Test Suites

Once you have a few test pages, you’ll want to organize them into test suites, which are just like the similarly named TestSuites from JUnit. Test suites group various test pages, allowing you to run similar tests together by running a single suite. Test suites are nothing more than special test pages that include a list of test pages or other test suites (thereby allowing you to have a master suite) that run in sequence.

You can define test suites in a similar fashion to test pages with two exceptions. First, they cannot contain any test functions. Second, your test suite must contain a suite() function that returns a JsUnitTestSuite object. You can use two methods to add items to a test suite: addTestPage(testPage), which adds an individual test page to your suite, and addTestSuite(testSuite), which adds a suite to your, well, suite. Keep in mind that when you add a test page to a test suite, you need to provide a fully qualified or relative pathname to your file in relation to the test runner. In other words, if your jsunit folder is in the same directory as your test pages, then the test runner is one folder deeper than your tests. If you see an error like the one shown in Figure 6-8, make sure you’ve provided the path relative to the test runner. When you want to add other test suites to your test suite, keep in mind that the argument to addTestSuite must be of type JsUnitTestSuite, which is declared in the same page as the suite function. Helpfully, you will find a test suite example in the jsunit/tests directory. Listing 6-8 also shows an example.

Figure 6-7. Running the setUpPage() test