

SE 420-01 Mini-Project 3 - Issue Tracking with Eclipse  
Spring 2015  
Sean Holden, Casey Layne, Chris Silvano, Alex Spradlin

**Youtube Video Tutorial:** <https://www.youtube.com/watch?v=ws1-6Q2DQP4>

**Tool features:**

- Supports creating, updating, commenting on, searching, and monitoring issues from within Eclipse
- Supports importing and exporting local tasks into and out of Eclipse
- Provides a task-focused interface, allowing developers to focus in on the code related to unresolved issues
- Monitors the user's activities to identify relevant information
- Allows users to connect to bug tracking software such as Github, Jira, or Bugzilla

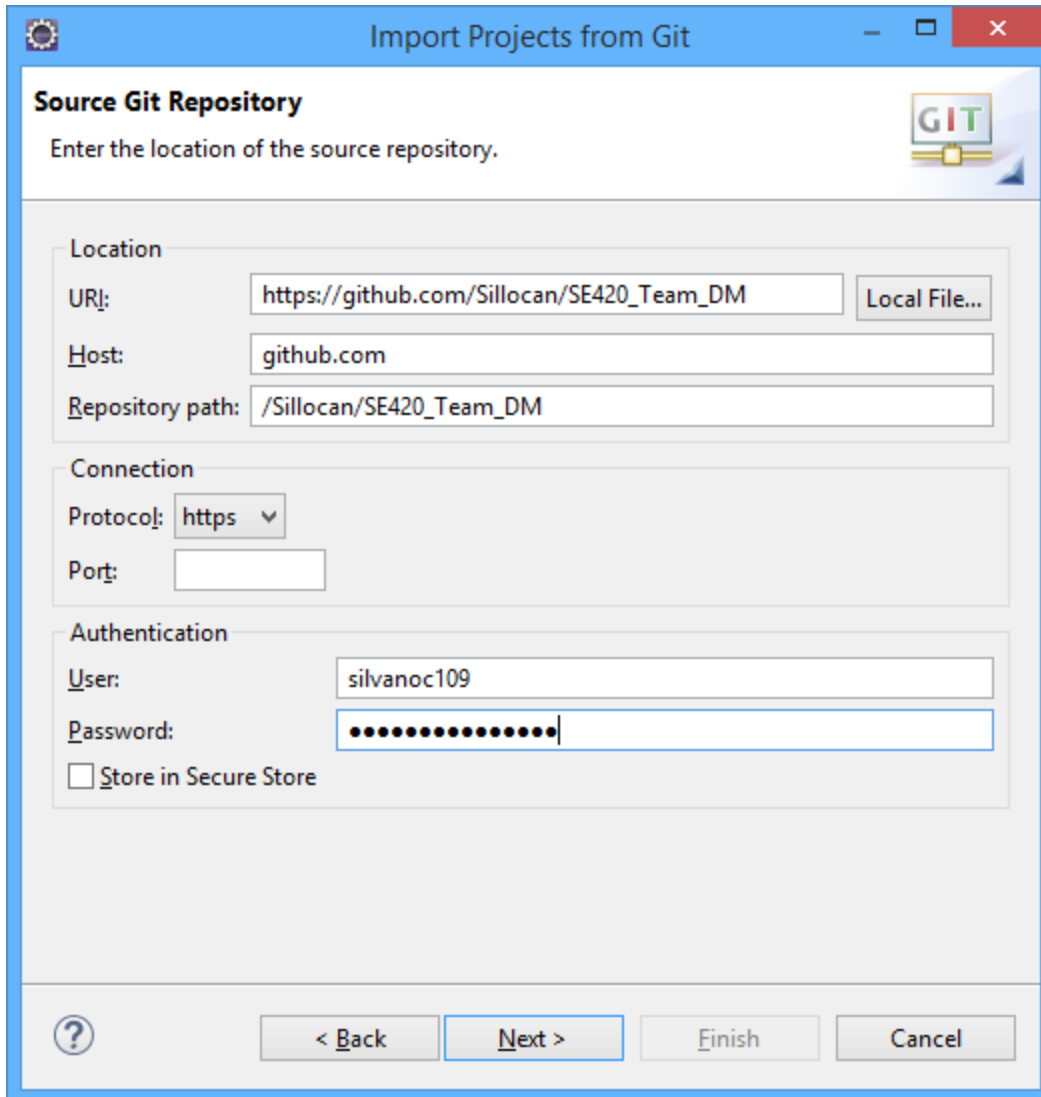
**Installation Instructions - GitHub Mylyn Connector:**

1. In Eclipse, go to Help > Install New Software
2. In the 'Work with' box input the following link:  
<http://download.eclipse.org/egit/github/updates/>
3. Under the search results, tick the checkbox labeled 'Collaboration' and hit Next
4. Select Next when the Install Details screen appears
5. Accept the Eclipse Foundation Software User Agreement and hit Finish
6. Allow the software to install within Eclipse, this may take a few minutes
7. When prompted, select Yes when asked if you want to restart Eclipse
8. Wait for Eclipse to restart
9. When Eclipse restarts, if no error messages are displayed, the plugin was installed successfully. If any errors occur, validate that the installation was performed correctly using these instructions

## User Manual:

### Getting Started - Pull from Repository

1. Go to File > Import > Git > Projects from Git
2. Select Clone URI
3. Use our repository and your credentials for this part:



**Import Projects from Git**

**Source Git Repository**  
Enter the location of the source repository.

**Location**

URI:

Host:

Repository path:

**Connection**

Protocol:  ▼

Port:

**Authentication**

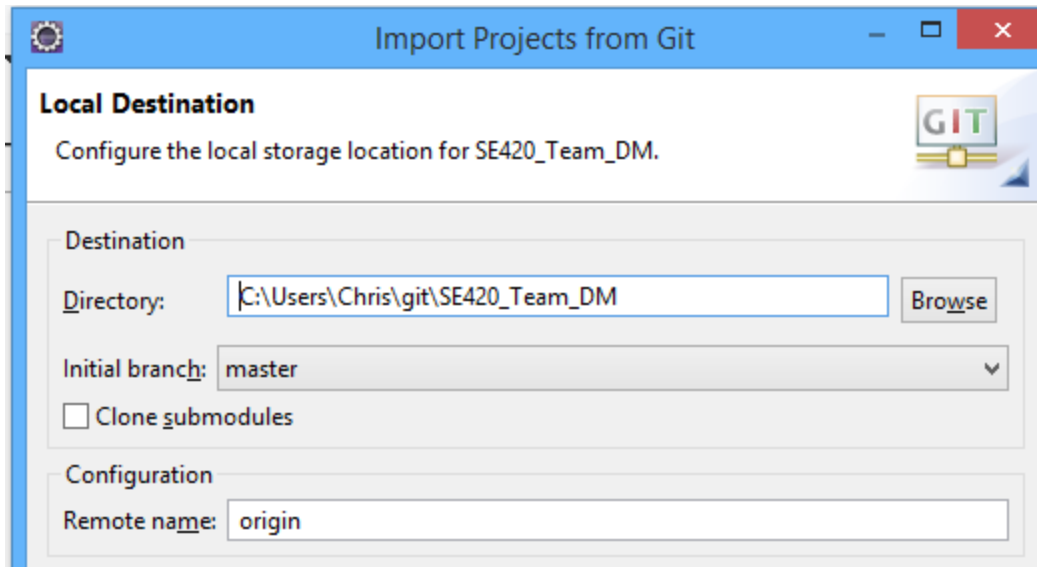
User:

Password:

☐ Store in Secure Store

4. Select the "master" branch then choose Next

5. Use your preferred directory, but leave Initial branch and Remote name the same:



**Import Projects from Git**

**Local Destination**  
Configure the local storage location for SE420\_Team\_DM.

**Destination**

Directory:

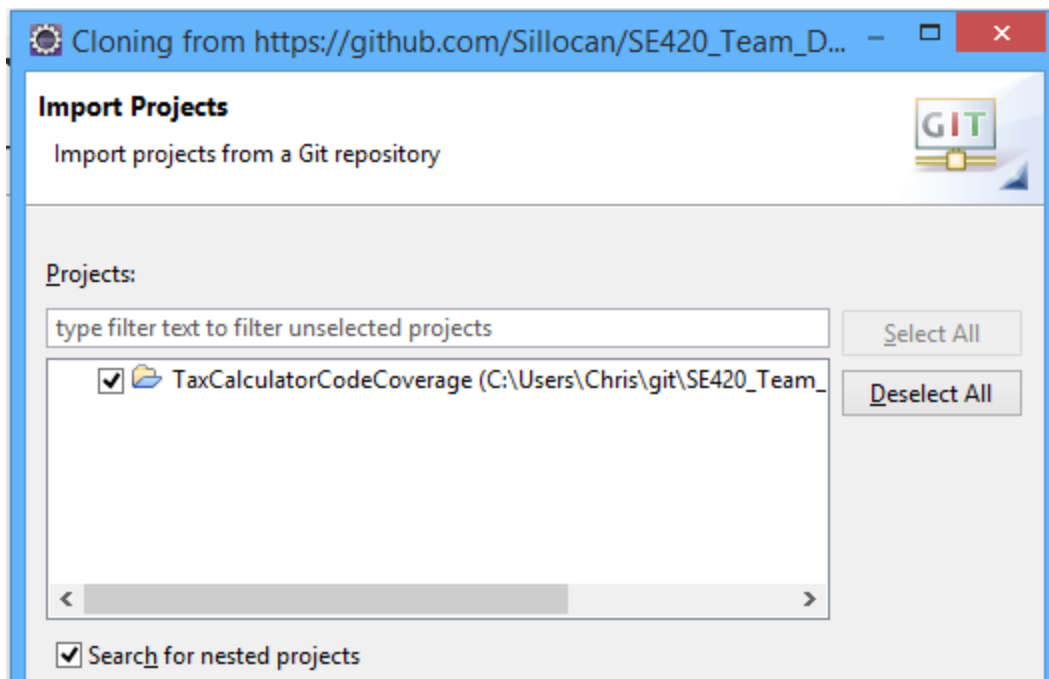
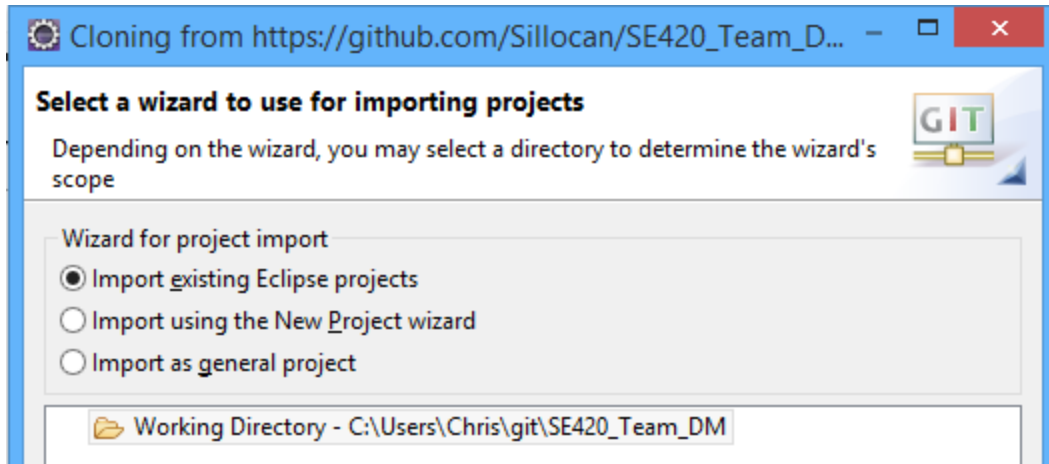
Initial branch:

☐ Clone submodules

**Configuration**

Remote name:

6. We'll check out the TaxCalculatorCodeCoverage project on our GitHub with this example:

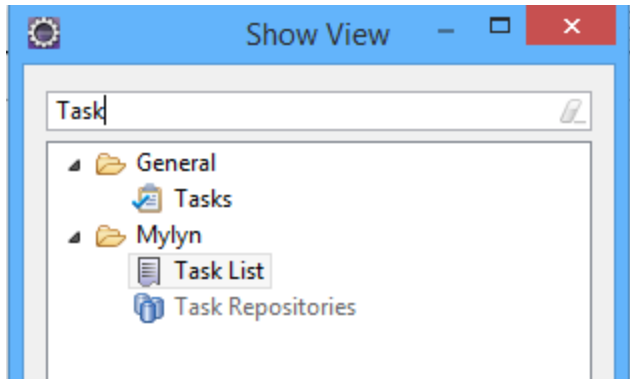


7. If successful, your package explorer should have something like this:

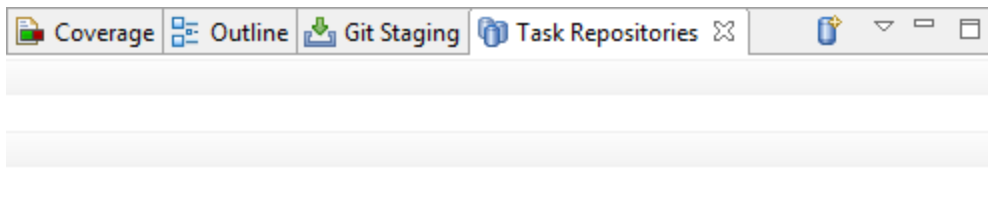


### Using GitHub Issues with Eclipse

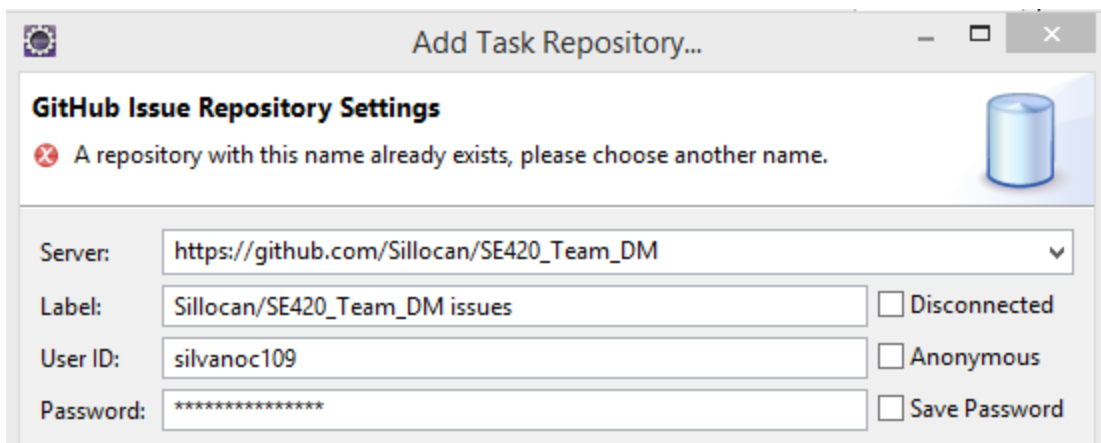
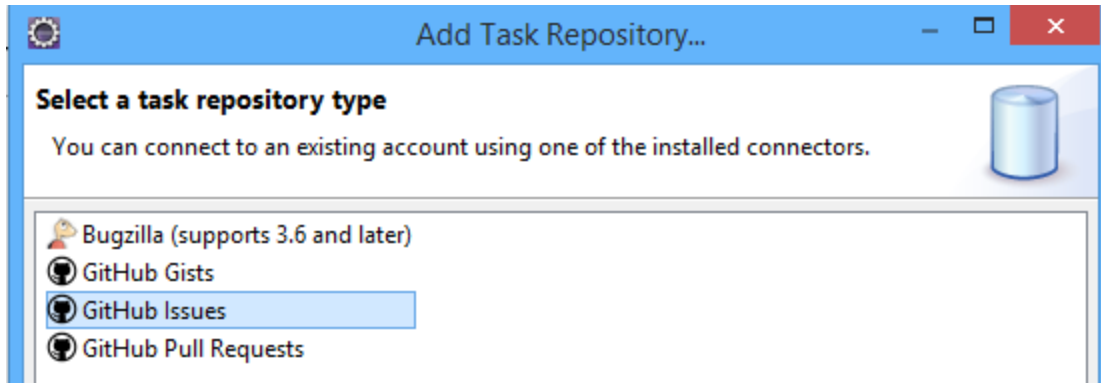
1. Install the GitHub Mylyn Connector
2. Make sure your Task List is visible  
(Window > Show View > Task List)
3. Make sure your Task Repositories tab is visible  
(Window > Show View > Search under Mylyn)



4. Select Add Task Repository under the Task Repositories tab:  
(Button next to drop down arrow)

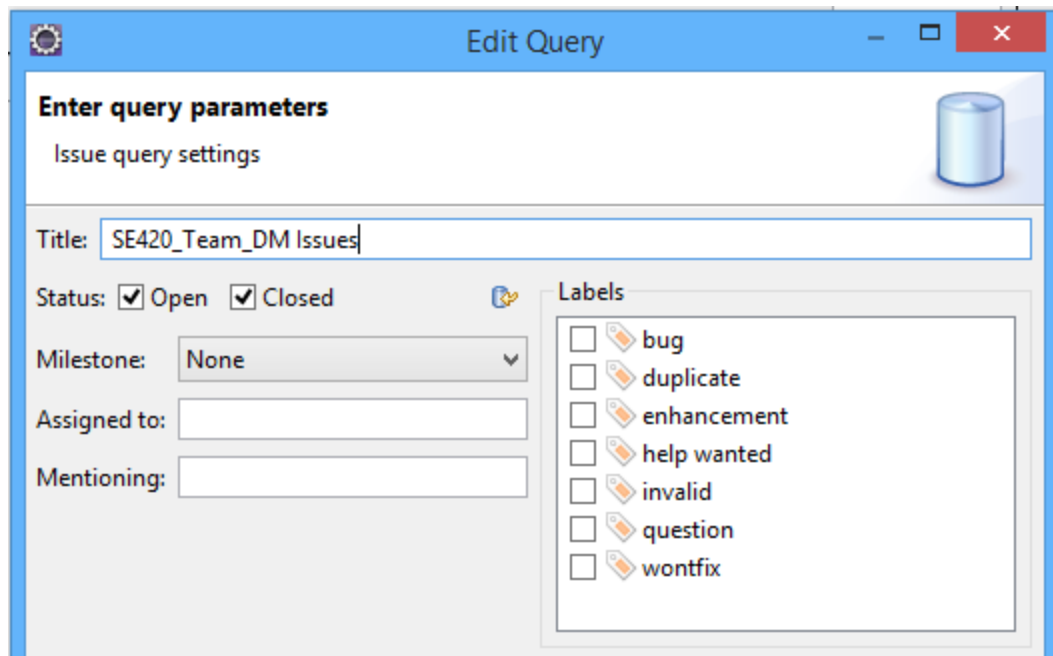
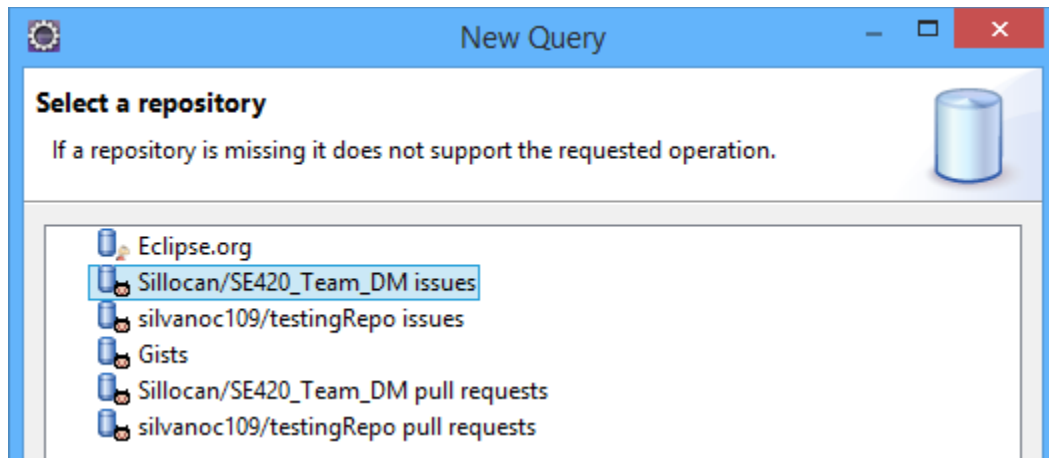


5. Add the Task Repository:

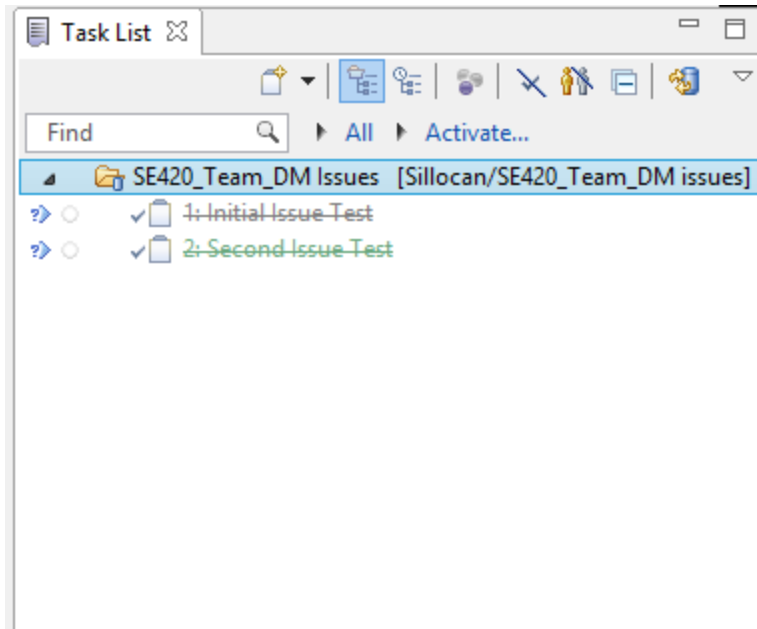


### Managing Tasks in Our Repository

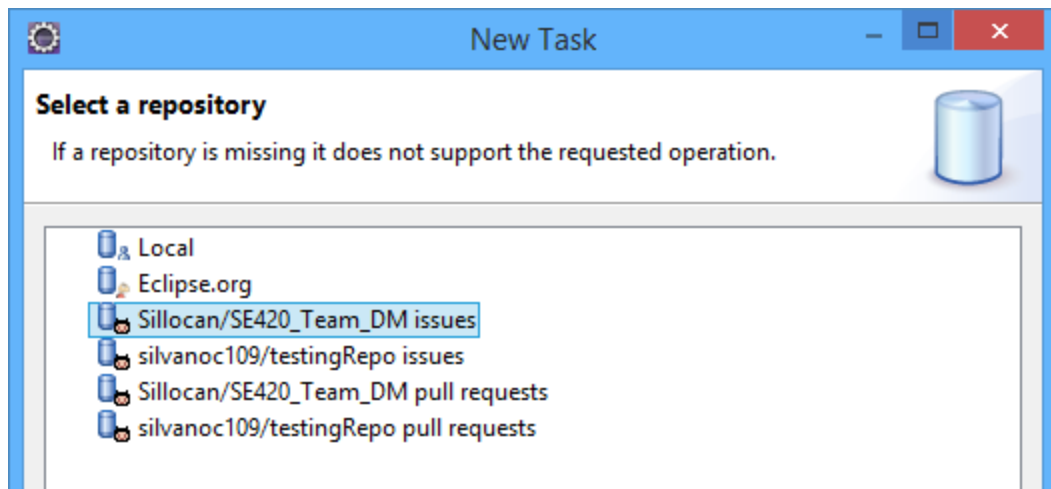
1. Create a query of the tasks you want to see:  
(Right Click in the Task List > New > Query)
2. Query our repository and create a title for the query:



3. If successful, you will see this in the Task List:



4. You can create a new issue using the Task List and add issues to our repository:  
(First button on left of the Task List spawns this window)



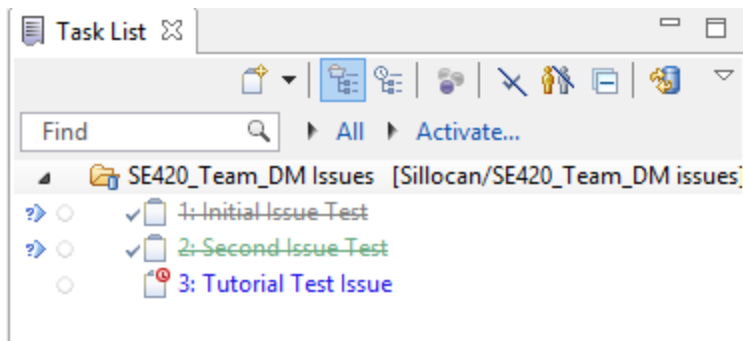


5. Report the issue, categorize it with a label, assign it (if applicable), then hit ***SUBMIT***:

The screenshot shows a 'New Task' form with the following fields and sections:

- Issue:** A header bar with the text 'Sillocan/SE420\_Team\_DM issues' and a 'Submit' button.
- Title:** A text input field containing 'Tutorial Test Issue'.
- Attributes:** A section with a 'Milestone' dropdown set to 'None' and 'Labels' with a green plus icon and the text 'question'.
- Private:** A section with 'Scheduled' set to 'This Week', 'Due' set to 'Feb 24, 2015 6:', and 'Estimate' set to '0'.
- Description:** A large text area containing the text 'Will this work for the tutorial?'.
- Actions:** A section with a checkbox 'Add to Category' set to 'Uncategorized' and a 'Submit' button.
- People:** A section with an 'Assignee' dropdown set to 'silvanoc109'.
- Context:** A footer bar with a 'Context' dropdown set to 'GitHub'.

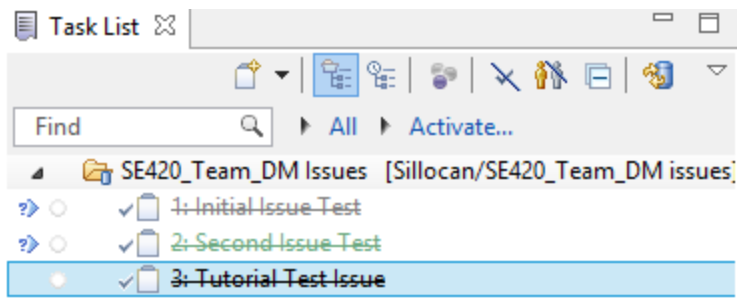
6. Check your Task List:



7. In an issue, ask questions and **leave the issue open or close the issue** when it is resolved:  
(Don't forget to hit **SUBMIT**)

The screenshot shows a Jira issue page for 'Tutorial Test Issue' (Issue 3) under the project 'Sillocan/SE420\_Team\_DM issues'. The issue status is 'open', created on Feb 24, 2015 at 5:09 PM, and last modified on the same date. The reporter and assignee are both 'silvanoc109'. The page includes sections for 'Attributes', 'Private' (set to 'This Week'), 'Description', and 'Comments' (currently empty). A 'New Comment' section is visible with the text 'Yes! I believe that this will work for the tutorial! Closing issue!'. At the bottom left, the 'Actions' menu is highlighted with a red box, showing options to 'Leave open' (unselected) or 'Close' (selected), with a 'Submit' button below. The bottom of the page shows 'Context' and 'GitHub' tabs.

8. Check your Task List:



9. To check on new issues, right click on the query folder you created and hit Synchronize

10. Issues can also be viewed on our GitHub repository:

