

Borland®

Silk Test 16.0

Silk4J Quick Start Tutorial

**Borland Software Corporation
700 King Farm Blvd, Suite 400
Rockville, MD 20850**

Copyright © Micro Focus 2015. All rights reserved. Portions Copyright © 1992-2009 Borland Software Corporation (a Micro Focus company).

MICRO FOCUS, the Micro Focus logo, and Micro Focus product names are trademarks or registered trademarks of Micro Focus IP Development Limited or its subsidiaries or affiliated companies in the United States, United Kingdom, and other countries.

BORLAND, the Borland logo, and Borland product names are trademarks or registered trademarks of Borland Software Corporation or its subsidiaries or affiliated companies in the United States, United Kingdom, and other countries.

All other marks are the property of their respective owners.

2015-02-12

Contents

Silk4J Quick Start Tutorial	4
Starting Silk4J	4
Creating a Silk4J Project	4
Recording a Test for the Insurance Company Web Application	5
Replaying the Test for the Insurance Company Web Application	6

Silk4J Quick Start Tutorial

This tutorial provides a step-by-step introduction to using Silk4J to test a Web application using dynamic object recognition. Dynamic object recognition enables you to write test cases that use XPath queries to find and identify objects.



Important: To successfully complete this tutorial you need basic knowledge of Java and JUnit.

For the sake of simplicity, this guide assumes that you have installed Silk4J and are using the sample Insurance Company Web application, available from <http://demo.borland.com/InsuranceWebExtJS/>.



Note: You must have local administrator privileges to run Silk4J.

For additional information about Silk4J, including information about sample scripts and applications, refer to the *Silk4J User Guide*. To view the guide, in Eclipse choose **Help > Help Contents** and then click **Silk4J User Guide**.

Starting Silk4J

1. Click **Start > Programs > Silk > Silk Test > Clients > Silk4J**.


If you have not specified a default workspace location or if this is the first time that you are launching Silk4J, the **Workspace Launcher** dialog box opens.

2. Specify the location of your workspace and click **OK**.

Creating a Silk4J Project

When you create a Silk4J project using the **New Silk4J Project** wizard, the wizard contains the same options that are available when you create a Java project using the **New Java Project** wizard. Additionally, the Silk4J wizard automatically makes the Java project a Silk4J project.

1. In the Eclipse workspace, perform one of the following steps:

- Click the drop-down arrow next to the Silk Test toolbar icon  and choose **New Silk4J Project**.
- Right click in the **Package Explorer** and select **New > Other**. Expand the Silk4J folder and double-click **Silk4J Project**.
- If you installed or updated Silk4J to an existing Eclipse location, choose **File > New > Other**. Expand the Silk4J folder and double-click **Silk4J Project**.

The **New Silk4J Project** wizard opens.

2. In the **Project Name** text box, type a name for your project.

For example, type *Tutorial*.

3. If you want to perform keyword-driven testing or configuration testing with Silk Central and you have a valid Silk Central license, check the **Connect to Silk Central** check box to configure the connection to Silk Central for keyword-driven testing.

The Silk Central server is configured for all your projects, not only for the new project.

- a) To use the project for configuration testing with Silk Central Connect, check the **Store project in Silk Central Connect** check box.

For additional information about Silk Central Connect, refer to the [Silk Central Connect User Guide](#).

4. Click **Next**. The **Select an application** page opens.
5. If you have not set an application configuration for the current project, select the tab that corresponds to the type of application that you are testing:
 - If you are testing a standard application that does not run in a browser, select the **Windows** tab.
 - If you are testing a Web application or a mobile Web application, select the **Web** tab.
6. To test a standard application, if you have not set an application configuration for the current project, select the application from the list.
7. To test a Web application or a mobile Web application, if you have not set an application configuration for the current project, select one of the installed browsers or mobile browsers from the list.
Specify the Web page to open in the **Enter URL to navigate** text box. If an instance of the selected browser is already running, you can click **Use URL from running browser** to record against the URL currently displayed in the running browser instance. For the tutorial, select **Internet Explorer** and specify <http://demo.borland.com/InsuranceWebExtJS/> in the **Enter URL to navigate** text box.
8. Click **Finish**. A new Silk4J project is created that includes the JRE system library and the required `.jar` files, `silktest-jtf-nodeps.jar` and the `junit.jar`. The **Project Created** dialog box opens.
9. *Optional:* Expand the **Test Type Selection** list to select the type of test that you want to record:
 - To bundle the recorded actions into one or more keywords, select **Keyword-Driven Test**. This is the default setting.
 - To record the test without creating keywords, select **Silk Test JUnit Test**.
10. Click **Yes** to start recording a new Silk4J test or click **No** to return to the Eclipse workspace.
For the tutorial, click **No**.

Recording a Test for the Insurance Company Web Application

Before you can create a Silk4J test, you must have created a Silk4J project.

Record a new test that navigates to the **Agent Lookup** page in the Insurance Company Web application. For a detailed version of how to record a test and how to configure test applications for each technology type, see the *Creating Tests* section of the Silk4J User Guide.

1. In the toolbar, click **Record Actions**. The application under test and the **Recording** window open. Silk4J creates a base state and starts recording.
2. In the Insurance Company Web site, perform the following steps:
 - a) From the **Select a Service or login** list box, select **Auto Quote**. The **Automobile Instant Quote** page opens.
 - b) Type a zip code and email address in the appropriate text boxes, click an automobile type, and then click **Next**.
For example, type 92121 as the zip code, `jsmith@gmail.com` as the email address and specify `Car` as the automobile type.
 - c) Specify an age, click a gender and driving record type, and then click **Next**.
For example, type 42 as the age, specify the gender as `Male` and `Good` as the driving record type.
 - d) Specify a year, make, and model, click the financial info type, and then click **Next**.
For example, type 2010 as the year, specify `Lexus` and `RX400` as the make and model, and `Lease` as the financial info type.
A summary of the information you specified appears.
 - e) Point to the **Zip Code** that you specified and press `Ctrl+Alt` to add a verification to the script.
You can add a verification for any of the information that appears.
The **Select Verification Type** dialog box opens.

- f) Select whether you want to create a verification of a property or an image verification.
For the tutorial, select **Verify properties of the TestObject**.
The **Verify Properties** dialog box opens.
- g) Check the **TextContents** check box and then click **OK**. A verification action is added to the script for the zip code text.
- h) Click **Home**.

An action that corresponds with each step is recorded.

3. Click **Stop**. The **Record Complete** dialog box opens.
4. The **Source folder** field is automatically populated with the source file location for the project that you selected. To use a different source folder, click **Select** and navigate to the folder that you want to use.
5. *Optional:* In the **Package** text box, specify the package name.
For example, type: `com.example`.
To use an existing package, click **Select** and select the package that you want to use.
6. In the **Test class** text box, specify the name for the test class.
For example, type: `AutoQuoteInput`.
To use an existing class, click **Select** and select the class that you want to use.
7. In the **Test method** text box, specify a name for the test method.
For example, type `autoQuote`.
8. Click **OK**.

Replay the test to ensure that it works as expected. You can modify the test to make changes if necessary.

Replaying the Test for the Insurance Company Web Application

1. Expand the **Tutorial** project in the Package Explorer.
2. Right-click the **AutoQuoteInput** class and choose **Run As > Silk4J Test** . If multiple browsers that are supported for replay are installed on the machine, the **Select Browser** dialog box opens.
3. Select the browser and click **Run**. When the test execution is complete, the **Playback Complete** dialog box opens.
4. Click **Explore Results** to review the TrueLog for the completed test. In this example, the verification will fail, because the **Zip Code** field in the test application is not cleaned.

Index

Q

Quick Start tutorial
 creating test 5
 introduction 4
 replaying tests 6

S

Silk4J
 creating project 4
 quick start tutorial 4

T

test class
 creating 5
test method
 recording 5
tests
 replaying 6
tutorial
 quick start 4