Notes to try to document how I got the JUnit tests incorporated as part of the GIT.

Not sure exactly what I did but here are some of the steps that I think I went through:

------------------------------------------------------------------------------------------------------------------------------------------------------

Part 1. - 20150507

NEW - ProgSumListItemTest - JUnit Tests Currently only ProgSumListItemTest under a new tests directory.

Not exactly sure how I got this to be created a where I wanted it to (which is within the SLM-OspiUI project mostly so that it would fall under the source controlled directory. I tried a bunch if different ways to create the JUnit test project as stackoverflow, Lars and others suggested but I kept getting projects that were separate from SLM-OspiUI and therefore not under GIT. Or I'd get 2 projects, one within the project and one not. I wasn't able to access / create a new file in the test directory within the project and so I just kept trying different things, adding and deleting projects and directories and finally I think I found a way that worked, unfortunately, I have no idea what I did:(

I do know that I created the tests directory first. Then somehow got a package under src to be created but it had the same name as com.slm.ospiui. For some reason Eclipse didn't complain about the duplicate names. I think I renamed the new com.slm.ospiui package to com.slm.ospiui.test and then I was able to add a test file (ProgSumListItemTest.java) to it (but I could have added the file to the original com.slm.ospiui package then moved it to the new package and renamed it sometime in there). I had created a test case many times before all this , when the Test Project was NOT part of the SLM-OspiUI project and then copied the code into the file in several places and tested it in each place before moving it to the different locations. Eventually I moved the entire test package to the tests directory. I have a note in the ProgSumListItemTest file that tells how to run the tests so far. Hopefully I can figure out how to do this again.

Part 2. - 20150508

So I tried to create another Android Test Project but didn't complete it. Here are the steps I tried in hopes of reproducing the results from above....

Trying to create a AndroidTestProject for SLM-OspiUI

* New -> Source folder at same level as src, called it "testAA"
* Selected SLM-OspiUI project
  + File -> New -> Other ...
    - Selected "Android Test Project"
    - Next
  + Project Name "testAA"
    - Uncheck "Use default location"
    - Set to: ...AppWorkspace/SLM-OspiUI/testAA
    - Next
  + Highlight SLM\_OspiUI
    - Check "This project"
    - Next or Finish

So... This time the following were created (Don' think this is quite what happened last time but may be)

1. A testAA-test folder / project at same level as src which contains folders for:
   1. assets , bin, gen, res
   2. also src with following sub directories
      1. com
      2. slm-ospiui
      3. test
   3. not quite how I wanted the directory structure but okay...
   4. So Now I tried to create a JUnit Test Case under the test directory:
      1. Source folder = SLM-OspiUI.src
      2. Package com.slm.ospiui
      3. New JUnit 3 test checked
      4. Name: ProgSumListItemTestB
      5. super class: junit.framework.TestCase
      6. class under test -> Browse:: selected ProgSumListItem (from list)
      7. Next
      8. Selected getDays0String
      9. finish
   5. This seemed to create the test template for me.
   6. Problem is, This is confusing because I think that the 2 directory structures are linked to each other in some way and I don't like having both because I know it's going to confuse me later. Also, this new file is not created in the testAA project so are they linked or not??
   7. Also, a second com.slm.ospiui package was created. That's where I created ProgSumListItemTestB. So there are 2 packages with the name com.slm.ospiui. One is the normal src directory and one contains the test file. I can rename this package com.slm.ospiui.test and now there is a new directory under com.slm.ospiui called test which contains the new ProgSumListItemTestB.java file. This may have been what I did in Part 1. and then I moved that package to the tests directory that is at the same level as the src directory.
   8. Now can I delete the outside project testAA without removing the com.slm.ospiui.test directory??
2. testAA - A new project at level of SLM-OspiUI. This too is under version control (so that's good)
   1. This project also has a src dir and under that is the package:
      1. com.slm.ospiui.test -- this is the format that I like, however,
      2. When I tried to create a File -> New -> JUnit Test Case -- it couldn't reference the package that ProgSumListItem.java was in (I'm guessing since it is in a different package).
      3. I think whatever I did in Part 1 was easier than this even tough the project structure seems right. May be a way to fix this with class paths or build paths.
      4. \*\* BUT WAIT -- I looked at the other generated directory, the testAA-test folder that was created at the same time. I guess that ProgSumListItemTest.java file was also created in this directory when I created it in the testAA project. SO SEE

Just wondering if I leave out the name when I uncheck the "Use default location" if a default name/directory will be created.

So if I look at the directory structure in Sprinklers/AppWorkspace I have an SLM-OspiUI directory and a tests directory.

* tests/src/com/slm\_ospiui/test
  + IS EMPTY
* SLM-OspiUI has (among others)
  + src - Normal, but
    - under src/com/slm/ospiui the is a ProgSumListItemTestB.java
  + testAA-test
    - has a src/com/slm\_ospiui/test with ProgSumListItemTest.java
  + tests - I created as part of Part 1.
    - has a com/slm/ospiui/tests with ProgSumListItemTest.java
  + testsAA
    - IS EMPTY - this is the directory I self created before I started / created the Android Test Project. So it makes sense that it's empty.

So in Part 1, I may have tried to import project from existing code or something but not sure it worked.

Also, remember the stackoverflow saying to rename the xTestTest directory to test and do something, but not sure that worked either.

So I'm now going to delete the testsAA project (the separate project that is empty) and see what happens:

* Got error saying that some of the contents of project testsAA was not synced with ?? I just hit continue and selected that all elements should be removed from disk knowing that this operation cannot be undone.
  + Separate Project testsAA and testAA-test directory were removed. But, still have the com.slm.ospiui.test package so all is good I think.

So I put this in the code as a comment on How to Run the JUnit tests but I'll add it to this file as well:

/\*

\* To run this test (which is in SLM-OspiUI/tests, package name = com.slm.ospiui.tests) ProgSumListItemTest

\* - Right click on folder name tests

\* - Select Run As JUnit Test

\* - Check "use configuration specific settings" and

\* - select "Android JUnit Test Launcher" then

\* - Click OK

\* Should run all tests.

\*

\* OR

\* - Right click on package com.slm.ospiui.test

\* - Select Run As JUnit Test

\* - Check "use configuration specific settings" and

\* - select "Android JUnit Test Launcher" then

\* - Select any and all tests you wish to run and

\* - Click OK

\* Should run all selected tests.

\*/

\*\*\*\*\*\*\*

A few websites:

http://developer.android.com/tools/testing/testing\_android.html#TestProjectPaths

http://developer.android.com/tools/testing/testing\_eclipse.html

http://stackoverflow.com/questions/5395216/creating-an-android-test-project-in-eclipse