Borland
A MICRO FOCUS COMPANY

Silk Test Workbench 15.5

ビジュアル テスト入門ガイド

Micro Focus
575 Anton Blvd., Suite 510
Costa Mesa, CA 92626

Copyright © Micro Focus 2014. All rights reserved. Silk Test は Borland Software Corporation に由来する成果物を含んでいます, Copyright © 2014 Borland Software Corporation (a Micro Focus company).

MICRO FOCUS, Micro Focus ロゴ、及びその他は Micro Focus IP Development Limited またはその米国、英国、その他の国に存在する子会社・関連会社の商標または登録商標です。

その他、記載の各名称は、各所有社の知的所有財産です。

2014-06-06

目次

Silk Test Workbench ビジュアル テスト チュートリアル 4

GUI の概要 4

Silk Test Workbench のメイン画面 4

開始画面 5

ビジュアル ナビゲータ 6

ビジュアル テストの記録：概要 7

サンプル Web アプリケーションの起動 8

サンプル Web アプリケーションに対するビジュアル テストの記録 8

ビジュアル テストの保存と名前の指定 9

記録されたテスト ステップの確認 9

記録されたビジュアル テストの再生 11

結果の分析：概要 11

[結果] ウィンドウ 12

[結果] ウィンドウのタブの使用 14

[結果] ウィンドウのツールバーの使用 15

[プロパティ] ペインの使用 15

画面プレビューの使用 15

ビジュアル テストの拡張：概要 16

画面プレビューからの更新 17

検証の挿入 17

アプリケーション データを保存するローカル変数の作成 18

ローカル変数へのアプリケーション データの格納 19

拡張したビジュアル テストの再生と分析 20

ビジュアル テスト内でのビジュアル テストの実行：概要 20

モジュール式テスト 21

2 つめのビジュアル テストの記録 21

別のビジュアル テストへのビジュアル テストの挿入 22

再生エラーへの対応：概要 23

モジュール式テストの再生 23

エラーのデバッグ 23

再生中の変数の追跡 24

結果の確認 25

アクティブ データの使用 25

アクティブ データ ファイルの確認 26

アクティブ データ テスト資産の作成 27

アクティブ データ ファイルの繰り返しロジックの作成 27

繰り返しステップの定義 28

アクティブ データのリテラル データへのマッピング 29

アクティブ データのビジュアル テストの再生と分析 30

ビジュアル テストからのスクリプトの再生 30

乱数を生成するスクリプトの作成 31

スクリプトの入力パラメータの定義 32

スクリプトの出力パラメータの定義 33

スクリプトのデータを使用するビジュアル テストの設定 33

ビジュアル テストでのスクリプト データの使用 34

テスト結果の再生と確認 34

Silk Test Workbench ビジュアル テスト チュートリアル

Silk Test Workbench ビジュアル テスト チュートリアルにようこそ。ご自身のペースで学習できるこのチュートリアルでは、Silk Test Workbench のビジュアルでストーリーボードベースのインターフェイスを使用して、パワフルで柔軟性の高い機能テストを作成する方法について説明します。 このチュートリアルでは、ビジュアル テストの作成、ビジュアル テストの再生、および再生結果の分析に必要な基本ステップについて学習します。 また、多数の機能を使用して、記録したビジュアル テストをすばやく更新および拡張する方法についても学習します。

このチュートリアルでは、Silk Test Workbench を使用して繰り返し可能なテストを作成する練習をするための Silk Test サンプル Web アプリケーション *http://demo.borland.com/InsuranceWebExtJS/* を使用します。

各レッスンは前のレッスンの出力に基づいて進められるため、チュートリアルのレッスンは順番に完了してください。

GUI の概要

このセクションでは、メイン画面、開始画面、およびビジュアル ナビゲータを含めた GUI について説明します。 このセクションはオプションです。 すでに開発環境の基本要素をご存知の場合は、次のレッスンに進んでください。

Silk Test Workbench のメイン画面

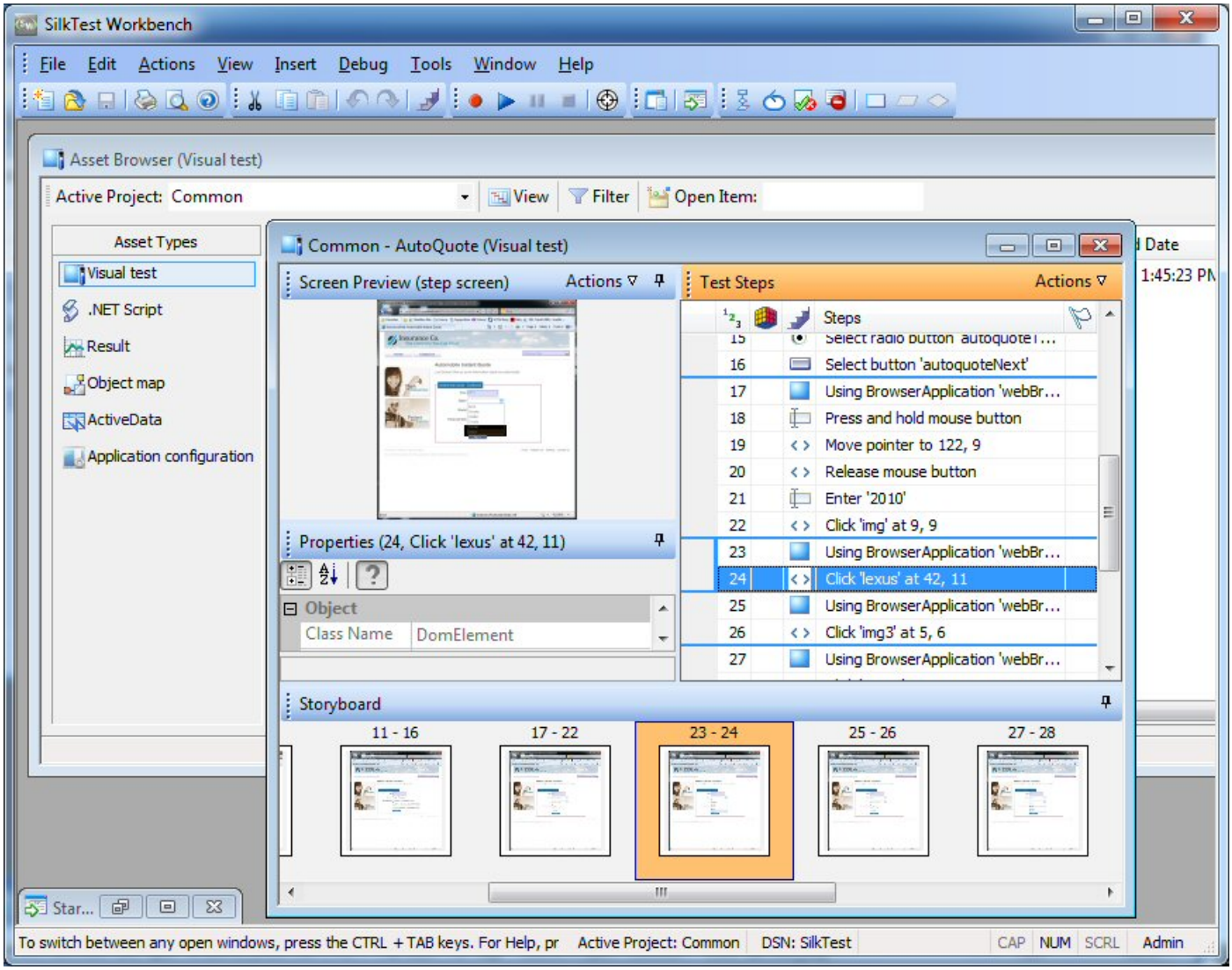
Silk Test Workbench のメイン画面は、テストにかかわるすべての操作の開始点になります。 メニューバーを除き、デスクトップに表示される項目はすべてユーザーが制御できます。

開始画面、ビジュアル ナビゲータ、アセット ブラウザなど、Silk Test Workbench のその他すべてのウィンドウは、Silk Test Workbench のメイン画面の子ウィンドウとして表示されます。

Silk Test Workbench のメイン画面には以下の要素が含まれます。

- メニューバー
- ツールバー
- メインのテスト資産表示領域
- ステータスバー

以下の図は、Silk Test Workbench のメイン画面のこれらの要素を示しています。 メインのテスト資産表示領域では、ビジュアル ナビゲータにビジュアル テストが表示され、背景にアセット ブラウザが表示され、開始画面が最小化されて表示されています。



開始画面

開始画面 では、あらゆるアプリケーションのテスト ソリューションの作成作業をすばやく開始できます。 **開始画面** は、ビジュアル テストと .NET スクリプトを作成したり管理したりする際の開始点であり、共通して行うすべてのテスト作業にリンクしています。 **開始画面** には 4 つのペインがあります。

スクリプト 新しいビジュアル テストと .NET スクリプトにアクセスして記録するには、**スクリプト** ペインを使用します。 また、最近アクセスしたビジュアル テストと .NET スクリプトの結果を開いたり、再生したり、表示したりすることもできます。

タスク **タスク** ペインを使用して、頻繁に使用する機能にアクセスします。

フラグ **フラグ** ペインを使用して、データベース内のテスト プロジェクトの他のテスト担当者やユーザーと、テスト スイートやテスト プロジェクト情報を共有します。

ヘルプ **ヘルプ** ペインからは、すばやく習得するためのアシスタンスに直接アクセスしたり、必要に応じてすばやく製品情報にアクセスしたりできます。

Silk Test Workbench の起動時に **開始画面** を表示するには、**カスタマイズ** をクリックしてオプションを変更します。

開始画面 を非表示にするには、**閉じる** をクリックします。

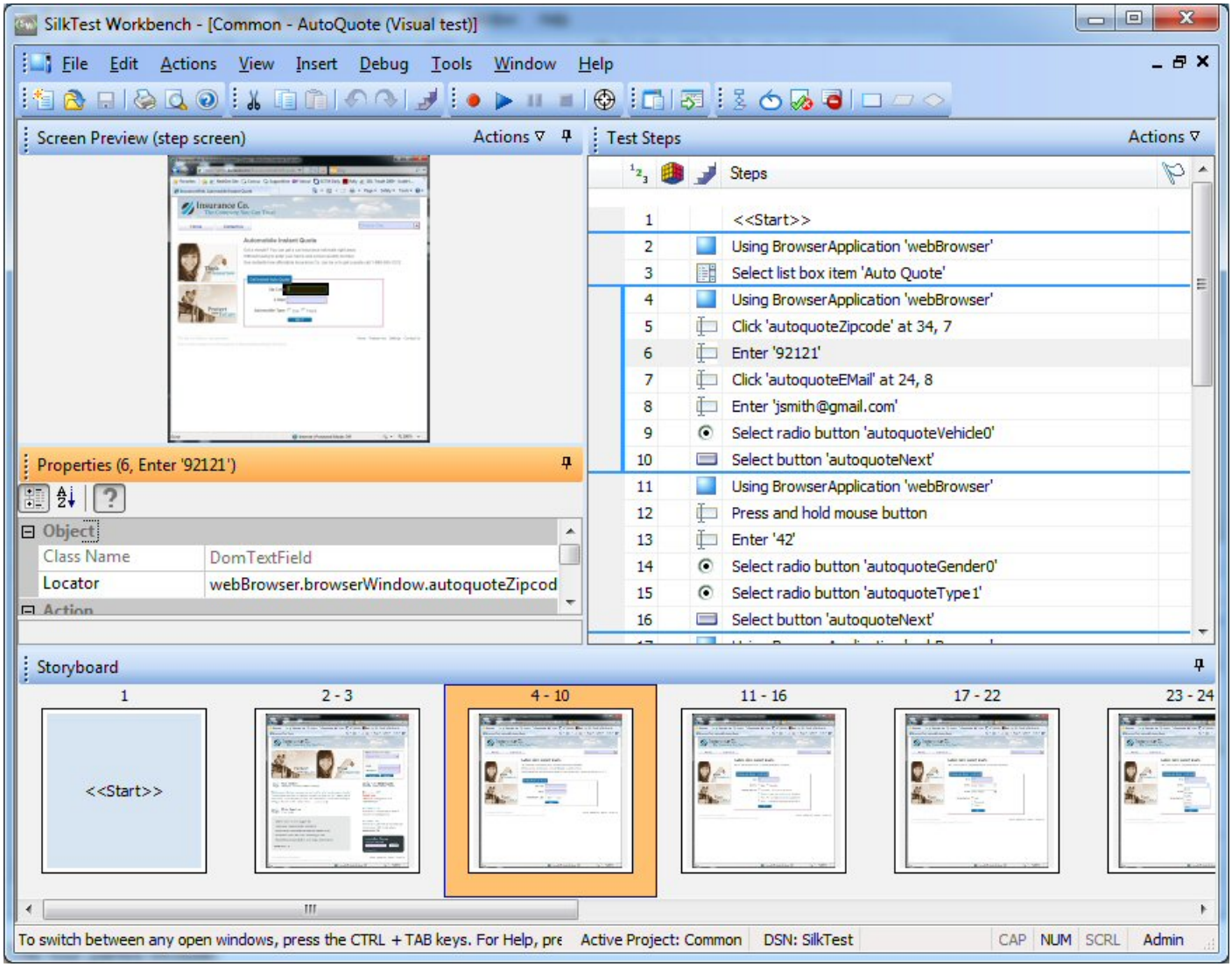
開始画面 の **タスク** ペインと **ヘルプ** ペインを非表示にする場合は **下部のパネルの縮小** を、表示する場合は **下部のパネルの拡張** をクリックします。

ヒント: この画面は、透かし表示されている **開始画面** をクリックすることで、メイン画面からいつでもアクセスできます。 また、いつでも Ctrl+Alt+S を押すか、または **表示** > **開始画面** を選択して、開始画面を表示できます。

注: ユーザー インターフェイスは、Windows 標準の大きいフォントや小さいフォントを使用します。 カスタムのフォント サイズを使用すると、ユーザー インターフェイスのテキスト表示が不正確になる可能性があります。

ビジュアル ナビゲータ

ビジュアル ナビゲータ は、ビジュアル テストの要素をグラフィカルに表すものであり、ポイント アンド クリック インターフェイスを通して各要素を操作できます。 **ビジュアル ナビゲータ** では、ビジュアル テストに関する情報が 4 つのペインに表示されます。4 つのペイン全体でビジュアル テストの各ステップを総合的に把握できます。



4 つのペインは以下のとおりです。

テスト ステップ	ビジュアル テストの各ステップが技術的ではない明確な言語で示されます。
画面プレビュー	ビジュアル テストの再生中に実行されるステップに応じてテスト対象のアプリケーションのスナップショットが表示されます。
プロパティ	ビジュアル テストのステップのプロパティが表示されます。
ストーリーボード	サムネイル イメージを使用してビジュアル テストのフローが表示されます。サムネイルは、ビジュアル テストのステップの論理グループを表します。

注:

Silk Test Workbench では、スナップショットは以下のような状況のときに取得されます。

- 記録中のすべての自動テスト ステップの前。

  注: SAP アプリケーションの場合、スナップショットはすべての自動テスト ステップの前ではなく、画面が変わったときに発生します。
- ビジュアル テストで Using ステップを実行した結果のスナップショット。
- 再生エラーが発生した場合。

画面プレビュー、**ストーリーボード**、および **プロパティ** ペインは **テスト ステップ** ペインと同期され、**テスト ステップ** ペインで選択したステップに関する情報が表示されます。 このように、**テスト ステップ** ペインでステップを選択し、他のペインでそのステップに関する情報を確認することで、ステップのすべての側面を簡単に把握できます。

ビジュアル ナビゲータ では、ビジュアル テストを表示するだけでなく、**画面プレビュー** ペインと **プロパティ** ペインを使用して既存のビジュアル テストを拡張または更新することもできます。 たとえば、ビジュアル テストを記録したあと、**プロパティ** ペインで、記録したプロパティのリテラル値を変数に置き換えることができます。 また、テスト対象アプリケーションに変更が発生したときにビジュアル テストをすばやく更新する場合は、**画面プレビュー** の **画面の更新** 機能を使用して、以前にキャプチャした画面を更新できます。

ビジュアル ナビゲータ では、ビジュアル テストに使用するのと同じペインを使用して、ビジュアル テストの再生結果も表示します。 結果の場合、これらのペインには機能が付加され、**結果** ウィンドウ内に表示されます。ウィンドウにはツールバー オプションと複数のタブが表示されます。各タブには結果のさまざまな内容が表示されます。 結果に固有の追加機能として、たとえば、**テスト ステップ** ペインに各ステップのステータス（合格または失敗）を表示する機能があります。 さらに、**画面プレビュー** では、記録中にキャプチャされた画面と再生中にキャプチャされた画面の違いを比較し、テスト アプリケーションにアクセスすることなく、既存のビジュアル テストを更新できます。

ビジュアル テストの記録：概要

サンプル Web アプリケーションで保険の見積もりを作成するための操作を行うと、Silk Test Workbench はそれらの操作をステップとして記録します。 ステップはビジュアル テストの基礎となります。 テストに必要な操作の記録を完了すると、ビジュアル ナビゲータに記録されたテストが表示されます。 ステップは、ビジュアル ナビゲータの[テスト ステップ] ペインに表示されます。

注: このチュートリアルで使用されているサンプル アプリケーションは、Internet Explorer に合わせて設計され、最適化されています。 チュートリアルのレッスンと同じユーザー体験を得るには、Micro Focus は Internet Explorer 以外のサポート対象ブラウザでチュートリアル サンプル アプリケーションを実行することをお勧めしません。

注: Web アプリケーションを記録または再生する前に、システムにインストールされているすべてのブラウザ アドオンを無効にします。 Internet Explorer でアドオンを無効にするには、**ツール** > **インターネット オプション** をクリックし、**プログラム** タブをクリックし、**アドオンの管理** をクリックし、アドオンを選択してから **無効にする** をクリックします。

サンプル Web アプリケーションの起動

このチュートリアルでは、Silk Test サンプル Web アプリケーションを使用します。 この Web アプリケーションは、デモ用に提供されています。

Internet Explorer で、Silk Test サンプル Web アプリケーションを使用します。 チュートリアルのレッスンと同じユーザー体験を得るには、Internet Explorer 以外のサポート対象ブラウザでサンプル Web アプリケーションを実行しないでください。

1. テストの速度と信頼性を高めるために DOM 関数を記録するには、以下のステップを実行します。
   a) **ツール** > **オプション** をクリックします。
   b) **オプション** メニュー ツリーの **記録** の隣にあるプラス記号（+）をクリックします。 **記録** オプションが右側のパネルに表示されます。
   c) **xBrowser** をクリックします。
   d) **ネイティブなユーザー入力を記録する** リスト ボックスから、**いいえ** を選択します。
   e) **OK** をクリックします。

   注: 通常、Web アプリケーションをテストする場合、DOM 関数ではなく、ユーザーの入力そのものを使用します。 Flash および Java アプレットなどのプラグインや、AJAX を使用するアプリケーションは、ネイティブなユーザー入力ではサポートされますが、高レベルの API 記録ではサポートされません。
2. ブラウザのすべてのアドオンを確実に無効にするには、以下のステップを実行します。
   Web アプリケーションの記録または再生を行う前に、ブラウザのアドオンをすべて無効にする必要があります。
   a) Internet Explorer で**ツール** > **インターネット オプション** を選択します。 **インターネット オプション** ダイアログ ボックスが開きます。
   b) **プログラム** タブをクリックし、**アドオンの管理** をクリックします。 **アドオンの管理** ダイアログ ボックスが開きます。
   c) アドオンのリストの **状態** 列で、各アドオンの状態が **無効** になっていることを確認します。
   状態 列に **有効** と表示されている場合は、アドオンを選択してから **無効にする** をクリックします。
   d) **閉じる** をクリックし、**OK** をクリックします。
3. サンプル アプリケーションにリモートでアクセスするには、*http://demo.borland.com/InsuranceWebExtJS* をクリックします。 サンプル アプリケーションの Web ページが開きます。

サンプル Web アプリケーションに対するビジュアル テストの記録

記録中は、記録を停止するまで、テスト アプリケーションでのすべての操作（Silk Test Workbench の操作は除く）が Silk Test Workbench によって記録されます。記録が完了したら、生成したビジュアル テストを変更して、ステップを追加したり、不要なステップを削除したりできます。

1. **ファイル** > **新規作成** を選択します。**資産の新規作成** ダイアログ ボックスが開きます。
2. 資産の種類のリストから **ビジュアル テスト** を選択します。
3. 記録をすぐに開始するために、**記録の開始** チェック ボックスをオンにします。
4. **OK** をクリックして、ビジュアル テストを資産として保存し、記録を開始します。**テストの構成** ダイアログ ボックスが開きます。
5. **Web** タブを選択します。
6. リストから **Internet Explorer** を選択して、**OK** をクリックします。Silk Test Workbench が最小化され、**記録中** ダイアログ ボックスが開きます。
7. Insurance Company Web サイトでは、次のステップのいずれかを行います：

記録中は、タスクバーの Silk Test Workbench アイコンが点滅します。**記録中** ダイアログ ボックスでは、現在操作中のオブジェクト、および最後に記録された操作を確認できます。

a) **Select a Service or login** リスト ボックスから **Auto Quote** を選択します。**Automobile Instant Quote** ページが開きます。

b) 郵便番号と電子メール アドレスを適切なテキスト ボックスに入力し、自動車タイプをクリックして、**Next** をクリックします。
たとえば、郵便番号に 92121、電子メール アドレスに jsmith@gmail.com をそれぞれ入力し、自動車タイプとして Car を指定します。

c) 年齢を指定し、性別と運転履歴タイプをクリックして、**Next** をクリックします。
たとえば、年齢に 42 を入力し、性別と運転履歴タイプに Male および Good をそれぞれ指定します。

d) 製造年、車種、モデルを指定し、財務情報タイプをクリックして、**Next** をクリックします。
たとえば、製造年に 2010 と入力し、車種とモデルに Lexus および RX400 をそれぞれ指定し、財務情報タイプとして Lease を指定します。
指定した情報の概要が現れます。

e) **Purchase** をクリックします。
Purchase A Quote ページが開きます。

f) ページ上部にある **Home** をクリックして、記録を開始したホーム ページに戻ります。

8. Alt+F10 を押すか、**記録中** または **モバイルの記録** ダイアログ ボックスで **記録の停止** をクリックするか、タスクバーの Silk Test Workbench アイコンをクリックして、記録を停止します。**記録完了** ダイアログ ボックスが開きます。**記録完了** ダイアログ ボックスで **次回からこのメッセージを表示しない** チェック ボックスがオンになっている場合、記録が停止したあとにこのダイアログ ボックスは表示されません。この場合、ビジュアル テストは **テスト ステップ** ペインに表示されます。

ビジュアル テストの保存と名前の指定

テストを記録すると、**記録完了** ダイアログ ボックスが開き、以下の操作を実行できます。

- 記録したビジュアル テストを再生する
- 記録した操作をビジュアル ナビゲータで確認する
- ビジュアル テストを保存し、記録したテストをビジュアル ナビゲータで確認する

ビジュアル テストをはじめて記録したばかりなので、テストを再生したり、記録した操作を確認したりする前にテストを保存し、名前を付けます。

1. **記録完了** ダイアログ ボックスで、**保存** をクリックします。

   ヒント: 名前を指定せずに資産を作成すると、Silk Test Workbench により「無題_」に連番の付いた一時的な名前が割り当てられます。

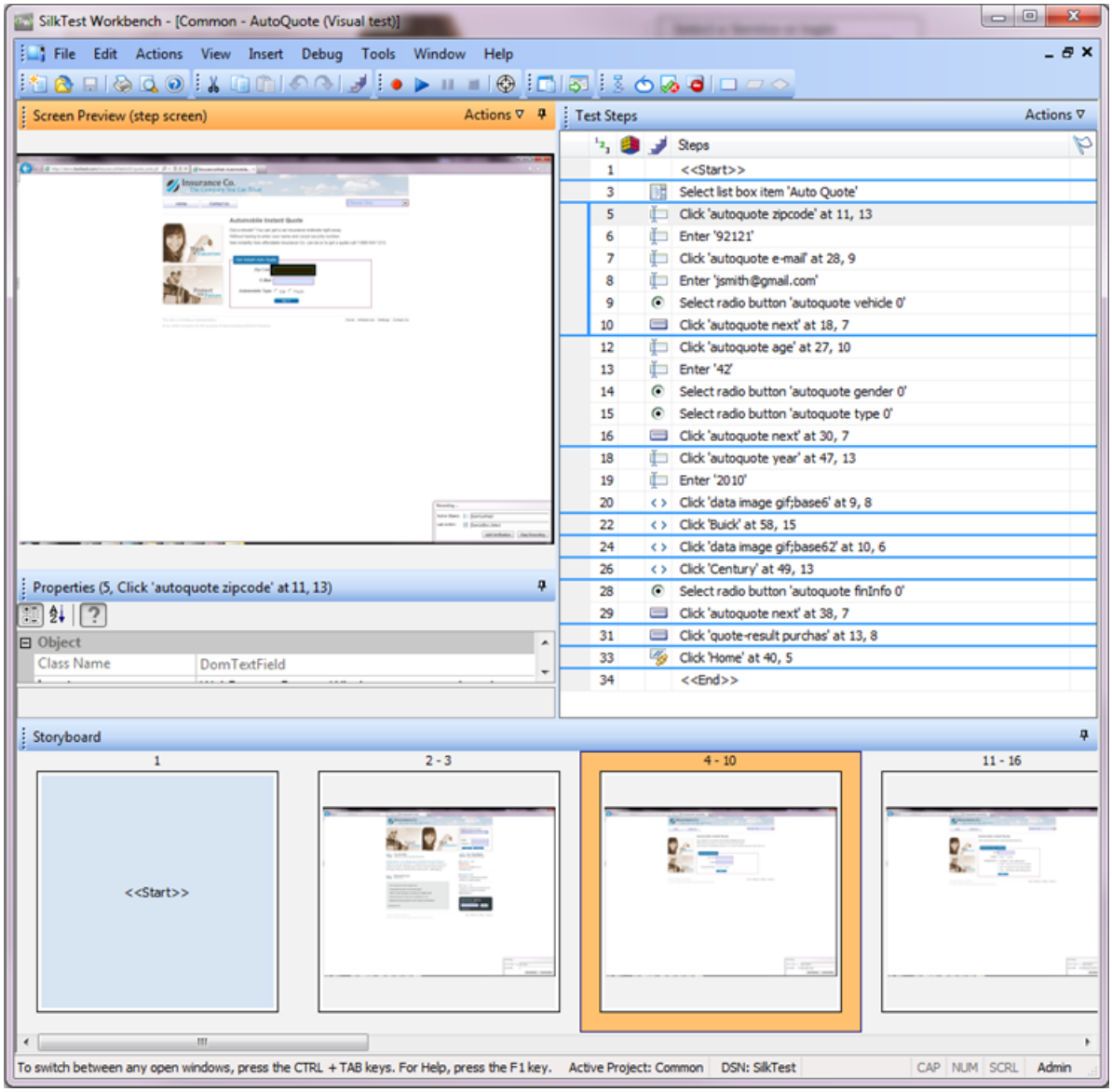
   テストにまだ名前を付けていないため、**名前を付けて保存** ダイアログ ボックスが開きます。
2. **名前** テキスト ボックスで、名前を AutoQuote に変更します。
3. **説明** テキスト ボックスに Visual test tutorial と入力します。
4. **OK** をクリックします。 ビジュアル ナビゲータにビジュアル テストが表示されます。

記録されたテスト ステップの確認

ビジュアル テストを記録し、保存すると、ビジュアル テストがビジュアル ナビゲータに表示されます。ビジュアル ナビゲータには 4 つのペインがあり、ビジュアル テストの包括的なビューが提供されます。

[テスト ステップ] ペインのステップは、Web アプリケーションで見積もりを完了するときにアクセスした画面と実行した操作を表します。

記録されたステップは、以下の図に示すステップと同じようになるはずです。



[テスト ステップ] ペインには以下の列が表示されます。

列	説明
（番号）*	ステップが記録され、再生される順番を表します。 デフォルトでは、ステップはこの順番で表示されます。 ステップにデバッグのためのブレークポイントが追加されている場合は、この列の隣の列にブレークポイントが表示されます。 デフォルトのビュー（ステップと画面）から画面のみのビューまたはステップのみのビューにビューを変更した場合、番号付けの方法では、記録の順序で間隔が示されるようにステップまたは画面は順番にスキップされます。
（ロジック）*	ステップにロジックが含まれている場合、ステップのロジックのタイプを表すアイコンを表示します。
（ステップ タイプ）*	ステップで実行される操作のタイプを表すアイコンを表示します。
ステップ	ステップで実行される操作について説明します。
（フラグ）*	**割り当てられたフラグ** のアイコンを表示します。

列	説明
	割り当てられたフラグ のアイコンは、関連付けられたステップが割り当て先のユーザーの **テスト ステップ** ペインおよびオプションで **開始画面** ペインに表示されるように、フラグが付けられていることを示します。 さらに、ポインタをアイコンの上に移動すると、フラグの説明、更新日、および割り当て日を示すツール ヒントを表示できます。
（ステップの説明）*	ユーザー定義のステップの説明を表示します。 この列はデフォルトのビューには表示されません。**テスト ステップ** ペインのタイトル バーから **操作** をクリックし、次に **表示** > **ステップの説明** をクリックすると表示できます。 ステップの説明を作成するには、ステップを選択します。 **プロパティ** ペインで、**ステップの説明** プロパティを更新します。 **ステップの説明** アイコンがこの列に表示され、ステップに説明があることが示されます。 ポインタをアイコンの上に移動して説明を含むツール ヒントを表示するか、ステップを選択して **プロパティ** ペインの説明を読みます。

ヒント: ビジュアル ナビゲータの他のペインは、**テスト ステップ** ペインと同期しています。 上の図では、記録したステップのうち、リスト ボックスから **Auto Quote** を選択するステップが **テスト ステップ** ペインで選択されています。 結果は以下のようになります。

- **画面プレビュー** には、**Auto Quote** を選択する前のアプリケーションの状態が示されます。
- **プロパティ** ペインには、選択されたステップのプロパティが示されます。
- ストーリーボードでは、**Auto Quote** リスト項目の選択に関連するステップのグループを表すサムネイルが強調表示されます。

ビジュアル テストのステップをスクロールし、さまざまなステップを選択すると、他のペインに更新した情報が表示されます。

記録されたビジュアル テストの再生

ビジュアル テストを記録し、保存したら、再生して、ビジュアル テストが適切に動作することを確認できます。

1. 次のいずれか 1 つのステップを行います：
   - **操作** > **再生** を選択します。
   - ツール バーで **再生** をクリックします。

   再生 ダイアログ ボックスが開きます。このダイアログ ボックスで、結果をどのように保存するかを指定できます。
2. **結果の説明** テキスト ボックスに Initial test results for the recorded test と入力します。
3. **OK** をクリックします。
4. 再生をサポートしている複数のブラウザーがマシンにインストールされている場合、**ブラウザーの選択** ダイアログ ボックスが開きます。ブラウザーを選択して、 **実行** をクリックします。

各結果は、一意のテスト実行番号によって識別されます。

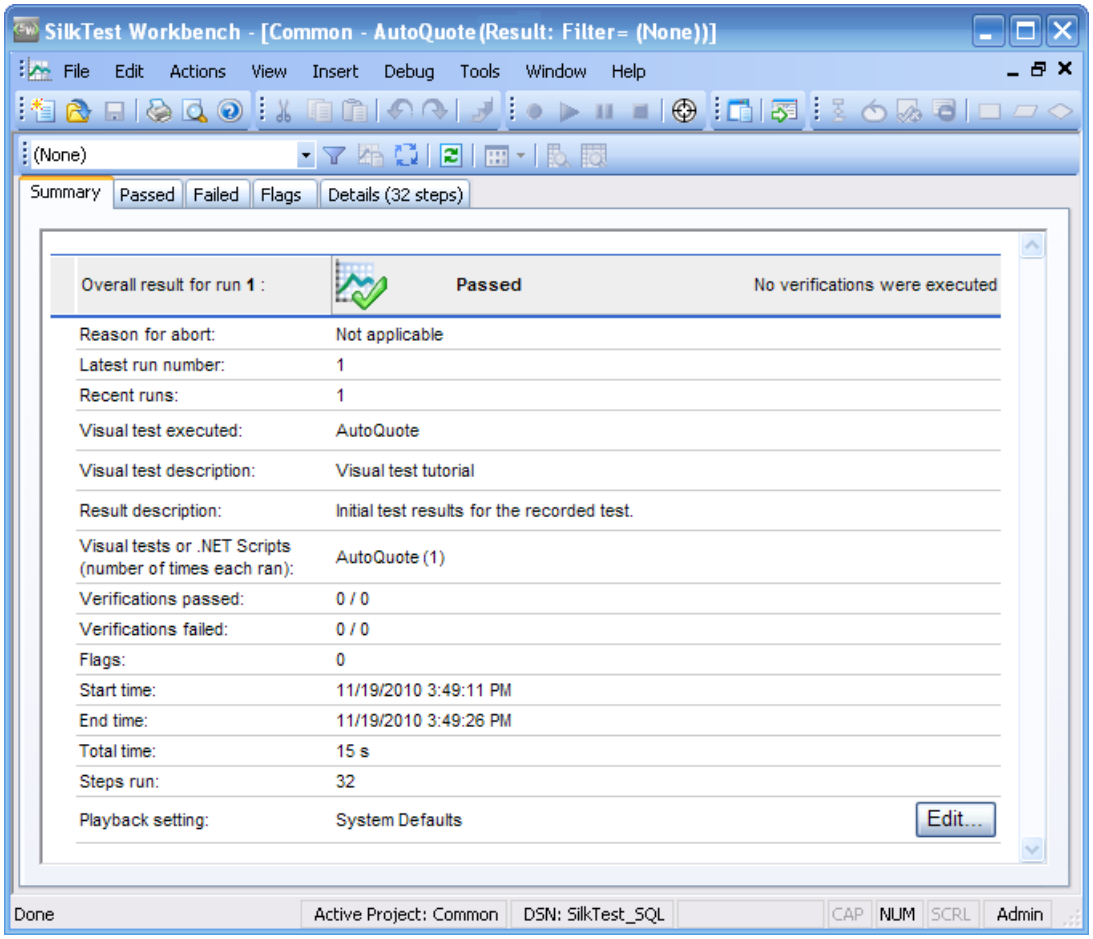
Silk Test Workbench が最小化され、ビジュアル テストが再生されます。再生中、サンプル アプリケーションアプリケーションに対して、ビジュアル テストの記録中に行った操作が画面上で再生されます。再生が正常に完了すると、**再生完了** ダイアログ ボックスが開きます。

結果の分析：概要

テストを再生すると、Silk Test Workbench によってテスト結果が生成されます。 テスト結果にはビジュアル テストの再生に関する情報が含まれています。 ビジュアル テストの名前、実行番号、各ステップが実行された日付と時刻、各ステップの合格／失敗のステータスなどの重要な情報が含まれます。

[結果] ウィンドウ

ビジュアル テストまたはスクリプトを再生したあと、再生結果を **結果** ウィンドウに表示することができます。 **結果** ウィンドウには [ビジュアル ナビゲータ] があり、テストの再生をあらゆる面からすばやく確認できます。 **結果** ウィンドウには、[ビジュアル ナビゲータ] 以外に以下の機能もあります。



[結果] ウィンドウのツールバー

結果 ウィンドウのツールバーには、結果に含まれる内容の表示と種類をカスタマイズするための以下のボタンとリスト ボックスがあります。

- **種類の選択によるフィルタ処理の結果**：すべての事前定義済みおよびユーザー定義の結果フィルタにすばやくアクセスできます。 リストからフィルタを選択します。
- **フィルタの管理**：**フィルタの管理** ダイアログ ボックスを開き、そこから結果フィルタを作成、編集、および適用できます。
- **条件**：**条件** ダイアログ ボックスが表示されます。このダイアログ ボックスでは、今後のすべての実行の成功を定義する条件として、合格した検証のパーセンテージを設定できます。 たとえば、合格条件を 90% に設定した場合は、ビジュアル テストまたはスクリプトにおける 10 件の検証のうち 9 件以上に合格しないと、再生の結果が合格になりません。 このオプションを設定すると、再生結果オプションの **結果合格条件（パーセンテージ）** が更新されます。 このパーセンテージは、今後のすべての結果に適用されます。
- **すべての実行を表示**：各実行結果の詳細を表示する **実行の詳細** ダイアログ ボックスを開きます。 このダイアログ ボックスで、以前の実行を開いたり削除したりすることができます。

- **更新**：現在の結果を更新します。
- **表示**：ビジュアル テストについて、**テスト ステップ** ペインに表示するステップの種類を設定します。このボタンの横にあるドロップダウン矢印をクリックして、ステップのみを表示するか、画面のステップのみを表示するか、またはその両方を表示するかを選択します。 **ステップの説明** 列を表示するように選択することもできます。 選択された表示は、**結果** ウィンドウの各タブ（**合格**、**失敗**、**フラグ**、および **詳細**）に適用されます。
- **基本表示**：標準的な **テスト ステップ** ペインの情報、および **結果** と **結果の詳細** の追加の列を表示します。 **要約** タブを表示している間は無効になります。
- **詳細表示**：各ステップの詳細情報を表示します。 **要約** タブを表示している間は無効になります。

[結果] ウィンドウのタブ

結果 ウィンドウには、結果の内容を特定の種類に分類する 5 つのタブがあります。

- **[要約] タブ**：以下の情報を示す全体的な概要レポートを表示します。
  - **総実行結果**：ビジュアル テストまたはスクリプトが正常に再生されて、結果合格条件（パーセンテージ）を満たしている場合は、[合格] と表示されます。結果合格条件（パーセンテージ）を満たしていない場合は、[失敗] と表示されます。ステップを正しく実行できなかった場合は、[再生エラー] と表示されます。
  - **中止の理由**：ビジュアル テストまたはスクリプトの再生が中止された理由を表示します。
  - **最新の実行番号**：最新の結果の実行番号を表示します。
  - **最近の実行**：最近の実行を表示します。 前の実行を表示するには、表示する実行をクリックします。 このフィールドに表示されていない以前の実行を開くには、ツールバーの **すべての実行を表示** をクリックして **実行の詳細** ダイアログ ボックスを開きます。ここで、前の実行を開いたり削除したりできます。
  - **実行されたビジュアル テスト／.NET スクリプト**：結果のビジュアル テストまたはスクリプトの名前を表示します。
  - **ビジュアル テスト／.NET スクリプトの説明**：結果のビジュアル テストまたはスクリプトの説明を表示します。
  - **結果の説明**：結果の説明を表示します。
  - **ビジュアル テスト** または **.NET スクリプト**：再生の一部として正しく実行されたビジュアル テストまたはスクリプトをすべてリストします。これには、Workbench.RunScript( ) メソッドを使用して実行された、挿入されたビジュアル テストまたはスクリプトも含まれます。 たとえば、ドライバ スクリプトでは、一度の再生により複数のスクリプトが実行されることがあります。
  - **合格した検証**：すべてのビジュアル テストまたはスクリプトで、正常に実行され合格した検証の合計数を表示します。 このフィールドの数字をクリックすると **合格** タブが開くため、合格したすべての検証を確認できます。
  - **失敗した検証**：すべてのビジュアル テストまたはスクリプトで、正常に実行されたが失敗した検証の合計数を表示します。 このフィールドの数字をクリックすると **失敗** タブが開くため、失敗したすべての検証を確認できます。
  - **フラグ**：ビジュアル テストについて、結果でフラグを付けられた検証ステップの総数です。 このフィールドの数値をクリックして、**フラグ** タブを開きます。 フラグは、スクリプトでは使用できません。
  - **開始時刻**：最初のビジュアル テストまたはスクリプトの再生が開始された時刻です。
  - **終了時刻**：最後のビジュアル テストまたはスクリプトの再生が終了した時刻です。
  - **合計時間**：ビジュアル テストまたはスクリプトを再生した合計時間です。
  - **実行したステップ数**：実行したステップまたはコード行の総数です。
  - **再生設定**：結果の作成に使用される再生オプションのグループである再生設定を表示します。 **編集** をクリックして再生オプションを表示し、再生設定を設定することができます。
- **[合格] タブ**：合格した検証をすべて表示します。
- **[失敗] タブ**：失敗した検証をすべて表示します。 再生エラーとなったステップはこのタブには表示されません。

- **[フラグ] タブ**：検証ロジックで作成された、フラグの付いたステップのみを表示します。 ビジュアル テストのフラグ付きのステップは、再生後はこのタブに表示されません。 フラグは、スクリプトでは使用できません。
- **[詳細] タブ**：ビジュアル テストの各ステップまたはスクリプトの各コード行に関する情報を表示します。 テスト ステップやコード行の名前、合格／失敗のステータス、説明、フラグのステータスなどの情報が表示されます。

ビジュアル ナビゲータのペイン

ビジュアル テストの場合は、**合格** タブ、**失敗** タブ、**フラグ** タブ、**詳細** タブのすべてに [ビジュアル ナビゲータ] の以下の 4 つのペインが表示されます。

- テスト ステップ
- 画面プレビュー
- プロパティ
- ストーリーボード

プロパティ ペインには **再生前のビジュアル テストのステップ プロパティを表示/非表示** ツールバー ボタンがあり、選択したステップのビジュアル テストのプロパティを表示または非表示にすることができます。

スクリプトでは、**結果** ウィンドウには **テスト ステップ** ペインおよび **プロパティ** ペインのみ表示されます。 **テスト ステップ** ペインには、各テスト ステップの再生ステータスと結果の詳細情報を示す列があります。

結果データの表示をカスタマイズするには、**全般** オプションから **デフォルトの結果表示** を変更します。 また、再生結果オプションを変更し、目的のデータを表示するフィルタを作成できます。

[結果] ウィンドウのタブの使用

テスト結果情報をすばやく参照するために、**結果** ウィンドウには、結果情報を種類別に分けて表示するフィルタの役目を果たす 5 つのタブがあります。

1. **再生完了** ダイアログ ボックスで **結果へ移動** をクリックします。

   結果 ウィンドウが開き、デフォルトで **要約** タブが表示されます。 **要約** タブには、再生が成功したかどうか、最新の実行番号、合格または失敗となったチェックまたは検証の数、テストの開始時刻と終了時刻、テストの実行結果に関するその他の基本情報などを含むテスト実行の概要が示されます。
2. **詳細** タブをクリックします。

   詳細 タブでは、ビジュアル ナビゲータの以下の 4 つのペインを使用して各ステップの結果が示されます。

テスト ステップ	ビジュアル テストの各ステップの再生結果に関する情報がリストされます。
画面プレビュー	再生中にキャプチャされた Web アプリケーションの画面が表示されます。
プロパティ	ステップのプロパティが表示されます。
ストーリーボード	結果のアウトラインと概要がグラフィカルに示されます。

   注: 合格、**失敗**、**フラグ** タブでもビジュアル ナビゲータを使用して結果情報が示されます。 違いは、**詳細** タブではすべてのステップが示されるのに対し、これらのタブでは特定の種類のステップが示されるという点だけです。
3. 任意のステップを選択します。

   ステップを選択すると、Silk Test Workbench により他のペインの情報がそのステップの情報に更新されます。 **画面プレビュー** では、再生中にキャプチャされた画面が、ビジュアル テストを最初に記録したときにキャプチャされた画面と比較されます。 **プロパティ** ペインには、選択したステップのプロパティが示されます。 **ストーリーボード** では、ステップが属すグループが強調表示されます。

ヒント: ステップの名前全体を表示するには、**ステップ** 列を広げるか、ステップの上にマウス ポインタを置いて、名前全体を示すツール ヒントを表示する必要がある場合があります。

[結果] ウィンドウのツールバーの使用

結果 ウィンドウのツールバーには、結果の内容の表示と種類をカスタマイズするためのいくつかのオプションがあります。

1. 結果のツールバーの **詳細表示** をクリックします。 Silk Test Workbench により、**テスト ステップ** ペインに追加の列が表示され、**プロパティ** ペインに追加のプロパティが表示されます。
2. **テスト ステップ** ペインで、右にスクロールして追加の列を表示します。
   これらの列には、ステップの実行に要したミリ秒単位の時間、テストを実行した人のユーザー名など、各ステップに関する特定の情報が示されます。
3. **プロパティ** ペインの [結果] プロパティ カテゴリには、**テスト ステップ** ペインの各列に対応するプロパティが示されます。 下にスクロールして、リスト全体を表示します。
4. **表示** ドロップダウン矢印をクリックし、**ステップのみ** を選択します。 Silk Test Workbench によりすべての画面が除外されるので、ステップだけをすばやく参照できます。
5. **表示** ドロップダウン矢印をクリックし、**ステップと画面** を選択します。 Silk Test Workbench により結果の画面とステップがすべて表示されます。

   結果 ウィンドウのツールバーには、この他にも結果の内容の表示と種類をカスタマイズするためのオプションがあります。

[プロパティ] ペインの使用

プロパティ ペインには、ステップの基本特性を示すプロパティが表示されます。ステップの名前、実行ステータス、再生の詳細などの情報が示されます。

1. ステップ「'42' を入力します」をクリックします。
   このステップは、**Age** テキスト ボックスに値「42」を入力する操作を実行します。
   プロパティ ペインが更新され、選択したステップのプロパティが表示されます。
2. **カテゴリ** アイコンまだ選択されていない場合は、これをクリックします。
   プロパティが以下のメイン カテゴリにグループ分けされます。

   結果 — **テスト ステップ** ペインの列に対応するプロパティが含まれます。 ステップの名前、ステップ実行の日付と時刻、ステップ実行に要した時間などのプロパティです。

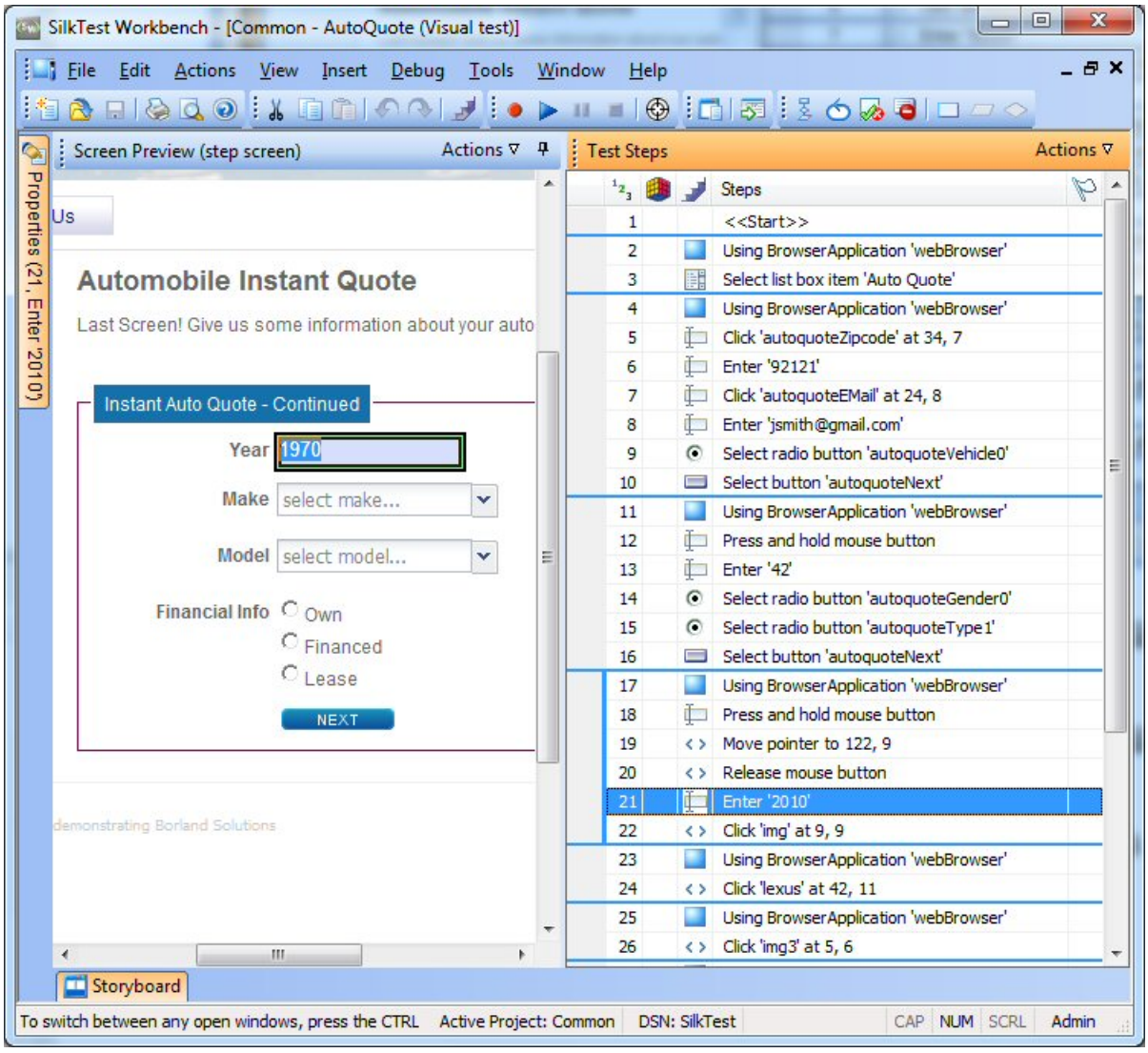
   拡張プロパティ — Silk Test Workbench が、入力 操作ステップによって入力されたテキストのコントロールまたは値を識別するために使用するロケーター名など、追加の再生の詳細が含まれます。 拡張プロパティは、操作ステップで変数や式が使用されたときにその内容を表示するのに便利です。 たとえば、入力 ステップで変数 *textVar* が値として使用された場合、[拡張プロパティ] カテゴリに *textVar* の内容が表示されます。
3. **拡張プロパティ** ノードを展開します。 Text プロパティおよびロケーター プロパティがその値と共に示されます。 Text プロパティの値は '42' です。これは、このステップで **Age** テキスト ボックスに入力された値です。
4. **再生前のビジュアル テストのステップ プロパティを表示/非表示** アイコンをクリックします。 [ビジュアル テストの詳細]ノードが表示され、ビジュアル テストに存在するステップの元のプロパティがすべて表示されます。

画面プレビューの使用

画面プレビュー には、**テスト ステップ** ペインに表示された、コントロールと対話する各ステップについてキャプチャされたテスト アプリケーションのイメージが表示されます。 キャプチャされたイメージとして、デスクトップ全体、アプリケーション ウィンドウ、またはアクティブなウィンドウのみを示すことが

できます。 キャプチャされたイメージは、関連するステップが実行される前のアプリケーションの状態を示します。

1. **テスト ステップ** ペインで、車の製造年を含むステップを選択します。 「2010」と入力します。 **画面プレビュー** に、再生中にキャプチャされた画面とビジュアル テストの記録中にキャプチャされた画面が並んで表示されます。 このステップのコントロールであるブラウザ ウィンドウが黒のボックスで強調表示されます。



2. **画面プレビュー** で、**操作** > **相違点の表示** > **オフ** をクリックします。 ビジュアル テストの画面が消え、再生の画面が開きます。
3. **操作** > **ズーム** > **75%** をクリックしたあと、スクロールバーを使用して製造年がはっきりと表示されるところまで移動します。
4. ビジュアル テストに切り替えます。 **テスト ステップ** ペインで、**操作** > **ビジュアル テストのウィンドウ** をクリックします。

ビジュアル テストの拡張：概要

ビジュアル テストの拡張には、既存のビジュアル テストが新しいバージョンのテスト アプリケーションで使用できるようにテストを更新することなどが含まれます。 たとえば、テスト アプリケーションのさまざまな状態を処理し、検証するために、テスト ロジックを挿入することができます。 さらに、ビジュアル テ

ストの内容をわかりやすくしたり、重要な点について自分自身や他のユーザーに注意を促すために、フラグ、メッセージ ボックス、またはステップの説明を挿入することができます。

これらは、Silk Test Workbench で既存のビジュアル テストを拡張し、よりパワフルで堅牢かつ柔軟なテストを作成する、ほんの数例です。

画面プレビューからの更新

Silk Test Workbench はビジュアル テストを記録するとき、テスト アプリケーションの各画面の関連するコントロールだけでなく、画面もキャプチャします。 Silk Test Workbench により、**画面プレビュー** にはキャプチャされた各画面が表示され、ビジュアル テストのステップによって識別されるコントロールが強調表示されます。 **画面プレビュー** で、キャプチャされた画面の別のコントロールを識別することにより、テスト アプリケーションにアクセスすることなくステップを更新できます。

このレッスンでは、Silk Test Workbench の **画面プレビューからのコントロールの挿入** 機能を使用して、**画面プレビュー** のキャプチャされた画面から別のボタンを選択します。

1. AutoQuote ビジュアル テストの **テスト ステップ** ペインで、<<終了>> ステップの前にあるステップを選択します。

   ステップ テキストは、次のようになります：「リンク 'Home' を選択します」

   画面プレビュー に **Purchase A Quote** ページが表示され、**Home** ボタンが強調表示されます。
2. **プロパティ** ペインで、**ロケーター** テキスト ボックスをクリックします。 ロケーターの値領域に、ロケーター選択ボタンが表示されます。
3. **画面プレビュー** で **操作** > **ズーム** > **75%** をクリックし、スクロールバーを使用して **Contact Us** ボタンがはっきり表示されるところまで移動します。
4. **プロパティ** ペインで、**画面プレビューから識別** をクリックします。 ポインタが **画面プレビュー** に移動します。
5. ポインタを **Contact Us** ボタンの上に移動し、**Contact Us** ボタンをクリックします。

   ボタンをクリックすると、ボタンの新しいロケーター名を表示するように **プロパティ** ペインの **ロケーター** プロパティが変わり、**テスト ステップ** ペインのステップ テキストが次のように変わります：「リンク 'contactUs' を選択します」
6. **保存** をクリックします。

次にこのビジュアル テストを再生すると、Silk Test Workbench により **Home** ボタンではなく **Contact Us** ボタンがクリックされます。

検証の挿入

検証とは、ユーザー定義の条件を評価し、合格／失敗のメッセージや、オプションとしてフラグをビジュアル テストの再生結果に送信するテスト ロジックです。

このレッスンでは、検証を挿入して、見積もりで正しい車両モデルが使用されていることを確認します。

1. **テスト ステップ** ペインで **操作** > **表示** > **ステップと画面** をクリックして、ステップと画面の両方が表示されていることを確認します。
2. 次のステップを選択します：「(91,11) の場所で 'rx400' をクリックします」
3. **画面プレビュー** で **操作** > **ズーム** > **75%** をクリックし、スクロールバーを使用して **Model** がはっきり表示されるところまで移動します。
4. 次のいずれか 1 つのステップを行います：

   - ツールバーの **検証タイプのロジックの作成** をクリックします。
   - **挿入** > **テスト ロジック** > **検証** を選択します。

   テスト ロジック デザイナ ウィザードの **ようこそ** ページが開きます。

5. **次へ** をクリックします。 **ロジックの種類の選択** ページが開きます。
6. **コントロールのプロパティ** をクリックし、**次へ** をクリックします。 **プロパティに基づく条件の定義** ページが開きます。
7. **画面プレビューから識別** をクリックします。 ポインタが **画面プレビュー** に移動します。
8. **Model** コンボ ボックスを選択します。 **プロパティに基づく条件の定義** ページに、選択したコントロールの名前が、コントロールのプロパティと共に表示されます。 コントロール名は、以下のように表示されます。

   ```
   webBrowser.browserWindow.modelCombo
   ```
9. **プロパティの選択** グリッドで、**テキスト** にスクロールして選択します。
10. **条件の選択** が **等しい** に設定されていることを確認します。
11. **プロパティの選択** から Text プロパティーを選択して、**期待値** テキスト ボックスに **RX400** が入力されていることを確認します。
12. **次へ** をクリックします。 **検証の構築** ページが開きます。 このページから、再生結果に送信する合格／失敗メッセージを定義できます。 ページの上部に Silk Test Workbench が検証する条件が以下のように表示されます。

    ```
    If "webBrowser.browserWindow.modelCombo"."Text" 等しい "RX400"
    ```

    この条件は検証のロジックを定義しています。 条件では、モデル タイプと選択されたタイプが比較されます。
13. デフォルトの合格の説明を The model type is correct に置き換え、デフォルトの失敗の説明を The model type is NOT correct に置き換えます。
14. **次へ** をクリックします。 **要約** ページが開きます。

    検証ステップは選択したステップのあとに挿入されます。 検証ステップのテキストは以下のようになります。「

    ```
    "webBrowser.browserWindow.modelCombo"."Text" 等しい "RX400"
    ```

    」を検証します。
15. 検証の内容を確認したら、**完了** をクリックします。

サンプル アプリケーションのプロパティの値を検証するテスト ロジックを挿入することで、記録したビジュアル テストを拡張できました。

アプリケーション データを保存するローカル変数の作成

変数を使用すると、テストの別の場所や別のビジュアル テストまたはスクリプトで使用するデータ値を保存できるため、ビジュアル テストを拡張できます。 データは他の種類のファイルに出力することもできます。

このレッスンでは、チュートリアルのあとのレッスンで使用できるように、電子メール アドレスを表示するコントロールのテキストを保存します。 このためには、まずテキストを保存するローカル変数を作成する必要があります。

1. **テスト ステップ** ペインで、**操作** > **挿入** > **変数** > **ローカルの追加** をクリックします。 **ローカル変数の追加** ダイアログ ボックスが開きます。
2. **変数名** テキスト ボックスに strEmailAddress と入力します。
3. **型** リストから **テキスト** を選択します。

   このレッスンでは、**初期値** テキスト ボックスを空のままにします。これは、後続のレッスンで変数に値を保存するためです。

   テキスト 型では、変数値がテキスト、つまり文字列データ型として保存されます。
4. **OK** をクリックします。 ビジュアル テストに対して新しい変数が保存されます。 変数を作成したあとは、<<開始>> ステップからそれを参照し、定義を編集することができます。

5. *strEmailAddress* 変数を表示するには、**テスト ステップ** ペインで <<開始>> ステップを選択します。

   ヒント: <<開始>> ステップは、すべてのビジュアル テストで常に最初のステップになります。

   ステップのプロパティが **プロパティ** ペインに表示されます。 <<開始>> ステップを選択している場合、**変数** カテゴリに *strEmailAddress* が **テキスト** 変数として表示されます。

各変数型の値領域は、現在定義されている各変数型の数を示します。 このレッスンではテキスト変数を 1 つ作成したので、その種類の値領域には 1 項目がこのテストに関連付けられていることが示されています。

見積もりの電子メール アドレスを保存するローカル変数を作成したので、アプリケーションの電子メール アドレスを変数に保存します。

ローカル変数へのアプリケーション データの格納

サンプル アプリケーションの **Get Instant Auto Quote page** ページには、一意の電子メール アドレスが表示されます。 このページの電子メール アドレスを含むテキストは、このページのコントロールのプロパティ値です。

このレッスンでは、前のレッスンで作成したローカル変数 *strEmailAddress* にこのテキストを保存します。

1. **テスト ステップ** ペインで、電子メール アドレスの値を表示するステップを選択します。
   ステップ テキストは、次のようになります：「'jsmith@gmail.com' を入力します」

   ステップを選択すると、**Get Instant Auto Quote page** ページのキャプチャされた画面が **画面プレビュー** に表示されます。
2. **テスト ステップ** ペインで、**操作** > **挿入** > **コントロールのプロパティ** をクリックします。
   これにより、ビジュアル テストの選択したステップの直後にステップが挿入されます。 ステップ テキストは、次のようになります：「コントロールの '' プロパティを取得します」

   このステップのプロパティを編集して、**Get Instant Auto Quote** ページの電子メール アドレスのテキストをローカル変数に保存します。 使用するコントロール、そのコントロールのプロパティ、およびプロパティの値を保存する変数を指定するように、このステップのプロパティを編集します。 変数は、前のレッスンで作成した *strEmailAddress* です。
3. **画面プレビュー** で **操作** > **ズーム** > **50%** をクリックし、スクロールバーを使用して電子メール アドレスのテキストがはっきり表示されるところまで移動します。
4. **プロパティ** ペインで、**ロケーター** テキスト ボックスをクリックします。 ロケーターの値領域に、ロケーター選択ボタンが表示されます。
5. **画面プレビューから識別** をクリックします。 ポインタが **画面プレビュー** に移動します。
6. ポインタをページの **E-Mail** テキストの上に移動します。
   強調表示のボックスがページのテキストを囲んでいることを確認し、強調表示された領域をクリックしてコントロールを識別します。

   Silk Test Workbench により、**プロパティ** ペインの **ロケーター** 情報が次の値に更新されます：webBrowser.browserWindow.autoquoteEMail
7. **プロパティ** ペインで **プロパティ** テキスト ボックスをクリックし、その値リストから **Text** を選択します。
8. **(ローカル変数の選択...)** をクリックし、リストから **strEmailAddress** を選択します。

   これで、コントロール、必要な値を含むコントロールのプロパティ、およびプロパティの値を保存する変数の指定が終了したので、ステップのテキストは次のようになるはずです：「コントロールの 'Text' プロパティを変数 'strEmailAddress' に格納します」

   プロパティ ペインとステップ テキストの両方の **ローカル変数名** の値が *strEmailAddress* に変わります。

テストでプロパティ値が正しくキャプチャされ、保存されることを確認するために、テストを再生して結果を確認します。

拡張したビジュアル テストの再生と分析

記録したテストにいくつかの拡張を行ったので、ビジュアル テストを再生し、結果を分析してみます。

1. 次のいずれか 1 つのステップを行います：
   - **操作** > **再生** を選択します。
   - ツール バーで **再生** をクリックします。

   再生 ダイアログ ボックスが開きます。
2. **結果の説明** テキスト ボックスに、Enhanced test results for the recorded visual test と入力します。
3. **OK** をクリックします。
4. 再生をサポートしている複数のブラウザーがマシンにインストールされている場合、**ブラウザーの選択** ダイアログ ボックスが開きます。ブラウザーを選択して、 **実行** をクリックします。 Silk Test Workbench によって、拡張したテストが再生されます。
5. **再生完了** ダイアログ ボックスで **結果へ移動** をクリックします。

   結果 ウィンドウが開き、デフォルトで **要約** タブが表示されます。

   要約 タブに、ビジュアル テストに合格した（エラーなしで正常に再生されたことを意味します）、または失敗した検証が示されます。
6. **合格（1）** タブをクリックします。

   かっこ内の数は、合格した検証の合計数を示します。**テスト ステップ** ペインには検証ステップが表示され、**結果の詳細** 列には検証の合格の説明が表示されます。
7. **詳細** タブをクリックして、各ステップの結果を表示します。
8. **結果** ウィンドウの **テスト ステップ** ペインで、変数に電子メール アドレスを保存した結果を示すステップまでスクロールします。ステップ テキストは、次のようになります：「コントロールの 'Text' プロパティを変数 'strEmailAddress' に格納します」
9. *strEmailAddress* 変数の内容が **結果の詳細** 列に表示されます。**Text** プロパティの値全体を表示するには、カーソルをそのステップの **結果の詳細** 列の上に置きます。ツールヒントにプロパティの内容全体が表示されます。

お疲れ様でした。これで、サンプル アプリケーションを確実にテストするビジュアル テストを作成できました。次のレッスンでは、ビジュアル テストを他のビジュアル テストですばやく簡単に実行する方法など、より高度なテストの概念と機能について学習します。

ビジュアル テスト内でのビジュアル テストの実行：概要

このチュートリアルでは、Web アプリケーションから自動車保険の見積もりを取得するために必要な各操作を実行する単一のビジュアル テストを作成しました。 単一のビジュアル テストは、単純なアプリケーションに対して基本的なテスト ケースを実行する場合に便利です。 ただし、ほとんどのソフトウェア テストでは、アプリケーションのすべての側面をテストするなど、より厳格なアプローチが必要となります。 さらに、テスト アプリケーションが変化したときに既存のビジュアル テストをすばやく更新できることが必要です。

これらのテストの課題を解決する効率的な手段を提供するために、Silk Test Workbench はモジュール式テストをサポートしており、特定のテスト ソリューションの共通の操作を単一のテストにまとめ、同じ操作セットを必要とする他のビジュアル テストでビジュアル テストを再利用できるようにしています。

モジュール式テスト

ビジュアル テストやスクリプトなどの Silk Test Workbench 資産を作成してアプリケーションのテスト ソリューションを構築する前に、テスト戦略を立てることをお勧めします。

1 つのビジュアル テストまたはスクリプトに特定のテスト ソリューションのすべての部分を含める必要はありません。また、通常、そうすることは有益ではありません。

通常、最も効率のよいテスト方法は、モジュール式アプローチを採用したものです。 アプリケーション テストを一連のトランザクション単位として考えます。

たとえば、オンライン発注システムのテストには、以下のようなトランザクション単位が含まれるかもしれません。

- オンライン システムへのログオン
- 顧客プロファイルの作成
- 注文の発注
- オンライン システムのログオフ

1 つのテストでこれらの単位をすべて処理し、このテストを使用するシナリオが 10 通りある場合、それらのシナリオを処理するために 10 個の別々のテストを記録する必要があります。 アプリケーションに何らかの変更があった場合、たとえば、ログオン ウィンドウにフィールドが 1 つ追加された場合、新しいフィールドへのデータ入力を処理するために 10 個の別々のテストに変更が必要になります。

これらのトランザクション単位をすべてテストするビジュアル テストまたはスクリプトを 1 つ作成して、それをシナリオごとに 10 個作成するよりも、これらのトランザクション単位を 1 つずつ処理する別個のテストをテスト「モジュール」として作成する方が有益です。 トランザクション単位ごとに別々のテストを作成し、それをテスト シナリオごとに再利用すれば、ログオン トランザクション単位を処理するテストだけを変更すればよいことになります。

モジュール式テストの基本が理解できたので、2 つめのテストを作成し、前のレッスンで作成したテストに追加してみます。

2 つめのビジュアル テストの記録

このセクションでは、2 つめのビジュアル テストを記録し、ビジュアル テスト資産の別の作成方法を学習します。

1. **ファイル** > **新規作成** を選択します。**資産の新規作成** ダイアログ ボックスが開きます。
2. [資産の種類] リストから **ビジュアル テスト** を選択し、**資産名** テキスト ボックスにビジュアル テストの名前を入力します。
   このチュートリアルでは、名前として AddAccount と入力します。
3. 記録をすぐに開始するために、**記録の開始** チェック ボックスをオンにします。
4. **OK** をクリックして、ビジュアル テストを資産として保存し、記録を開始します。**テストの構成** ダイアログ ボックスが開きます。
5. **Web** タブを選択します。
6. リストから **Internet Explorer** を選択して、**OK** をクリックします。Silk Test Workbench が最小化され、**記録中** ダイアログ ボックスが開きます。
7. サンプル アプリケーションの **Home** ページで、**Login** セクションの **Sign Up** をクリックします。**Create A New Account** ページが開きます。
8. 適切なフィールドに、以下の情報を入力します。
   次のフィールドに移動するには Tab キーを押します。

フィールド名	値
名 (First Name)	Pat

フィールド名	値
姓 (Last Name)	Smith
生年月日 (Birthday)	12/02/1990 **注:** カレンダー コントロールの月と年の隣にある下矢印をクリックして月と年を変更し、カレンダーで 12 を選択します。
電子メール アドレス (E-mail Address)	smith@test.com
住所 (Mailing Address)	1212 Test Way
市 (City)	San Diego
州 (State)	CA
郵便番号 (Postal Code)	92121
パスワード (Password)	test

9. **Sign Up** をクリックします。
10. **Continue** をクリックします。連絡先情報が表示されます。
11. ページ上部にある **Home** をクリックして、記録を開始したホーム ページに戻ります。
12. **Log Out** をクリックします。
13. Alt+F10 を押して記録を完了します。**記録完了** ダイアログ ボックスが開きます。
14. **保存** をクリックします。 ビジュアル ナビゲータにビジュアル テストが表示されます。

別のビジュアル テストへのビジュアル テストの挿入

このセクションでは、ユーザー アカウントを追加する 2 つめのビジュアル テストを、元のビジュアル テストの自動見積もりの要求を行うステップの前に挿入します。

ビジュアル テスト内でビジュアル テストを実行すると、複数のビジュアル テストで効率的に同じ基本ステップをテストできます。

ヒント: ビジュアル テストを別のビジュアル テストに挿入する際、テスト アプリケーションが正しい初期再生状態になっていることを確認することが重要です。

1. **開始画面** の **最新** リストで、AutoQuote をクリックして開きます。
   AutoQuote はこのチュートリアルで最初に作成したビジュアル テストです。
2. **テスト ステップ** ペインで <<開始>> ステップを選択します。
3. **操作** > **挿入** > **ビジュアル テスト** をクリックします。 **ビジュアル テストの参照** ダイアログ ボックスが開きます。
4. **資産の選択** リストから **AddAccount** という名前のビジュアル テストを選択し、**OK** をクリックします。

   Silk Test Workbench によって、選択したステップの前にステップが挿入されます。 挿入されたステップによって、選択したビジュアル テストが呼び出されます。 このステップ テキストは、以下のとおりです。

   ビジュアル テスト 'AddAccount' の再生

   ヒント: 再生中に上記のステップが実行されると、挿入されたビジュアル テストが再生される前に、元のビジュアル テストが最後まで再生されます。

次のレッスンでは、このモジュール式ビジュアル テストを再生し、再生エラーに対処する方法を学習します。

再生エラーへの対応：概要

再生中のエラーは、テスト アプリケーションの変更、ビジュアル テストの不適切なステップ フローなど、さまざまな要因により発生します。 このようなエラーをデバッグ機能を使用してすばやく診断し、修正することにより、テストの保守を最小限にし、チームによるテスト作業の効率を上げることができます。

まず、前のレッスンで作成したモジュール式テストを再生します。

モジュール式テストの再生

前のレッスンでは、ビジュアル テスト AddAccount を別のビジュアル テスト AutoQuote に挿入することにより、モジュール式テストを作成しました。

このセクションでは、このモジュール式テストを再生し、再生中のエラーを確認します。

1. AutoQuote ビジュアル テストが開いている状態で、ツールバーの **再生** をクリックします。**再生** ダイアログ ボックスが開きます。
2. **結果の説明** テキスト ボックスに Responding to errors in a modular test と入力します。
3. **OK** をクリックします。
4. 再生をサポートしている複数のブラウザーがマシンにインストールされている場合、**ブラウザーの選択** ダイアログ ボックスが開きます。ブラウザーを選択して、 **実行** をクリックします。

   再生中に、**Create A New Account** ページでテストが停止し、エラー メッセージが表示されます。

   データベースでは、各顧客レコードに対して一意の電子メール アドレスが必要なため、このエラーが発生します。AddAccount ビジュアル テストの記録中にすでに電子メール アドレスを入力しているため、この電子メール アドレスはデータベースにすでに存在しており、テストが失敗します。

再生エラーを確認したので、テストのデバッグを開始します。

エラーのデバッグ

Silk Test Workbench で再生エラーが発生すると、Silk Test Workbench の **再生エラー** ダイアログ ボックスが表示され、デバッグ モードを開始するオプションが提供されます。 デバッグ モードでは再生が一時停止され、Silk Test Workbench のデバッグ機能を使用して再生エラーを診断し、修正することができます。

このセクションでは、前のセクションでモジュール式テストを再生しているときに発生したエラーをデバッグしてみます。

1. **再生エラー** ダイアログ ボックスで、**デバッグ** をクリックします。

   デバッグ モードでは再生が一時停止されます。 そのため、エラーを発生させたステップのプロパティを編集したり、ステップを削除または無効にしたり、他のビジュアル テストからステップをコピー アンド ペーストしたりしてエラーを修正してから、再生を再開できます。

   Silk Test Workbench はデバッグ モードに入り、AddAccount ビジュアル テストを表示します。エラーを発生させたステップが黄色で強調表示されます。
2. **編集** > **有効化/無効化** を選択してステップ「'signup continue' をクリックします」を無効化します。

   このステップを無効にすると、次回テストを再生するときにエラーを回避できます。

   ステップのテキストがグレーの斜体になります。これはステップが無効にされたことを示します。
3. ツール バーで **再生** をクリックします。 **再生エラー** ダイアログ ボックスが表示されます。
4. **デバッグ** をクリックします。
5. **編集** > **有効化/無効化** を選択してステップ「'logout-form logout' をクリックします」を無効化します。

テストでは Home ページで **Log Out** ボタンを探しますが、Sign Up Continue ステップが無効になっておりボタンがページに表示されないために、このエラーが発生します。 このエラーは、チュートリアルのあとの方で修正します。

6. メニューから **デバッグ** を選択します。

   メニューの上部に表示される以下のステップ コマンドを使って、ステップ実行を制御できます。

   ステップ イン（F8）
   1 つずつステップを再生します。**ステップ イン** は各ステップのトレースに有効であり、また再生しているビジュアル テストに挿入された別のビジュアル テストにもステップ インします。挿入された各ビジュアル テストについても、一度に 1 ステップずつ実行されます。

   ステップ イン はテストの詳細分析に有効であり、変数の使用やターゲット アプリケーションとの対話に対する各ステップの影響を確認することができます。

   ステップ オーバー（Shift +F8）
   別のビジュアル テストに挿入されたビジュアル テスト全体またはスクリプト全体を、1 つのステップとして実行します。**ステップ オーバー** は、デバッグ モードの再生中に、ビジュアル テストまたはスクリプトを 1 ステップとして再生する場合に使用します。これにより、挿入されたビジュアル テスト全体またはスクリプト全体が再生されます。挿入されたビジュアル テスト全体またはスクリプト全体が再生されると、元のビジュアル テスト内の次のステップで再生が停止し、デバッグ モードになります。

   別のビジュアル テストまたはスクリプトを再生するステップ以外のステップで **ステップ オーバー** を使用すると、**ステップ イン** を使用した場合と同じになります。次のステップのみが実行されたあと、再生は停止し、再びデバッグ モードになります。

   ステップ アウト (Ctrl + Shift + F8)
   別のビジュアル テストから再生しているビジュアル テストまたはスクリプト内の残りのステップをすべて実行したあと、元のビジュアル テスト内の次のステップで再生が一時停止します。

   ステップ アウト は、別のビジュアル テストに挿入されたビジュアル テスト内のステップまたはコード行で再生が停止されているときに、残りのビジュアル テストを再生して元のビジュアル テストに戻る場合に使用します。挿入されたビジュアル テストまたはスクリプトの残りの部分の再生が実行されたら、元のビジュアル テスト内の次の行で再生が停止し、再びデバッグ モードになります。

   カーソルまで実行 (Ctrl + F8)
   再生を一時停止するステップを選択できます。これにより、ビジュアル テストまたはスクリプト内の選択したセクションを「ステップ オーバー」することができます。

   カーソルまで実行 は、ビジュアル テストまたはスクリプトを再生して、実行時エラーが発生したポイントの直前で停止する場合に使用します。これにより、ブレークポイントを挿入することなく特定の行またはステートメントで再生を停止できます。再生が停止したら、別のデバッグ オプションのいずれかを使用して再生を継続できます。

   カーソルから実行
   現在選択されているテスト ステップからビジュアル テストを再生します。

7. **ステップ アウト** を選択します。

   このコマンドにより、AddAccount ビジュアル テストの残りのステップが実行され、その後、AutoQuote ビジュアル テストの次のステップで再生が一時停止されます。

   ステップ アウト を選択すると、Silk Test Workbench により AddAccount の最後のステップとしてホーム ページが表示され、再生が一時停止します。そして、Silk Test Workbench により AutoQuote がデバッグ モードで開かれ、次のステップが強調表示されます。

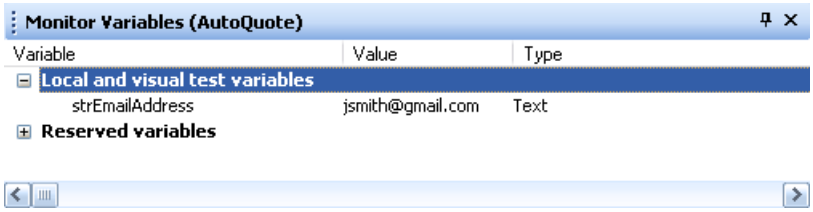
次に、ビジュアル テストで使用されている変数の値を監視する方法について学習します。

再生中の変数の追跡

デバッグ モードでは、ローカル変数 ウィンドウを使用して、ローカル変数と予約変数の両方の再生値を追跡できます。

このセクションでは、ローカル変数 ウィンドウを使用して、前のレッスンで作成したローカル変数の値を参照します。

1. **デバッグ** > **ローカル変数** を選択します。 ローカル変数 ウィンドウが開きます。
2. **ローカルおよびビジュアル テスト変数** ノードを展開します。
   前のレッスンで作成したローカル変数 *strEmailAddress* が表示されます。 再生中、この変数を使用するステップが実行されると、変数の値が **値** 列に表示されます。 結果は、以下の図のようになります。



   注: 変数の結果を ローカル変数 ウィンドウで確認する場合にテストでエラーがない場合は、ブレークポイントをビジュアル テストに挿入します。 ブレークポイントを挿入するには、variable ステップのあとにあるステップを選択し、**デバッグ** > **ブレークポイントの設定／クリア** を選択してからテストを再生します。
3. **カーソルまで実行** デバッグの動作を確認するには、「(24, 9) の場所で 'Lexus' をクリックします」ステップを選択し、**デバッグ** > **カーソルまで実行** を選択します。 Silk Test Workbench は、残りのステップを「(24, 9) の場所で 'Lexus' をクリックします」ステップまで再生したあと、AutoQuote ビジュアル テストを表示します。

次に、モジュール式テストの再生を完了し、テスト結果を確認します。

結果の確認

ビジュアル テストのデバッグが終了したら、ビジュアル テストの結果を確認します。

1. ツールバーの **再生** をクリックして、AutoQuote ビジュアル テストの再生を完了します。**再生完了** ダイアログ ボックスが開きます。
2. **結果へ移動** をクリックします。AutoQuote の結果が表示され、デフォルトで **要約** タブが表示されます。

   要約 タブにはテスト実行の全体的な詳細が表示されます。**ビジュアル テストまたは .NET スクリプト (実行回数)** フィールドには、「AutoQuote(1)」と挿入されたビジュアル テスト「AddAccount(1)」がリストされます。
3. **詳細** タブをクリックします。
4. **テスト ステップ** ペインで、青色のテキストのステップまでスクロールします。

   結果 列と **結果の詳細** 列を確認すると、再生中に発生したエラーに関する情報をすばやく参照できます。

   注: テスト ステップ ペインにすべてのステップを表示するには、[ステップと画面] が選択されていることを確認します。**操作** > **表示** > **ステップと画面** をクリックします。**失敗** タブには、再生エラーを含むステップは表示されません。失敗した検証のみが表示されます。

再生エラーを診断し、デバッグする方法を学びました。

アクティブ データの使用

アプリケーションの使用を効率よく模倣するために、アプリケーション テストでは異なるデータ セットを使用して同じ操作を繰り返すことがよく行われます。 たとえば、前のレッスンでは顧客レコードを作成するテストを記録しました。 10 件の顧客レコードを作成するには、テストごとに顧客データ セットを用意

し、10 個の異なるテストを記録できます。 ただし、Silk Test Workbench では元のテストを拡張し、実行ごとに異なるデータ セットを使用してテストを 10 回繰り返すことができます。

アクティブ データ テストでは、外部ファイルのデータをテスト アプリケーションへの入力として使用し、実行ごとに異なるデータを使用して、指定したステップを繰り返し実行することができます。

このレッスンでは、以下の操作方法を学びます。

- アクティブ データ テスト資産を作成し、ビジュアル テストに関連付ける
- 実行ごとに異なるデータを使用し、指定したステップを特定の回数実行する繰り返しロジックを作成する
- ビジュアル テストでアクティブ データ ファイルを使用する方法を定義する
- 繰り返し実行するステップを定義する
- データ ファイルのアクティブ データをビジュアル テストのリテラル データにマップする

前のレッスンでは、Pat Smith の顧客情報を顧客データベースに入力するビジュアル テストを記録しました。 このビジュアル テストを再生するたびに、最初の記録中にキャプチャしたリテラル データ値が Silk Test Workbench で使用されます。

このレッスンでは、Pat Smith の顧客情報を入力するために使用されるリテラル データ値をアクティブ データに置き換え、Silk Test Workbench がビジュアル テストを再生するときに外部ファイルに含まれる別の顧客情報が使用されるようにします。

ビジュアル テスト用のアクティブ データを作成する前に、アクティブ データ ファイルについて確認します。

アクティブ データ ファイルの確認

アクティブ データ資産を作成する際、使用するデータが含まれる既存のデータ ファイルを選択するか、資産によって使用されるデータを入れる新しいファイルを作成することができます。 このチュートリアルでは、customers.csv という名前の、顧客情報を含むカンマ区切りの既存のファイルを使用します。 このファイルは、Silk Test Workbench のインストール ディレクトリの Examples フォルダにあります。

customers.csv ファイル内の各列は、Age 列を除き、データベースに顧客情報を入力するために使用されるフィールドに対応しています。Age 列は、このチュートリアルのあとのレッスンで使用します。

このテストにアクティブ データを使用すると、Silk Test Workbench はテストを再生し、ファイル内の各顧客がデータベースに入力されるまで、毎回別の顧客のデータを使って顧客情報を入力するステップを繰り返します。

ヒント: アクティブ データ ファイルでは、パスワード値はコード化されたテキストになっています。

アクティブ データ ファイルには以下の列が含まれます。

- 名（First Name）
- 姓（Last Name）
- 生年月日（Birthday）
- 電子メール アドレス（Email Address）
- 住所（Mailing Address）
- 市（City）
- 州（State）
- 郵便番号（Postal Code）
- パスワード（Password）
- 年齢（Age）（後続のレッスンで使用）

次に、アクティブ データ テスト資産を作成し、ビジュアル テストに関連付けます。

アクティブ データ テスト資産の作成

アクティブ データ テストでデータ ファイルを使用するには、そのファイルを使用するアクティブ データ資産を作成し、そのアクティブ データ資産をビジュアル テストに関連付ける必要があります。

1. 開始画面の **最新** リストで、**AddAccount** というビジュアル テストをダブルクリックして開きます。 ビジュアル ナビゲータに AddAccount ビジュアル テストが表示されます。
2. **テスト ステップ** ペインで、**操作** > **挿入** > **アクティブ データ** > **新規作成** をクリックします。 **アクティブ データ資産の設定** ウィンドウが表示されます。
3. **名前** テキスト ボックスにアクティブ データ資産の名前を customers と入力します。
4. **参照** をクリックして、アクティブ データ資産と関連付ける customers.csv ファイルを探します。 **アクティブ データ資産の選択** ダイアログ ボックスが表示されます。
5. サンプル ファイルの場所に移動し、アクティブ データのサンプル ファイル customers.csv を選択します。

   デフォルトの場所は、次のようになります： C:¥Program Files¥Silk¥Silk Test¥examples ¥customers.csv
6. **開く** をクリックします。 **全般** タブの **ファイル** テキスト ボックスにパスとファイル名が表示されます。
7. **アクティブ データ資産の設定** ウィンドウで、**オプション** タブをクリックします。
8. **最初の行を見出しとして使用** チェック ボックスをオンにします。

   ヒント: 詳細 タブをクリックすると、アクティブ データ ファイルの内容を確認することができます。

   この設定により、アクティブ データ ファイルのデータの最初の行が、データではなく見出し行として扱われます。
9. **保存して閉じる** をクリックしてアクティブ データ資産を作成し、AddAccount ビジュアル テストに関連付けます。

ビジュアル テストでは、関連付けられているアクティブ データ資産の情報はすべて、<<開始>> ステップの **プロパティ** ペインに格納されます。 ビジュアル テストに関連付けられている各アクティブ データ資産を確認するには、<<開始>> ステップを選択し、**プロパティ** ペインの **アクティブ データ** プロパティを確認します。

これで、アクティブ データ資産が作成され、ビジュアル テストと関連付けられました。次に、テストでアクティブ データを使用する繰り返しロジックを作成します。

アクティブ データ ファイルの繰り返しロジックの作成

ビジュアル テストでアクティブ データを使用する場合、通常は一連のステップを繰り返し、その際毎回リテラル データをアクティブ データに置き換えます。

このレッスンでは、データベースに顧客を追加するリテラル データを、10 件の顧客レコードを含むアクティブ データ ファイル customers.csv 内のデータに置き換えます。 繰り返しごとに、アクティブ データ ファイルの顧客レコードのデータが使用されて、InsuranceCo Web サイトの **Create a New Account** ページの各フィールドにデータが入力されます。

1. ビジュアル ナビゲータに AddAccount ビジュアル テストが表示されていることを確認します。
2. Silk Test Workbench のツールバーの **繰り返しタイプのロジックの作成** をクリックします。 **ようこそ** ページが表示された **テスト ロジック デザイナ** ウィザードが表示されます。
3. **次へ** をクリックします。 **ロジックの種類の選択** ページが開きます。
4. **アクティブ データ ファイルを使用してステップ シーケンスを繰り返す** をクリックし、**次へ** をクリックします。 **使用するアクティブ データ資産の定義** ページが表示されます。

5. このビジュアル テストに関連付けられているアクティブ データ資産は 1 つだけ（customers）なので、**アクティブ データの資産** リストにこの資産が表示されます。 そうでない場合は、リストからこの資産を選択します。
6. **開始行** テキスト ボックスで、デフォルト値の **1** をそのままにします。
7. **データを含む最後の行で終了** チェック ボックスをオンにします。
8. **順序通りにすべての行を取得** をクリックします。

   これで、繰り返しロジックでデータ ファイルを使用する方法、および繰り返し回数が定義できました。

   - 繰り返しロジックの実行時、アクティブ データファイルの最初の行が最初に使用されます。
   - 各行を使って、最後の行まで繰り返されます。
   - 繰り返しロジックではすべてのデータ行が順番に使用され、アクティブ データ ファイルにはデータ行が 10 行あるため、データベースに顧客を追加するステップは 10 回繰り返されます。 毎回、アクティブ データ ファイルの異なる顧客データ行が入力として使用されます。
9. **次へ** をクリックします。 **繰り返しの構築** ページが開きます。

アクティブ データの使用方法が定義できたので、次にビジュアル テスト内の繰り返すステップを指定します。

繰り返しステップの定義

繰り返しロジックで繰り返すステップを決定するときは、リテラル データを置き換えるステップだけでなく、繰り返しが必要なすべての操作を考慮します。

このレッスンでは、リテラル データを置き換えるステップがいくつかありますが、アクティブ データ ファイル内の各顧客を入力するには、InsuranceCo Web サイトの **Create a New Account** ページにアクセスするプロセス全体を繰り返す必要があります。 そのため、ページにアクセスし **Home** ページにテストを戻すステップも含め、実際のテスト プロセスのすべてのステップを繰り返す必要があります。

1. **繰り返しの構築** ページで **ステップの開始** リンクをクリックします。 **ステップの選択** ダイアログ ボックスが開きます。
2. InsuranceCo Web サイトとの対話を開始するステップを選択します。 これは <<開始>> ステップのすぐあとのステップです。

   ビジュアル テストには顧客を追加するプロセスで必要なステップがすべて含まれるため、すべてのステップを繰り返す必要があります。
3. **OK** をクリックします。 **ステップの選択** ダイアログ ボックスが閉じられ、選択したステップのテキストが **ステップの開始** リンクに表示されます。
4. **繰り返しの構築** ページで **ステップの終了** リンクをクリックします。 **ステップの選択** ダイアログ ボックスが開きます。
5. InsuranceCo Web サイトとの対話を終了するステップを選択します。 これは <<終了>> ステップのすぐ前のステップです。
6. **OK** をクリックします。 **ステップの選択** ダイアログ ボックスが閉じられ、選択したステップのテキストが **ステップの終了** リンクに表示されます。
7. **次へ** をクリックします。 **テスト ロジック デザイナ** の **要約** ページが表示されます。 このページには、ビジュアル テストに定義されたテスト ロジックが表示されます。
8. **終了** をクリックして、ビジュアル テストのテスト ステップにテスト ロジックを挿入します。

   Silk Test Workbench により、ビジュアル テストの <<開始>> ステップのあとに、繰り返しロジックを開始するステップが挿入されます。 ステップのテキストは次のようになります：「アクティブ データ 'customers' を使用して繰り返します」

   また、Silk Test Workbench により、<<終了>> ステップのあとに、繰り返しロジックを終了するステップも挿入されます。 ステップのテキストは以下のようになります。

   繰り返しの終了

   新しく挿入された繰り返しロジック ステップの間のすべてのステップがインデントされ、繰り返しロジック ステップの間にネストされて表示されます。 ビジュアル テストでアクティブ データを使用する

ための最後の作業は、アクティブ データ ファイル内のデータをビジュアル テストのリテラル データにマップすることです。

アクティブ データのリテラル データへのマッピング

ビジュアル テストでアクティブ データ ファイル内のデータを使用するには、該当するテスト ステップのデータをアクティブ データ ファイル内のデータ列にマップする必要があります。

1. ビジュアル テストで、フィールドにデータを入力する最初のステップを選択します。

   これは、ビジュアル テストで最初に記録された、InsuranceCo Web サイトの **Create a New Account** ページの **First Name** テキスト ボックスに「Pat」と入力するステップとなります。

   ステップのプロパティが **プロパティ** ペインに表示されます。
2. **プロパティ** ペインの **パラメーター** カテゴリーで、**text** パラメーターを選択します。これは、**Pat** に設定されたテキストを示します。
3. 値領域の **選択** をクリックし、**アクティブ データ** をクリックします。

   ヒント: customers が **アクティブ データの資産** リストにない場合、この資産はこのビジュアル テストに関連付けられていません。

   プロパティ/変数/アクティブ データの選択、またはリテラルの編集 ダイアログ ボックスが開きます。このダイアログ ボックスで、テスト ステップの選択したプロパティのデータをアクティブ データ ファイル内の列のデータにマップします。

   アクティブ データの資産 テキスト ボックスには、ビジュアル テストに関連付けられたアクティブ データ資産 **customers** が表示されます。 **列** リストには、customers アクティブ データ ファイルの列名が表示されます。
4. **列** リストで、**First Name** 列を選択します。

   これが名前を含んだファイル内の列であるため、この列をマップして、リテラル データ「Pat」を置き換えます。
5. **OK** をクリックしてダイアログ ボックスを閉じ、データをマップします。 Silk Test Workbench によって、テスト ステップの実際のデータが、アクティブ データ ファイル内の選択したデータへのマップ式に置き換えられます。 既存のステップ テキストは、以下のとおりです。

   'Pat' を入力します

   これが、アクティブ データ資産名、置換するデータ型、および列名を示す、以下のテキストに更新されます。

   '[[customers].Text("First Name")]' を入力します
6. ビジュアル テストに関連付けたアクティブ データ ファイル内のデータに置換するデータを含む残りのテスト ステップすべてについて、ステップ 1 から 5 を繰り返します。 次のステップは、「'Smith' を入力します」というステップ テキストが含まれるステップです。

   下の表は以下の内容を示します。

   - このビジュアル テスト内の、アクティブ データ ファイルのデータで置き換えるリテラル データを含む各ステップのテキスト
   - 置き換えるデータを含むステップのプロパティの名前
   - リテラル データの代わりに使用するデータを含むアクティブ データの列

   注: アクティブ データを使用して更新された、パスワード保護テキスト ボックスにテキストを入力するステップについては、ステップのテキストにアクティブ データ情報が示されません。

ステップ テキスト	プロパティ	アクティブ データ列
'Smith' を入力します	テキスト	Last Name
'smith@test.com' を入力します	テキスト	Email Address

ステップ テキスト	プロパティ	アクティブ データ列
'1212 Test Way' を入力します	テキスト	Mailing Address
'San Diego' を入力します	テキスト	City
リスト ボックス項目 'California' を選択します	リスト ボックス項目	State
'92121' を入力します	テキスト	Postal Code
'test' を入力します	テキスト	Password

アクティブ データ ファイルのデータを使用するようにビジュアル テストを設定できたので、これを再生し、結果を確認します。

アクティブ データのビジュアル テストの再生と分析

顧客を追加するビジュアル テストを、アクティブ データ ファイルの顧客レコードを使用するように拡張できたので、再生してみます。このビジュアル テストを再生すると、アクティブ データ ファイルの顧客レコードが実際に顧客データベースに入力されます。

1. 次のいずれか 1 つのステップを行います：
   - **操作** > **再生** を選択します。
   - ツール バーで **再生** をクリックします。

   再生 ダイアログ ボックスが開きます。
2. **結果の説明** テキスト ボックスに AddCustomers using ActiveData と入力します。
3. **OK** をクリックします。
4. 再生をサポートしている複数のブラウザーがマシンにインストールされている場合、**ブラウザーの選択** ダイアログ ボックスが開きます。ブラウザーを選択して、 **実行** をクリックします。 Silk Test Workbench によって、拡張したテストが再生されます。
5. **再生完了** ダイアログ ボックスで **結果へ移動** をクリックします。

   結果 ウィンドウが開き、デフォルトで **要約** タブが表示されます。
6. **詳細** タブをクリックして、各ステップの結果を表示します。

   詳細 タブには、再生中に実行された各ステップの結果が表示されます。スクロールバーを使用して、実行されたステップをスクロールします。

   結果 ウィンドウの **テスト ステップ** ペインには、ビジュアル テストの実際のステップよりもかなり多くのステップがあります。これは、繰り返しごとに、それぞれ実行ステップのセットがあるためです。

注: また、式を使用して、ビジュアル テストでアクティブ データを更新することもできます。**式デザイナ** を使用すると、アクティブ データ ファイルの列のデータを入力として使用でき、データの値を変更する式を作成できます。更新されたデータは、同じアクティブ データ ファイルの同じ列に保存できます。詳細については、オンライン ヘルプの「ビジュアル テストでの式を使用したアクティブ データの更新」を参照してください。

ビジュアル テストからのスクリプトの再生

スクリプトとビジュアル テストは、資産を使用してテスト アプリケーションでのメニュー項目の選択やデータの入力などの手動のユーザー操作を自動化するという点で同じです。 違いは、ユーザーの操作がどのように表現されるかです。 スクリプトでは、Microsoft .NET Framework で稼動するスクリプト記述言語である Microsoft の Visual Basic が使用されますが、ビジュアル テストでは、ビジュアル ナビゲータのポイントアンドクリック インターフェイスによって生成されるステップが使用されます。

いずれの場合も、互いに独立して実行できるパワフルで柔軟性の高い自動テストを作成できます。 どちらを選ぶかは、ユーザーのニーズと好みによります。 また、両方を組み合わせて使用することもできます。

たとえば、特殊なタスクを実行するスクリプトを作成し、ビジュアル テストにそのスクリプトを再生するステップを挿入することができます。 このようにして、スクリプトで使用されるスクリプト記述言語の力を利用してビジュアル テストを補強することができます。

ビジュアル テストからスクリプトを再生すると、スクリプトは、繰り返しテスト タスクまたは特殊なテスト タスクを実行するために必要に応じて呼び出せる関数と同様に機能します。 このアプローチはチームによるテスト環境で役立ちます。たとえば、経験の豊富なテスト担当者が共通するテスト機能を実行するスクリプトのライブラリを作成し、経験の浅い開発者がそこからスクリプトを選択して、ビジュアル テストを作成することができます。

乱数を生成するスクリプトの作成

まず最初に、ユーザーの年齢に使用できる乱数を生成するスクリプトを作成します。

1. **ファイル** > **新規作成** を選択します。**資産の新規作成** ダイアログ ボックスが開きます。
2. [資産の種類] リストから **.NET スクリプト** を選択し、**資産名** テキスト ボックスに function_randomAge と入力して **OK** をクリックします。

   ヒント: スクリプトに「function」と名前を付けることにより、共通して使用されるテスト タスクのライブラリを整理でき、ここからすばやくタスクを選択して、ビジュアル テストに挿入することができます。

   スクリプトが **コード** ウィンドウに表示されます。
3. Sub Main() の下の行にカーソルを置き、以下のように入力します。

```
Dim rand As New Random()
Dim TSrandomAge As Integer = rand.Next(10000, 99999)
MsgBox(TSrandomAge)
```

4. ツール バーで **再生** をクリックします。 **再生** ダイアログ ボックスが開きます。
5. **OK** をクリックします。 メッセージ ボックスが表示され、10,000 から 99,999 の生成された乱数が表示されます。
6. **OK** をクリックします。 **再生完了** ダイアログ ボックスが開きます。
7. **.NET スクリプトへ移動** をクリックします。
8. 数値範囲パラメータの (10000, 99999) をパラメータ *MinVal* と *MaxVal* に置き換えます。 これら 2 つのパラメータを使用することにより、スクリプトを開いて値を変更するのではなく、このスクリプトを再生するビジュアル テストから乱数の範囲を設定することができます。 以下のように入力します。

```
Dim TSrandomAge As Integer = rand.Next(MinVal, MaxVal)
```

9. スクリプトが適切に動作することを確認するために、Sub Main() の直後で、以下の値をパラメータに割り当てます。

```
Dim MinVal=16
Dim MaxVal=105
```

   スクリプトは以下のコードのようになっているはずです。

```
Public Sub Main()
   Dim rand As New Random()
   Dim MinVal=16
   Dim MaxVal=105
   Dim TSrandomAge As Integer = rand.Next(MinVal, MaxVal)

   MsgBox(TSrandomAge)
End Sub
```

10. ツール バーで **再生** をクリックします。 **再生** ダイアログ ボックスが開きます。
11. **OK** をクリックします。 メッセージ ボックスが表示され、16 から 105 の生成された乱数が表示されます。
12. **OK** をクリックします。 **再生完了** ダイアログ ボックスが開きます。

13. **.NET スクリプトへ移動** をクリックします。
14. 入力パラメータを使用して *MinVal* および *MaxVal* パラメータを設定するには、以下のステップを実行します。
    a) 作成する入力パラメータが含まれるように Main() sub を変更します。

```
Public Sub Main(args As IDictionary(Of String, Object))
   Dim MinVal=args ("MinVal")
   Dim MaxVal=args ("MaxVal")
```

    "MinVal"と "MaxVal" は、次の手順で作成する入力パラメーターの名前です。
    b) 作成する出力パラメータが含まれるように Main() sub を変更します。

```
Public Sub Main(args As IDictionary(Of String, Object))

   args ("TSrandomAge")=TSrandomAge
```

    "TSrandomAge" は、次の手順で作成する出力パラメーターの名前です。
    c) 年齢はスクリプトからビジュアル テストに渡されるため、ステートメントの先頭にアポストロフィを入力してメッセージ ボックスのコードをコメントにします。

```
'MsgBox(TSrandomAge)
```

    スクリプト全体は以下のようになっているはずです。

```
Public Module Main
    Dim _desktop As Desktop = Agent.Desktop

    Public Sub Main(args As IDictionary(Of String, Object))

        Dim rand As New Random()

        Dim MinVal= args("MinVal")
        Dim MaxVal= args("MaxVal")

        Dim TSrandomAge As Integer = rand.Next(MinVal, MaxVal)
        args ("TSrandomAge")=TSrandomAge
        'MsgBox(TSrandomAge)

    End Sub
End Module
```

次に、ランダム年齢をビジュアル テストに渡すために使用するスクリプトの入力パラメータと出力パラメータを定義します。

スクリプトの入力パラメータの定義

スクリプトは、入力パラメータを使用してビジュアル テストからデータを受け取ったり、逆に出力パラメータを使用してビジュアル テストにデータを渡したりできます。 このタスクでは、function_randomAge スクリプトで作成された乱数の範囲を設定する 2 つの入力パラメータを定義します。

1. **プロパティ** ペインで、**入力パラメータの追加** を右クリックして選択します。 **スクリプト入力パラメータの追加** ダイアログ ボックスが開きます。
2. **名前** テキスト ボックスに **MinVal** と入力します。
3. **型** リストで、**数値 (Double)** を選択します。
4. **デフォルト値** テキスト ボックスに 16 と入力します。
   必要に応じて、ビジュアル テストにデフォルト値を設定することもできます。
5. **OK** をクリックします。 **プロパティ** ペインの入力パラメータのリストに、入力パラメータが表示されます。
6. **プロパティ** ペインで、**入力パラメータの追加** を右クリックして選択します。 **スクリプト入力パラメータの追加** ダイアログ ボックスが開きます。

7. **名前** テキスト ボックスに **MaxVal** と入力します。
8. **型** リストで、**数値 (Double)** を選択します。
9. **デフォルト値** テキスト ボックスに 105 と入力します。
   必要に応じて、ビジュアル テストにデフォルト値を設定することもできます。
10. **OK** をクリックします。 **プロパティ** ペインの入力パラメータのリストに、入力パラメータが表示されます。

スクリプトの出力パラメータの定義

スクリプトは、入力パラメータを使用してビジュアル テストからデータを受け取ったり、逆に出力パラメータを使用してビジュアル テストにデータを渡したりできます。 このタスクでは、function_randomAge スクリプトで作成された乱数をビジュアル テストに渡すパラメータを定義します。

1. **プロパティ** ペインで、**出力パラメータの追加** を右クリックして選択します。 **スクリプト出力パラメータの追加** ダイアログ ボックスが開きます。
2. **名前** テキスト ボックスに **TSrandomAge** と入力します。
3. **型** リストで、**数値 (Double)** を選択します。
4. **デフォルト値** テキスト ボックスは空のままにします。
5. **OK** をクリックします。 **プロパティ** ペインの出力パラメータのリストに、出力パラメータが表示されます。

ビジュアル テストで出力パラメータを受け取るローカル変数を定義します。

スクリプトのデータを使用するビジュアル テストの設定

スクリプトのデータを使用するようにビジュアル テストを設定するには、スクリプトを再生し、スクリプトのデータを格納するローカル変数を作成し、このローカル変数をスクリプトの出力変数と関連付けるステップを挿入する必要があります。 さらに、ビジュアル テストからスクリプトにデータを渡すために、スクリプトの入力変数の値を設定する必要があります。

1. AutoQuote ビジュアル テストを開きます。
2. **ファイル** > **名前を付けて保存** を選択します。 **名前を付けて保存** ダイアログ ボックスが開きます。
3. AutoQuote ビジュアル テストを変更するため、元のテストと区別できるよう **AutoQuote_Modified** と名前を変更し、**OK** をクリックします。
4. 以下のステップ テキストのあとにあるステップを選択します。
   ビジュアル テスト 'AddAccount' の再生
5. **テスト ステップ** ペインで、**操作** > **挿入** > **.NET スクリプト** を選択します。 **.NET スクリプトの参照** ダイアログ ボックスが開きます。
6. **function_randomAge** を選択し、**OK** をクリックします。 Silk Test Workbench により、スクリプトを再生するステップが挿入されます。
7. スクリプトのデータを格納するローカル変数を追加するには、以下のステップを実行します。
   a) **テスト ステップ** ペインで、**操作** > **挿入** > **変数** > **ローカルの追加** をクリックします。 **ローカル変数の追加** ダイアログ ボックスが開きます。
   b) **変数名** テキスト ボックスに VTrandomAge と入力します。
   c) **型** リストから **数値（Double）** を選択し、**初期値** を 0 に設定します。
   d) **OK** をクリックします。 <<開始>> ステップから変数を参照し、定義を編集することができます。
8. スクリプトのステップを選択します。
   ステップは次のようになります：「.NET スクリプト 'function_randomAge' の再生」
9. 挿入されたスクリプト ステップの **プロパティ** ペインで、**'TSrandomAge' の内容の引き渡し先** プロパティの値領域をクリックし、ローカル変数 **VTrandomAge** を選択します。

入力パラメータをスクリプトで追加したため、その値が **プロパティ** ペインに表示されます。 ただし、値を編集することで、ビジュアル テストで使用される入力パラメータを変更できます。 これらの値は、スクリプトでは変更されません。

これで、スクリプトのデータを使用するようにビジュアル テストが設定されました。AutoQuote_Modified ビジュアル テストが再生されるたびに、function_randomAge スクリプトが再生され、ビジュアル テストのローカル変数 VTrandomAge に固有の乱数が渡されます。

次に、function_randomAge スクリプトで生成された乱数を含むビジュアル テストのローカル パラメータを使用するように、年齢情報を入力するステップを変更します。

ビジュアル テストでのスクリプト データの使用

このセクションでは、function_randomAge スクリプトで生成された乱数を含むビジュアル テストのローカル パラメータを使用するように、年齢情報を入力するステップを変更します。

1. **Age** テキスト ボックスで、データを入力するステップを選択します。 このステップ テキストは、以下のとおりです。

   '42' を入力します
2. **プロパティ** ペインで、**テキスト** プロパティを選択します。
3. **選択** ボタンをクリックして、**変数** を選択します。 **プロパティ/変数/アクティブ データの選択、またはリテラルの編集** ダイアログ ボックスが開きます。
4. **VTrandomAge** を選択し、**OK** をクリックします。 ステップ テキストは、次のようになります：「'[VTrandomAge]' を入力します」

このステップで、*VTrandomAge* ローカル変数の値が **Age** テキスト ボックスに入力されるようになりました。

テスト結果の再生と確認

1. 次のいずれか 1 つのステップを行います：

   - **操作** > **再生** を選択します。
   - ツール バーで **再生** をクリックします。

   再生 ダイアログ ボックスが開きます。
2. **結果の説明** テキスト ボックスに Adding Customers using data passed from a script と入力します。
3. **OK** をクリックします。
4. 再生をサポートしている複数のブラウザーがマシンにインストールされている場合、**ブラウザーの選択** ダイアログ ボックスが開きます。ブラウザーを選択して、 **実行** をクリックします。 Silk Test Workbench によって、拡張されたテストが再生され、スクリプトから渡された乱数とビジュアル テストで定義した追加の顧客情報を使用して 10 人の新規顧客が作成されます。
5. **再生完了** ダイアログ ボックスで **結果へ移動** をクリックします。

   結果 ウィンドウが開き、デフォルトで **要約** タブが表示されます。
6. **詳細** タブをクリックして、各ステップの結果を表示します。

索引

G

GUI
概要 4
メイン画面 4

U

UI
概要 4

あ

アクティブ データ
繰り返しロジック 27–29
資産の作成 27
チュートリアル 25, 30
データ ファイル 26
データ ファイルの確認 26

え

エラー
概要 23
確認 25
調べる 23
デバッグ 23

か

開始画面
概要 5
画面プレビュー
ビジュアル テストの更新 17

く

繰り返しロジック
アクティブ データ 27–29

け

結果
エラーの確認 25
概要 11
[結果] ウィンドウ 12
[結果] ウィンドウ
画面プレビュー 15
タブ 14
ツールバー 15
[プロパティ] ペイン 15
検証
追加 17

さ

再生
ビジュアル テスト 11
再生エラー
概要 23
確認 25
調べる 23
デバッグ 23
変数の追跡 24
サンプル アプリケーション
起動 8
記録 8

し

出力パラメータ
定義 33

す

スクリプト
サンプル アプリケーション 8
出力パラメータ 33
入力パラメータ 32
ビジュアル テストでの使用 33, 34
乱数の生成 31

ち

チュートリアル
アクティブ データ 25, 30
再生 34
乱数の生成 31

て

テスト
モジュール式 21
テスト結果
概要 11
画面プレビュー 15
タブ 14
ツールバー 15
[プロパティ] ペイン 15

に

入力パラメータ
定義 32

ひ

ビジュアル テスト
2 つめのテストの記録 21
拡張 16
拡張したテストの再生 20
確認 9
記録 7, 8
検証の追加 17

更新 17
再生 11
スクリプトでの使用 33, 34
挿入 22
変数の格納 19
変数の追加 18
保存 9
命名 9
モジュール式の概要 20, 21
ビジュアル テストの拡張
概要 16
画面プレビューからの更新 17
検証の追加 17
再生 20
変数の追加 18, 19
ビジュアル テストの記録
概要 7
確認 9
サンプル アプリケーション 8
保存 9
ビジュアル ナビゲータ
概要 6

へ

変数
追加 18
データの格納 19
デバッグ 24

め

メイン画面
GUI 4

も

モジュール式テスト
2 つめのテストの記録 21
概要 20, 21
挿入 22

ろ

ローカル変数
追加 18
データの格納 19