: ● Before current step ○ After current step

The dialog box displays the object's properties:

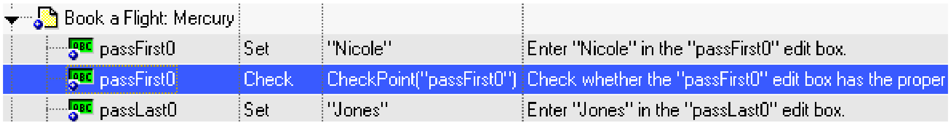
- The **Name** is the name of the object as defined in the HTML code of the Web page.
- The **Class** is the type of object. **WebEdit** indicates that the object is an edit box.
- The **ABC** icon in the type column indicates that the value of the property is a constant.

For each object class, QuickTest recommends default property checks. The table below describes the default checks.

Property	Value	Explanation
html tag	INPUT	**INPUT** is the html tag as defined in the HTML source code.
innertext		In this case, the value of innertext is empty. The checkpoint checks that the value is empty.
name	passFirst0	**passFirst0** is the name of the edit box.
type	text	**text** is the type of object as defined in the HTML source code.
value	<FirstName> (The name you entered while recording)	The value entered in the edit box.

The first name is entered in the First Name box when the **passFirst0 Set**... step is performed. Therefore, in the **Insert statement** area of the Checkpoint Properties dialog box, select **After current step**. This inserts the checkpoint after the **passFirst0 Set**... step in which the first name is entered.

Accept the rest of the settings as default and click **OK**. QuickTest adds a standard checkpoint step to your test below the selected step.



Book a Flight: Mercury			
passFirst0	Set	"Nicole"	Enter "Nicole" in the "passFirst0" edit box.
passFirst0	Check	CheckPoint("passFirst0")	Check whether the "passFirst0" edit box has the proper
passLast0	Set	"Jones"	Enter "Jones" in the "passLast0" edit box.

6 Save the test.

Choose **File** > **Save** or click the **Save** button.

You insert most checkpoints in the manner described above. Several special types of checkpoints are explained in the following sections.

Checking Pages

In this section, you will add a page checkpoint to your test. The page checkpoint checks that the number of links and images in the page when you run your test is the same as when you recorded your test.

1 Locate the page where you want to add a page checkpoint.

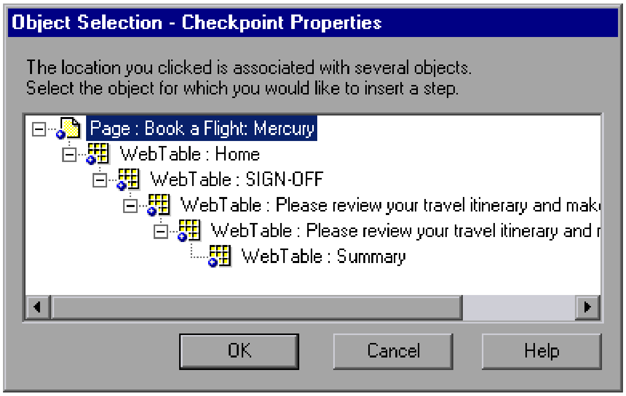
In the Keyword View, expand **(+) Action1** > **Welcome: Mercury Tours**.

Highlight the **Book a Flight: Mercury** row in the Keyword View. The page is displayed in the Active Screen.

2 Create a page checkpoint.

Right-click anywhere in the Active Screen, and choose **Insert Standard Checkpoint**. The Object Selection – Checkpoint Properties dialog box opens. Note that this dialog box may include different elements, depending on where you click in the Active Screen.

Highlight **Page: Book a Flight: Mercury** (the top level) and click **OK**.

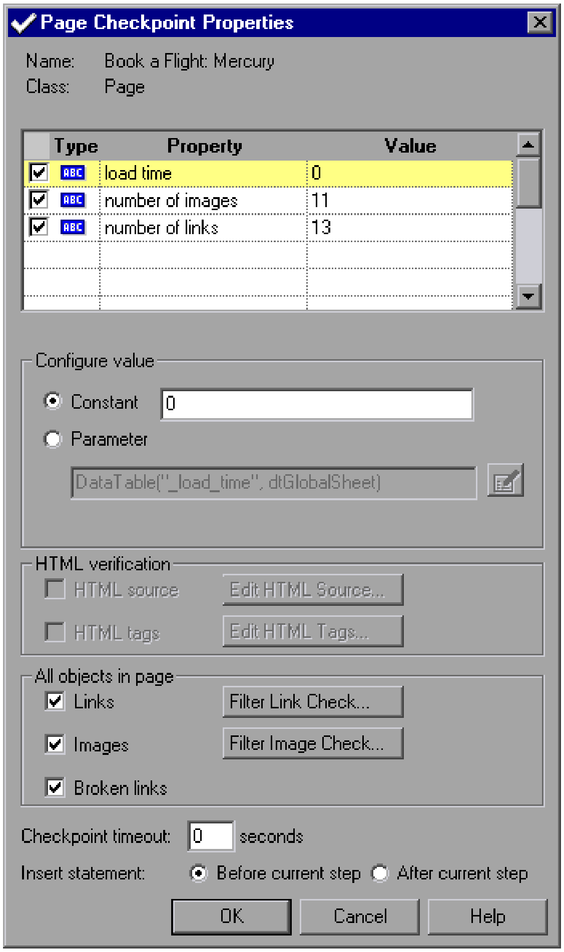


Object Selection - Checkpoint Properties

The location you clicked is associated with several objects.
Select the object for which you would like to insert a step.

- Page : Book a Flight: Mercury
  - WebTable : Home
    - WebTable : SIGN-OFF
      - WebTable : Please review your travel itinerary and mak
        - WebTable : Please review your travel itinerary and r
          - WebTable : Summary

The Page Checkpoint Properties dialog box opens.



Name: Book a Flight: Mercury
Class: Page

	Type	Property	Value
☑	ABC	load time	0
☑	ABC	number of images	11
☑	ABC	number of links	13

Configure value

● Constant 0

○ Parameter DataTable("_load_time", dtGlobalSheet)

HTML verification

□ HTML source

□ HTML tags

All objects in page

☑ Links

☑ Images

☑ Broken links

Checkpoint timeout: 0 seconds

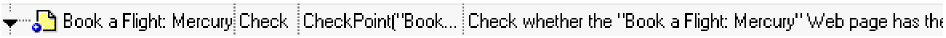
Insert statement: ● Before current step ○ After current step

When you run the test, QuickTest checks the number of links and images in the page and the load time, as described in the pane at the top of the dialog box.

QuickTest also checks the actual target URL of each link and the actual source of each image.

Accept the default settings and click **OK**.

QuickTest adds the page checkpoint to your test. It is displayed in the Keyword View as a checkpoint operation on the **Book a Flight: Mercury** page.



Book a Flight: Mercury | Check | CheckPoint("Book... | Check whether the "Book a Flight: Mercury" Web page has the

3 Save the test.

Choose **File** > **Save** or click the **Save** button.

Checking Text

In this section, you will add a text checkpoint to your test to check whether **New York** is displayed in the **Flight Confirmation** page.

1 Locate the page where you want to add a text checkpoint.

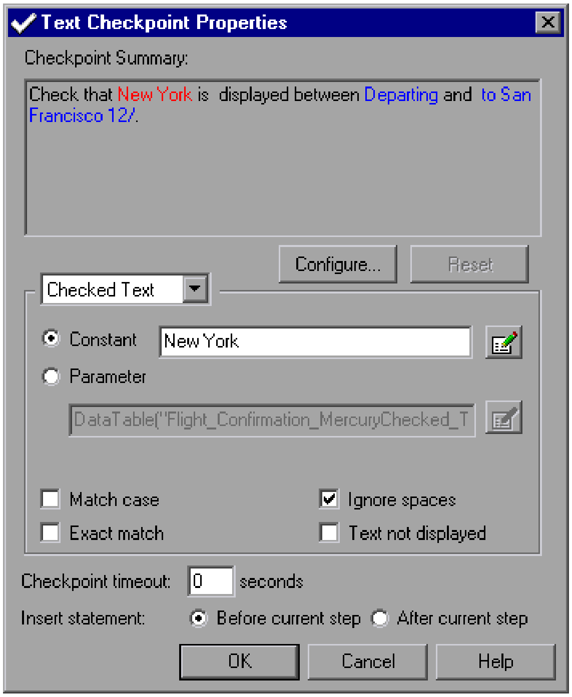
In the Keyword View, expand **(+) Action1** > **Welcome: Mercury Tours**.

Highlight the **Flight Confirmation: Mercury** page in the Keyword View. The page is displayed in the Active Screen.

2 Create a text checkpoint.

In the Active Screen, under "Departing," highlight the text **New York**.

Right-click the highlighted text and choose **Insert Text Checkpoint.** The Text Checkpoint Properties dialog box opens.



Text Checkpoint Properties

Checkpoint Summary:

Check that New York is displayed between Departing and to San Francisco 12/.

Checked Text

● Constant: New York

○ Parameter: DataTable("Flight_Confirmation_MercuryChecked_T

□ Match case ☑ Ignore spaces

□ Exact match □ Text not displayed

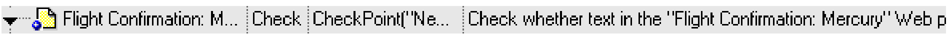
Checkpoint timeout: 0 seconds

Insert statement: ● Before current step ○ After current step

When **Checked Text** appears in the list box, the **Constant** field displays the text string you highlighted. This is the text QuickTest looks for when running the test.

Click **OK** to accept the default settings in this dialog box.

QuickTest adds the text checkpoint to your test. It is displayed in the Keyword View as a checkpoint operation on the **Flight Confirmation: Mercury** page.



Flight Confirmation: M... | Check | CheckPoint("Ne... | Check whether text in the "Flight Confirmation: Mercury" Web p

3 **Save the test.**

Choose **File** > **Save** or click the **Save** button.

Checking Tables

In this section, you will add a table checkpoint to check the cost of the outbound flight, as displayed in the **Book a Flight: Mercury** page.

1 Locate the page where you want to add a table checkpoint.

In the Keyword View, expand **(+) Welcome: Mercury Tours** > **Book a Flight: Mercury**.

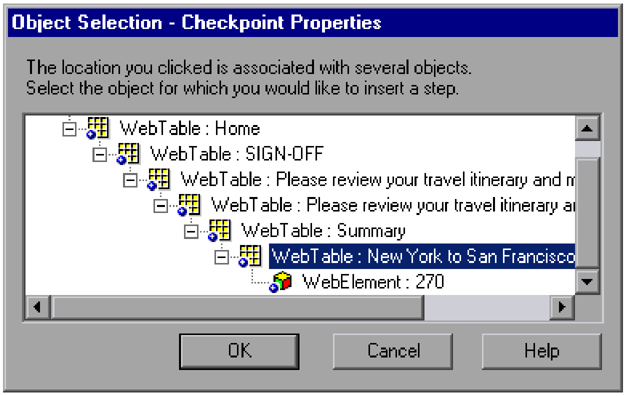
Highlight the **passFirst0** step in the Keyword View. The page is displayed in the Active Screen.

2 Create a table checkpoint.

In the Active Screen, right-click the price displayed for the first flight (New York to San Francisco)—**270**—and choose **Insert Standard Checkpoint**.

The Object Selection – Checkpoint Properties dialog box opens.

Select **WebTable: New York to San Fransisco**.



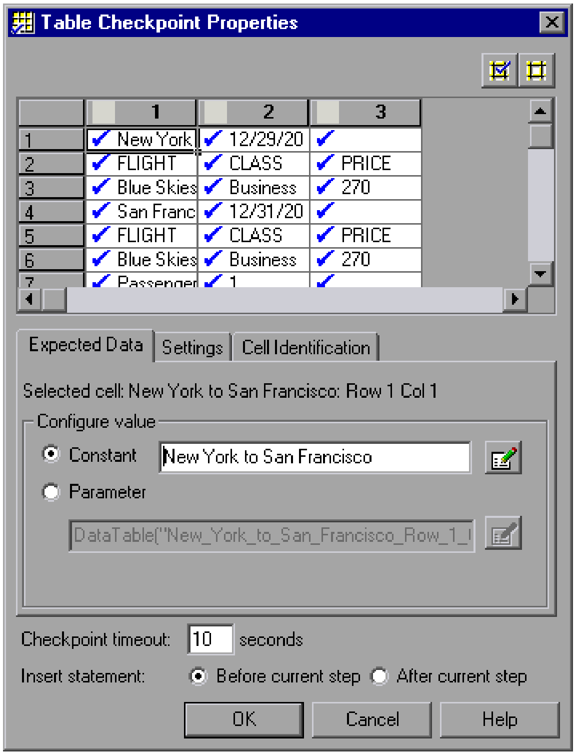
Object Selection - Checkpoint Properties

The location you clicked is associated with several objects.
Select the object for which you would like to insert a step.

- WebTable : Home
  - WebTable : SIGN-OFF
    - WebTable : Please review your travel itinerary and m
      - WebTable : Please review your travel itinerary ar
        - WebTable : Summary
          - WebTable : New York to San Francisco
            - WebElement : 270

Note that the table you selected is highlighted in the Active Screen. Click **OK**.

The Table Checkpoint Properties dialog box opens, showing the rows and columns of the table.



	1	2	3
1	✓ New York	✓ 12/29/20	✓
2	✓ FLIGHT	✓ CLASS	✓ PRICE
3	✓ Blue Skies	✓ Business	✓ 270
4	✓ San Franc	✓ 12/31/20	✓
5	✓ FLIGHT	✓ CLASS	✓ PRICE
6	✓ Blue Skies	✓ Business	✓ 270
7	✓ Passenger	✓ 1	✓

Selected cell: New York to San Francisco: Row 1 Col 1

Configure value

● Constant: New York to San Francisco

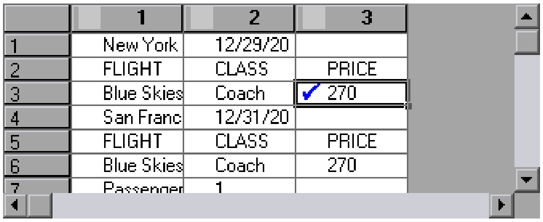
○ Parameter: DataTable("New_York_to_San_Francisco_Row_1_I

Checkpoint timeout: 10 seconds

Insert statement: ● Before current step ○ After current step

Note that by default, check marks appear in all cells. You can double-click a cell to toggle the cell selection, or double-click a row or column header to toggle the selection for all the cells in the selected row or column.

Double-click each column header to clear the check marks. Double-click column 3, row 3 to check the value of this cell. (QuickTest checks only cells containing check marks.)

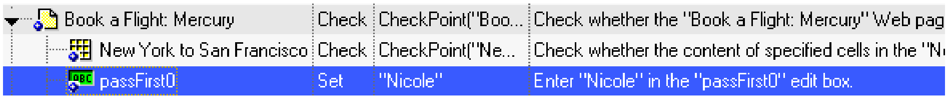


	1	2	3
1	New York	12/29/20	
2	FLIGHT	CLASS	PRICE
3	Blue Skies	Coach	✓ 270
4	San Franc	12/31/20	
5	FLIGHT	CLASS	PRICE
6	Blue Skies	Coach	270
7	Passenger	1	

Tip: You can change the column widths and row heights by dragging the boundaries of the column and row headers.

Click **OK** to close the dialog box.

QuickTest adds the table checkpoint to your test. It is displayed in the Keyword View as a new step under the **Book a Flight: Mercury** page.



Book a Flight: Mercury	Check	CheckPoint("Boo...	Check whether the "Book a Flight: Mercury" Web pag
New York to San Francisco	Check	CheckPoint("Ne...	Check whether the content of specified cells in the "N
passFirst0	Set	"Nicole"	Enter "Nicole" in the "passFirst0" edit box.

3 Save the test.

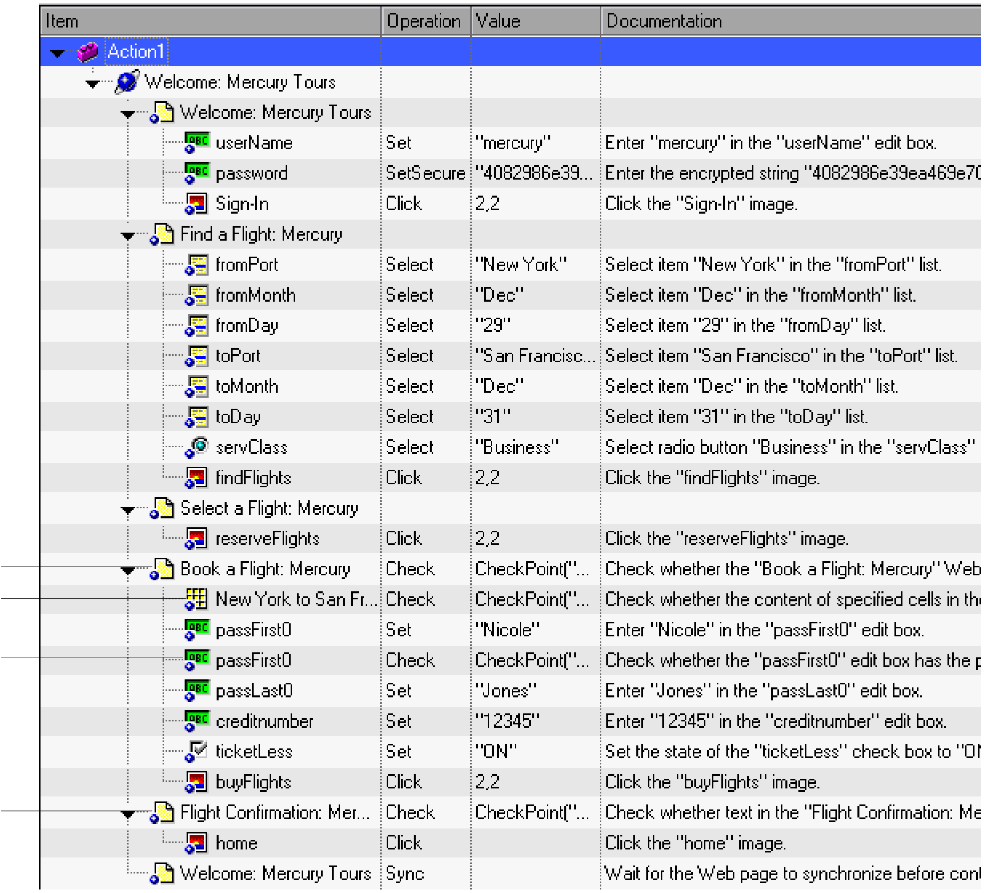
Choose **File** > **Save** or click the **Save** button.

Running and Analyzing a Test with Checkpoints

In this section, you will review your test with checkpoints, run the test, and analyze the checkpoint results.

1 Expand the test and review your test.

Choose **View** > **Expand All** or use the * shortcut key on your number keypad. The Keyword View displayed is similar to the following:



Item	Operation	Value	Documentation
Action1			
Welcome: Mercury Tours			
Welcome: Mercury Tours			
userName	Set	"mercury"	Enter "mercury" in the "userName" edit box.
password	SetSecure	"4082986e39...	Enter the encrypted string "4082986e39ea469e70
Sign-In	Click	2,2	Click the "Sign-In" image.
Find a Flight: Mercury			
fromPort	Select	"New York"	Select item "New York" in the "fromPort" list.
fromMonth	Select	"Dec"	Select item "Dec" in the "fromMonth" list.
fromDay	Select	"29"	Select item "29" in the "fromDay" list.
toPort	Select	"San Francisc...	Select item "San Francisco" in the "toPort" list.
toMonth	Select	"Dec"	Select item "Dec" in the "toMonth" list.
toDay	Select	"31"	Select item "31" in the "toDay" list.
servClass	Select	"Business"	Select radio button "Business" in the "servClass"
findFlights	Click	2,2	Click the "findFlights" image.
Select a Flight: Mercury			
reserveFlights	Click	2,2	Click the "reserveFlights" image.
Book a Flight: Mercury	Check	CheckPoint("...	Check whether the "Book a Flight: Mercury" Web
New York to San Fr...	Check	CheckPoint("...	Check whether the content of specified cells in th
passFirst0	Set	"Nicole"	Enter "Nicole" in the "passFirst0" edit box.
passFirst0	Check	CheckPoint("...	Check whether the "passFirst0" edit box has the p
passLast0	Set	"Jones"	Enter "Jones" in the "passLast0" edit box.
creditnumber	Set	"12345"	Enter "12345" in the "creditnumber" edit box.
ticketLess	Set	"ON"	Set the state of the "ticketLess" check box to "ON
buyFlights	Click	2,2	Click the "buyFlights" image.
Flight Confirmation: Mer...	Check	CheckPoint("...	Check whether text in the "Flight Confirmation: Me
home	Click		Click the "home" image.
Welcome: Mercury Tours	Sync		Wait for the Web page to synchronize before cont

Page checkpoint

Table checkpoint

Standard checkpoint

Text checkpoint

2 Start running your test.

Click **Run** or choose **Test** > **Run**. The Run dialog box opens. Ensure that **New run results folder** is selected. Accept the default results folder name.

Click **OK**. When the test run is completed, the Test Results window opens.

3 View the test results.

When QuickTest finishes running the test, the Test Results window opens. The test result should be **Passed**, indicating that all checkpoints passed. If one or more checkpoints had failed, the test result would be **Failed**.

4 View the results of the page checkpoint.

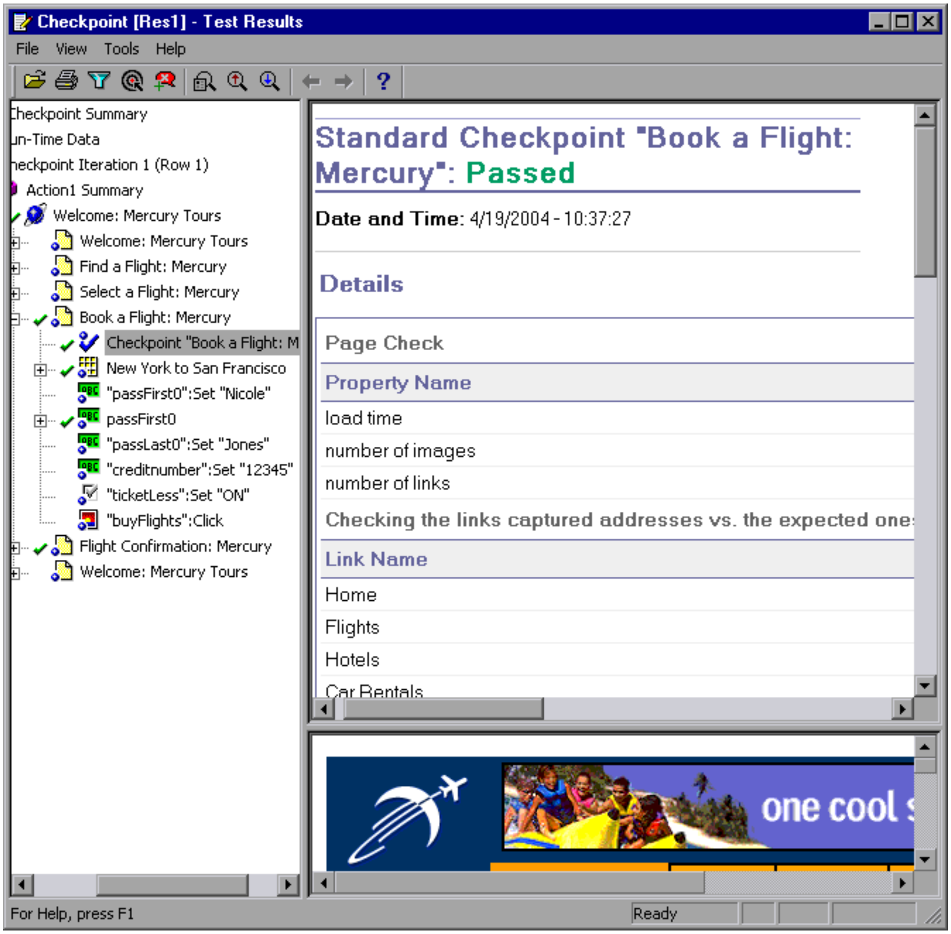
In the results tree, expand **(+) Checkpoint Iteration 1 (Row 1)** > **Action1 Summary** > **Welcome: Mercury Tours** > **Book a Flight: Mercury**.

Highlight **Checkpoint "Book a Flight: Mercury"**.

In the Details pane, you can review the details of the page checkpoint, which lists the items checked.

Tip: You can drag the bottom of the Details pane to enlarge it and/or scroll within the pane to view all of the checkpoint details.

The checkpoint passed because the actual values of the checked properties match the expected values.



heckpoint Summary
un-Time Data
heckpoint Iteration 1 (Row 1)
Action1 Summary
Welcome: Mercury Tours
Welcome: Mercury Tours
Find a Flight: Mercury
Select a Flight: Mercury
Book a Flight: Mercury
Checkpoint "Book a Flight: M
New York to San Francisco
"passFirst0":Set "Nicole"
passFirst0
"passLast0":Set "Jones"
"creditnumber":Set "12345"
"ticketLess":Set "ON"
"buyFlights":Click
Flight Confirmation: Mercury
Welcome: Mercury Tours

Standard Checkpoint "Book a Flight: Mercury": Passed

Date and Time: 4/19/2004 - 10:37:27

Details

Page Check

Property Name
load time
number of images
number of links

Checking the links captured addresses vs. the expected ones

Link Name
Home
Flights
Hotels
Car Rentals

one cool s

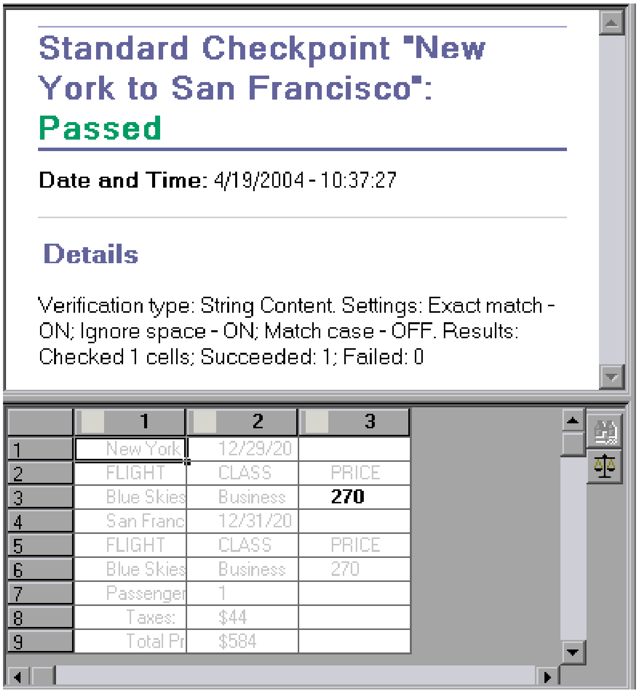
5 View the results of the table checkpoint.

In the results tree, **Book a Flight:** page, expand **(+) New York to San Francisco**.

Highlight **Checkpoint** "**New York to San Francisco**".

In the Details pane, you can review the details of the table checkpoint. You can also review the values of the table cells (cell values that were checked are displayed in black; cell values that were not checked are displayed in gray).

The checkpoint passed because the actual value of the checked cell matches the expected value.



Standard Checkpoint "New York to San Francisco": Passed

Date and Time: 4/19/2004 - 10:37:27

Details

Verification type: String Content. Settings: Exact match - ON; Ignore space - ON; Match case - OFF. Results: Checked 1 cells; Succeeded: 1; Failed: 0

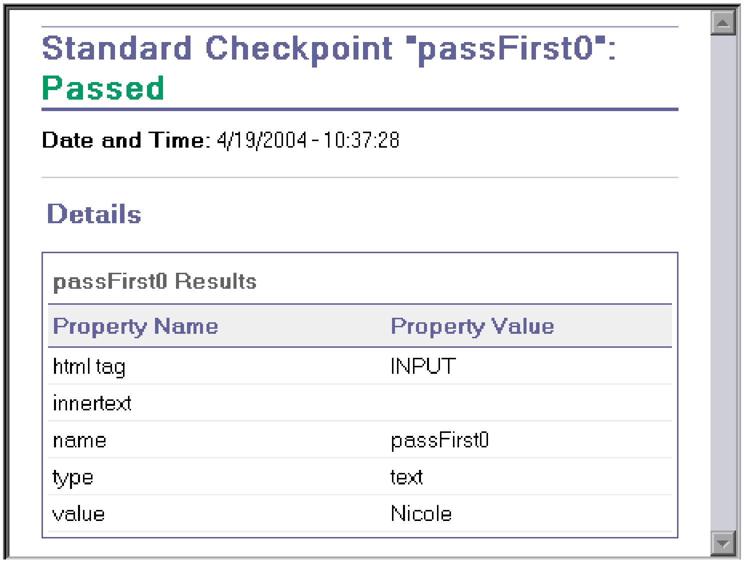
	1	2	3
1	New York	12/29/20	
2	FLIGHT	CLASS	PRICE
3	Blue Skies	Business	**270**
4	San Franc	12/31/20	
5	FLIGHT	CLASS	PRICE
6	Blue Skies	Business	270
7	Passenger	1	
8	Taxes:	$44	
9	Total Pr	$584	

6 View the results of the standard checkpoint.

In the results tree, **Book a Flight: Mercury** page, expand **(+) passFirst0**.

Highlight **Checkpoint "passFirst0"**.

In the Details pane, you can review the details of the standard checkpoint, which lists the properties that were checked and their values. The checkpoint passed because the actual values of the checked properties match the expected values.



Standard Checkpoint "passFirst0": Passed

Date and Time: 4/19/2004 - 10:37:28

Details

passFirst0 Results

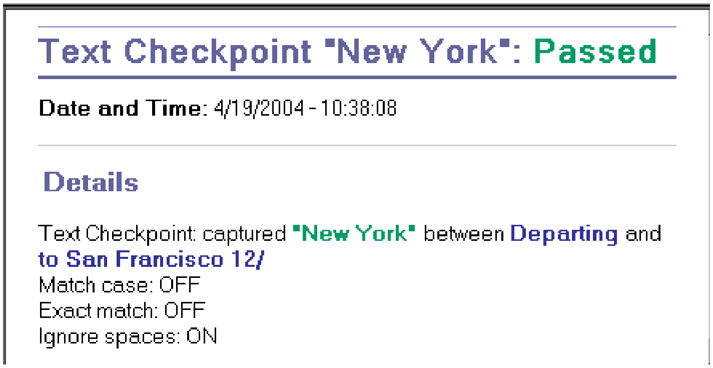
Property Name	Property Value
html tag	INPUT
innertext	
name	passFirst0
type	text
value	Nicole

7 View the results of the text checkpoint.

In the results tree, expand **(+) Checkpoint Iteration 1 (Row 1)** > **Action1 Summary** > **Welcome: Mercury Tours**> **Flight Confirmation: Mercury.**

Highlight **Checkpoint** "**New York**".

In the Details pane, you can review the details of the text checkpoint. The checkpoint passed because the actual text matches the expected text.



Text Checkpoint "New York": Passed

Date and Time: 4/19/2004 - 10:38:08

Details

Text Checkpoint: captured **"New York"** between **Departing** and **to San Francisco 12/**
Match case: OFF
Exact match: OFF
Ignore spaces: ON

8 Close the Test Results window.

Choose **File** > **Exit**.

5

Parameterizing Tests

When you test your applications, you may want to check how the application performs the same operations with multiple sets of data. For example, suppose you want to check how your Web site responds to ten separate sets of data. You could record ten separate tests, each with its own set of data. Alternatively, you can create Data Table parameters so that your test runs ten times, each time using a different set of data.

In this lesson you will learn about:

- Defining a Data Table Parameter
- Adding Parameter Values to a Data Table
- Modifying Steps Affected by Parameterization
- Running and Analyzing a Parameterized Test

Defining a Data Table Parameter

In the previous lessons, you reserved a flight from New York to San Francisco. New York is a constant value, which means that New York is the departure city each time you run the test. In this exercise you will make the departure city a parameter so that you can use a different departure city for each test run.

1 Start QuickTest and open the Checkpoint test.

If QuickTest is not already open, choose **Start** > **Programs** > **QuickTest Professional** > **QuickTest Professional**.

- If the Welcome window opens, click **Open Existing**.
- If QuickTest opens without displaying the Welcome window, choose **File** > **Open** or click the **Open** button.

In the Open Test dialog box, locate and select the **Checkpoint** test, then click **Open**.

2 Save the test as Parameter.

Select **File** > **Save As**. Save the test as Parameter.

3 Confirm that the Active Screen option is enabled.

If you do not see the Active Screen at the bottom of the QuickTest window, click the **Active Screen** button, or choose **View** > **Active Screen**.

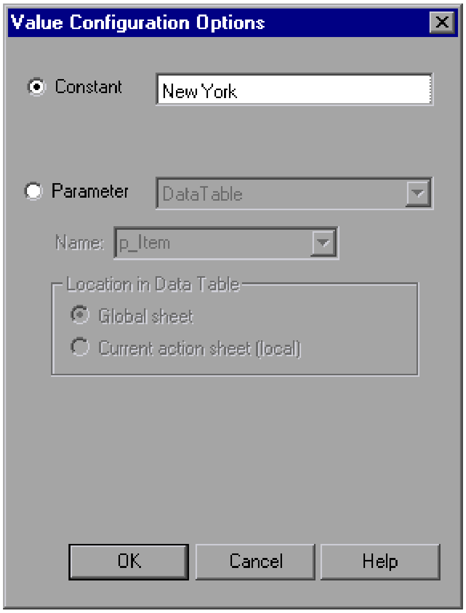
4 Confirm that the Data Table option is enabled.

If you do not see the Data Table at the bottom of the QuickTest window, click the **Data Table** button, or choose **View** > **Data Table**.

5 Select the text to parameterize.

In the Keyword View, expand (+) **Action1** > **Welcome: Mercury Tours** > **Find a Flight: Mercury**.

In the **fromPort** row in the Keyword View, click in the **Value** cell and then click the parameterization icon . The Value Configuration Options dialog box opens.


Value Configuration Options

● Constant: New York

○ Parameter: DataTable

Name: p_Item

Location in Data Table

● Global sheet

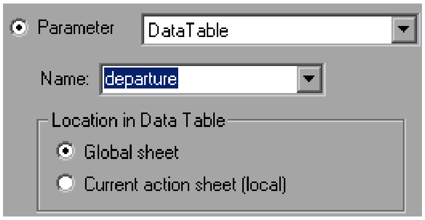
○ Current action sheet (local)

6 Set the parameterization properties.

Select the **Parameter** radio button. This lets you replace the constant value (**New York**) with a parameter.

Confirm that the **DataTable** option is selected. This means that the value of the parameter will be taken from the QuickTest Data Table.

The **Name** box is enabled and displays **p_Item**. Change this to departure.


● Parameter: DataTable

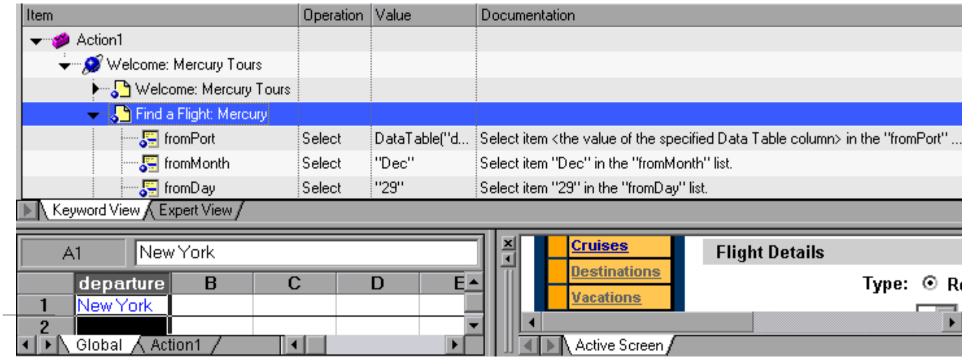
Name: departure

Location in Data Table

● Global sheet

○ Current action sheet (local)

Click **OK** to close the dialog box. QuickTest adds the **departure** parameter to the Data Table as a new column and inserts New York in the first row under it. New York will be the first of several departure cities that QuickTest will use during test runs of the application.



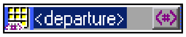
Item	Operation	Value	Documentation
Action1			
Welcome: Mercury Tours			
Welcome: Mercury Tours			
Find a Flight: Mercury			
fromPort	Select	DataTable("d...	Select item <the value of the specified Data Table column> in the "fromPort" ...
fromMonth	Select	"Dec"	Select item "Dec" in the "fromMonth" list.
fromDay	Select	"29"	Select item "29" in the "fromDay" list.

A1 New York

	departure	B	C	D	E
1	New York				
2					

New column in Data Table

Note the change in the step's appearance in the Keyword View. Previously, the step was displayed as fromPort Select New York. Now, the step is displayed as fromPort Select DataTable("departure", DTGlobalSheet). When you click the **Value** cell, the following information is displayed, indicating that the value is parameterized using a Data Table parameter called departure:



<departure>

Adding Parameter Values to a Data Table

As you saw, QuickTest displays parameter values in the Data Table. In this section, you will add two more departure cities to the Data Table, so that QuickTest can test the application with this data.

1 Enter additional cities in the departure column.

Click row 2 in the departure column and type Portland.

Click row 3 and type Seattle.

Press **Enter**.

2 Save the test.

Select **File** > **Save** or click the **Save** button.

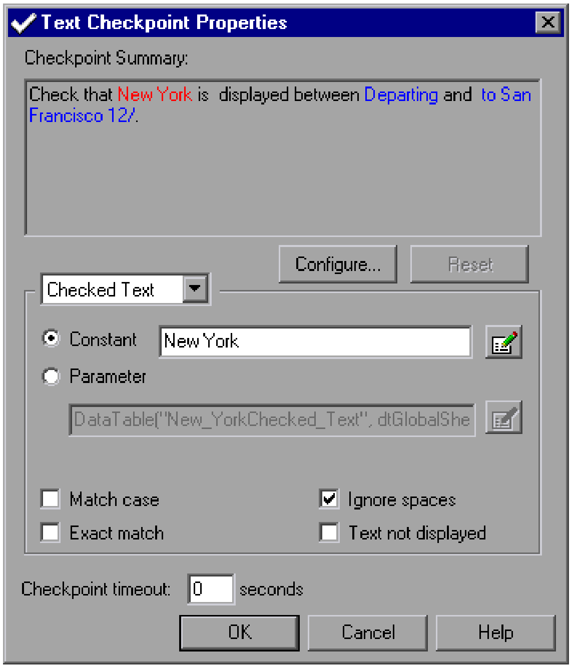
Modifying Steps Affected by Parameterization

After parameterizing one step in a test, other test objects might be affected when the value of the parameterized step changes. If so, you must modify the expected values of those objects to match the value resulting from the parameterized step. In this section, you will modify the text checkpoint so that when running the test, QuickTest checks for the text that matches the current departure city.

1 Locate the text checkpoint to modify.

In the Keyword View, expand (+) **Welcome: Mercury Tours**.

Right-click **Flight Confirmation: Mercury** and select **Checkpoint Properties**. The Text Checkpoint Properties dialog box opens.



Text Checkpoint Properties

Checkpoint Summary:

Check that New York is displayed between Departing and to San Francisco 12/.

Checked Text

● Constant: New York

○ Parameter: DataTable("New_YorkChecked_Text", dtGlobalShe

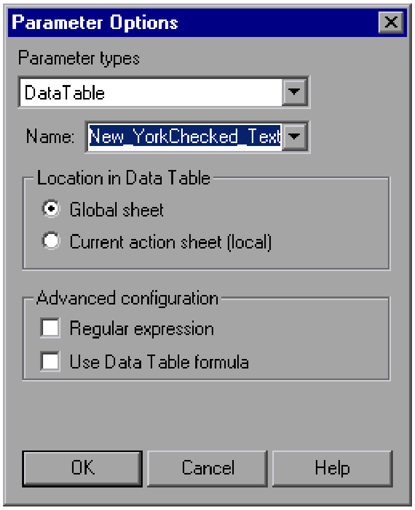
□ Match case
☑ Ignore spaces
□ Exact match
□ Text not displayed

Checkpoint timeout: 0 seconds

2 Parameterize the text checkpoint.

In the **Checked Text** area, **New York** is displayed in the **Constant** box. **New York** is the expected value for the checkpoint for every iteration.

Select **Parameter** and click the **Parameter Options** button. The Parameter Options dialog box opens.



Parameter Options

Parameter types: DataTable

Name: New_YorkChecked_Text

Location in Data Table

● Global sheet

○ Current action sheet (local)

Advanced configuration

□ Regular expression

□ Use Data Table formula

In the **Name** box, select **departure**. This instructs the checkpoint to use the departure parameter values in the Data Table for the expected results.

Click **OK** to close the Parameter Options dialog box, and then click **OK** again to close the Text Checkpoint Properties dialog box. The checkpoint is now parameterized.

3 Save the test.

Select **File** > **Save** or click the **Save** button.

Running and Analyzing a Parameterized Test

You will now run the modified **Parameter** test. QuickTest will run the test three times, once for each departure city in the Data Table. Each test run is called an *iteration*.

1 **Run the Parameter test.**

Click **Run** on the Testing toolbar or choose **Test** > **Run**. The Run dialog box opens.

Select **New run results folder** and accept the default results folder name.

Click **OK**. When the test run is completed, the Test Results window opens.

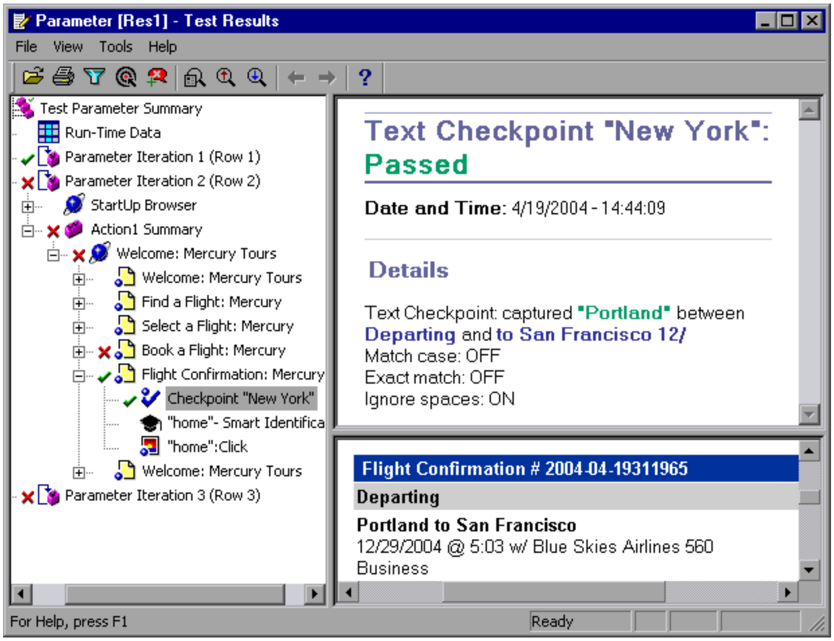
2 **Examine the results summary.**

The Test Results window shows that the second and third iterations of the test failed, even though the text checkpoint passed in all three iterations. See below for further information about why the iterations failed.

➤ Iteration 2:

In the results tree, expand (+) **Parameter Iteration 2** > **Action1 Summary** > **Welcome Mercury Tours** > **Flight Confirmation: Mercury**.

Highlight **Checkpoint "New York"**.



Text Checkpoint "New York": Passed

Date and Time: 4/19/2004 - 14:44:09

Details

Text Checkpoint: captured **"Portland"** between **Departing** and **to San Francisco 12/**
Match case: OFF
Exact match: OFF
Ignore spaces: ON

Flight Confirmation # 2004-04-19311965
Departing
Portland to San Francisco
12/29/2004 @ 5:03 w/ Blue Skies Airlines 560
Business

The checkpoint results in the Details pane show that Portland was the expected result and the actual result. Therefore, the checkpoint passed. You can also see, in the Application pane, that the flight information reflects the flight from Portland.

➤ Iteration 3:

In the results tree, expand (+) **Parameter Iteration 3** > **Action1 Summary** > **Welcome Mercury Tours** > **Flight Confirmation: Mercury**.

Highlight **Checkpoint "New York"**.

The checkpoint results in the Details pane show that Seattle was the expected result and the actual result. Therefore, the checkpoint passed. The Application pane shows the flight information for the flight from Seattle.

The table below summarizes the results of the text checkpoint in each iteration:

Iteration	Expected Text	Actual Text	Result
#1	New York	New York	Checkpoint Passed
#2	Portland	Portland	Checkpoint Passed
#3	Seattle	Seattle	Checkpoint Passed

Note: Although the text checkpoints passed in all three iterations, the second and third iterations of the test still failed because the table checkpoint failed in these iterations—the ticket price changed when the departure city changed. In the next lesson, you will modify the table checkpoint so that when you run your test, QuickTest will update the expected ticket price in the checkpoint with the correct ticket price for the current departure city.

3 Close the Test Results window.

Choose **File** > **Exit** to close the Test Results window.

6

Creating Output Values

In the previous lesson, you created parameters that inserted different data into each iteration of a test run. You can also retrieve data from your application and output it to the Data Table, using output values. This data can then be used at a later stage in the test. QuickTest displays the retrieved data, following the test run, in the Runtime Data Table.

For example, you can use an output value to verify that the date or flight number is correctly displayed in two different Web pages, by using the value obtained in one page as the expected text that QuickTest checks for in the other page.

In this lesson you will learn about:

- ➤ Creating an Output Value
- ➤ Running and Analyzing a Test with Output Values

Creating an Output Value

In the previous lesson, the second and third iterations of your test failed because the ticket price changed when the departure city changed. The checkpoint that checked the fare of the outbound flight in the **Book a Flight: Mercury** page did not update its expected value as the fare changed.

In this lesson, you will create an output value that retrieves the outbound fare from the **Select a Flight: Mercury** page in each test iteration. You will then modify the table checkpoint you created, so that it checks that the price displayed in the **Book a Flight: Mercury** page matches the price captured in the **Select a Flight: Mercury** page.

1 Start QuickTest and open the Parameter test.

If QuickTest is not already open, choose **Start** > **Programs** > **QuickTest Professional** > **QuickTest Professional**.

- If the Welcome window opens, click **Open Existing**.
- If QuickTest opens without displaying the Welcome window, choose **File** > **Open** or click the **Open** button.

In the Open Test dialog box, locate and select the **Parameter** test, then click **Open**.

2 Save the test as Output.

Select **File** > **Save As**. Save the test as Output.

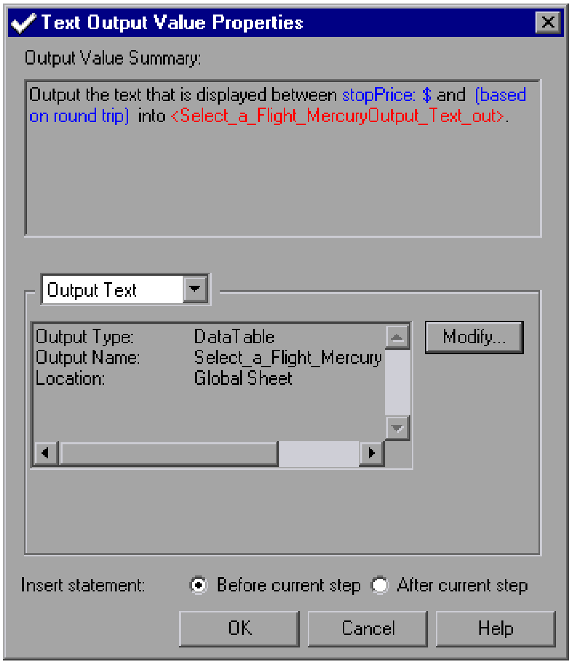
3 Confirm that the Active Screen option is enabled.

If you do not see the Active Screen at the bottom of the QuickTest window, click the **Active Screen** button, or choose **View** > **Active Screen**.

4 Select the text you want to use as an output value.

In the Keyword View, expand (+) **Welcome: Mercury Tours** and click the **Select a Flight: Mercury** page. The page is displayed in the Active Screen.

In the Active Screen, highlight the fare, without the **$** sign, for the outbound flight (New York to San Francisco)—**270**. Right-click the highlighted text and select **Insert Text Output**. The Text Output Value Properties dialog box opens.



Output Value Summary:

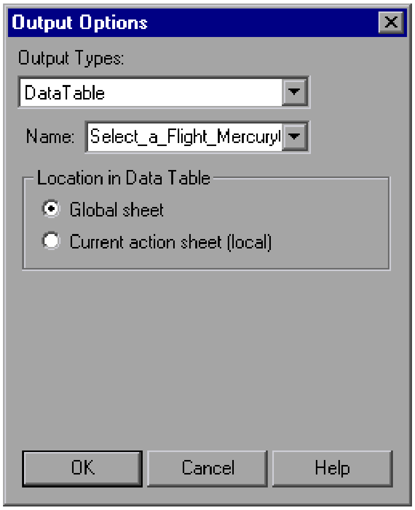
Output the text that is displayed between stopPrice: $ and (based on round trip) into <Select_a_Flight_MercuryOutput_Text_out>.

Output Type: DataTable
Output Name: Select_a_Flight_Mercury
Location: Global Sheet

Insert statement: ● Before current step ○ After current step

5 Set the output value settings.

Click the **Modify** button to open the Output Options dialog box.



Output Options

Output Types: DataTable

Name: Select_a_Flight_Mercuryl

Location in Data Table

● Global sheet

○ Current action sheet (local)

The **Name** box displays **Select_a_Flight_MercuryOutput_Text_out**. Change this to depart_flight_price.

Click **OK** to accept the other default settings and close the Output Options dialog box. Click **OK** again to close the Text Output Value Properties dialog box. QuickTest adds the **depart_flight_price** column to the Data Table.

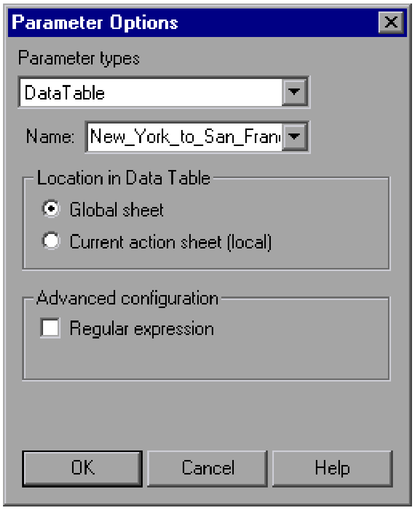
QuickTest adds the output value operation to the **Select a Flight: Mercury** page in the Keyword View.

6 Modify the table checkpoint that checks the ticket price displayed in the flight Summary table.

In the Keyword View, expand (+) **Welcome:Mercury Tours** >
Book a Flight: Mercury.

Right-click **Checkpoint "New York to San Francisco"**, and select **Checkpoint Properties**. The Table Checkpoint Properties dialog box opens.

Click column 3 in row 3 (the checked cell). In the **Configure value** area, select **Parameter** and click the **Parameter Options** button. The Parameter Options dialog box opens.



Parameter Options

Parameter types: DataTable

Name: New_York_to_San_Fran

Location in Data Table

● Global sheet

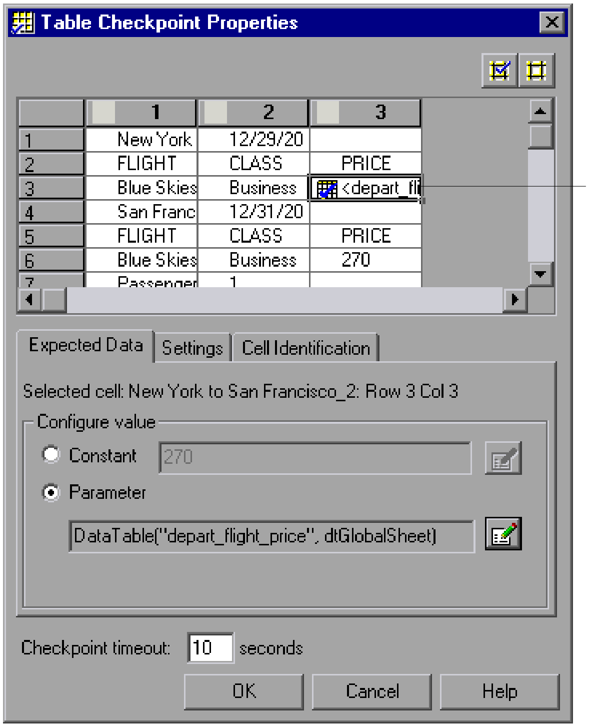
○ Current action sheet (local)

Advanced configuration

□ Regular expression

In the **Name** box, select **depart_flight_price**.

Click **OK** to close the Parameter Options dialog box. The Table Checkpoint Properties dialog box is redisplayed, with a parameterization icon next to the table value you parameterized.



	1	2	3
1	New York	12/29/20	
2	FLIGHT	CLASS	PRICE
3	Blue Skies	Business	<depart_fli
4	San Franc	12/31/20	
5	FLIGHT	CLASS	PRICE
6	Blue Skies	Business	270
7	Passenger	1	

Selected cell: New York to San Francisco_2: Row 3 Col 3

Configure value

○ Constant 270

● Parameter

DataTable("depart_flight_price", dtGlobalSheet)

Checkpoint timeout: 10 seconds

Parameterized value

Click **OK** to close the Table Checkpoint Properties dialog box. The checkpoint is now parameterized.

7 Save the test.

Select **File** > **Save** or click the **Save** button.

Running and Analyzing a Test with Output Values

You will now run the test and examine the results.

1 Run the Output test.

▶ Run

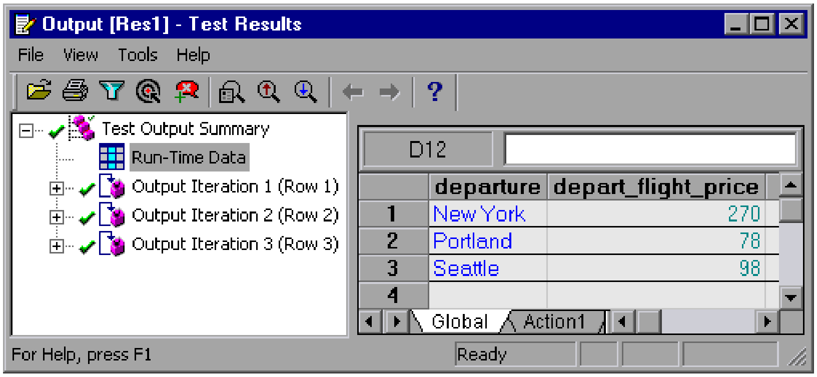
Click the **Run** button or choose **Test** > **Run**. The Run dialog box opens.

Select **New run results folder** and accept the default results folder name.

Click **OK**. When the test run is completed, the Test Results window opens.

2 Examine the run-time data results.

In the Test Results window, select **Run-Time Data** from the results tree. The output values used during the test run are displayed in a grid. Note that a different price is shown in the **depart_flight_price** column for each iteration.



- Test Output Summary
  - Run-Time Data
  - Output Iteration 1 (Row 1)
  - Output Iteration 2 (Row 2)
  - Output Iteration 3 (Row 3)

	departure	depart_flight_price
1	New York	270
2	Portland	78
3	Seattle	98
4		

3 Examine the checkpoint results.

Choose **View** > **Expand All**.

In **Output Iteration 1 (Row 1)**, under the **Book a Flight: Mercury** page, click **Checkpoint "New York to San Francisco"**.

Note that in the Details pane, the checkpoint is marked in green as passed because the price of the ticket for the first flight in the Summary table matched the expected value obtained from the Select a Flight page.

Examine the checkpoints for Iterations 2 and 3 as well. In the Details pane, these checkpoints are also marked in green as "Passed," for the same reason.

4 Close the Test Results window.

Choose **File** > **Exit** to close the Test Results window.

7

Using Regular Expressions

In Lesson 4, "Creating Checkpoints," you created a text checkpoint that searched for a specific text string. You can use regular expressions to increase the flexibility and adaptability of your tests.

In this lesson you will learn about:

- Regular Expression Syntax
- Working with Regular Expressions
- Running and Analyzing a Test with Regular Expressions

Regular Expression Syntax

Regular expressions enable QuickTest to identify objects and text strings with varying values. You can use regular expressions when defining the properties of an object, the methods of an argument, when parameterizing a step, and when creating checkpoints with varying values.

A regular expression is a string that specifies a complex search phrase. By using special characters such as a period (.), asterisk (*), caret (^), and brackets ([ ]), you define the conditions of the search. For more information on regular expression syntax, refer to the *QuickTest Professional User's Guide*.

Working with Regular Expressions

In this tutorial, you will create a text checkpoint on a date text string that changes according to the selected flight date. You can define the date as a regular expression so that the checkpoint checks that the captured text string matches the expected format, rather than checking the exact text.

To do this, you will create a text checkpoint with a regular expression that will match any single character within a defined range.

1 Start QuickTest and open the Recording test.

If QuickTest is not already open, choose **Start** > **Programs** > **QuickTest Professional** > **QuickTest Professional**.

➤ If the Welcome window opens, click **Open Existing**.

➤ If QuickTest opens without displaying the Welcome window, choose **File** > **Open** or click the **Open** button.

In the Open Test dialog box, locate and select the **Recording** test, then click **Open**.

2 Save the test as RegExpression.

Select **File** > **Save As**. Save the test as RegExpression.

3 Confirm that the Active Screen option is enabled.

If you do not see the Active Screen at the bottom of the QuickTest window, click the **Active Screen** button, or choose **View** > **Active Screen**.

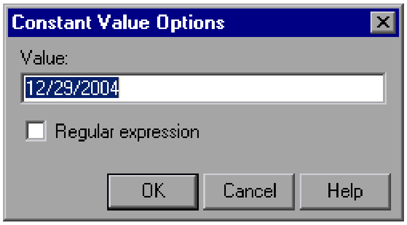
4 Select the text for which you will create the checkpoint.

In the Keyword View, expand (+) **Welcome: Mercury: Tours** and click the **Select a Flight: Mercury** page. The page is displayed in the Active Screen.

In the Active Screen, scroll up and highlight the date for the outbound flight, New York to San Francisco (12/29/2004). Right-click the highlighted string and select **Insert Text Checkpoint**. The Text Checkpoint Properties dialog box opens.

5 Enter the regular expression for the text you want to check.

In the **Checked Text** area, click the **Constant Value Options** button. The Constant Value Options dialog box opens.



Value: 12/29/2004

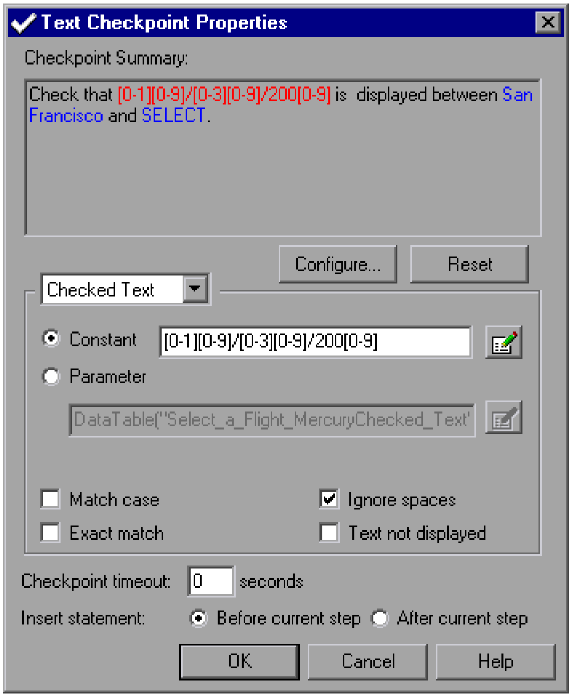
☐ Regular expression

In the **Value** box, replace the current date with [0-1][0-9]/[0-3][0-9]/200[0-9]. This instructs QuickTest to check that each character in the selected text matches the number-range format defined by the regular expression. The expression checks for the following format: MM/DD/200Y.

Select the **Regular expression** check box. A message box prompts you to insert the backslash character before each special character. Click **No**.

Note: Clicking **Yes** would cause QuickTest to treat the special characters ([ ), (-) and ( ]) as literal characters and not as a regular expression.

Click **OK** to accept the other default settings and close the Constant Value Options dialog box. The Text Checkpoint Properties dialog box opens.



Text Checkpoint Properties

Checkpoint Summary:

Check that [0-1][0-9]/[0-3][0-9]/200[0-9] is displayed between San Francisco and SELECT.

Checked Text

● Constant [0-1][0-9]/[0-3][0-9]/200[0-9]

○ Parameter

DataTable("Select_a_Flight_MercuryChecked_Text"

□ Match case ☑ Ignore spaces

□ Exact match □ Text not displayed

Checkpoint timeout: 0 seconds

Insert statement: ● Before current step ○ After current step

Click **OK** to close the Text Checkpoint Properties dialog box.

QuickTest adds the text checkpoint to your test. It is displayed as a check operation on the **Select a Flight: Mercury** page in the Keyword View.

6 Save the test.

Select **File** > **Save** or click the **Save** button.

Running and Analyzing a Test with Regular Expressions

In this exercise you will run the test and examine the checkpoint results.

1 Run the RegExpression test.

▶ Run

Click the **Run** button or choose **Test** > **Run**. The Run dialog box opens.

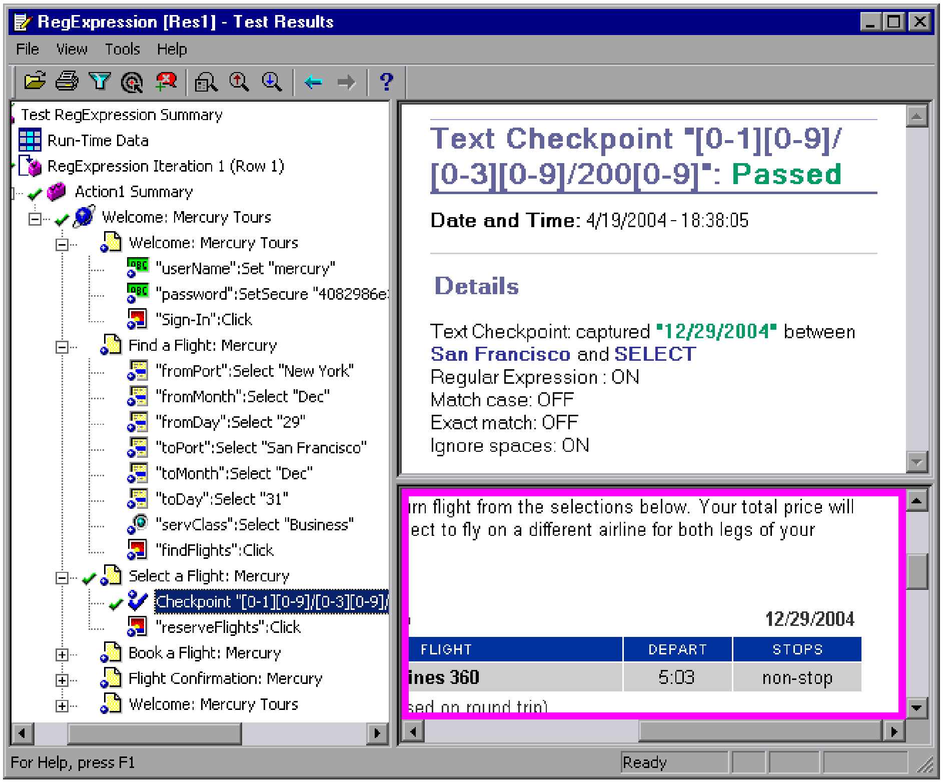
Select **New run results folder** and accept the default results folder name.

Click **OK**. When the test run is completed, the Test Results window opens.

2 Examine the checkpoint results.

In the results tree, expand (+) **Test RegExpression Summary** > **RegExpression Iteration 1 (Row 1)** > **Action1 Summary** > **Welcome: Mercury Tours** > **Select a Flight: Mercury**

Select **CheckPoint** "**[0-1][0-9]/[0-3][0-9]/200[0-9]**".



Test RegExpression Summary
- Run-Time Data
- RegExpression Iteration 1 (Row 1)
  - Action1 Summary
    - Welcome: Mercury Tours
      - Welcome: Mercury Tours
        - "userName":Set "mercury"
        - "password":SetSecure "4082986e
        - "Sign-In":Click
      - Find a Flight: Mercury
        - "fromPort":Select "New York"
        - "fromMonth":Select "Dec"
        - "fromDay":Select "29"
        - "toPort":Select "San Francisco"
        - "toMonth":Select "Dec"
        - "toDay":Select "31"
        - "servClass":Select "Business"
        - "findFlights":Click
      - Select a Flight: Mercury
        - Checkpoint "[0-1][0-9]/[0-3][0-9]/
        - "reserveFlights":Click
      - Book a Flight: Mercury
      - Flight Confirmation: Mercury
      - Welcome: Mercury Tours

Text Checkpoint "[0-1][0-9]/[0-3][0-9]/200[0-9]": Passed

Date and Time: 4/19/2004 - 18:38:05

Details

Text Checkpoint: captured "12/29/2004" between San Francisco and SELECT
Regular Expression : ON
Match case: OFF
Exact match: OFF
Ignore spaces: ON

rn flight from the selections below. Your total price will
ect to fly on a different airline for both legs of your

12/29/2004

FLIGHT	DEPART	STOPS
ines 360	5:03	non-stop

sed on round trip)

The checkpoint passed because the text was displayed in the format specified by the regular expression.

3 **Exit the Test Results window.**

Choose **File** > **Exit** to close the Test Results window.

8

Dividing Tests into Multiple Actions

Actions divide your test into logical sections. When you create a new test, it contains a call to one action. By dividing your tests into calls to multiple actions, you can design more modular and efficient tests.

In this lesson you will learn about:

- ➤ Working with Multiple Actions
- ➤ Creating New Actions
- ➤ Inserting Existing Actions
- ➤ Parameterizing an Action
- ➤ Running and Analyzing a Multi-action Test

Working with Multiple Actions

If you examine one of the tests you created in the previous lessons, you will see that it can be divided into several distinct processes:

- ➤ You logged into the Mercury Tours site.
- ➤ You submitted a flight order.
- ➤ You logged out.

Assume that you wanted to run your test for five different flight orders. As we saw in Lesson 5, "Parameterizing Tests", you could parameterize your test so that it would run the test five times using five different sets of data. You can also organize your test so that only the second procedure runs five times, simulating a single user logging in, ordering five flights, and logging out. You do this by dividing your test into calls to different actions.

To divide your test into calls to different actions, you can insert a call to a new action, split an existing action into calls to two actions, insert a call to a copy of an existing action, or insert a call to an existing action.

You can insert calls to actions into your test during your recording session or afterwards. Use one of the following menu options or toolbar buttons to add actions to your test:

- **Insert** > **Call to New Action** or use the **Insert Call to New Action** button.
- **Step** > **Split Action** or use the **Split Action** button.
- **Insert** > **Call to Copy of Action** or right-click an action and choose **Insert Call to Copy of Action**.
- **Insert** > **Call to Existing Action** or right-click an action and choose **Insert Call to Existing Action**.

Creating New Actions

In this exercise you will create a test and divide it into three action calls.

Recording the Test

1 Start QuickTest and open a new test.

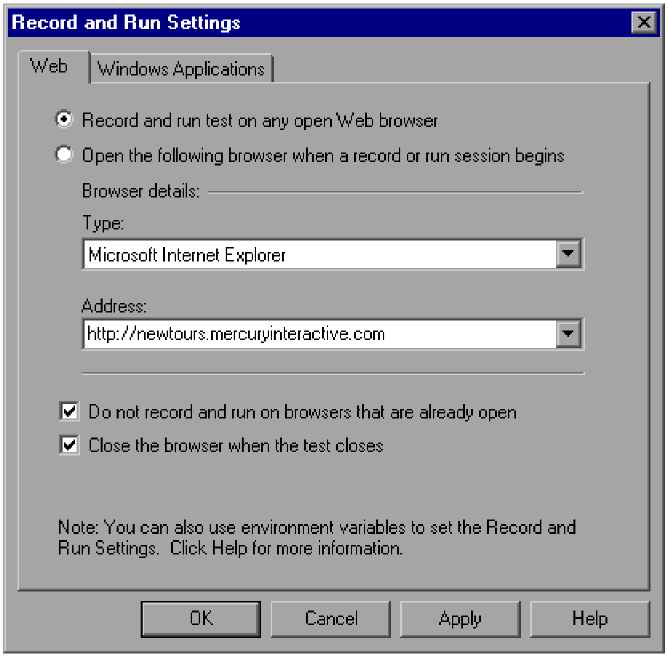
For more information, see step 1 of “Recording a Test” on page 10.

2 Start recording on the Mercury Tours Web site.

In the coming steps, you will record a test similar to the one recorded in the “Recording Tests” lesson. However, this test is designed slightly differently to optimize the test for a multi-action test.

Confirm that all Web browsers are closed.

Choose **Test** > **Record** or click the **Record** button. The Record and Run Settings dialog box opens.



Record and Run Settings

Web | Windows Applications

● Record and run test on any open Web browser

○ Open the following browser when a record or run session begins

Browser details:

Type: Microsoft Internet Explorer

Address: http://newtours.mercuryinteractive.com

☑ Do not record and run on browsers that are already open

☑ Close the browser when the test closes

Note: You can also use environment variables to set the Record and Run Settings. Click Help for more information.

- In the Web tab, select **Open the following browser when a record or run session begins**.

  Choose a browser from the **Type** list and confirm that the URL in the **Address** box is http://newtours.mercuryinteractive.com.

  Confirm that the **Do not record and run on browsers that are already open** and **Close the browser when the test is closed** options are selected.

- In the Windows Applications tab, confirm that **Record and run on these applications (opened on session start)** is selected, and that there are no applications listed.

Click **OK**.

QuickTest begins recording, and your browser opens to the Mercury Tours Web site.

3 Sign in to the Mercury Tours Web site.

In the **User Name** and **Password** boxes, type the name and password you registered with Mercury Tours.

Click **Sign-In**.

The Flight Finder page opens.

4 Enter flight details.

Change the following selections:

- Departing From: **New York**
- On: **Dec 29**
- Arriving In: **San Francisco**
- Returning: **Dec 31**

Click **CONTINUE** to accept the other default selections. The Select Flight page opens.

5 Select a flight.

Click **CONTINUE** to accept the default flight selections. The Book a Flight page opens.

6 Enter required passenger and purchase information.

Enter the required information (fields with red text labels) in the Passengers and Credit Card sections.

At the bottom of the page, click **SECURE PURCHASE**. The Flight Confirmation page opens.

7 Return the browser to the Flight Finder page.

At the bottom of the Flight Confirmation page, click **BACK TO FLIGHTS** button. The Flight Finder page opens.

Ending the recording on the Flight Finder page enables you to create an action that starts and ends on the same page, so that you can call the action multiple times if you wish.

8 Stop Recording.

In QuickTest, click **Stop** to stop the recording process.

9 Save the test as ActionA.

Select **File** > **Save As**. Save the test as ActionA.

Dividing the Test into Two Actions

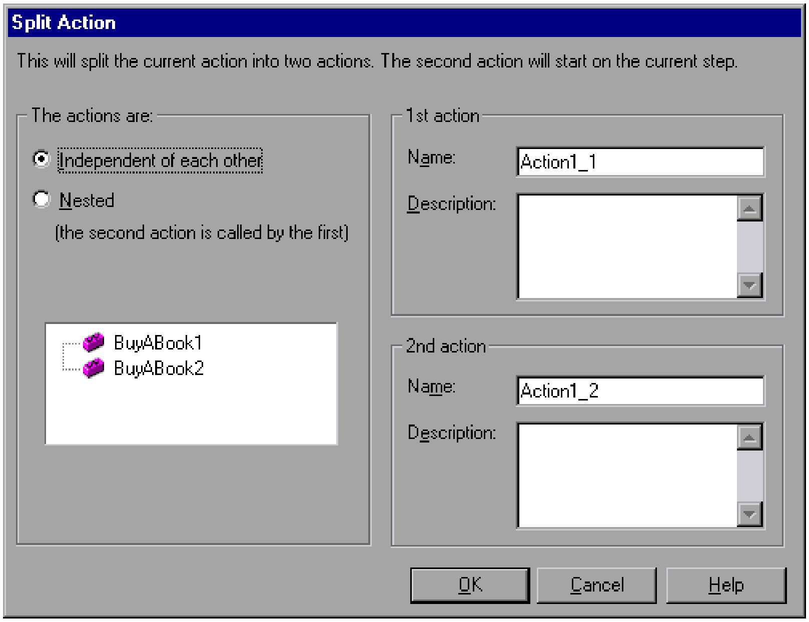
Since your test currently contains two main processes: signing-in and purchasing a flight, it makes sense to divide the test into separate actions.

1 Select the page where you want the second action to begin.

In the Keyword View, expand **(+) Action1** > **Welcome: Mercury Tours**.

2 Split the test into two actions.

Select the first **Find a Flight: Mercury** page and choose **Step** > **Split Action** or click the **Split Action** button. The Split Action dialog box opens.



Split Action

This will split the current action into two actions. The second action will start on the current step.

The actions are:

● Independent of each other

○ Nested

(the second action is called by the first)

BuyABook1
BuyABook2

1st action

Name: Action1_1

Description:

2nd action

Name: Action1_2

Description:

3 Enter names and descriptions for the two actions.

In the **Name** box of the **1st action** type: Sign_in.

In the **Description** box of the **1st action**, type: Enters user name and password to sign in.

In the **Name** box of the **2nd action** type: FlightOrder.

In the **Description** box of the **2nd action**, type: Chooses and purchases a flight.

Click **OK** to close the Split Action dialog box and accept the other default selections. The two actions are displayed in the Keyword View.

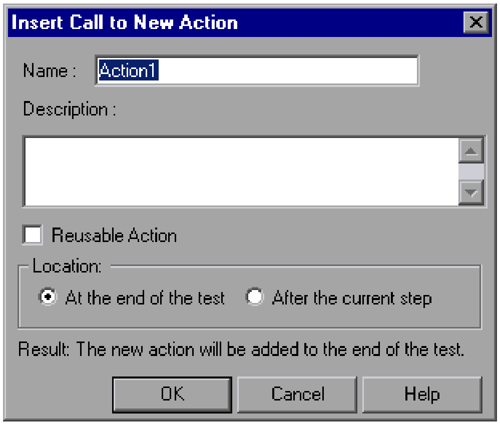
Choose **View** > **Expand All** to examine the contents of each action. Note that the **FlightOrder** action begins and ends with the **Find a Flight: Mercury** page.

Inserting a Call to a New Action

Now you will insert a call to a new action and record the step of returning to the Mercury Tours home page so that the entire test begins and ends on the same page.

1 Insert the ReturnHome action.

Choose **Insert** > **Call to New Action** or click the **Insert Call to New Action** button. The Insert Call to New Action dialog box opens.



Insert Call to New Action

Name : Action1

Description :

□ Reusable Action

Location:

● At the end of the test ○ After the current step

Result: The new action will be added to the end of the test.

Enter ReturnHome in the **Name** box, confirm that **At the end of the test** is selected, and click **OK**. The dialog box closes and the **ReturnHome** action is added to the end of your test.

2 Prepare the application for recording.

Confirm that your Web browser is open to the Flight Finder page.

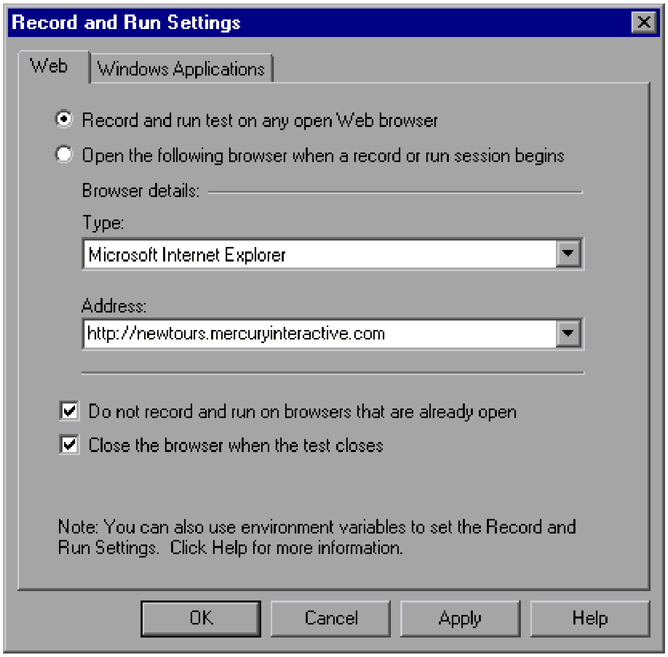
If your browser is not open, open it to the Mercury Tours site (http://newtours.mercuryinteractive.com) and sign in. The Flight Finder page opens.

Position the applications on your desktop so that you can see both QuickTest and the Web browser.

3 Change the Record and Run settings.

Currently, your Record and Run settings are set so that a new browser opens each time you begin recording your test. However, because you want to record a step on a browser that you have already opened and prepared, you need to change the Record and Run settings before you begin recording.

In QuickTest, choose **Test** > **Record and Run Settings**. The Record and Run Settings dialog box opens.



Record and Run Settings

Web | Windows Applications

● Record and run test on any open Web browser

○ Open the following browser when a record or run session begins

Browser details:

Type: Microsoft Internet Explorer

Address: http://newtours.mercuryinteractive.com

☑ Do not record and run on browsers that are already open

☑ Close the browser when the test closes

Note: You can also use environment variables to set the Record and Run Settings. Click Help for more information.

In the **Web** tab, select **Record and run test on any open Web browser** and click **OK**.

4 Record the ReturnHome action.

In the Keyword View, select the new **ReturnHome** action icon.

● Record

Choose **Test** > **Record** or click the **Record** button.

Click the **Home** navigation link on the left side of the Flight Finder page. The Mercury Tours home page opens.

■ Stop

In QuickTest, click **Stop** to stop the recording process.

5 Return the Record and Run Settings to their original settings.

You changed the record and run settings for the purposes of recording the **ReturnHome** action, but this change also affects the way QuickTest runs the test. You want QuickTest to open a new browser to the Mercury Tours home page when you run the test, so you need to reapply the original settings.

Choose **Test** > **Run and Record Settings**. The Record and Run Settings dialog box opens.

In the Web tab, select **Open the following browser when a record or run session begins**. Confirm that http://newtours.mercuryinteractive.com is displayed in the **Address** box and click **OK**.

6 Save the test.

Choose **File** > **Save** or click the **Save** button.

Setting Actions as Reusable Actions

Reusable actions are like functions in programming. When you have a process that you may need to include in several tests, you record, modify, and enhance the steps of the process and save them in a reusable action. Then you can call the action from other tests, rather than recording, modifying, and enhancing the same steps each time.

To call an action from another test, you must make it a reusable action in the test with which it was stored. The **Sign_in** and **ReturnHome** action you created in this lesson can apply to almost any test you might create on the Mercury Tours site, so it makes sense to make them reusable actions.

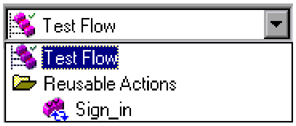
1 Make the Sign_in action reusable.

In the Keyword View, right-click the **Sign_in** action and choose **Action Properties**. The Action Properties dialog box opens. Select the **Reusable action** check box and click **OK**. If a message box is displayed, read the message, click **Do not show this dialog box again**, and click **OK**. The action icon changes to a reusable action icon .

2 View the details of the Sign_in action.

Note that you no longer have the option to expand the **Sign_in** action within the Keyword View.

If the Action List was not previously displayed above the Keyword View, it is displayed after you convert the action to a reusable action. The Action List displays a list of all reusable and external actions in the test.



View the details of the **Sign_in** action by selecting **Sign_in** from the Action List. The details of the **Sign_in** action are displayed.

In the Action List, select **Test Flow** to return to the main flow of the test in the Keyword View.

3 Make the ReturnHome action reusable.

Repeat steps 1 and 2 for the **ReturnHome** action.

4 Save the test.

Choose **File** > **Save** or click the **Save** button.

Inserting Existing Actions

When you plan a suite of tests, you may realize that each test requires one or more identical activities, such as signing in. Once you have created the action and stored it with one test, you can insert either a call to a copy of the existing action, or a call to the existing action, into other tests.

When you insert a call to copy of an existing action, you can make changes to the copied action, and your changes will neither affect, nor be affected by, any other test. Calls to existing actions, however, are read-only in the calling test. They can be modified only in the test with which they were stored. Calls to existing actions enable you to call the same action from several tests and make it easy to maintain tests, because when your application changes you only have to update the existing action stored with the original test.

In the following exercises you will create a new test that is similar to the **ActionA** test, except that the **Sign_in** and **ReturnHome** actions are external actions (calls to existing actions stored with other tests) and the **FlightOrder** action is slightly modified.

Inserting Calls to Actions

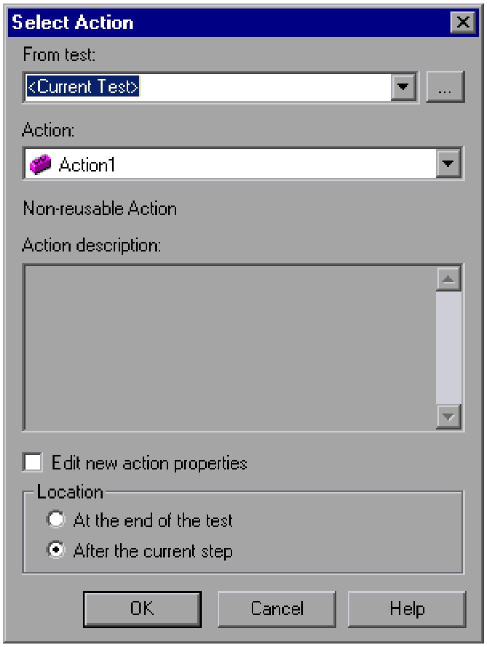
First, you will insert calls to the reusable **Sign_in** and **ReturnHome** actions from **ActionA** into **ActionB**.

1 Open a new test.

For more information, see step 1 of "Recording a Test," on page 10.

2 Insert a call to the Sign_in action.

Choose **Insert** > **Call to Existing Action**. The Select Action dialog box opens.



Select Action

From test: <Current Test>

Action: Action1

Non-reusable Action

Action description:

☐ Edit new action properties

Location

○ At the end of the test

● After the current step

Click the browse button next to the **From test** box. Select the **ActionA** test in the Open Test dialog box, and click **Open**. Note that only reusable actions from the **ActionA** test are displayed in the Select Action dialog box.

Select the **Sign_in** action. In the **Location** area, select **At the end of the test**. Click **OK**. The call to **Sign_in [ActionA]** action is added to the end of the test as a call to an external action with the icon.

Note that you cannot expand the **Sign_in** action call in the test flow in the Keyword View. To view the steps in the **Sign_in** action, select it from the Action List. The action call you inserted is read-only. To modify the action, you must open the original (**ActionA**) test and modify the action there.

Select **Test Flow** in the Action List to return to the main flow of the test in the Keyword View.

3 Insert a call to the ReturnHome action.

Use the instructions in step 2 to insert a call to the **ReturnHome** action from **ActionA**. The **ReturnHome** action is added to the end of your test.

4 Save the test.

Select **File** > **Save** or click the **Save** button. Save the test as ActionB.

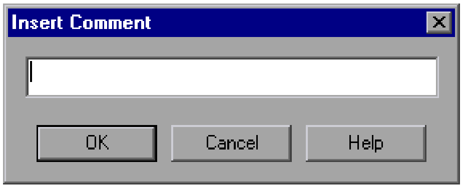
5 Modify the original ReturnHome action.

Choose **File** > **Open**. Browse to **ActionA**, and click **Open**.

Double-click the **ReturnHome** action icon to display its contents. (This is equivalent to selecting the **ReturnHome** action from the Action List.)

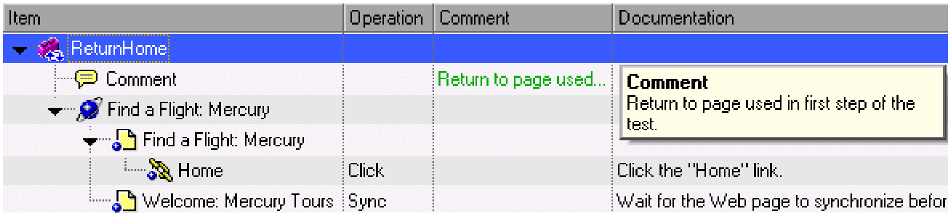
Make sure that the **Comment** column is displayed in the Keyword View. If it is not, right-click on the Keyword View column header and select **Comment** from the displayed list.

In the Keyword View, select the **ReturnHome** action icon and choose **Insert** > **Step** > **Comment** to add a comment about the purpose of the **Home** step. The Insert Comment dialog box opens.



In the edit box, enter: Return to page used in first step of the test. Click **OK**.

The comment step is added as the first step in the action, and you can see the comment text in the **Comment** column of that step. If the comment is too long to see, it is displayed in a tooltip when you move your cursor over the **Comment** column.



Item	Operation	Comment	Documentation
ReturnHome			
Comment		Return to page used...	
Find a Flight: Mercury			
Find a Flight: Mercury			
Home	Click		Click the "Home" link.
Welcome: Mercury Tours	Sync		Wait for the Web page to synchronize befor

Comment
Return to page used in first step of the test.

6 Save the test.

Select **File** > **Save** or click the **Save** button.

7 Confirm that the modification made in the original ReturnHome action is also included in the calling test.

Choose **File** > **Open**. Browse to **ActionB** and click **Open**.

Double-click the **ReturnHome [ActionA]** action icon to display it. Note that the comment you added to the original action (**ActionA**) also exists in the calling test (**ActionB**).

In the Action toolbar, select **Test Flow** from the Action List to return to the main flow of the test in the Keyword View.

Inserting a Copy of an Action

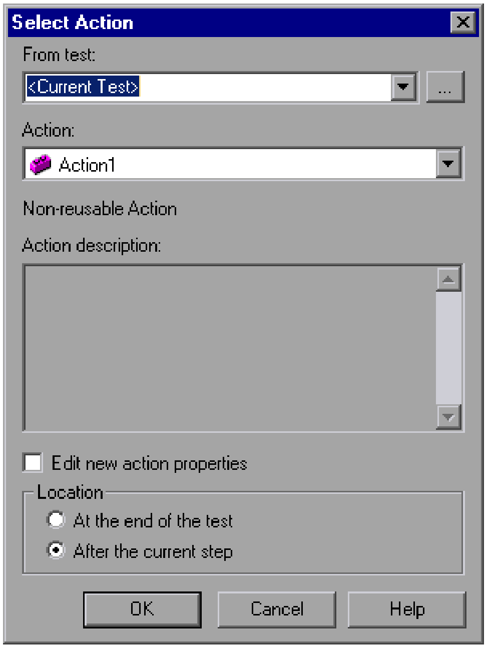
Now, you will insert a copy of the **FlightOrder** action.

1 Open the ActionB test.

Choose **File** > **Open**. Browse to **ActionB** and click **Open**.

2 Insert a copy of the FlightOrder action.

Choose **Insert** > **Call to Copy of Action**. The Select Action dialog box opens.



Select Action

From test: <Current Test>

Action: Action1

Non-reusable Action

Action description:

□ Edit new action properties

Location

○ At the end of the test

● After the current step

Click the browse button next to the **From test** box. Select the **ActionA** test in the Open Test dialog box, and click **Open**. The actions from the **ActionA** test are displayed in the Action List. Note that the description of the highlighted action is displayed in the **Action description** box.

Select the **FlightOrder** action. In the **Location** area, select **At the end of the test**. Click **OK**. The **Copy of FlightOrder** action is added to the end of the test. Note that the call to the copy of the action you inserted is just like any other action call in your test. You can modify the steps in the call to the copied action just as you would those of any other action call.

3 Modify the Copy of FlightOrder action.

In the Keyword View, expand **(+) Copy of FlightOrder** > **Welcome: Mercury Tours** > **Find a Flight: Mercury**. Select **fromPort** and click the **Value** cell.

Change the value in the **Value** cell to London and click any other row in the Keyword View.

4 Call the Copy of FlightOrder action after the Sign_in action.

You want the **Copy of FlightOrder** action to be immediately after the **Sign_in** action. Since you cannot drag any action to be immediately after a reusable action (because it is read-only in the calling test), you must perform the following steps to change the order of the action calls.

Select the **Copy of FlightOrder** action and drag it immediately below the **Action1** action in the Keyword View. Then, select the **Sign_in** action and drag it immediately below the **Action1** action in the Keyword View. The **Copy of FlightOrder** action is now immediately after the **Sign_in** action.

5 Delete Action1 from the test.

Every new test contains a default **Action1** action call. You can now delete the **Action1** action call.

In the Keyword View, select **Action1** and choose **Edit** > **Delete** or press the **Delete** key on your keyboard. A confirmation message opens. Click **Yes** to delete the action call.

6 Save the test.

Choose **File** > **Save** or click the **Save** button.

7 Confirm that the original FlightOrder action was not modified.

Choose **File** > **Open**. Browse to **ActionA**, and click **Open**.

Expand **FlightOrder** > **Welcome: Mercury Tours** > **Find a Flight: Mercury**. Note that the change of departure city you made in the **ActionB** test did not affect the original action in **ActionA** (New York is still the departure city in **ActionA**).

Parameterizing an Action

If you look at the Data Table at the bottom of the QuickTest window in the **ActionB** test, you will see four tabs: **Global**, **Copy of FlightOrder**, **Sign_in [ActionA]**, and **ReturnHome [ActionA]**.

Note: If the Data Table is not displayed, choose **View** > **Data Table** to display it.

The Global tab is a data sheet whose data is used for the entire test. If five rows of data are displayed in the Global table, the test will run five times. In addition, you can create data sets for each action, using the relevant action sheet. If you parameterize a step using an action parameter and enter five rows of data in the corresponding sheet, then that action will run five times within each test iteration.

Note: The **Sign_in [ActionA]** and **ReturnHome [ActionA]** data sheets are displayed in gray and cannot be edited because each of these data sheets belong to the corresponding called action and can be edited from only the called action's original test.

In this section, you will create a data set for the **FlightOrder** action to check how the application handles multiple orders during a single user session.

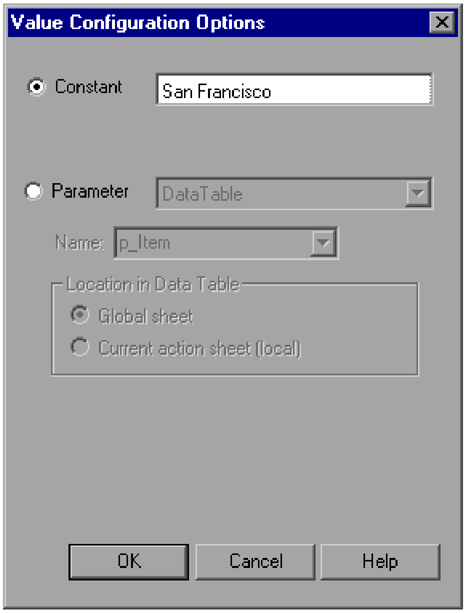
1 Open the ActionB test.

If the **ActionB** test is not already open, choose **File** > **Open** and browse to **ActionB**. Click **Open**.

2 Select a step to parameterize.

Expand **(+) Copy of FlightOrder** > **Welcome: Mercury Tours** > **Find a Flight: Mercury.**

In the **toPort** row in the Keyword View, click in the **Value** cell and then click the parameterization icon . The Value Configuration Options dialog box opens.



Value Configuration Options

● Constant San Francisco

○ Parameter DataTable

Name: p_Item

Location in Data Table

● Global sheet

○ Current action sheet (local)

Enter the parameter information.

Select **Parameter** and confirm that the **DataTable** option is selected.

In the **Name** box, type ArrivalCity.

You are creating this data set exclusively for the **Copy of FlightOrder** action. Therefore, select **Current action sheet (local)**. Click **OK** to close the Value Configuration Options dialog box.

QuickTest automatically adds the **ArrivalCity** column to the **Copy of FlightOrder** data sheet and inserts **San Francisco** into the first cell of the **ArrivalCity** column.

3 Parameterize additional steps.

It is not logical for a single customer to fly to two different locations on the same dates, so it makes sense to parameterize the departure and arrival dates.

In the Keyword View, select **fromMonth**. Use the procedures in step 2 to parameterize the departure date. Make sure to select **Current action sheet (local)** in the Value Configuration Options dialog box.

In the Keyword View, select **toMonth.** Use the explanations in step 2 to parameterize the arrival date. Make sure to select **Current action sheet (local)** in the Value Configuration Options dialog box.

4 Enter the parameterization data in the Data Table.

In the Data Table, click the **Copy of FlightOrder** tab.

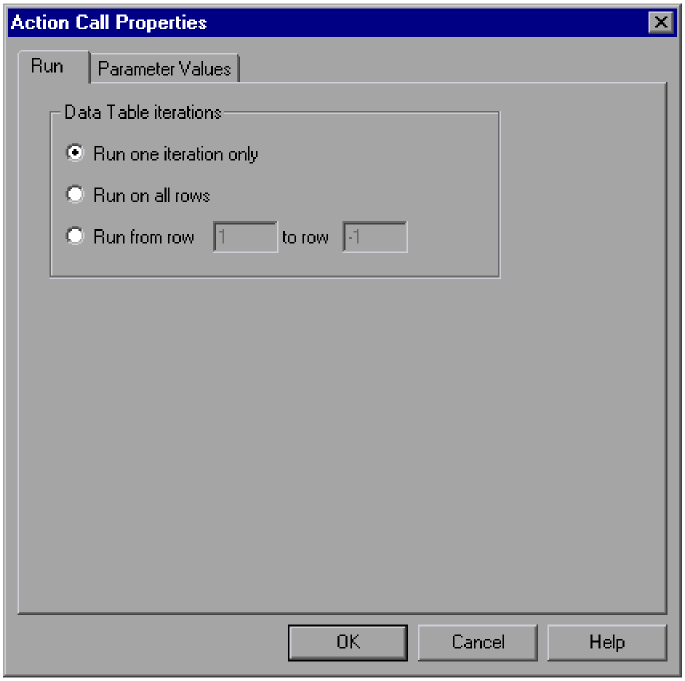
Type Portland in the second row of the **ArrivalCity** column. Press **Enter**.

Type Oct in the second row of the departure and arrival date columns.

5 Set the action to run all the data in the local Data Table.

By default, QuickTest only runs one iteration of each action in a test. You want QuickTest to use all the data in the **Copy of FlightOrder** tab.

Right-click the **Copy of FlightOrder** action in the Keyword View and choose **Action Call Properties**. The Action Call Properties dialog box opens.



Data Table iterations

● Run one iteration only

○ Run on all rows

○ Run from row 1 to row -1

Select **Run on all rows** in the Run tab. If a message box opens, read it and then click **OK** to close it. Click **OK** to close the Action Call Properties dialog box.

6 Save the test.

Select **File** > **Save** or click the **Save** button.

You have now created a test that signs-in to the Mercury Tours site, submits two flight orders: a flight from London to San Francisco from Dec 29 to Dec 31, and a flight from London to Portland from Oct 29 to Oct 31, and then logs out of the system by clicking the **Home** link.

Running and Analyzing a Multi-action Test

You will now run the **ActionB** test. The entire test will run only once, but the **FlightOrder** action will run twice; one time for each set of data in the **Copy of FlightOrder** data sheet.

1 **Run the ActionB test.**

Confirm that all Web browsers are closed.

Click **Run** or choose **Test** > **Run**. The Run dialog box opens.

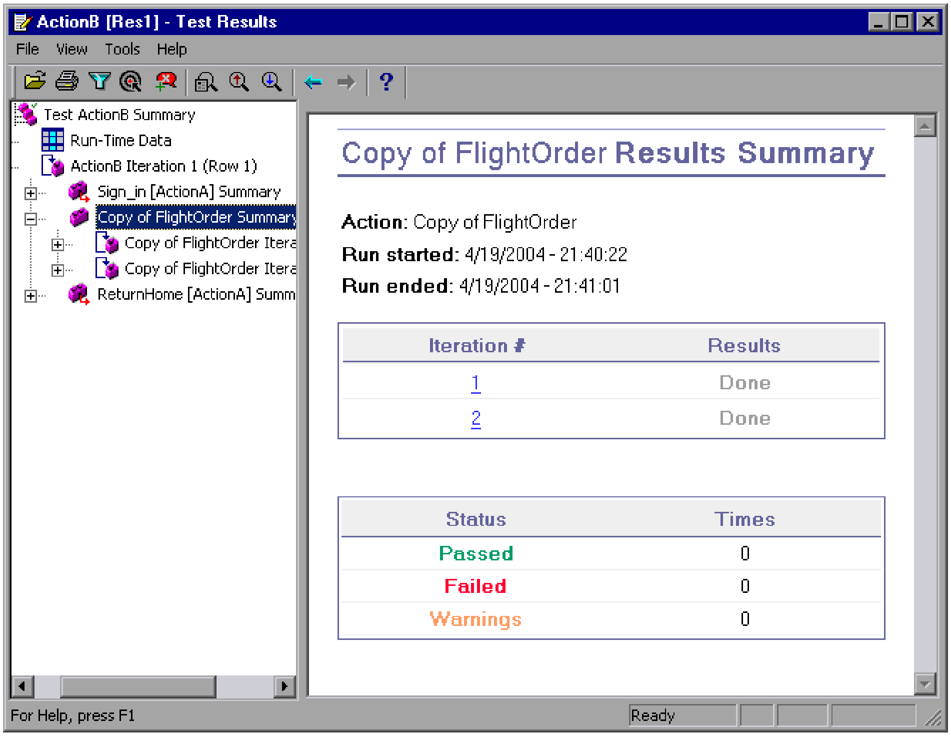
Select **New run results folder** and accept the default results folder name.

Click **OK**. When the test run is completed, the Test Results window opens.

2 **Examine the test results.**

Examine the Results Summary. The test is marked as **Done**. This indicates that the test ran without any failures (there were no checkpoints to "pass").

Expand **Test ActionB Summary** > **ActionB Iteration 1 (Row 1)** in the Test Results window. Select the **Copy of FlightOrder Summary** action in the results tree. Note that within the summary, the results of both iterations are displayed.


ActionB [Res1] - Test Results
File
View
Tools
Help
Test ActionB Summary
Run-Time Data
ActionB Iteration 1 (Row 1)
Sign_in [ActionA] Summary
Copy of FlightOrder Summary
Copy of FlightOrder Itera
Copy of FlightOrder Itera
ReturnHome [ActionA] Summ
For Help, press F1
Ready

Copy of FlightOrder Results Summary

Action: Copy of FlightOrder
Run started: 4/19/2004 - 21:40:22
Run ended: 4/19/2004 - 21:41:01

Iteration #	Results
1	Done
2	Done

Status	Times
Passed	0
Failed	0
Warnings	0

3 **Close the Test Results window.**

Choose **File** > **Exit** to close the Test Results window.

9

Where Do You Go from Here?

Now that you have completed the exercises in this tutorial, you are ready to apply the QuickTest concepts and the skills you learned to testing your own application.

In this lesson you will learn about:

- Testing Your Own Application—Getting Started
- Getting Additional Information
- Documentation Updates

Testing Your Own Application—Getting Started

In this tutorial, you learned about the basic tools needed for testing applications and Web sites.

We suggest that you follow the procedure outlined below when testing your own application.

1 **Plan your test.**

Decide how to organize your test. Consider what users will want to accomplish when deciding which operations to record. Confirm that your application and QuickTest are set to match the needs of your test.

Decide which object repository mode to use for your test. The object repository mode you choose determines the way that QuickTest stores information about your application when you record tests. You can have a separate action repository for each action and store the objects for each action in its corresponding action repository, or you can store all the objects in your test in a common (shared) object repository file. You can also use the same shared object repository file for multiple tests.

If you are new to testing, you may want to use the object repository per-action mode, as you can use this mode without making any other changes to your test.

If you are familiar with testing, it is probably most efficient to work in the shared object repository mode. In this mode, you can use one object repository file for multiple tests. This means that object information that applies to many tests is kept in one central location. When the objects in your application change, you can update them in one location for all of your tests.

For more information on choosing an object repository mode, refer to the *QuickTest Professional User's Guide*.

2 Record your test.

While recording (described in “Recording Tests” on page 9), follow the steps you expect users to perform as they navigate within your application or Web site.

3 Enhance your recorded test.

- ➤ Add checkpoints (described in “Creating Checkpoints” on page 25) to search for specific values of a page, object, text string, or table.
- ➤ Replace fixed values in your test with parameters (described in “Parameterizing Tests” on page 45) to check how your application performs the same operations with multiple sets of data.
- ➤ Create output values (described in “Creating Output Values” on page 55) to retrieve data during a test run.
- ➤ Where applicable, use regular expressions (described in “Using Regular Expressions” on page 63) to increase the flexibility and adaptability of your tests.

You can further enhance your test with programming and conditional and loop statements, which add logic to your test. These are described in the *QuickTest Professional User's Guide*.

4 **Debug your test.**

Debug your test to check that it operates smoothly and without interruption. For additional information about debugging, refer to the *QuickTest Professional User's Guide*.

5 **Run your test.**

Run your test on your application or Web site (described in "Running and Analyzing Tests" on page 19) to check that the application functions as expected.

6 **Analyze the test results.**

Examine the results of your test to pinpoint defects in your application. (Refer to the appropriate sections of this tutorial to understand what to look for in the test results.)

7 **Report defects.**

If you have Quality Center installed, you can submit any defects discovered to a Quality Center database. (Quality Center is Mercury Interactive's tool for software test management. For additional information, refer to the *QuickTest Professional User's Guide* and the documentation included with Quality Center.)

Getting Additional Information

For more information about QuickTest Professional, refer to the printed documentation and online resources provided with QuickTest Professional.

Printed Documentation

In addition to this tutorial, QuickTest Professional comes with the following printed documentation:

- **QuickTest Professional Installation Guide**

  Explains how to install QuickTest Professional.

- **QuickTest Professional User's Guide**

  Provides step-by-step instructions for using QuickTest Professional to test your application or Web site. The User's Guide describes many useful testing tasks and options not covered in this tutorial.

- **QuickTest Professional Shortcut Key Reference Card**

  Lists commands that you can execute using shortcut keys.

Online Resources

In addition to this tutorial, the following online documentation is included with QuickTest Professional:

- **Readme** (available from the QuickTest Professional Start menu program folder)

  Provides the latest news and information about QuickTest Professional.

- **What's New in QuickTest Professional** (available from **Help** > **What's New in QuickTest**)

  Describes the newest features, enhancements, and supported environments in the current version of QuickTest Professional.

- **Printer-Friendly Documentation**

  Displays the complete documentation set in Adobe portable document format (PDF). Online books can be read and printed using Adobe Reader, which can be downloaded from the Adobe Web site (http://www.adobe.com).

- **QuickTest Professional Context-Sensitive Help** (available from dialog boxes and windows)

  Describes specific QuickTest dialog boxes and windows.

- **QuickTest Professional Tutorial** (available from the QuickTest Professional Welcome window, the **Help** menu, and the QuickTest Professional Start menu program folder)

  Teaches you basic QuickTest skills and shows you how to start designing tests for your applications.

- **QuickTest Professional User's Guide** (available from **Help** > **QuickTest Professional Help**)

  Provides step-by-step instructions for using QuickTest Professional to test your application or Web site. The User's Guide describes many useful testing tasks and options not covered in this tutorial.

- **QuickTest Professional Object Model Reference** (available from **Help** > **QuickTest Professional Help**)

  Describes QuickTest Professional test objects, lists the methods and properties associated with each object, and provides syntax information and examples for the methods.

- **QuickTest Professional Automation Object Model Reference** (available from the QuickTest Professional Start menu program folder and from **Help** > **QuickTest Automation Object Model Reference**)

  Provides syntax, descriptive information, and examples for automation objects, methods, and properties. It also contains a detailed overview to help you get started writing QuickTest automation scripts. The automation object model assists you in automating test management, by providing objects, methods and properties that enable you to control virtually every QuickTest feature and capability.

- **VBScript Reference** (available from **Help** > **QuickTest Professional Help**)

  Contains Microsoft VBScript documentation, including VBScript, Script Runtime, and Windows Script Host.

- **Mercury Tours** sample Web site and the **Mercury Tours** Windows sample flight application (both available from the QuickTest Professional Start menu program folder)

  The sample applications for QuickTest Professional are the basis for many of the examples in the documentation. The examples and instructions in this tutorial are based on the Mercury Tours Web site (http://newtours.mercuryinteractive.com).

- **Customer Support Online** (available from **Help** > **Customer Support Online**)

  Uses your default Web browser to open the Mercury Customer Support Web site. This site enables you to browse the knowledge base and add your own articles, post to and search user discussion forums, submit support requests, download patches and updated documentation, and more. The URL for this Web site is http://support.mercury.com.

- **Send Feedback** (available from **Help** > **Send Feedback**)

  Enables you to send online feedback about QuickTest Professional to the product team.

- **Mercury Interactive on the Web** (available from **Help** > **Mercury Interactive on the Web**)

  Opens Mercury Interactive’s home page in your default Web browser. This site provides you with the latest information about Mercury Interactive and its products, including new software releases, seminars and trade shows, customer support, and educational services. The URL for this Web site is http://www.mercury.com.

Documentation Updates

Mercury Interactive is continuously updating its product documentation with new information. You can download the latest version of this document from the Customer Support Web site (http://support.mercury.com).

To download updated documentation:

1. In the Customer Support Web site, click the **Documentation** link.
2. Under **Select Product Name**, select **QuickTest Professional**.

   Note that if **QuickTest Professional** does not appear in the list, you must add it to your customer profile. Click **My Account** to update your profile.
3. Click **Retrieve**. The Documentation page opens and lists the documentation available for the current release and for previous releases. If a document was recently updated, **Updated** appears next to the document name.
4. Click a document link to download the documentation.

Index