FlexMonkey 1.0 QuickStart
Updated 7/16/09

Welcome to the FlexMonkey 1.0 QuickStart.

In this tutorial, you'll see how easy it is to start testing with the monkey. We'll guide you through setting up your first FlexMonkey project, and creating your first test. Then we'll add an assertion to your test, and run the test to see how it all works. When we're done, we'll save our work and close down the monkey. We assume you have already installed FlexMonkey by clicking the installer on http://flexmonkey.gorillalogic.com. Let's get started.

Basic FlexMonkey Options.

There are two basic ways to use FlexMonkey. You can run the application you would like to test in a browser, or you can run the application in a FlexMonkey window. Running in the browser is necessary for many applications, but running in a FlexMonkey window is easier and faster. The rest of the tutorial applies to either method. If you want to run in a FlexMonkey window, you can skip the *Running with the MonkeyAgent* section and move on to *Setting up your first FlexMonkey project*.

Running with the MonkeyAgent.

If you want to run the application you would like to test in a browser, you need the MonkeyAgent. (You can download the MonkeyAgent from http://flexmonkey.gorillalogic.com -- it's in MonkeyAccesories.) The agent is a Flex application that will host your application during testing. It consists of three files, *MonkeyAgent.swf*, *MonkeyAgent.html*, and *AC_OETags.js*.

To install the MonkeyAgent, all you need to do is to place these files together in the same domain as the application you will be testing. If your application is on a server, you'll need to put the MonkeyAgent (SWF, HTML and Javascript) there (If *AC_OETags.js* is already there, you don't need the one that comes with the Agent.) If you will be running locally, the easiest method is to put the MonkeyAgent in the Flex Builder *bin-debug* directory of the application you are testing. If you place the Monkey Agent anywhere that is not a Flex Builder *bin-debug* directory, then you will need to change the Flash Player Global Security settings for that directory. Please see http://www.macromedia.com/support/documentation/en/flashplayer/help/settings_manager04.html#119065 for more details. Once the MonkeyAgent is installed, you will need to run the agent any time you use FlexMonkey to test an application running in the browser.

When the MonkeyAgent opens, you'll see it is divided into two parts; the upper part is where your application will appear, and the lower part is the *MonkeyAgent Console Window*, which will log communication with FlexMonkey. If you don't want to see the log, just pull down the divider bar to the bottom of the browser window.
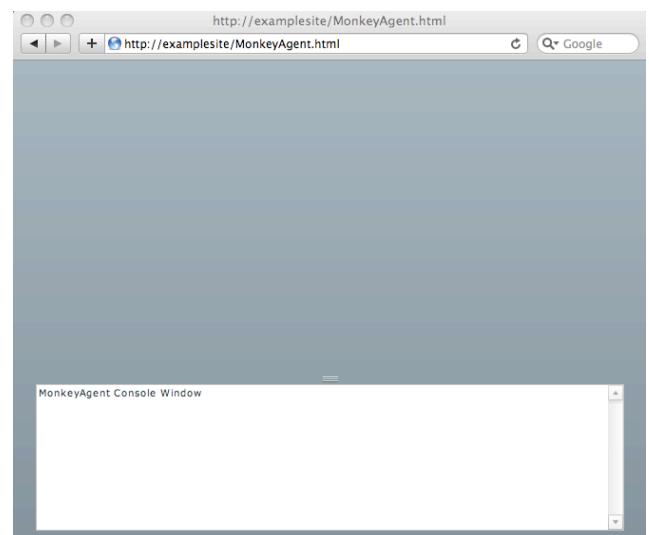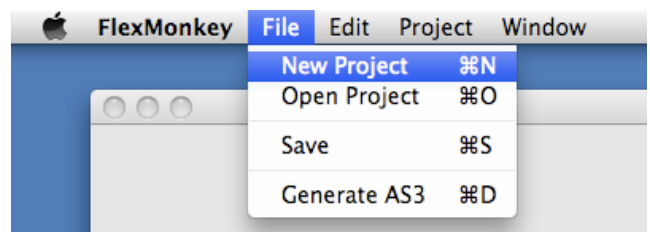
Setting up your first FlexMonkey project.

When the monkey opens for the first time, you'll see a dialog like this one. Just follow the instructions to create your first FlexMonkey project. Click *OK*.
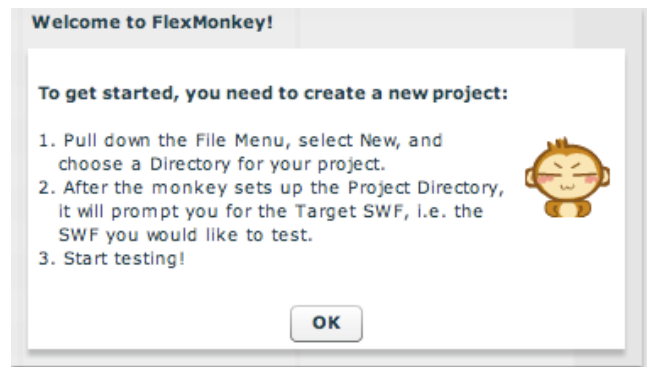

Pull down the File Menu and select *New Project*.

In the *New Project* directory browser, choose a directory to hold your FlexMonkey project files. Here, we are choosing the FlexBuilder project directory for the MonkeyContacts project.
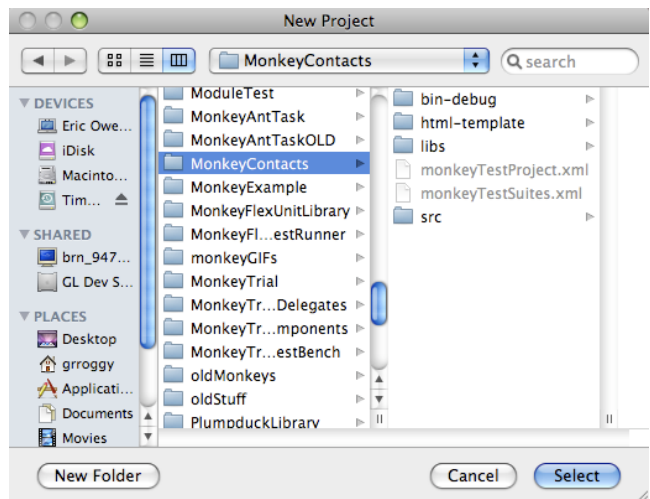

Next, the monkey pops open the *Project Properties Window*. If you are using the MonkeyAgent, click the *Use Browser* box. If the SWF you would like to test is a local file, click the Test Target SWF *Browse* Button. If the SWF is remote, type in the URL. Here, we click the *Browse* button.
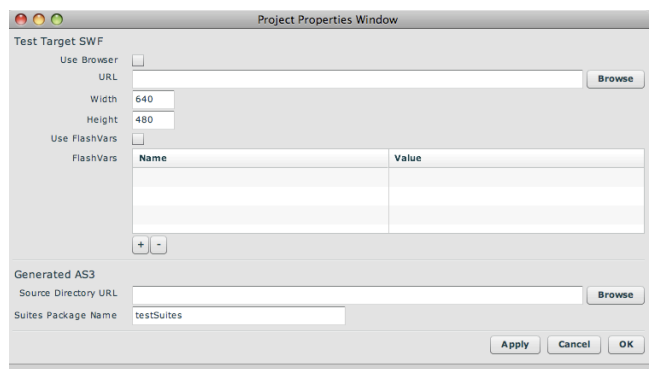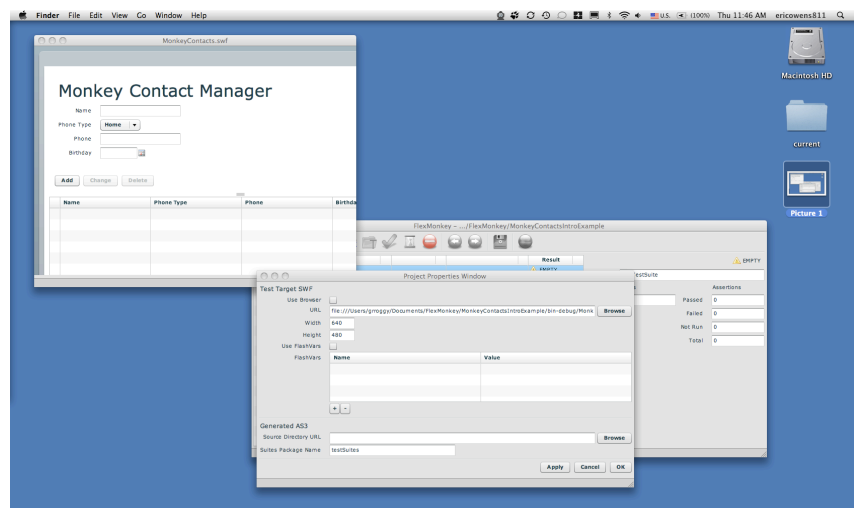
In the *Select Test Target SWF* Browser Window, choose the SWF file for the application you would like to test. Here, we are choosing the MonkeyContacts.swf file.
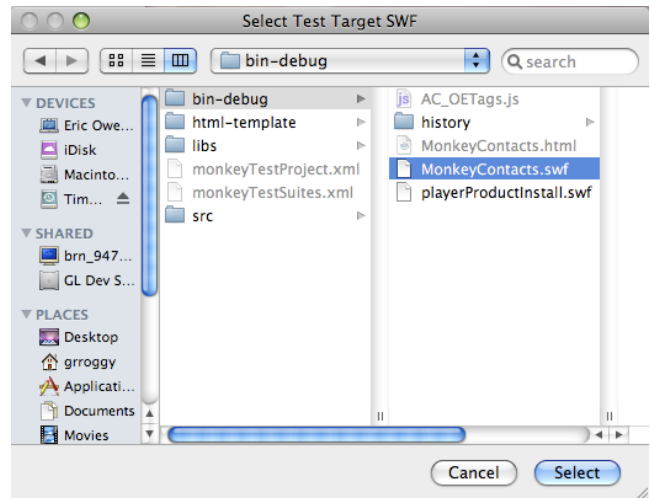
After you have filled in the Test Target SWF URL, click the *Apply* button and your Target SWF should appear in the *FlexMonkey TargetSWF* Window.
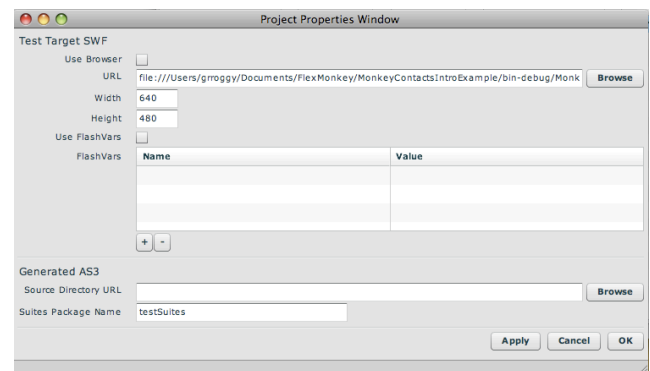
Now, your desktop should look something like this.

If your application is too large for the default size of the *FlexMonkey TargetSWF* Window, you can adjust its height and width in the *Project Properties Window*. If you make a change to height or width, as we did here, you can click the *Apply* button to see the effect. When you are satisfied with the appearance of your Target SWF, click OK to close the *Project Properties Window.* (It's OK to leave the *FlashVars* and the *Generated AS3 Source Directory* empty for now.) FlexMonkey will save your project properties, and will automatically load them the next time you start the monkey.
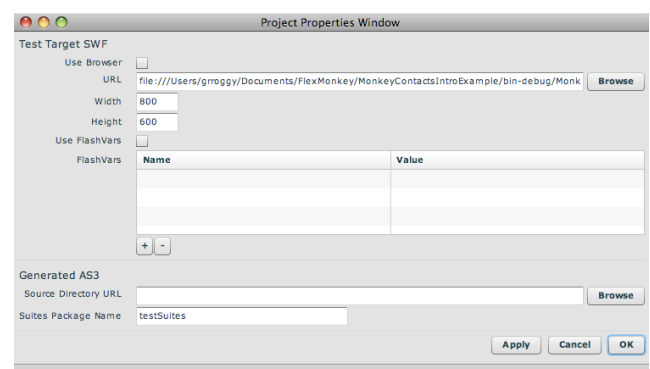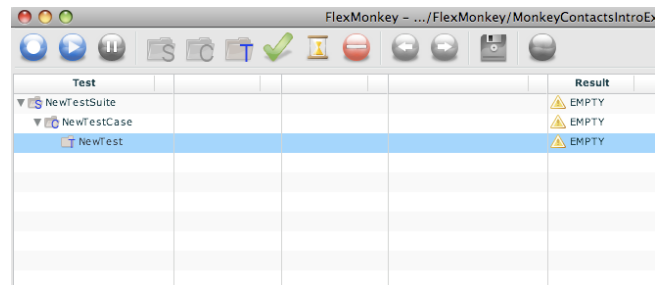
Creating your first test.

After you create a new project, your *FlexMonkey Window* will look like this.
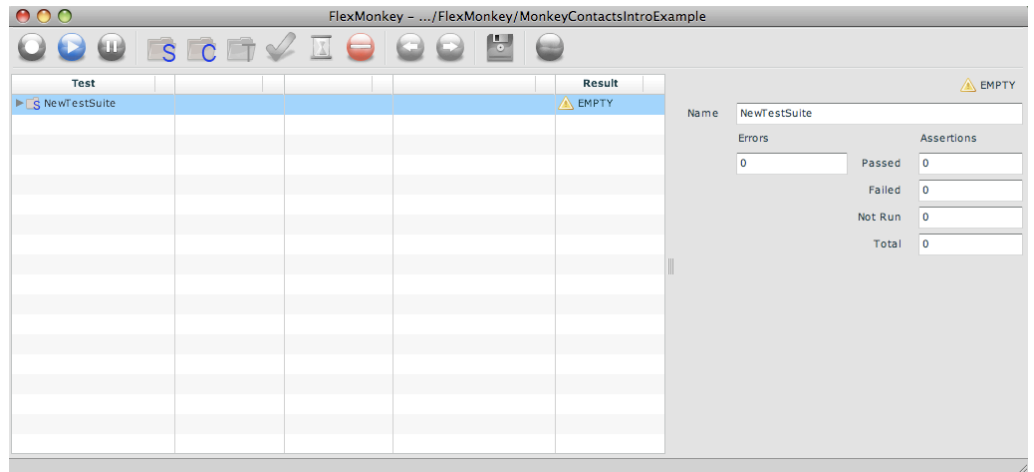
Click on the little triangle next to the *NewTestSuite* and then the *NewTestCase* to expose the *NewTest*. Select the *NewTest* as shown here. NewTest is *Empty* because you haven't added any UI commands to it yet.
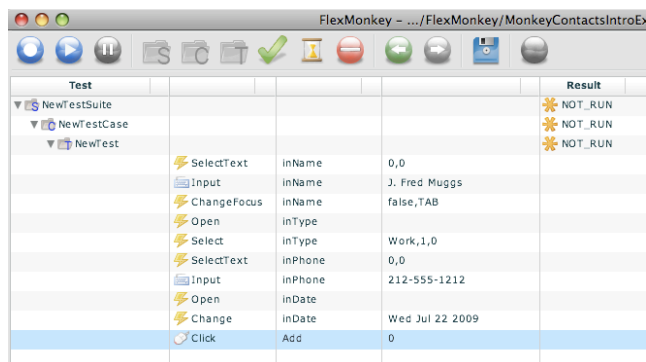

The button in the upper left is the *Record* button. To start recording a test, click Record. The Record button will glow red to indicate that recording is active.
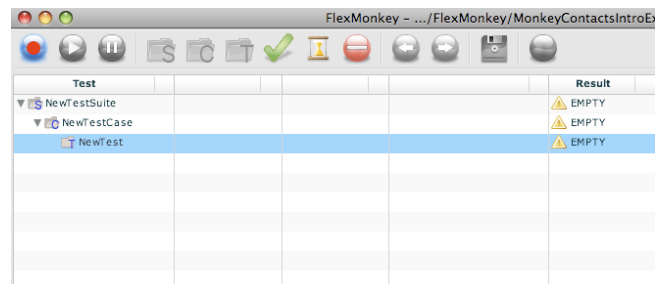
Our application is the *Monkey Contact Manager.* (You can download it as a FlexBuilder project from http://flexmonkey.gorillalogic.com -- it's in MonkeyAccessories.) We are going to test adding a new contact. While recording is active, we'll type a new name, select the phone type from the combo box, enter the Birthday, enter the phone number, and click the *Add* button. When we are done, we'll click the *Record* button in the *FlexMonkey* Window to turn recording off.
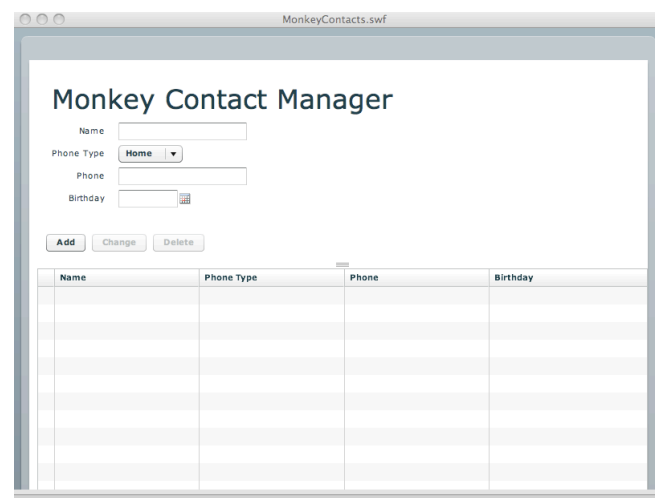
Here's the result. As you can see, the monkey
inserted a sequence of UI commands into
NewTest. Be sure you clicked the *Record* button
again to turn recording off.

## Adding assertions to your test.

With the last command in
the test still selected,
click the *Add Verify*
button, which is the
green check mark in the
control bar. The *Take
Snapshot* dialog will
appear over the *Flex
Monkey Window*
indicating that the
monkey is in Snapshot
mode.
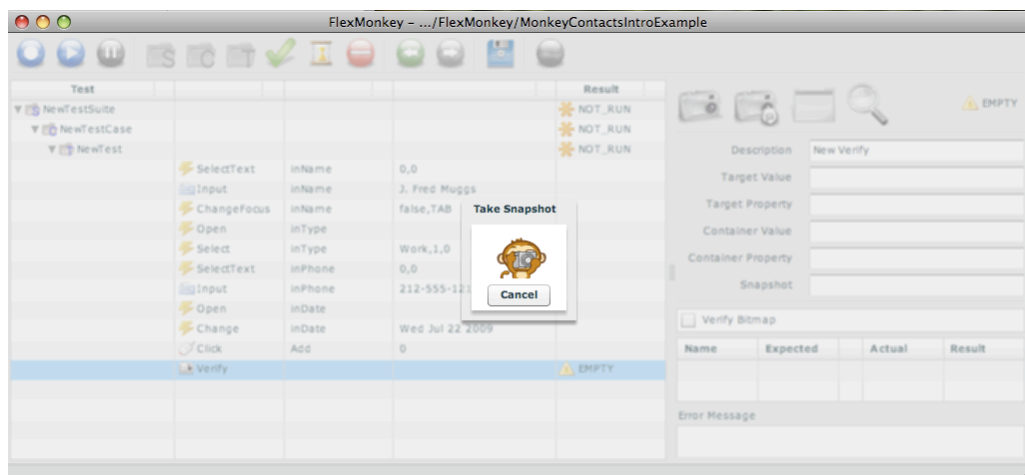
While in Snapshot mode, roll your cursor over the
Target SWF window. The monkey highlights the
visible Flex components in red to make it easy to
select one. Clicking a red-highlighted component
makes it the target for your assertions. Here, we
are selecting the DataGrid.
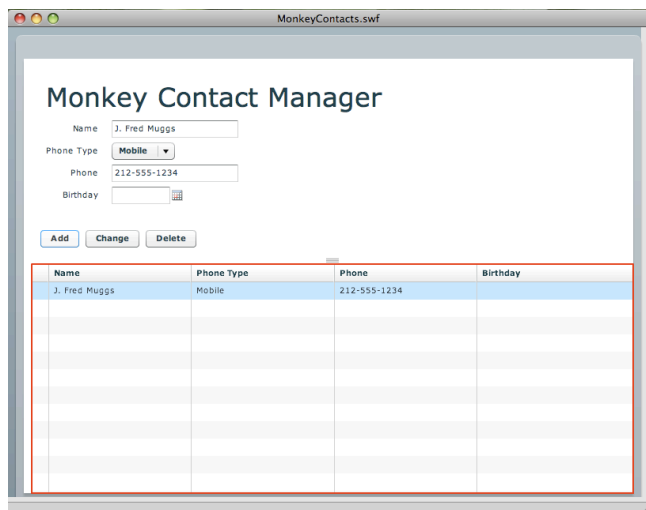
After you select a target component for your assertions, FlexMonkey pops up the *FlexMonkey Spy Window.* The window shows the current properties and styles for the target component. Here we have checked the selectedIndex property, which is currently set to 0. Click the OK button and the monkey will set an assertion that the selectedIndex will be 0 when the Verify Command executes.
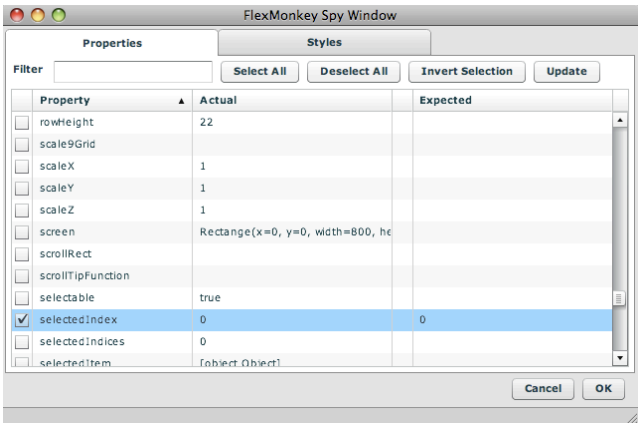

Back in the *FlexMonkey Window*, you can see the result of clicking the OK button in the *FlexMonkey Spy Window.* The detail view for the Verify command shows that a property named selectedIndex is expected to be equal to 0 when the command executes.
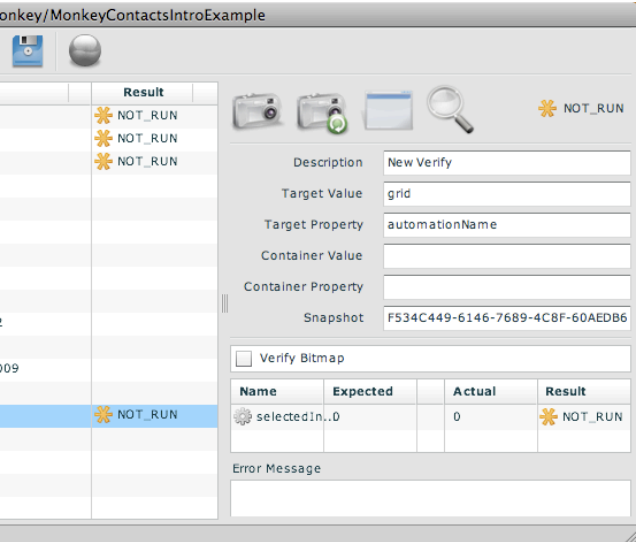

Running your test.

Now let's have FlexMonkey play back the test. First, select *NewTest* as shown here. The *Play* button is just to the right of the record button. Click *Play*, and the monkey will execute all of the UI commands in NewTest.
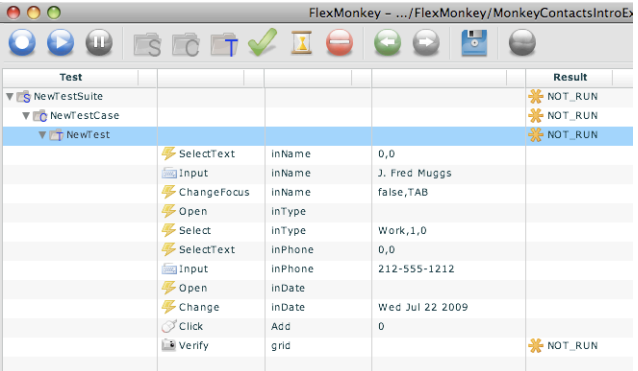
When FlexMonkey is playing, the *Play* button will glow green as shown here. As the monkey plays, it highlights each UI command in the *FlexMonkey Window* as it is executing. Naturally, you can also watch the commands executing in the Target SWF Window.
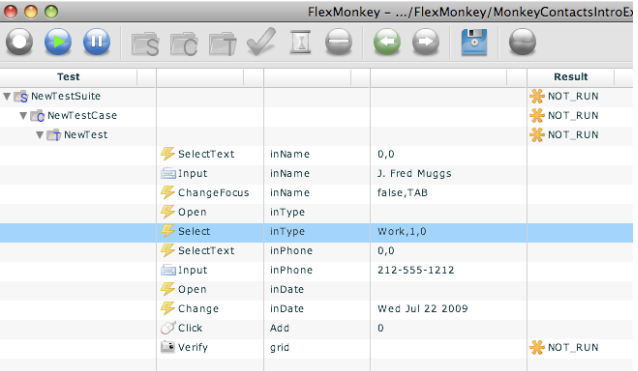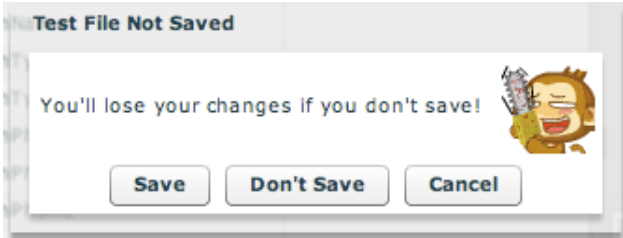
When FlexMonkey is finished playing the last command of the test, the *Play* button will stop glowing green and the results for the test will be updated.

Save your work and shut down.

You can save your work at anytime by clicking the blue diskette at the far right of the control bar. If you don't, and you quit FlexMonkey, you'll be prompted to save your work.

That's it! You're testing with FlexMonkey!