Telerik Corp.

Test Studio Standalone & Visual Studio Plug-In Quick-Start Guide

Contents

Create your First Test
Standalone Web Test
Standalone WPF Test
Standalone Silverlight Test
Visual Studio Plug-In Web Test12
Visual Studio Plug-In WPF Test16
Visual Studio Plug-In Silverlight Test20
Recording Toolbar Overview
Add Quick Verification Steps
Create Advanced Test Verifications
Create a Data Driven Test
Test Execution (Standalone)
Test Execution (VS Plug-In)
How to Resolve Test Step Failures
Elements Pane Overview
Change How an Element Is Found44
Use the 3D Viewer
Perform Common Automation Tasks 51
Telerik Test Run Configuration (Standalone)
Telerik Test Run Configuration (Standalone)
Telerik Test Run Configuration (Standalone)
Telerik Test Run Configuration (Standalone) 52 Telerik Test Run Configuration (VS Plug-In) 53 Steps Pane Overview (Standalone) 54 Steps Pane Overview (VS Plug-In) 55
Telerik Test Run Configuration (Standalone)52Telerik Test Run Configuration (VS Plug-In)53Steps Pane Overview (Standalone)54Steps Pane Overview (VS Plug-In)55DOM Explorer Tool Window Overview57
Telerik Test Run Configuration (Standalone)52Telerik Test Run Configuration (VS Plug-In)53Steps Pane Overview (Standalone)54Steps Pane Overview (VS Plug-In)55DOM Explorer Tool Window Overview57VS Plugin Telerik Test Tab Overview60
Telerik Test Run Configuration (Standalone)52Telerik Test Run Configuration (VS Plug-In)53Steps Pane Overview (Standalone)54Steps Pane Overview (VS Plug-In)55DOM Explorer Tool Window Overview57VS Plugin Telerik Test Tab Overview60Customize Test Using C# or VB.NET Code62
Telerik Test Run Configuration (Standalone)52Telerik Test Run Configuration (VS Plug-In)53Steps Pane Overview (Standalone)54Steps Pane Overview (VS Plug-In)55DOM Explorer Tool Window Overview57VS Plugin Telerik Test Tab Overview60Customize Test Using C# or VB.NET Code62Create a Test with a Custom Coded Step62
Telerik Test Run Configuration (Standalone)52Telerik Test Run Configuration (VS Plug-In)53Steps Pane Overview (Standalone)54Steps Pane Overview (VS Plug-In)55DOM Explorer Tool Window Overview57VS Plugin Telerik Test Tab Overview60Customize Test Using C# or VB.NET Code62Create a Test with a Custom Coded Step62Create a Test That Uses a Code Behind File65

Create your First Test

Let's jump right in and show you how easy it is to record test. We'll show you how to record a Web, WPF, and Silverlight test with the Standalone and Visual Studio plugin versions. Ready? Here we go.

Standalone Web Test

- 1. Launch Telerik Test Studio.
- 2. Click "Record New Test" and then "Record New Web Test."



3. Recording is activated and an IE recording window opens.



- 4. Type <u>www.google.com</u> in the URL box and hit Enter.
- 5. Notice a recording step is added to the Steps pane.

Image: Constraint Read: Re	Co Co	Imp//www.gooptic.com. D = G X Goopt X Web Images Video Maps	ロー開留 の☆® Sign in 🛊 *
Import - Weinfest tablest* Import - Weinfest tablest* Import - Weinfest tablest* Import - Weinfest tablest*	Properties	<image/> <image/> <image/> <image/> <image/> <image/> <text></text>	
•••••		<u> </u>	0 🖽 😼 9.42 AM 4/12/2011

6. Type "Telerik" in the Google search box and click the Search button.

7. Notice that two more steps are added.

Steps - WebTest.tstest*									
	9	.	ſ.	î	5	¢		View Class	
		1	V				4	Navigate to : 'http://www.google.com/'	-
		2	V				4	Set 'QText' text to 'Telerik'	-
	>	3					4	Click 'BtnGSubmit'	•

Close the browser to stop recording. That's how easy it is to create your first Web Test!



Standalone WPF Test

- 1. Launch Telerik Test Studio.
- 2. Click "Record New Test" and then "Record New WPF Test."



- 3. The "Configure WPF Application Path" window appears. There are two options to determine the default application to launch when recording or executing this test.
 - a. WPF Application Path drag and drop the shortcut to your WPF application into this text box, or click "Browse" and find it manually.
 - Active WPF Applications Telerik Test Studio detects all WPF apps currently running and lists them. Highlight the desired app and press "Select Application." In this example FamilyShow is used, available from <u>http://vertigo.com/Lab.aspx</u>.

Configure WPF Application Path X						
WPF Application Path Determines the default application to launch when recording or executing this test WPF Application Path (Drag & Drop the Link to your Application)	Browse					
Active WPF Applications To select a currently running application, click 'Select Application'						
devenv (1052): TestProject1 - Microsoft Visual Studio (Administrator)	Select Application					
FamilyShow (5248): Family.Show	Refresh					
	OK Cancel					

4. Click OK. If you need to later modify the WPF application path, choose "Configure" from the toolbar.

17	🔒 Save 🔵	•		
	Project	Record	Test L	ists Re
Record	Configure	👗 Cut 🌒 Copy [Paste	Clear	Steps
App Step	Configure Configure execution	Edit application r options.	ecording	and

- 5. Hit "Record" to launch the app with the recording toolbar docked at the top.
- 6. Notice that steps are added as actions are taken within the application.



7. Close the browser to stop recording. That's how easy it is to create your first WPF Test!

Standalone Silverlight Test

- 1. Launch Telerik Test Studio.
- 2. Click "Create New Project," name the project, and click "OK."

	w Project			×	A product by ☆telerik
	Project Name:	OOB			deliver more than expected
Recent Projects	Location:	C:\Users\Admin\Doo	cuments\Test Studio Projec	ts Cancel	1 Packages
 C:\\TestProje C:\\TestProject1 C:\\TestProject1 		Create New Project	Open Existing Project	Z Test	JStudio for Web Studio for WPF
		Record New Test	Manage License		

3. Right click the project node and click "Add New Test."

Project Files	
- 👩 ООВ	
	Add New Test
	Add Existing Test
	Create Folder
	Close
	Rename
	Paste
	Open Project Folder in Windows Explorer
	Source Control

4. Select "Web Test," name the test, press Enter, and double click it to open it.

Select Test Type		x
Test Types:	WPF Test	Web Tests can be used to perform actions against web elements, verify web elements' states and content, and to also perform test synchronization to help automate rich internet applications (RIAs) that utilize dynamic technologies like AJAX, MVC, or Silverlight, Web Tests can be run against Internet Explorer, Firefox, Safari, and Chrome.
Test Name: 00	в	
		OK Cancel

- 5. Click the "Out-Of-Browser" icon in the "Silverlight" ribbon.
- 6. Check the "Configure this test to run against a Silverlight application" box in Out-Of-Browser options.
- 7. Enter either the Web URL, browse for the Local Application Directory, or drag and drop a link to your Silverlight application.
- 8. Select the "Recording Host."
 - a. Select "Internet Explorer" if you entered a Web URL.
 - b. Select "Silverlight Out-of-Browser" if you entered a Local Application Directory.



igure Silverlight Out-Of-Browser Options	
ilverlight Application	
Configure this test to run against a Silverlight application	
Silverlight Applications can be run both in the web browser and directly on the desktop. Enablin feature requires that you specify a Web Url or Local Application Directory for the Silverlight Appl When running in a web browser, this test will automatically navigate to the specified Web Url.	ng this lication.
ilverlight Application Path	
Web Url	
Local Application Directory (Drag & Drop the Link to your Application)	
4075161818.www.microsoft.com	rowse
Execution Hosts:	
ecording Host	
Determines which application is launched when clicking 'Record'	
© Internet Explorer	
Silverlight Out-of-Browser	

- 9. Click "Record" to launch the browser or out-of-browser app with the recording toolbar docked at the top.
- 10. Notice that steps are added to the test as actions are taken within the browser/application.

Step	s - (OOB.tstest*		🕼 Contact Editor - www.microsoft.com	• • • •
	{	ل 🕆 🛛 🕈	C View Class		
	1		Click NewButtonButton	Bob Smith	CANCEL SAVE
	2		✓ LeftClick on Item0Textboxview		
	3		🐬 Type 'Bob' into Item0Textbox	First: Bob Last: Smith	
	4		JeftClick on Item1Textboxview	Company: Telerik	
	5		🐬 Type 'Smith' into Item1Textbox	Joh Titles Silverlight Developer	
	6		✓ LeftClick on Item2Textboxview		
	7		🗲 Type 'Telerik' into Item2Textbox	E-mail: Bob@mydomain.com	
	8		✓ LeftClick on Item3Textboxview	Phone:	
	9		🐬 Type 'Silverlight Developer' into Item3Textbox	Notes:	
	10		JeftClick on Item4Textboxview		
>	11		🐬 Type 'Bob@mydomain.com' into Item4Textbox		
				You are offline, Network is available	
				the server	CREATE DESKTOP SHORTCUT

11. Close the browser/application to stop recording. That's how easy it is to create your first Silverlight Test!

Visual Studio Plug-In Web Test

1. Within Visual Studio, click "Telerik"->"Test Studio"->"Create New Test Project."



2. Choose "Test Projects" -> "Test Documents" -> "Test Project," name the project, and click OK.

New Project					? 🗙
Recent Templates		.NET Framework 4 Sort by: Default	- II 🔲	Search Installed Templates	Q
Installed Templates Visual C# Windows Web Cloud Reporting Silverlight Test WCF Workflow Telerik Test Other Languages Other Project Type Database Modeling Projects Test Document Online Templates	5	Test Project	Test Documents	Type: Test Documents A project that contains tests.	
<u>N</u> ame:	TestProject1				
Location:	ation: c:\users\admin\documents\visual studio 2010\Projects 🗸				
Solution na <u>m</u> e:	TestProject1			Create directory for solution Add to source control	
				ОК	Cancel

- τų× Solution Explorer 🗟 🚯 🛃 🖧 ÷ 🌄 Solution 'TestProject1' (1 project Solution Items 둳 Local.testsettings 🚮 TestProject1.vsmdi TraceAndTestImpact.test: stProject1 🔛 Build Properties Rebuild References Clean Microsoft.VisualStudio 💷 System Run Code Analysis - System.Core **Calculate Code Metrics** UnitTest1.cs Add ۲ 1 New Test... Add Reference... 🔛 🛛 New Item... Ctrl+Shift+A Add Service Reference... ::: Existing Item... Shift+Alt+A 📔 🛛 New Folder 🖧 🛛 View Class Diagram 📔 🛛 Unit Test... Set as StartUp Project 1 Load Test... Debug ۲ 2 Web Performance Test... 🧳 Add Solution to Source Control... 8 Coded UI Test... Ж Cut Ctrl+X Ordered Test **1** B Paste Ctrl+V 🗿 🛛 Generic Test × Remove Del 🛅 🛛 Windows Form... Rename 📷 Tea.. 🌉 Class.. User Control... Unload Project 🔁 Component... Open Folder in Windows Explorer Ĩ t1 Project Properties -**9**3 Class... Shift+Alt+C Ŀ, Properties Alt+Enter •=: Z ¥ 🛅 Project File TestProject1.csproj
- 3. Right click on the project node in the solution explorer and select "Add -> New Test."

4. Click on the "Web Test" template, name the test, and select OK.

Add New Fest	Code Te Q Ordere	d UI st d Test	Database Unit Test Unit Test Unit Test Wpf Test	Generic Test Unit Test Wizard	Description Use a WebAii Test to record browser based scenario tests for your web applications. WebAii tests can be used to record actions against web elements, verify web elements' states and content and to also perform test synchronization to help automate rich internet applications (RIAs) that utilize dynamic technologies like ASP.NET AJAX, ASP.NET MVC or Silverlight. WebAii tests can be recorded once and then run against InternetExplorer, Firefox, Chrome or Safari.
Test <u>N</u> ame:		WebTes	st1.tstest		
Add to Test P	roject:	🐼 Test	Project1		V
					OK Cancel

5. The new, empty test opens automatically.

13

Project Folder c:\users\admin\do

6. Click on the "Record" button (in red below) to begin recording test steps.



7. Internet Explorer will automatically open in record mode.



- 8. Type <u>www.google.com</u> in the URL box and press Enter.
- 9. Notice a recording step is added to the Steps tab.

ee Testhijesti-Moseshi kowi Duda (Adminimato) Die Edit View Telek, Beier Build Debug Tesm. Dies Zools Architecture Test Apslyse Window Heip Die Test View Telek, Brief Build Debug eine Dies Architecture Test Apslyse Window Heip	() : : : : : : : : : : : : : : : : : :	
	Web Images Videos Maps News Shopping Gmail more -	Sign in 🛛 🍳
W Walf Cataland X S 2 (1) (2000 + (-)) Trand - S 2 (1) (2000 + (-)) Trand - S 2 (1) (2) (1) (2) (1) (2) (1) (2) (2) (2) (2) (2) (2) (2) (2) (2) (2	Google	ra 1
	Advertising Programs Business Solutions About Google © 2011- Prevary	
Eveneth Equiver		
A Page	Ounge lackground image	-
👸 🖉 🔮 🎬 🧭 🛷		- 4) 🖽 😥 11.57 AM

- 10. Type "Telerik" in the Google search box and click the search button.
- 11. Notice that two more steps are added.

Web	Test	t1.tstesť	×						
3	🖻 🎎 🔰 (none) 👻 🥪 Record 🐱								
Step	Steps Storyboard Data								
1	🗑 🍠 🖡 🕆 🗳 🦿 Browser: Internet Explorer 🛛 🗞 🦘 400 💽 💽 Add Dialogs Logical 🕸 🗟								
	1			4	Navigate to : 'http://www.google.com/'				
	2			4	Set 'QText' text to 'Telerik'				
>	3			4	Click 'BtnGSubmit'				

12. Close the recording browser window. That's how easy it is to create your first Web Test! Save and build your project.

Visual Studio Plug-In WPF Test

1. Within Visual Studio, click "Telerik"->"Test Studio"->"Create New Test Project."



2. Choose "Test Projects" -> "Test Documents" -> "Test Project," name the project, and click OK.

New Project					?
Recent Templates		.NET Framework 4 Sort by: Defau	lt 🔹 🔢 🛄	Search Installed Templates	Q
Installed Templates Visual C# Windows Web Cloud Reporting Silverlight Test WCF Workflow Telenik Test D Other Languages Other Project Type Database Modeling Projects Test Projects Test Document Online Templates	5	Test Project	Test Documents	Type: Test Documents A project that contains tests.	
<u>N</u> ame:	TestProject1				
Location:	c:\users\admin\	documents\visual studio 2010\Projects	•	Browse	
Solution na <u>m</u> e:	TestProject1			Create directory for solution Add to source control	
				ок	Cancel

- Solution Explorer т Д) 🛅 | 🛐 🛃 | 🖧 Solution 'TestProject2' (1 project) ÷ Solution Items Local.testsettings TestProject2.vsmdi TraceAndTestImpact.testsettings pject2 🔛 Build perties Rebuild erences Microsoft.VisualStudio.QualityTools.UnitTest Clean System Run Code Analysis System.Core Calculate Code Metrics Test1.cs 🚺 New Test... Add Add Reference... New Item... Ctrl+Shift+A Add Service Reference. 💷 Existing Item.. Shift+Alt+A 📔 New Folder 🖧 View Class Diagram 뒿 Unit Test... Set as StartUp Project 🛃 Load Test... Debug Web Performance Test... Add Solution to Source Control... 뒿 Coded UI Test... * Cut Ctrl+X 3 Ordered Test Paste Ctrl+V 👌 Generic Test X Remove Del 🛅 Windows Form... Rename lorer 🛛 📷 Team Explorer 🛛 🗟 Class Vie 🔠 User Control... Unload Project - u × Component... Open Folder in Windows Explorer oject Properties 🍕 Class... Shift+Alt+C Properties Alt+Enter - **Z**¥ E TestProject2.csproj Project File Project Folder c:\users\admin\documents\v
- 3. Right click on the project node in the solution explorer and select "Add -> New Test..."

4. Click on the "WPF Test" template, name the test, and select OK.

Add New Test				? 💌
Templates:			<u>_</u>	
Basic Unit (Test	Coded UI Test	Database Unit Test	Generic Test	Description Use a WpfTest to record and playback scenarios for your Wpf applications.
Load Test Or	vdered Test	Unit Test	Unit Test Wizard	
Web Performan	Web Test	Wpf Test		
Test <u>N</u> ame:	WpfTes	t1.tstest		
<u>A</u> dd to Test Proje	ect: 🕼 Test	Project2		OK Cancel

5. The new, empty test opens automatically. Click the "Configure WPF Application" icon in the toolbar.

📅 🕍 🚺 (none) 👻 🥌 Record 👻 💗							
Steps	Storyboard	Data					
🛱 🖪	• ↓ ↑	5	🦿 😫 🥎 40 Configure WPF Application Logical 🔹 💐				

- 6. The "Configure WPF Application Path" window appears. There are two options to determine the default application to launch when recording or executing this test.
 - a. WPF Application Path drag and drop the shortcut into this text box, or click "Browse" and find it manually.
 - Active WPF Applications Telerik Test Studio detects all WPF apps currently running and lists them. Highlight the desired app and press "Select Application." In this example FamilyShow is used, available from <u>http://vertigo.com/Lab.aspx</u>.

Configure WPF Application Path	×
WPF Application Path Determines the default application to launch when recording or executing this test WPF Application Path (Drag & Drop the Link to your Application)	Browse
Active WPF Applications To select a currently running application, click 'Select Application'	
devenv (1052): TestProject1 - Microsoft Visual Studio (Administrator)	Select Application
FamilyShow (5248): Family.Show	
	Refresh
	OK Cancel

7. Hit "Record" to launch the app with the recording toolbar docked at the top.



8. Notice that steps are added as actions are taken within the application.

00 TestProject2 - Microsoft Visual Studio (Administrator)		The Family Show	
File Edit View Telenik Project Build Debug Team Data Tools Architecture Test Analyze	Window Help	Encold Show	CREATED COLTOTIE
🔁 • 🔛 • 😂 🛃 🥔 🕺 🖧 🖄 '0 • 0 • 💭 • 🖏 🕨 Debug 🔹 Any CPU	• 🧐 🔹 • 🖓 😤 🖗	Family.Snow	VERIGUM
11 555 HBBB GG DEG G CON. 8Q.		New Open - Save - GEDCOM - Skins -	
WpfTestLtstest* ×	Solution Explorer + 0 ×	(
🝸 🎎 🔒 (none) + 🧶 Record + 📦	Sa 🔉 🖬		
Steps Stoyboard Data	Solution 'TestProject2' (1 project)		
😰 🍠 🌲 🕆 🄊 🦿 🔱 🦘 400 n 🜔 n 🕕 🗛 Kada kagical n 🕸 🗟	Local testsettings		
1 V Click NewButtonButton	TestProject2.vsmdi		
2 🔽 🔰 Type'John' into FirstNameInputTextBoxTextbox -	TraceAndTestImpact.testsettings		
3 📝 🍼 LeftClick on Item2Textboxview	Big Properties		
4 📝 Type 'Doe' into LastNameInputTextBoxTextbox	References		
5 📝 🍼 LeftClick on Item1Textboxview 🔹	Settings airs		
6 📝 🛛 🖉 Type '1/1/2011' into BirthDateInputTextBoxTextbox 🔹	UnitTestLcs		
7 📝 🍠 LeftClick on hem3Textboxview -	WpfTest1.tstest WofTest1.test		
> 8 📝 🧳 Type 'Austin, TX' into BirthPlaceInputTextBoxTextbox 🔹	S oprovide the		
	1 1	Add - forstly my other	
		Add a family member	
		Gender	
		Drag a photo O Male	
		here Female	
		First Name Last Name	
		John Doe	
	💫 Solution Explorer 🛛 🖏 Team Explorer 🛤 Class View	Date of Birth Place of Birth	
	Properties • 🖲 🗙	1/1/2011 Austin, TX	
	ArtOfTest.WebAii.Design.IntrinsicTranslators.Silverlight.C •		
	20 21 III		
	(Bindings) (Collection) ^		
	EnsureElementsClickable True		
	Offset 44, 16		
	Pause None E		
Elements Explorer of TestProject2 • 0.2	PrimaryTarget /FamilyShow.exe/Family.S		
3 3	RunsAgainst AllBrowsers		
- Fanily Show	RunsAgainstVersionCom Equals		
	SecondaryTarget		
- Item2Textboxview	StepType Action		
LooDianebipu(Textbox Item ITextboxview			
BrthOateInputTextBoxTextbox	(Bindings) Bind data driven properties against a datasource. Click the		
	drop down to see properties that support data binding		
Ready			
🚯 🥝 🔮 🚆 🛷 🌇 👁			- 🗢 🖶 🐞 413 PM

9. Close the application window. That's how easy it is to create your first WPF Test! Save and build your project.

Visual Studio Plug-In Silverlight Test

1. Within Visual Studio, click "Telerik"->"Test Studio"->"Create New Test Project."



2. Choose "Test Projects" -> "Test Documents" -> "Test Project," name the project, and click OK.

New Project					? 🗙
Recent Templates		.NET Framework 4 Sort by: Default	- II 🔲	Search Installed Templates	Q
Installed Templates Visual C# Windows Web Cloud Reporting Silverlight Test WCF Workflow Telerik Test Other Languages Other Project Type Database Modeling Projects Test Document Online Templates	5	Test Project	Test Documents	Type: Test Documents A project that contains tests.	
<u>N</u> ame:	TestProject1				
Location:	c:\users\admin\	documents\visual studio 2010\Projects	•	Browse	
Solution na <u>m</u> e:	TestProject1			Create directory for solution Add to source control	
				ОК	Cancel

- τų× Solution Explorer 🗟 🚯 🛃 🖧 ÷ 🌄 Solution 'TestProject1' (1 project 🔺 🚞 Solution Items 둳 Local.testsettings 🛺 TestProject1.vsmdi TraceAndTestImpact.test stProject1 🛗 Build Properties Rebuild References Clean Microsoft.VisualStudio 💷 System Run Code Analysis - System.Core Calculate Code Metrics UnitTest1.cs Add ۲ 1 New Test... Add Reference... 🔛 🛛 New Item... Ctrl+Shift+A Add Service Reference... ::: Existing Item... Shift+Alt+A 📔 🛛 New Folder 🖧 🛛 View Class Diagram 📔 🛛 Unit Test... Set as StartUp Project 1 Load Test... Debug ۲ Web Performance Test... 2 🧳 Add Solution to Source Control... 8 Coded UI Test... Ж Cut Ctrl+X **2** Ordered Test B Paste Ctrl+V 🗿 🛛 Generic Test X Remove Del -8 Windows Form... Rename 📷 Tea.. 🌉 Class.. User Control... 1 Unload Project 🔁 Component... Open Folder in Windows Explorer Ĩ t1 Project Properties -**9**3 Class... Shift+Alt+C Ŀ, Properties Alt+Enter •=: Z ¥ 🛅 Project File TestProject1.csproj Project Folder c:\users\admin\do
- 3. Right click on the project node in the solution explorer and select "Add -> New Test."

4. Click on the "Web Test" template, name the test, and select OK.

Add New Test					? 💌
Templates:					
Basic Unit Test Load Test Web Performan	Code Te	cf ed UI st ed Test	Database Unit Test Unit Test Unit Test Wpf Test	Generic Test Unit Test Wizard	Description Use a WebAii Test to record browser based scenario tests for your web applications. WebAii tests can be used to record actions against web elements, verify web elements' states and content and to also perform test synchronization to help automate rich internet applications (RIAs) that utilize dynamic technologies like ASP.NET AJAX, ASP.NET MVC or Silverlight. WebAii tests can be recorded once and then run against InternetExplorer, Firefox, Chrome or Safari.
Test <u>N</u> ame:		WebTe	st1.tstest		
<u>A</u> dd to Test P	Project:	🐼 Test	tProject1		*
					OK Cancel

5. The new, empty test opens automatically. Click the "Configure Silverlight App" icon in the toolbar.

3	🕻 🚺 (non	e) 🔻	🥪 🖲 Record 👻
Steps	Storyboard	Data	
🗑 🙍	• ↓ ↑	5	Configure Silverlight App

- 6. Check the "Configure this test to run against a Silverlight application" box in Out-Of-Browser options.
- 7. Enter either the Web URL, browse for the Local Application Directory, or drag and drop a link to your Silverlight application.
- 8. Select the "Recording Host."
 - a. Select "Internet Explorer" if you entered a Web URL.
 - b. Select "Silverlight Out-of-Browser" if you entered a Local Application Directory.

gure Silverlight Out-Of-Browser Options	
Iverlight Application	
Configure this test to run against a Silverlight application	
Silverlight Applications can be run both in the web browser and directly on the desktop. Enabling this	
feature requires that you specify a Web Url or Local Application Directory for the Silverlight Application.	
When running in a web browser, this test will automatically navigate to the specified Web Url.	
verlight Application Path	
Web Hr	
Local Application Directory (Drag & Drop the Link to your Application)	
4075161818.www.microsoft.com	
Execution Hosts: 🥪	
cording Host	
Determines which application is launched when clicking 'Record'	
Internet Explorer Situation to Control of Provided	
Silverlight Out-or-browser	
OK Cancel	

- 9. Click "Record" to launch the browser or out-of-browser app with the recording toolbar docked at the top.
- 10. Notice that steps are added to the test as actions are taken within the browser/application.

Steps	-	Storyboard	Data								
1	9	🦊 1	17	¢	Browser: Out-Of-Browser 🔹 毮 🦘 400	• • • •	Add Di	alogs Logic	al 🔹 🔇	9	
	1	V		4	Click ResetButtonButton	🔛 Contact Edito	r - www.m	icrosoft.com] II 🕘 😤 🔡 -	
	2			4	Click NewButtonButton						
	3	V		4	LeftClick on Item0Textboxview	Bob Smit	h				CANCEL SAVE
	4			4	Type 'Bob' into Item0Textbox						
	5			4	LeftClick on Item1Textboxview	First:	Bob		Last: Smith		
	6			4	Type 'Smith' into Item1Textbox	Company:	Telerik				
	7			4	LeftClick on Item2Textboxview		Cilcul inte	. D			-
	8			4	Type 'Telerik' into Item2Textbox	Job Title:	Silverligh	t Developer			
	9			4	LeftClick on Item3Textboxview	E-mail:	Bob@myo	domain.com			
	10			4	Type 'Silverlight Developer' into Item3Textbox	Phone:					
	11			4	LeftClick on Item4Textboxview	Notes:					
>	12	V		4	Type 'Bob@mydomain.com' into Item4Textbox	notabl					
						You are of	l	모 티 Network is avail	lable	-	
						disconnecte the serv	d from ver	techork is avail			CREATE DESKTOP SHORTCUT

11. Close the browser/application to stop recording. That's how easy it is to create your first Silverlight Test!

Recording Toolbar Overview

The recording toolbar attached to the IE browser or WPF app recording window contains (from left to right):



- a. Go back to host Makes Visual Studio active (locked in Standalone edition).
- b. Enable or disable hover over highlighting. Press 'Pause/Break' to toggle.
- c. Pause Recording. Press 'Print Screen' to toggle.
- d. Start Recording. Press 'Print Screen' to toggle.
- e. Reconfigure Recorder. (Restores communication between Test Studio and the recording browser window or WPF app if necessary)
- f. Show the DOM Explorer.

Add Quick Verification Steps

Adding verification steps is quick and simple using the Quick Tasks menu.

- 1. Enter record mode.
- 2. Enable hover over highlighting by clicking the icon (in red below) in the docked taskbar.

	II 🕚 🗇 📴 🗸	
	1	命余盛
Enable or	disable hover over highl	ighting. Press 'Pause/Break' to toggle

3. In the recording browser window, hover over the element against which to verify.



- 4. Wait for the blue circular nub to appear and click on it.
- 5. This opens the Element Menu.



RadControls for Silvenight include more than 35 advanced UI controls for build

6. Click the Quick Tasks button (in red above). This opens a list of available quick tasks. The type of element selected dictates what is listed.

Telerik Telerik is WinForms .NET ORM	Quick Tasks Generic Html Translator	T MVC, Silverlight, as .NET Reporting,
ASP NF	🔗 Verify - text contains 'Telerik'	
Demos	🔗 Verify - element is visible	
Silverligh	🔗 Verify - element is not visible	mo
ASP.INE	Wait - text contains 'Telerik'	
more res	Wait - element exists	
ASP.NE	Wait - element does not exist	nts Telerik 🔍
Meet the m	Wait - element is visible	eb 2.0 applications.
www.teleri	Wait - element is not visible	Similar
<u>Silverligh</u>	3	Telerik Q

RadControls for Silverlight include more than 35 advanced UI controls for buil

7. Double clicking one of these items will add that Verify/Wait step to the test.

Ste	Steps - WebTest.tstest*								
	Ι.	₽ 1	• •	າ	¢		View Class		
	1					4	Navigate to : 'http://www.google.com/'		•
	2					4	Set 'QText' text to 'Telerik'		-
	3					4	Click 'BtnGSubmit'		•
>	4					ø	Verify 'TextContent' 'Contains' 'Telerik' on 'TelerikEmTag'		•

Figure 1 - Standalone version

≫	🖻 🔐 🛿 (none) 👻 👳 🥌 Record 👻							
Step	s	Storybo	oard D	ata				
1		1	1	? 🤊	Browser: Internet Explorer 🔹 ዿ 🦘 400 🔹 🜔 🕶			
	1			4	Navigate to : 'http://www.google.com/'			
	2			4	Set 'QText' text to 'Telerik'			
	3			4	Click 'BtnGSubmit'			
>	4			\$	Verify 'TextContent' 'Contains' 'Telerik' on 'TelerikEmTag'			

Figure 2 - VS plugin

Create Advanced Test Verifications

Custom Verifications are created from the Sentence Verification Builder dialog.

1. Reopen the Element Menu and click "Build Verification."



RadControls for Silverlight include more than 35 advanced UI controls for buil

2. The verification builder offers sentence based verifications of elements. Start by selecting a type of Available Verifications.

Sentence Verification Builder		_ 🗆 X
Target Element:		ORER
	Change target element by selecting a different element in the DOM Explorer	CPLC
Available Verifications:	Content Style IsVisible	DOM B
Selected Sentences:		

- 3. When crafting verifications, content is dynamically built against the currently selected element. As selections are made, default values are populated according to values the element contains.
- 4. For example, choose "Content" as the verification type and three menu options appear. Click on the down arrow next to each option to see a list of possible values.

Sentence Verification Builder		_ 🗆 X
Target Element:	 Change target element by selecting a different element in the DOM Explorer	PLORER
Available Verifications: Selected Sentences:	Content Style IsVisible	DOMEX
🔽 🔡 🗙 Content Verify	InnerText Exact Telerik InnerText InnerMarkup OuterMarkup TextContent StartTagContent	_

5. Validate the verification by clicking the "Verification" icon.



6. Locate the current element in the DOM tree by clicking the "Locate in DOM" icon.



7. Delete or start over by clicking the "Delete" icon.



- Verifications can be crafted to verify many different values, styles, or attributes of an element. Craft multiple sentences by selecting a type and filling in the verification criteria. NOTE: Each sentence will add a separate verification step.
- 9. Once finished building the verification, select OK to add it as a step to the current test.

Ste	Steps - WebTest.tstest*						
Image:							
	1			4	Navigate to : 'http://www.google.com/'		•
	2			4	Set 'QText' text to 'Telerik'		•
	3			4	Click 'BtnGSubmit'		•
>	4			V	Verify 'InnerText' 'Exact' 'Telerik' on 'TelerikEmTag'		•
	5			\$	Verify 'TextContent' 'Contains' 'Telerik' on 'TelerikEmTag'		•

Figure 3 - Standalone version

>	***		BuiltInGr	id 👻 🔤	necord 👻
Step	s	Storyb	oard Da	ata	
1		🖡	1	? 🤁	Browser: Internet Explorer 🔹 ዿ 🦘 400 🔹 🜔 🗸 🔘
	1			4	Navigate to : 'http://www.google.com/'
	2			4	Set 'QText' text to 'Telerik'
	3			4	Click 'BtnGSubmit'
	4			\$	Verify 'InnerText' 'Exact' 'Telerik' on 'TelerikEmTag'
>	5			\$	Verify 'TextContent' 'Exact' 'Telerik' on 'TelerikEmTag'

Figure 4 - VS plugin

29

Create a Data Driven Test

Let's create a new, Data Driven test. We'll go through five iterations of the test, each with a different search text.

- 1. Create a new test as outlined above.
- 2. Select Web Test and alter Test Name if desired. Click OK.
- 3. Open the new test and click "Record."
- 4. Navigate to www.google.com, enter "Telerik" in the Google search box, and hit the Search button.
- 5. Click the "Pause" button in the docked IE taskbar.



6. Click the "Local Data" button in the Test Views ribbon or the Data tab in Visual Studio.

Steps Storyboard Local Data	WebTest1.tstest* × Image: storyboard Image: storyboard
Local Data View the local data source associated only with this test case. Use this data source to quickly setup your test to be data driven.	Columns:

7. There are three buttons at the top of this pane. Only "Create a new data table" is enabled. Click it to add a new grid for data.

Local Te	Local Test Data - WebTest2.tstest*					
Columns: 5 📚 🔒 🔒						
	Col1	Col2	Col3	Col4	Col5	
▶ 1						
*2						

- 8. The default grid will have five columns. This example will execute five iterations of the test with different search text for each.
- 9. Change the columns text box to "1" and click "Update."



- 10. Enter any text into the first grid cell and hit Enter or Tab. The input will move to the second row.
- 11. Continue entering text for the remaining grid cells. New rows are added as you type.

Local Te	Local Test Data - WebTest2.tstest*			
Columns	s: 1 📚 🗟 🔒			
	Col1			
1	One			
2	Two			
3	Three			
4	Four			
▶ 5	Five			
*6				

12. Right click "Col1" and choose "Rename Column." In this example, "Numbers" is the new name.

Local Test Data - WebTest2.tstest				
Columns	5: 1	📚 🔒 🔒		
	Coli			
1	One	Rename Column		
2	Two	Delete Column		
3	Three			
4	four			
<u>}</u>	Five			
₩g				

- 13. Save the test.
- 14. Data from an array can be used in recorded steps and code behind methods. To bind data from a data array to a recorded step, continue with the following steps. To use reference data from the data array in a code behind method, skip to step 24.
- 15. Press the F4 key to open the Properties pane.
- 16. Highlight the recorded step that sets the value of the Google search text box (test step 2 in this example).

17. The properties for this step will appear in the Properties pane.

Properties	
2 ↓	
🗆 Data Driven	<u> </u>
(Bindings)	(Collection)
🗆 Elements	
PrimaryTarget	/Google/QText
SecondaryTarget	
Execution	
Pause	None
WaitOnElements	True
WaitOnElementsTime	
(Bindings)	
Bind data driven proper drop down to see prope	ties against a datasource. Click the rties that support data binding

- 18. Click the drop-down arrow for "(Bindings)."
- 19. Click on the "Text" node in the displayed tree.
- 20. Enter the name of the column you want to draw data from into the text box. In our example we want the first and only column, so we'll enter "\$(Numbers)."

🛃 🛛 Data Driven Editor:	
Select property in treeview below and set the d binding expression: format: \$(columnIndex) or \$(columnName)	ata
🖃 🔲 Properties	
🔤 Text	
\$(Numbers) Set	Clear

- 21. Click the Set button.
- 22. The data for the column named "Numbers" from the data array is now bound to the Text property for that step. Instead of entering "Telerik" into the search box, the data stored in the data array will be entered.
- 23. Save and execute the test. Note that the test will execute for each row in the data array.
- 24. To use reference data from the data array in a code behind method, follow these steps.

- 25. Locate the step that sets the value of the Google search text box (test step 2 in this example). Right click the step and select "Customize Step in Code."
- 26. Choose Visual Basic or C# and click OK.
- 27. Use the "Data" property followed by the index of the column to reference data from the grid. For example:

```
// You can reference the column by index
Pages.Google.QText.Text = (string)Data[0];
// Or by name
Pages.Google.QText.Text = (string)Data["Numbers"];
```

- 28. Save and build the project.
- 29. Execute your test. Note that the test will be executed for each row in the data array.
- 30. If the test executes too quickly to validate visually, click the "Enable Annotation" button and set the delay in milliseconds before executing. These are located in the Quick Execution ribbon.



Test Execution (Standalone)

After recording your steps, it is time to execute your test. You have two options: Quick Execute or as part of a Test List.

The first method is Quick Execute.

1. Click the Execute button.



2. Afterwards, test results are automatically displayed. Click "Log" for more information.

Step	Steps - WebTest.tstest							
	View Class							
0	Pass - 3 passed out of total 3 executed.						×	
	0	1				🐬 Navigate to : 'http://www.google.com/'		•
		2				🕖 Set 'QText' text to 'Telerik'		-
>	0	3				🐬 Click 'BtnGSubmit'		-

The second method is through a test list. Add the test to a list and then execute the test list.

- 🗧 Save . Test Lists Project Record Results Reports Help O Schedule List Dynamic List Edit List Edit Settings Delete Clone Execute List Abort Run TestList Test Lists Tests Type Test List ৰ্ব Owner ব Tests ব T Date Test
- 1. Click the Test Lists tab:

2. Click "List" in the Add ribbon.

	s 🖬 👔	Save .	,				Te	Telerik Test Studio - TestProject		
	P	roject	Record	Test Lists	Results	Repor	ts Help			
Li	st Dyr	namic List	Edit List	Edit Settings	© Delete	Clone	Execute List	Abort Run	Schedule TestList	
	Add			Edit			Execu	tion	Scheduling	
T	est Lists Type To	est List	۲	Date		τow	ner T Tes	its T	Tests Test	

- 3. Give the test a name.
- 4. Add the test to the list on the right.

Add New Test	List	×
E	You can use test lists to manage groups of tests that can be executed together with the results summarized at the test list level	
Test list nam	ne	
TestList1		
Included ter	sts stProject1 WebTest	

- 5. Click OK to save the new test list.
- 6. Click "Execute List" in the Execution ribbon.



7. A browser window or WPF app opens and the test executes. Upon completion, the browser window closes and the Results tab opens.

💦 🗄 Save 🗸		т	elerik Test Studio - Tes	tProject1		
Project	Record Test Lists	Results Repo	orts Help			
Selected Run Analyze	JI Delete Selected Reload	Reload from Publis Server Serv Scheduling	th to Manage Ver Results Server	rd Excel VS Resu Export	It File TFS Build Server	Back Forward Navigate
Run Results and Sci	heduled TestLists for	Run for 6 - 12 April	2011			
<u>D</u> ay	Week	Month	imeline	_		
4-06-2011	4-07-2011	4-08-2011	4-09-2011	4-10-2011	4-11-2011	4-12-2011
			Scheduled			
suits are loaded.						

8. To view the test results, double click the test result entry in the Timeline view ("TestList1" in this example).

Teleri	rik Test Studio - TestProject1	
Project Record Test Lists Results Reports Help		
Selected Run Analyze common	Cel VS Result File TFS Build Server Navigate	
Run Results and Scheduled TestLists for Run for Tuesday, April 12, 2011	TestResults	
Calendar Report	TestList1 Tests	
	Pass - 1 tests passed out of total 1 executed. Executed on 'AOT-WIN7!	ULT' machine. 📋
12 Tuesday	Test Path Tend Time Passe	ed/Total Result
	WebTest.tstest 4/12/2011 10:20:01 AM 3/3	9
10 AM TestList1 4/12/2011 10:19 AM	<u>-</u>	
	_	
Pass Fail Scheduled		Page 1 of 1
Results are loaded.		41

Test Execution (VS Plug-In)

After recording your steps, it is time to execute your test. You have two options: Quick Execute or from Visual Studio Test View.

The first method is Quick Execute.

1. Click the Quick Execute button.



2. Afterwards, test results are automatically displayed. Click "Log" for more information.

WebAi	iiTe	st1.	aii* ×	Test	List Edito	r	•		
3	\$		(none	e) - 🗧	🖗 🔴 R	lecord 👻			
Steps	S	tory	board	Data					
†	👕 🥥 🐥 🏫 🏷 🦿 Browser: Internet Explorer 🔹 🏂 🦠 400 🔹 🜔 🕶 🕕								
🥝 P	ass	-4	passe	d out of	total 4 ex	ecuted.	۹ 🗙		
(0	1			4	Navigate to : 'http://www.google.com/'	*		
	0	2			4	Set 'QText' text to 'Telerik'	•		
	0	3	V		4	Click 'BtnGSubmit'	-		

The second method is from the Visual Studio Test View, using the VS testing framework.

 Ensure the Test View is visible. Click "Test"->"Windows"->"Test View." Test Analyze Window Help

resc	Analyze window help	_			
5	New Test		- 🗟 🚰 🎲 🐋		
\$	Load Metadata File				
2	Create New Test List				
	Run				
	Debug •				
7	Manage Test Controllers	-			
	Select Active Test Settings	<u><u><u></u></u></u>			
	Edit Test Settings				
	Windows •		Test View		
		14	Test List Editor		
		8=	Test Results		
		E	Code Coverage Results		
		Ξ.	Test Runs		
		5	Test Impact View		

2. Right click on your test and click "Run Selection."

st Name	Project TestBroiset1		
tMe	Run Selection		
Þ	Debug Selection		
	<u>O</u> pen Test		
5	New <u>T</u> est		
×	Cut	Ctrl+X	
	Сору	Ctrl+C	
	<u>P</u> aste	Ctrl+V	
\times	<u>D</u> elete	Del	
	Select <u>A</u> ll	Ctrl+A	
171	Add/Remove Columns		
	Open Test List Editor		
1	<u>P</u> roperties		
lution Exp	lorer 🛛 🚡 Team Explorer 💈	🍇 Class View 🛽	👃 Test View

3. A browser window or WPF app will open and the steps of your test will automatically execute. Test Results

0 0	🔠 🏣 🕼 🗛 🔤 🕮 💷 💷 🔠 🔠								
\bigcirc	Test run completed Results: 2/2 passed; Item(s) checked: 0								
	Result	Test Name	Project	Error Message					
	🖉 🕜 Passed	WebTest1	TestProject1	Overall Result: Pass					
	🗋 🥝 Passed	TestMethod1	TestProject1						

4. After the test runs, click the link (in red above) in the "Test Results" tool window to view the execution log.

Admin@AOT-WIN7UL...1-04-13 12:13:56 × WebTest1.tstest

 Result Summary

 Test run name:
 Admin@AOT-WIN7ULT 2011-04-13 12:13:56

 Run result:

 2/2 tests passed, 0 failed, 0 skipped
 Test settings:
 Local
 Submitted by:
 AOT-Win7Ult\Admin
 Started on:
 4/13/2011 12:14:03 PM
 Completed on:
 4/13/2011 12:15:07 PM

How to Resolve Test Step Failures

After running Quick Execution for a test, any failed steps will be marked with a red icon and the overall test will fail.

Step	os - V	Veb	Test.	tstes	t				
🥏 🦆 🎓 🏷 🤻 View Class									
0	3 Fail - 3 passed out of total 4 executed.							Log	×
	0	1				4	Navigate to : 'http://www.google.com/'		•
	0	2				4	Set 'QText' text to 'Telerik'		•
	0	3				4	Click 'BtnGSubmit'		•
>	3	4				\$	Verify 'InnerText' 'NotContain' 'Telerik' on 'TelerikEmTag	g' •	•

Figure 7 - Standalone version

▶	*		Buil	tInGri	d 🕶	🥪 🛑 Record 👻
Step	s S	Story	board	d Da	ta	
1		{	1	• •	୨ ୯	Browser: Internet Explorer 🛛 🗞 🦘 400 🜔 🔘
0	Fail	- 3 p	asse	d out	of total	4 executed.
	0	1				F Navigate to : "http://www.google.com/"
	0	2	V			🗲 Set 'QText' text to 'Telerik'
	0	3				🗲 Click 'BtnGSubmit'
>	3	4				Verify 'TextContent' 'NotContain' 'Telerik' on 'TelerikEmTag'

Figure 8 - VS plugin

1. Double click the X icon to launch the Step Failure Details UI.

2. The Step Failure Details UI gives a failure summary, exception details, and a complete test log.

ep Failure Details												
Failure Images	Page DOM Resolve Failure											
Test Name:	WebTest											
Step Description:	Verify 'InnerText' 'NotContain' 'Telerik' on 'TelerikEmTag'											
Failure Summary												
Content.InnerText Match Type: 'N Expected Resul Value at time	of 'TelerikEmTag' does not match! >tContain' :: 'Telerik' 5 failure: 'Telerik'											
torace are came												
Exception Details	View											
Complete Test Log	View											
	D 🚮 👘											
	Constant Constant Constant											
	Clipboard To File Failure											
		OK Cancel										

- 3. The Failure tab gives a summary. The Images tab shows the browser image at the time of failure versus the expected image. The Page DOM tab shows the page's object model at the time of failure. And finally, the Resolve Failure tab allows you to change Selected Sentences and reverify.
- 4. Upon failure, decide if the test needs updating or the tested page, Web application, or WPF application has a defect. If the test needs updating, simply modify the verification properties and click OK to update the test. Otherwise, the test has detected a product defect and a bug should be logged in your bug tracking system.

Elements Pane Overview

The Elements pane maintains a list of all Elements within the current project. It provides a one-stop shop to view elements and edit the way they are found during execution. The Elements menu bar has two buttons:

- a. Enable/Disable Highlighting this icon on the left controls highlighting of elements on the recording surface as they are selected in the tree view. Click this button to turn on element highlighting, click it again to turn it off.
- b. Refresh click this icon to refresh the display of the elements in Explorer. You seldom should have to do this because Test Studio normally refreshes the window properly.



The HTML tree view is organized by Page -> Frame -> Test Regions -> Elements.

The Silverlight tree view is organized by Page->Frame->SilverlightApp->Elements.

The hierarchy is maintained according to where the element is located on the page. For example, if there are no frames or regions, then elements for that particular page will be listed under the "Page" node.

Right click each Page node to see a context menu with the following:

- a. Validate All Elements validates that all elements can be located on the current page using the current Find Expression settings.
- b. Load Page loads the page in the recording surface.
- c. Properties makes the Properties pane active.

Elements		
9		
🖃 🗐 All Pages		
	Edit Element	
L= 🃎	Validate All Elements	
	Delete	
🗄 🖓 Те 🗎	View Error	
	Locate in Dom	
۰.	Load Page	
P	Properties	

Each Element node has a context menu with three active choices (when the page is loaded in the recording surface):

- a. Edit Element loads the Find Expression Builder.
- b. Locate in DOM highlights the element node in the DOM Explorer tool window.
- c. Load Page loads the URL to which the element belongs in the recording window.
- d. Properties makes the Properties pane active.



Change How an Element Is Found

When a web page element has an action recorded against it, or you explicitly add an element to the "Elements" pane, a "Find Expression" is generated that tells the framework how to find that specific element on the web page.

1. To change how an element is found, right click on the element in the Explorer and select "Edit Element."



In the VS Plug-In, click the "Show Element Explorer" icon in the toolbar (in red below) and locate the Explorer at the bottom of the screen.

	Elements Explorer	
	3	
	⊡… 🗐 All Pages 🛛 🕅	Edit Element
		Validate All Elements
Tools Architecture Test Analyze Window		Delete
Debug + Any CPU +	1	View Error
	a	Locate in Dom
	📰 Test Results Ele 🛸	Load Page
Show Element Explorer	y 🔤 🚰	Properties

2. The "Find Expression Builder" launches and displays how the element is currently found.



- 3. The "Find Expression Builder" consists of the following:
 - a. Web Page Information Friendly Name and Page URL.
 - b. Element Information Friendly Name, the number of tests and steps used in, Element type, Status, and DOM View.
 - c. Find Expression Validation Options three ways to connect to the browser and validate the find expression.
 - i. Connect to existing browser that has the page with the element on it. Click the drop-down to see open instances of IE.
 - ii. Launch a new browser and navigate to Web Page URL. Use this is IE is not open.
 - iii. Choose a step from the projects' tests and execute to this step.
 - d. Current Find Expression Narrative represents the find expression with a simple statement. Edit by hovering over the applicable words or phrases and click when the white pen appears. Delete a statement by clicking the "X" at the beginning of it.
 - i. Element Attribute the property or attribute of the element to examine. HTML element examples include TextContent, NodeIndexPath, TagName, TagIndex, and XPath. HTML attribute examples include name, id, and visibility. XAML element examples include TextContent, XamITag, Name, and TagIndex. XAML attribute examples include Foreground and Content.
 - ii. Change the compare type via the drop-down menu (e.g. Exact, Contains, StartsWith, etc.).
 - iii. Value to look for can be a string, number, index path (tag or XAML index), or a RegEx expression.
 - iv. And/Then Use multiple criteria for locating the correct element. "And" means the current line and the next line must be to true. "Then" equals a chained find

expression. Specify how to find element 'A,' and then underneath element 'A' find element 'B'. This is useful for pages in which multiple custom complex controls exist in different panels. Use a chained find expression to locate the correct panel, and then find the correct element (such as a button) within that panel.

- v. Switch to "Expert View" to validate.
- 4. After specifying the method and criteria used to find the element, click "Expert View" and then "Validate." This validates that the expression finds the element within the current DOM.

NOTE: For more information and a detailed discussion on crafting and using find expressions, please visit our <u>website</u>.



Use the 3D Viewer

The 3D viewer shows a hierarchical view of the elements contained in your application. It starts at a selected element, and then traverses the DOM tree up to the DOM root. It is used to identify/lock on elements and quickly build verifications.

Cock on Surface C
Lock on Surface poople j, bkampidamp, document, f.q. focus/j, lock/high/f.document, image(), social Google Search sarch empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #4272db, emdColosStr = #1254 R progid D.KinageTransform.Microsoft Gradient(start:ColosStr = #4272db, emdColosStr = #1254 empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #4272db, emdColosStr = #1254 Progid D.KinageTransform.Microsoft Gradient(start:ColosStr = #4272db, emdColosStr = #1254 empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #4272db, emdColosStr = #1254 Progid D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 Immediate empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 Immediate empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 Immediate empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 Immediate empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 Immediate empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 empoid/D.KinageTransform.Microsoft Gradient(start:ColosStr = #1254 Immediate
CNPUT class=lsbb/> Vero: Selected Only Add to Procest CNPUT class=lsbb/> Vero: Selected Only Add to Procest CSPAN class=lsbb/> CSPAN class=lsbb/> "PADIGe-Tick-tikic.checked=1 name=btnG value="Coogle Search" type=submic> F CSPAN class=lsbb/> CSPAN class=lsbb/> "PADIGe-Tick-tikic.checked=1 name=btnG value="Coogle Search" type=submic> F CSPAN class=lsbb/> "PADIGe-Tick-tikic.checked=1 name=btnG value="Coogle Search" type T CMAU "PADIGE-Tick-tikic.checked=1 n
Image: proving back procedure in a proving back prov
Indicates labor Add to Project SPAN class=labor SPAN cla
Image: Comparison of the comparison
R projidDXImageTransform.Microsoft.Gradient(tatr/ColorStr=#1222d)te,endColorStr=#1a54 COSPLAY: nore* id=wag(>> = ODSPLAY: nore* id=wag(>> = ODSP
Google Search arch arch </th
Coogle Search arch earch "USPLAY: none" idsugis> "USPLAY: none" idsugis> "USPLAY: none" idsug
Coogle Search arch erch erch </th
Comparison of the second
Find Element Available Verifications View: Selected Only Add to Project #AGKIN: 0px auto; DISPLAY: block; BACKGROUND: none transparent scroll repeat 0% 0%" Inne=sclient value=psy type=hidden> Tophr> ="COSITION: relative" class=tisf-p> #b> Selected Only Add to Project #b> Selected Only Add to Project #b> Selected Only Add to Project #b> Selected Only Add to Project #b> Selected Only Add to Project #b> Selected Only Add to Project #b> Selected Only B Selected Only B Selected Only B <
Find Element Available verifications View: Selected Only Add to Project toph? *DOING-TOR 2px" class=jtb> toph? <input class="lsb" name="btnG" onclick="this.checked=1" type="submit" value="Google Search"/> SPAN class=lsbb> SPAN class=lsb SPAN class=ds> View: View: <t< th=""></t<>
Add to Project "Marchik Opx attri Callson" Find Element Available verifications View: Selected Only Add to Project abb
Find Element Available verifications View: Selected Only Add to Project Find Element Available verifications View: Selected Only Add to Project SPAN class=lsb ondick=this,checked=1 name=btnG value="Google Search" type=submit> SPAN class=lsbb> SPAN class=lsbb> SPAN class=ds> View: Google Search type View: V
Constraint of the second
Find Element Available Verifications View: Selected Only Add to Project "PADDING-TOP: 2px" class=isb> <input class="lsb" name="btnG" onclick="this.checked=1" type="submit" value="Google Search"/> ER> View:
Find Element Available Verifications View: Selected Only Add to Project ="PADDING-TOP: 2px" class=ijb> <input class="lsb" name="btnG" onclick="this.checked=1" type="submit." value="Google Search"/> value=f type=hidden>
<input class="lsb" name="btnG" onclick="this.checked=1" type="submit" value="Google Search"/> ER> <all class="lsb" name="btnG" onclick="this.checked=1" td="" type<="" ut="" value="Google Search"> value=ftype=hidden> value=ftype=hidden></all>
<input class="lsb" name="btnG" onclick="this.checked=1" type="submit" value="Google Search"/> >AN class=ds> <span class="lsbb</td"> >AN class=ds> value=ftype=hidden> >AN class=ds>
 <input class="lsb" name="btnG" onlick="this.checked=1" type<br="" value="Google Search"/>^AN class=ds bondick=this.checked=1 name=btnG value="Google Search" type ^AN class=dsb> y value=f type=hidden>
 value=f type=hidden>
 yvalue=ftype=hidden>
<center> i type=hidden></center>
<div_child="padding_top: 2nv"="" <pre="">childen></div_child="padding_top:>
type=hidden>
<form id="tsf" role="searc</th" style="MARGIN: 0px auto; DISPLAY: block; BACKGROUND: none transparent scroll repeat 0% 0%"></form>
<div id="searchform"></div>
<BODY onload="try{lgoogle.j.b&&document.f.q.focus()}catch(e){};if(document.images)new Image().src='/image</th>
<html> TON: relative MARGIN: 0x 8or DISPLAY: block: HEIGHT: 4660* ideord class_tdgs></html>

There are two ways to open the 3D viewer:

- 1. Use the pop-out menu.
 - a. In the recording browser window, enable hover over highlighting and place the mouse cursor over an element to examine in the 3D viewer.
 - b. Wait for the blue circular nub to appear.



c. Drag this nub. It changes into an image of the highlighted element and a pop out panel appears on the left side of the screen with three buttons in it. Drop the image of the element onto the middle button (3D viewer). The 3D viewer will open and the dragged



- d. Drag the element onto the top button and it adds that element to the Elements Explorer tool window.
- e. Drag the element onto the bottom button and it locates that element in the DOM tree and highlights it in the DOM Explorer tool window.
- 2. Use the Element Menu.
 - a. Perform steps A & B above.
 - b. Click the blue nub. The Element Menu opens.
 - c. Click the View 3D button.



d. The 3D viewer opens and the specified element is selected in the viewer.

Once the 3D viewer is open, perform the following:

- 3. Traverse the DOM layers by:
 - a. Clicking on an element in the element list.
 - b. Dragging the blue nub left/right (shown below).

🔶 😔 http://www.google.com/ 🔎 – C 🗙 🚼 Google 🛛 🗙	舵 🛣 🐯
Lock on Surface	×
Googl Google Search	
Long Long Long Long Long Long Long Long	-

- c. Clicking on the left/right arrows (shown above).
- 4. Increase/decrease magnification of the screenshot by clicking the or + symbol (shown above).
- 5. Click the "Lock on Surface" button to:
 - a. Close the 3D viewer and return to the IE recording window.
 - b. Highlight that element.
 - c. Open the Element Menu.

A http://www.google.com/ D + C × C Google ×	ĥ ☆ ‡3
Lock on Surface	×
Good Good Goode Search	
coogie coulon	
	_
0000 0000	
(-) (-) (+)	
Find Element Available Verifications View: All Categories Selected Only Add to Preserve and the preserve and th	roject
Content Verify InnerText Contains (empty)	
Content Verify TextContent Contains (empty)	
Content Verity InnerMarkup Contains <input class="lsb" onclic<="" td=""/><td></td>	
Content Verity OuterMarkup Contains <input c<="" td=""/><td></td>	
Content Verify StartTagContent Contains 	

6. Click the "Available Verifications" tab to display a list of prebuilt verifications.

- 7. The View group boxes control which of the available verifications to display. Change the displayed category with the Categories drop-down. Limit the view to only selected verifications by ticking the "Selected Only" checkbox.
- 8. Quickly add verifications by ticking the checkbox(es) for the desired verification(s) and then clicking "Add to Project."
- 9. When a Verification is highlighted, two buttons appear next to it. The first button (on the left) will test whether or not the verification passes or fails against the currently loaded web page. The other button will locate the selected element in the DOM tree and highlight it in the DOM Explorer tool window.

	Fin	d Element	Availab	e Verifications	Vie	W:	All Categ	ories 🚿	Selected Only	Add to Project	
Ū		Content	Verify	InnerText 💌	Contains		(empty)	L		f	9
		Content	Verify	TextContent	Contains	(em	pty)				
		Content	Verify	InnerMarkup	Contains	(en	npty)				

10. You also have the option of modifying the verification before you add it to the project. Use the drop-downs and the edit boxes to modify the verification before clicking "Add to Project."

Perform Common Automation Tasks

This picture shows an expanded view of the Element Menu with all of the different tasks you can accomplish with it:



Telerik Test Run Configuration (Standalone)

To configure Telerik Framework settings for execution:

- 1. Click the Test Lists tab.
- 2. Highlight a test list to configure by clicking on it in the Test Lists pane on the left.
- 3. Click "Edit Settings" in the Edit ribbon.

17	🚦 Save 🔵 🗸													Telerik Te
	Project	Record	Test Lists	Results	Repo	rts Help								
List	Dynamic List	Z Edit List	Edit Settings	Delete	O O Clone	Execute List	Abort Run	5 Schedule TestList						
	Add		Edit			Execu	ution	Scheduling	1					
Test	: Lists	_	_			_	_	_		_			_	
Тур	e Test List										τ	Date T	Owner T	Tests T
	TestList1											4/12/2011 10:18:37 AM		1

4. This opens a dialog box that shows all of the possible configuration settings for that test list. Change the settings as needed for your test list.

Edit Test List Settings X						
General Web Wpf						
Annotation						
AnnotateExecution						
AnnotationMode	All 👻 🗉					
 Desktop 						
SimulatedMouseMoveSpeed	0.3					
▲ Execution						
ClientReadyTimeout	30000					
ExecuteCommandTimeout	20000					
ExecutionDelay	0 *					
	OK Cancel					

Telerik Test Run Configuration (VS Plug-In)

To configure Telerik Framework settings for execution:

1. Double click "Local.testsettings" under "Solution Items" in the Solution Explorer.

🤜 Solution 'TestProject1' (1 project)
a 🔛 Solution Items
😥 Local.testsettings
🚮 TestProject1.vsmdi
庭 TraceAndTestImpact.testsettings
IstProject1
Properties
省 AssemblyInfo.cs
References
🚞 Data
Settings.aiis
🖞 UnitTest1.cs
WebTest1.tstest
🔚 WebTest1.resx

- 2. On the left side, click "Telerik Test Studio."
- 3. This gives you a page with three tabs to configure the run time configuration settings for all test executions within this project.

es I D'	Test Studio tests run settings. Configure the execution settings for Test Studio tests.							
a and Diagnostics								
up and Cleanup Scripts	General Web Wpf							
its								
t Timeouts								
: lest Test	Annotation							
rik Test Studio	AnnotateExecution	False						
	AnnotationMode	All						
	✓ Desktop							
	SimulatedMouseMoveSpeed	0.3						
	✓ Execution							
	ClientReadyTimeout	30000						
	ExecuteCommandTimeout	20000						
	ExecutionDelay	0 HandleAndFailTest 500 True						
	UnexpectedDialogAction							
	WaitCheckInterval							
	XMultiMgr							
	⊿ Log							
	CreateLogFile	True						
	LogAnnotations	False						
	LogLocation	C:\WebAiiLog\						
	QueryEventLogErrorsOnExit	False						
	AnnotateExecution							
	Whether or not to highlight/annotate the ta	arget element that the current action is being executed against.						

Steps Pane Overview (Standalone)

St	tep)5 -	Weł	oTest	.tstes	st*		
4	3	.	ſ.	î	5	C		View Class
		1					4	Navigate to : 'http://www.google.com/'
		2					4	Set 'QText' text to 'Telerik'
		3		۲			4	Click 'BtnGSubmit'
>	>	4	V		Q		ø	Verify 'InnerText' 'Exact' 'Telerik' on 'TelerikEmTag'

The Steps pane provides the list of steps in the currently selected test. Each step has:

- a. Order of the Step.
- b. Enabled checkbox whether the step will run during execution.
- c. Breakpoint indicator whether the test will pause at that step and show Debug Options.
- d. Continue on failure indicator whether the test will stop if that step detects a failure.
- e. Type Icon represents the type of step: Action, Verification, Coded, etc.
- f. Step description.
- g. Delete Button deletes the step from the test.

The menu bar for the Steps tab contains (from left to right):

- a. Toggle Highlighting when enabled, the element of the selected step is highlighted in the recording surface.
- b. Move Step Down moves selected step down one in the sequence.
- c. Move Step Up moves selected step up one in the sequence.
- d. Undo restores a step you just deleted/changed to its previous state and location.
- e. Redo reapplies the edit that was undone with the Undo button.
- f. Class View –switches the view between the code for the currently selected coded step and the entire code behind file (if the test has a code behind file).



Steps Pane Overview (VS Plug-In)

3	**	: 📮 I	BuiltInG	òrid	- -	🖉 🔴 Record 👻
Step:	s	Storyb	oard D)ata		
*		1	1	5	¢	Browser: Internet Explorer - 1 🗞 🦘 400 - 🜔 - 🔘
	1				4	Navigate to : 'http://www.google.com/"
	2				4	Set 'QText' text to 'Telerik'
	3	V			4	Click 'BtnGSubmit'
>	4	V	Q		ø	Verify 'InnerText' 'Exact' 'Telerik' on 'TelerikEmTag'

The Steps pane provides the list of steps in the currently selected test. Each step has:

- a. Order of the Step.
- b. Enabled checkbox whether the step will run during execution.
- c. Breakpoint indicator whether the test will pause at that step and show Debug Options.
- d. Continue on failure indicator whether the test will stop if that step detects a failure.
- e. Type Icon represents the type of step: Action, Verification, Coded, etc.
- f. Step description.
- g. Delete Button deletes the step from the test.

The menu bar for the Steps tab contains (from left to right):

- a. Clear All Steps deletes all steps from test.
- b. Toggle Highlighting when enabled, the element of the selected step is highlighted in the recording surface.
- c. Move Step Down moves selected step down one in the sequence.
- d. Move Step Up moves selected step up one in the sequence.
- e. Undo restores a step you just deleted/changed to its previous state and location.
- f. Redo reapplies the edit that was undone with the Undo button.
- g. Browser Drop-Down selects which browser to use for Quick Execution.
- h. Show Debugger Options.
- i. Enable Test Annotation enables test annotation during Quick Execution. To change the execution across all tests, change the setting in the test configuration as previously explained.
- j. Execution Delay sets the time (in milliseconds) to delay between each step.
- k. Quick Execute immediately executes the current test in the selected browser.
- I. Stop Execution halts a running test.
- m. Add This drop-down adds special steps that cannot be recorded. They include:
 - i. Capture Desktop captures a screenshot of the desktop and stores it in a file.
 - ii. Capture Browser captures a screenshot of the browser and stores it in a file.
 - iii. Custom Annotation inserts personalized annotations into the test.

- iv. Test as Step include another test as a test step in this test. Design Canvas will run the other test when it comes to this step and, when complete, resumes running the rest of the steps of this test.
- v. Delay Execution causes the test to pause for the specified number of milliseconds.
- vi. Clear Cookies clears browser cookies.
- vii. Wait for URL waits for a specific URL to appear in the navigation bar of the browser. This is useful for page redirection or form posting where there can be a processing delay by the web server.
- viii. Inspection Point pauses the test and opens a window with the current DOM. The test resumes when the DOM window is closed.
- ix. Comment adds a comment to the test.
- x. Manual Step adds a manual step to the test.
- Dialog add specific dialog handlers as test steps. The dialogs that can be handled are Alert, Confirm, Logon, File Upload, File Download, Logon, and a special Generic dialog handler, which simply closes most dialogs.
- o. Logical inserts conditional statements as test steps. These include if...else, while...loop, and loop.
- p. Recapture Storyboard recaptures screen shots of each test step if the web site or application changes.
- q. Preview Code displays code with the ability to copy to clipboard.

DOM Explorer Tool Window Overview

DOM Explorer
🖅 🚠 💠 Page (telerik - Google Sear 👻 🛱
🔺 🗔 <body -1000em"="" absolute;="" display:="" href="/setprefs?sig=0_wZhLwhNe6ldLHLZ2Eem85aNT9f8=&suggon=2&</p></td></tr><tr><td>▷ 🟥 <TEXTAREA style=" id="csi" left:="" none"="" onload="try{!google.j.b&&document.f.q.focus()}catch(e){};if(document.images)new Image().src='/images/nav_logo</td></tr><tr><td>A </body>
▷ 😂 <script></script>

The DOM Explorer tool window displays the Document Object Model for the currently loaded page in the IE recording window or WPF app. The nodes are a hierarchical representation of all the HTML/XAML elements. Each node in the tree is listed by:

<[Tag Name] [attributes]>

The toolbar for the DOM Explorer contains (from left to right, top to bottom)

- a. Enable highlighting when enabled, the element currently selected in the DOM Explorer is highlighted in the IE recording window or WPF app by having a red rectangle drawn around it.
- b. Hierarchal View displays the DOM as a hierarchal tree in the order displayed.
- c. TagName displays the DOM sorted by tag names.

- d. Move to main element drop-down lists the root element, frames, and any Silverlight applications found on the current page. Selecting one of these from the drop-down highlights that element in the DOM Explorer tool window.
- e. Refresh DOM rebuilds the tree from the current page.
- f. Search enables another toolbar below the first. This second toolbar searches the DOM tree for elements within it.
 - i. Standard search does simple string search and match.
 - Search Using Find Expressions click the pencil icon (in red below) to change the search method. Use the following syntax, referenced from the Expert View in the Find Expression builder: element attribute=string (see below for example).



- g. Perform Search evaluates the find expression and highlights the first matching element in the DOM Explorer.
- h. Move to Previous Result
- i. Move to Next Result

Right click a node to bring up a context menu containing these options:

- a. Goto provides a shortcut for moving to a significant layer of the DOM tree (Page root element, embedded frames, Silverlight applications).
- b. Show Element Menu opens the Element Menu with options for the selected element.
- c. Add to Project Elements adds the currently selected element to the "Elements Explorer" tool window and craft a default FindParam for that element.
- d. Copy to Clipboard copies the HTML/XAML to the clipboard. Data is available to paste into another application or report.

e. Properties – opens a properties window showing detailed information about the selected element.

Properties	X
2↓ 🖾	
🗆 Misc	▲
AbsoluteSiblingTagInde	8
AbsoluteTagIndex	136
Attributes	(Collection)
ChildNodes	(Collection)
Children	(Collection)
Content	<TEXTAREA style="DISPL</th>
CssClassAttributeValue	
Data	
Depth	2
ElementType	TextArea 💌
AbsoluteSiblingTagInde	×

VS Plugin Telerik Test Tab Overview

00	TestPr	roje	ct1 -	Mic	rosof	ft Visual	Studio (Ad	Iministrat	or)									
File	Edit	t \	/iew	Τe	elerik	Proje	t Build	Debug	Team	Data	Tools	Architecture	Test	Analyze	Window	Help		
: 6	- 1	•	2		9	1 X D	1 1 1	- (° -	JE - E	\$ Þ	Debug	- A	ny CPU		-	2		·
: 5) %)	> 🐐	> *1		2 K	9 😥	🖏 🗔 🗄	= = =	5	Ċ	φ≣ <mark>,</mark> [[50 CD -						
ŧ۳	Web	Test	1.tst	est*	х													
Serve	3	***		Bui	iltInG	rid 👻	🥪 🌰 R	ecord 👻]									
ër Ex	Step	s	Story	boar	d D	ata												
plorer		0		6	1	? 🤆	Browse	r: Interne	t Explorer	•	<u>\$</u> 9	400 🔹 🜔	- 0	Add	Dialogs	 Logical 	 ۵	
		1	V			4	Navigate	to : 'http://	www.goog	gle.com	v"							
		2	V			4	Set 'QTe:	d' text to 'T	elerik'									
		3	V	۲		4	Click 'Btn	GSubmit'										
	>	4			Q	ø	Verify 'Inr	nerText' 'Ex	act' 'Tele	rik' on '	TelerikEn	nTag'						
																		-

Each Telerik test loaded in Visual Studio has its own document tab that allows for step visualization and test manipulation. The test tab has a menu bar with the following icons:

- a. Add code behind file adds a new .cs or .vb code behind file which is then logically attached to the test in Solution Explorer.
- b. Convert all steps to code.
- c. External data sources a drop-down menu used to bind the test to an external data source for data driven testing. Use Excel, CSV, XML, or external database files as data sources.
- d. Configure Silverlight App loads Silverlight Out-of-Browser options.
- e. Record opens the Recorder window and directly enables recording for the test.

Note: With the Storyboard tab selected, some recorded steps will not have an image associated with them. This is due to the nature of the action. For example, the "Navigate To" action has no image.



Click on any of the step images in the Storyboard tab to bring that image forward and move to that step on the Steps tab. Each recorded image has the element that it was recorded against highlighted within the image.

The test tab also contains a "Data" tab. It allows for the creation of a simple, Excel-like data array to be used by the test steps. See the "Creating a Data Driven Test" section for more details and an example.

Customize Test Using C# or VB.NET Code

Telerik Test Studio supports code behind files. This allows you to write code and have it executed as a test step. It is for scenarios that require a test step that is more complex than what can be composed with the Verification Builder or the actions provided by the Element Menu.



Create a Test with a Custom Coded Step

TestProject1 - Microsoft Visual Studio (Administrator)
ile Edit View Telerik Refactor Project Build Debug Team Data Tools Architecture Test Analyze Window Help
🛐 = 🖼 = 🧭 🚽 🐇 ங 🖄 🔊 = 🔍 - 💭 = 🏷 👂 🔜 💿 = -
▋ ∿∿ % % ≥@@@ ©@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
WebTest1.tstest.cs* × WebTest1.tstest*
😤 TestProject1.WebTest1 - 🔍 WebTest1_CodedStep()
<pre>// to access data for a specific data iteration. // // Example: // // [CodedStep("MyCustom Step Description")] // public void MyCustomStep() // { // // Custom code goes here // ActiveBrowser.NavigateTo("<u>http://www.google.com</u>"); //</pre>
<pre>// // Or // ActiveBrowser.NavigateTo(Data["url"]); // } // }</pre>
<pre>public class WebTest1 : BaseWebAilTest { [Dynamic Pages Reference] [CodedStep(@"Set 'QText' text to 'Telerik'")] public void WebTest1_CodedStep() { // Set 'QText' text to 'Telerik' Pages.Google.QText.Text = "Telerik"; } }</pre>
<pre>} // Add your test methods here } </pre>

- 1. Create a test as outlined above.
- 2. Record a few steps as outlined above.
- 3. There are two ways of creating a code behind file for your test.
 - a. Standalone click the "Script Step" button in the Add ribbon.



VS Plug-In – click the "Add code behind file" button in the test toolbar.

V		\$ []	🏮 B	uiltI	nG	rid	•	Ŷ	
Ste	ps	Sto	orybo	ard	D	ata			
	1		Ŷ	ſ		5	¢		e
	, ,								

- OR -

- b. Right click on any step and select "Customize Step in Code."
- 4. Step 3a creates a code behind file with an empty test method. Step 3b creates a code behind file with the selected step converted into a coded method matching the selected step name.

5. Note that a step with generated code is read-only in the Steps tab. The description can only be changed in the methods "CodedStep" attribute that appears in the code behind file. A coded step is indicated with a code icon.

Steps	5 -	WebTest.	tstesť	•
	4	l 🕇	5	View Class
:	1			🐬 Navigate to : 'http://www.google.com/'
	2			≫ [WebTest_CodedStep] : Set 'QText' text to 'Telerik'
3	3			🐬 Click 'BtnGSubmit'
> 4	4			🔗 Verify 'InnerText' 'Exact' 'Telerik' on 'TelerikEmTag'

▶	***	🚺 E	BuiltI	nGrid	🝷 🐷 🧶 Record 👻	
Step	\$	Storybo	bard	Data		
		I I	Ŷ	5	🦿 🛛 Browser: Internet Explorer 🛶 😫 🦘 400 💌 🜔	•
	1				🐬 Navigate to : 'http://www.google.com/'	
	2				[WebTest1_CodedStep] : @"Set 'QText' text to 'Telerik'	
	3				Click 'BtnGSubmit'	
>	4				🔗 Verify 'InnerText' 'Exact' 'Telerik' on 'TelerikEmTag'	

6. Once a step is converted to code, it cannot be converted back to a regular step.

(Th		Save).					Teler	rik Test Studio	- TestProjec	t1						
	/	Project		Record	Test Lists	Results	Reports	Help									
Recor	d T	ile ile		Cut Copy Paste Edit	Clear Steps	Storyboard Test View:	Local Data	(2) € (400)) 🙆 🥑 (0 🔹 🤽 Quick Er	Execution	te Abo	ort C B Sil	Out-Of- irowser	Script	s Step Step Logical	Dialogs	Recapture Storyboard Recapture
St	eps - `	WebTe	est.ts	test*	_	-	-	-	_			Prop	erties	-	_	_	
	1	l 🕈) C	View Class								21 🖻				
	1				Navigate to	: 'http://ww	w.google.co	m/'		*		🗆 Da	ita Drive	en -			_
	2				WebTest_C	odedStep] :	Set 'QText'	text to 'Tele	rik'	*		(Bi	indings)		(Collection)		- 11
	3				Click 'BtnGS	ubmit'				-		Pa	use		None		
>	4				🤣 Verify 'Inner	Text' 'Exact'	'Telerik' on	TelerikEmT	'ag'	-		🗆 Mi	isc				
												Pri Re Ru (Bind Bind drop	imaryTar quiresSil InsAcain fings) data driv down to	rget verlight st ven propert o see prope	False AllBrowsers ties against a da rties that suppo	tasource. rt data bir	▼ Click the nding
	PI Ref	ference									×	Elem	ients	_		_	
•		78 79 30 31 32 33 34 35 36 37 38		[p ())	CodedStep(8 ublic void // Set ' Pages.Go	"Set 'Q WebTest QText' bogle.QT	Text' te: _CodedSt text to ext.Text	xt to 'T =p() 'Telerik = "Tele	elerik'"; ;' ;';		4 III +		All Page Good Of the Cool Of	is Igle QText BthGSubmit erikGoogleSe BthGSubmit erikGoogleSe TelerikEmTa	: sarch : sarch0 ag		
Ready																	.4

Create a Test That Uses a Code Behind File

```
WebTest1.tstest.cs* 🗙 WebTest1.tstest*
✤TestProject1.WebTest1
         // to access data for a specific data iteration.
          11
         // Example:
          11
         // [CodedStep("MyCustom Step Description")]
         // public void MyCustomStep()
// {
          // Custom code goes here
ActiveBrowser.NavigateTo("<u>http://www.google.com</u>");
          11
                   // Or
          //
// }
                   ActiveBrowser.NavigateTo(Data["url"]);
          11
   ė
          public class WebTest1 : BaseWebAiiTest
          {
               [ Dynamic Pages Reference ]
   ŧ
              [CodedStep(@"Set 'QText' text to 'Telerik'")]
public void WebTest1_CodedStep()
{
   ė
                   // Set 'QText' text to 'Telerik'
Pages.Google.QText.Text = "Telerik";
               }
               // Add your test methods here...
          }
    [}
```

- 1. Create a test and add a code behind file using Step 3a outlined above.
- 2. To see the entire code behind file, click the "View Class" button below the test title.

S	Step)s -	Web	Test.ts	test*	_	
1	9	,	l 1	1 1	୨ ୯		View Class
		1				4	Navigate to View Entire Code Behind File
		2				1	[WebTest_CodedStep] : Set 'QText' text to 'Telerik'
		3				4	Click 'BtnGSubmit'
	>	4				ø	Verify 'InnerText' 'Exact' 'Telerik' on 'TelerikEmTag'

- 3. Just like a step that has been converted to code, a method in the code behind file is represented as a step in the Steps tab.
- 4. Notice a code method skeleton is created. Change the description and/or the method name to suite your needs.

Ste	ps -	WebTe	est.ts	test^		
0		₽	1	C	View Class	
	1				Ø Navigate to : 'http://www.google.com/'	*
	2				WebTest_CodedStep] : Set 'QText' text to 'Telerik'	*
	3				🗲 Click 'BtnGSubmit'	*
	4				Verify 'InnerText' 'Exact' 'Telerik' on 'TelerikEmTag'	*
>	5				≫ [WebTest_CodedStep1] : New Coded Step	*
A.D	I Ro	foronce				~
AP	I Re	ference	2			×
AP	I Re	ference	2		}	*
АР	I Re	ference 34 35 36	2		}	*
AP	<u>I Re</u> 8 8	ference 34 35 36 37	2) [CodedStep(@"New Coded Step")]	*
AP	I Re 8 8 8 8	ference 34 35 36 37 38	2		<pre>} [CodedStep(@"New Coded Step")] public void WebTest_CodedStep1()</pre>	*
АР	I Re	ferenci 34 35 36 37 38 39	2		<pre>} [CodedStep(@"New Coded Step")] public void WebTest_CodedStep1() {</pre>	*
AP	1 Re 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8	ference 34 35 36 37 38 39 90	2		<pre>} [CodedStep(@"New Coded Step")] public void WebTest_CodedStep1() { }</pre>	*
AP		ference 34 35 36 37 38 39 90 91 91	2		<pre>} [CodedStep(@"New Coded Step")] public void WebTest_CodedStep1() { }</pre>	*
AP	1 Re 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8	ference 34 35 36 37 38 39 90 91 91 92 91	2	}	<pre>} [CodedStep(@"New Coded Step")] public void WebTest_CodedStep1() { }</pre>	*
AP		ference 34 35 36 37 38 39 90 91 91 92 93 93 94	2	}	<pre>} [CodedStep(@"New Coded Step")] public void WebTest_CodedStep1() { }</pre>	

- 5. Do not remove the "CodedStep" attribute. This is how Telerik Test Studio recognizes custom coded steps versus other methods it should ignore.
- 6. In order to write custom steps in code with the VS Plug-In, add a method in the code behind file that takes no parameters and has "void" as its return type.
- 7. Now add code to the method. The method must be decorated with the "CodedStep" attribute:

```
[CodedStep("MyCustom Step Description")]
public void MyCustomStep()
{
    // Custom code goes here
    ActiveBrowser.NavigateTo("http://www.google.com");
}
```

- 8. Add the above method or one similar to the code behind and save.
- 9. The name of the method becomes the step name. The Coded Step attribute description is listed as the description in the steps tab.
- 10. Save and build the project.
- 11. Execute the test.
- 12. The code behind file has access to all Telerik run-time objects, like ActiveBrowser. This is an identical coding experience to the Telerik run-time automation framework.

How to Reference an Element in a Code Behind File

1. Pages can be referenced in the code behind file in the same way they are found hierarchically in Elements Explorer.



2. For example, in the image above, the "BtnGSubmit" element is referenced this way:

Pages.Google.BtnGSubmit

3. It returns the strongly typed Telerik framework "HtmlInputSubmit" type.