**Git version control with Eclipse**

**Table of Contents**

1. [Installation of Git support into Eclipse](http://www.vogella.com/tutorials/EclipseGit/article.html#installation-of-git-support-into-eclipse)
2. [Git user settings in Eclipse](http://www.vogella.com/tutorials/EclipseGit/article.html#git-user-settings-in-eclipse)
3. [Configuring the toolbar and the menu for Git usage](http://www.vogella.com/tutorials/EclipseGit/article.html#configuring-the-toolbar-and-the-menu-for-git-usage)
4. [Working with Eclipse projects in a Git repository](http://www.vogella.com/tutorials/EclipseGit/article.html#working-with-eclipse-projects-in-a-git-repository)

[4.1. Adding a new project to a local Git repository](http://www.vogella.com/tutorials/EclipseGit/article.html#adding-a-new-project-to-a-git-repository)

[4.2. Adding a new project to a remote Git repository](http://www.vogella.com/tutorials/EclipseGit/article.html#adding-a-new-project-to-a-git-repository)

[5. Import (clone) projects from a remote GIT repository](http://www.vogella.com/tutorials/EclipseGit/article.html#import-projects-from-an-existing-repository)

[6. Performing Git operations in Eclipse](http://www.vogella.com/tutorials/EclipseGit/article.html#performing-git-operations-in-eclipse)

[6.1. Pull, push and fetch](http://www.vogella.com/tutorials/EclipseGit/article.html#pull-push-and-fetch)

[6.2. Basic team operations](http://www.vogella.com/tutorials/EclipseGit/article.html#basic-team-operations)

[6.3. Team operations available on the project](http://www.vogella.com/tutorials/EclipseGit/article.html#team-operations-available-on-the-project)

[6.4. Amending a commit](http://www.vogella.com/tutorials/EclipseGit/article.html#amending-a-commit)

[7. Branching in Eclipse](http://www.vogella.com/tutorials/EclipseGit/article.html#branching-in-eclipse)

[8. Starting a merge operation in Eclipse](http://www.vogella.com/tutorials/EclipseGit/article.html#starting-a-merge-operation-in-eclipse)

[8.1. Merge](http://www.vogella.com/tutorials/EclipseGit/article.html#merge)

[8.2. Solving merge conflicts](http://www.vogella.com/tutorials/EclipseGit/article.html#solving-merge-conflicts)

[9. Rebasing a branch onto another branch](http://www.vogella.com/tutorials/EclipseGit/article.html#rebasing-a-branch-onto-another-branch)

[10. Git reset and Git reflog](http://www.vogella.com/tutorials/EclipseGit/article.html#git-reset-and-git-reflog)

[10.1. Moving the branch pointer with Git reset](http://www.vogella.com/tutorials/EclipseGit/article.html#moving-the-branch-pointer-with-git-reset)

[10.2. Finding "invisible" commits with the Reflog view](http://www.vogella.com/tutorials/EclipseGit/article.html#finding-invisible-commits-with-the-reflog-view)

[11. Using git cherry-pick](http://www.vogella.com/tutorials/EclipseGit/article.html#using-git-cherry-pick)

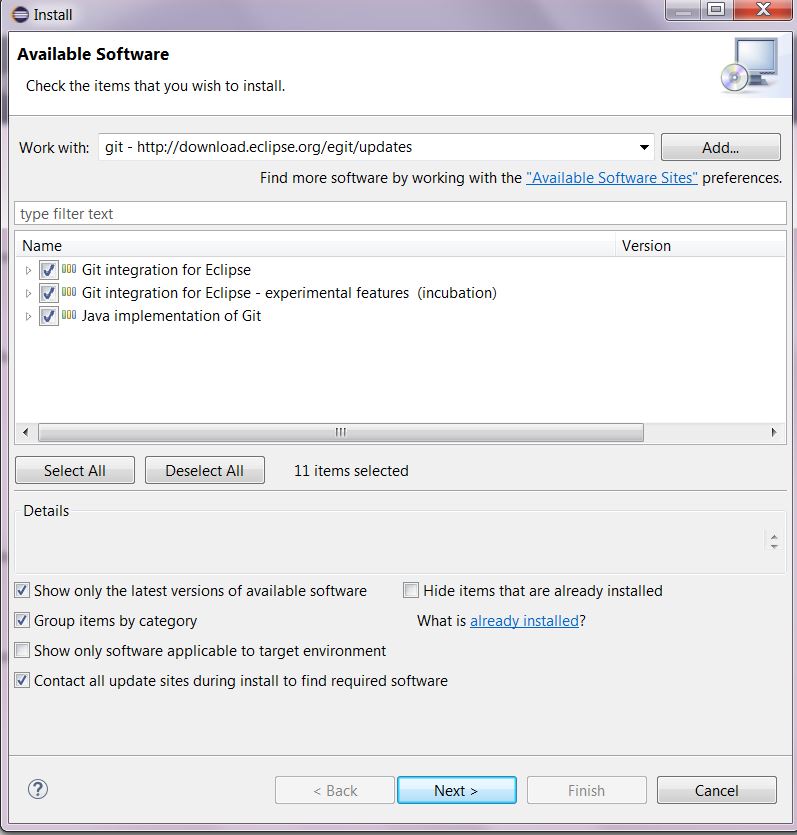
[12. Creating patches](http://www.vogella.com/tutorials/EclipseGit/article.html#creating-patches)

**1.** [**Installation of Git support into Eclipse**](http://www.vogella.com/tutorials/EclipseGit/article.html#installation-of-git-support-into-eclipse)**:**

* Most Eclipse IDE distributions from Eclipse.org already contain support for Git. In this case, no additional installation is required.
* Otherwise, you can install it via the Eclipse installation manager.
* Select the **Help ▸ Install new Software… ▸ menu entry**.

Enter the following update site URL:

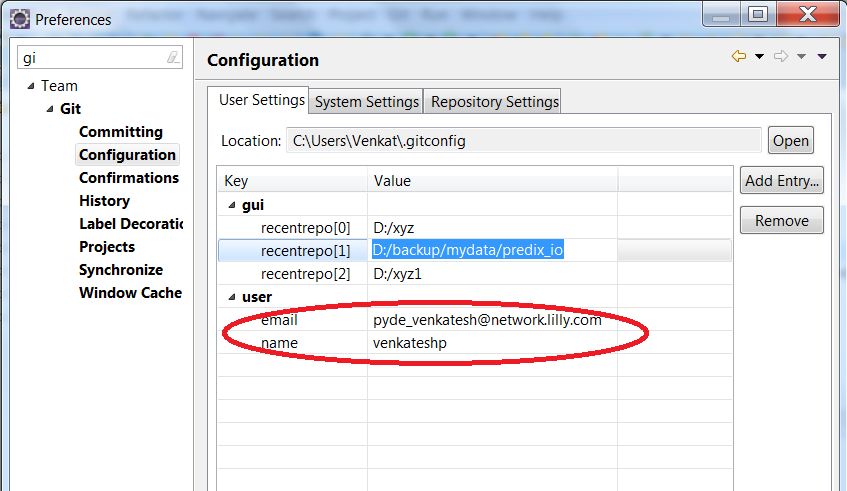
<http://download.eclipse.org/egit/updates>



1. [**Git user settings in Eclipse**](http://www.vogella.com/tutorials/EclipseGit/article.html#git-user-settings-in-eclipse)

* To use Git, you must configure your full name and email address.
* This information is used to fill the author and committer information of commits you create.
* These Git configuration settings can be adjusted via the Eclipse preference setting.

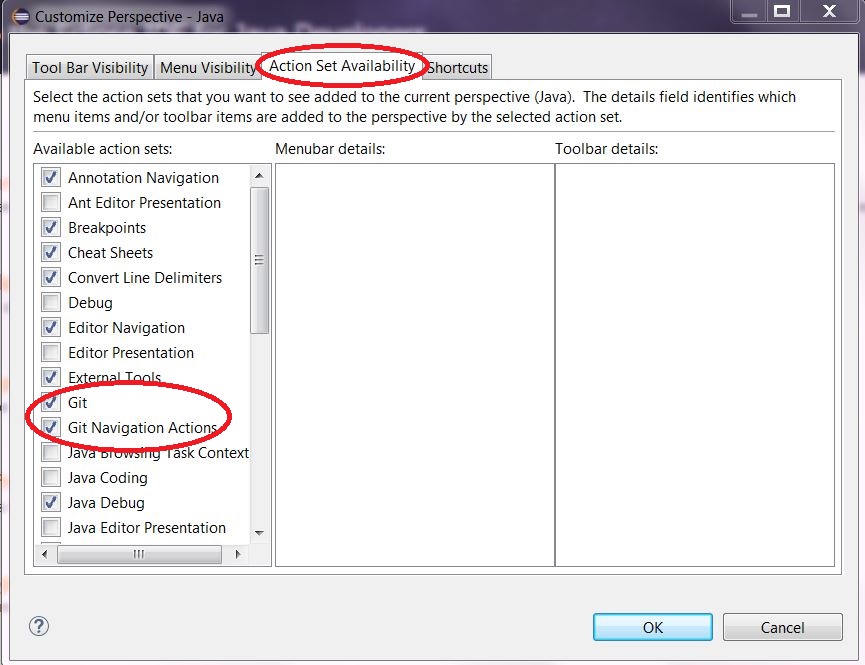
Select***Window ▸ Preferences ▸ Team ▸ Git ▸ Configuration***to see the current configuration and change email address and username.

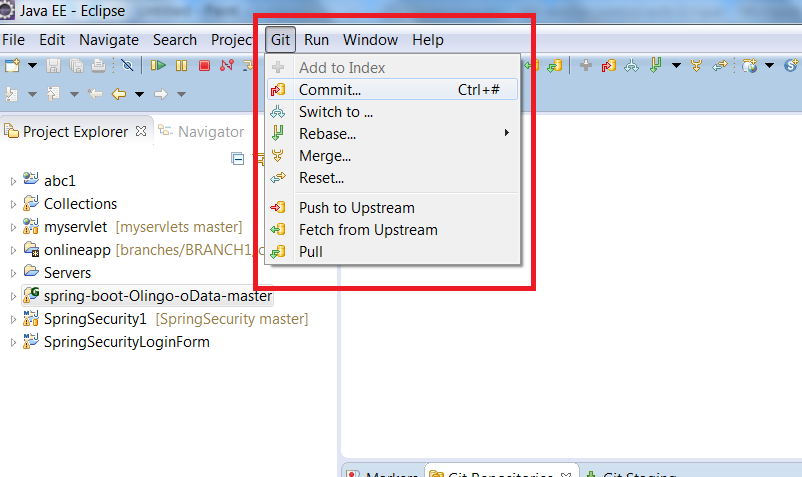


1. [**Configuring the toolbar and the menu for Git usage**](http://www.vogella.com/tutorials/EclipseGit/article.html#configuring-the-toolbar-and-the-menu-for-git-usage)

* To simplify access to the common Git operations you can activate the Git toolbar.

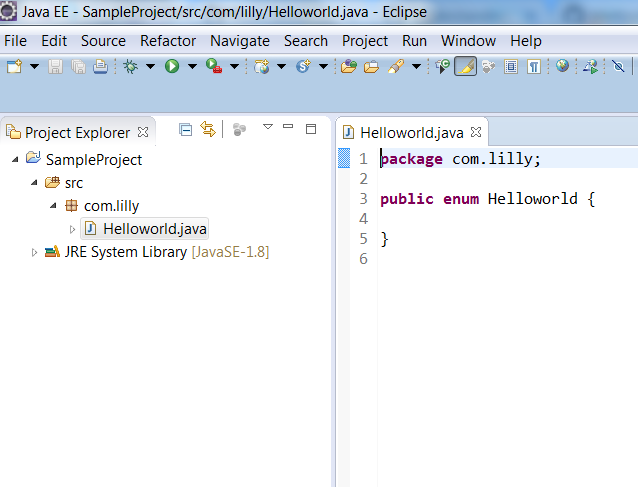
For this, select***Window ▸ Perspective ▸ Customize perspective*…​**and check the***Git* and *Git Navigation Actions* entries** in the***Action Set Availability*** tab.





1. [**Working with Eclipse projects in a Git repository**](http://www.vogella.com/tutorials/EclipseGit/article.html#working-with-eclipse-projects-in-a-git-repository)
   1. [**Adding a new project to a local Git repository**](http://www.vogella.com/tutorials/EclipseGit/article.html#adding-a-new-project-to-a-git-repository)**:**

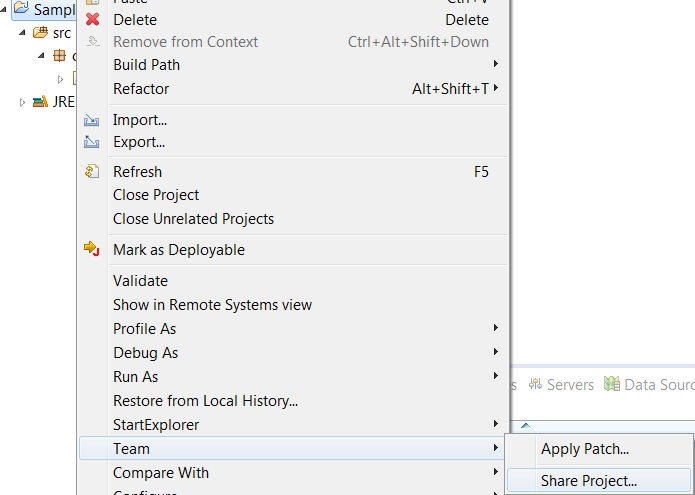
* Create sample project and move it to git remote repository.



To move above project into GIT remote repository, we need to follow below steps.

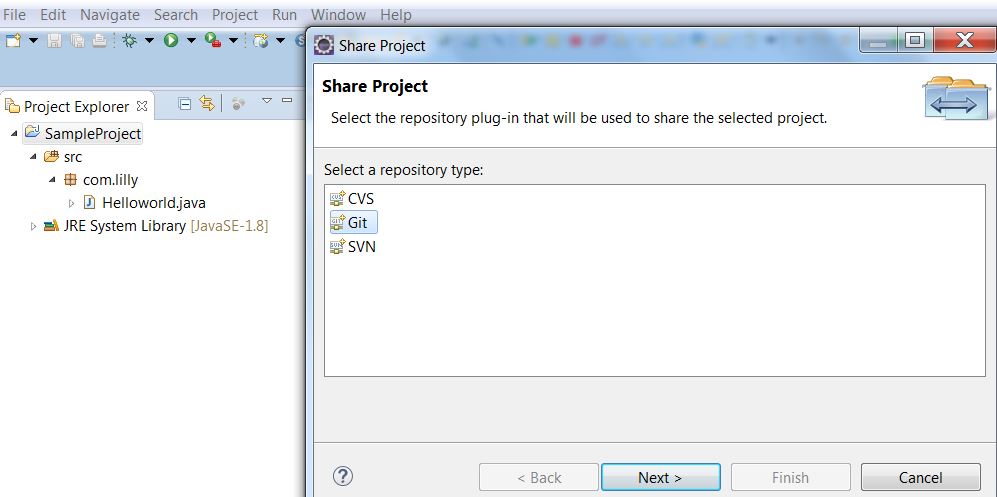
Step 1:

* Right click on ***project ▸ Team ▸ Share project***



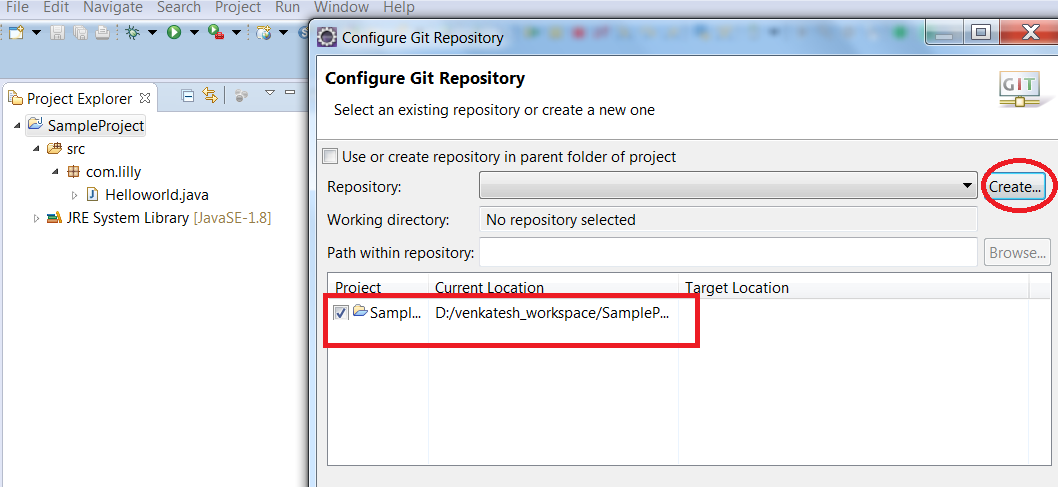
Step 2:

* Select ***GIT*** and click on ***Next button.***



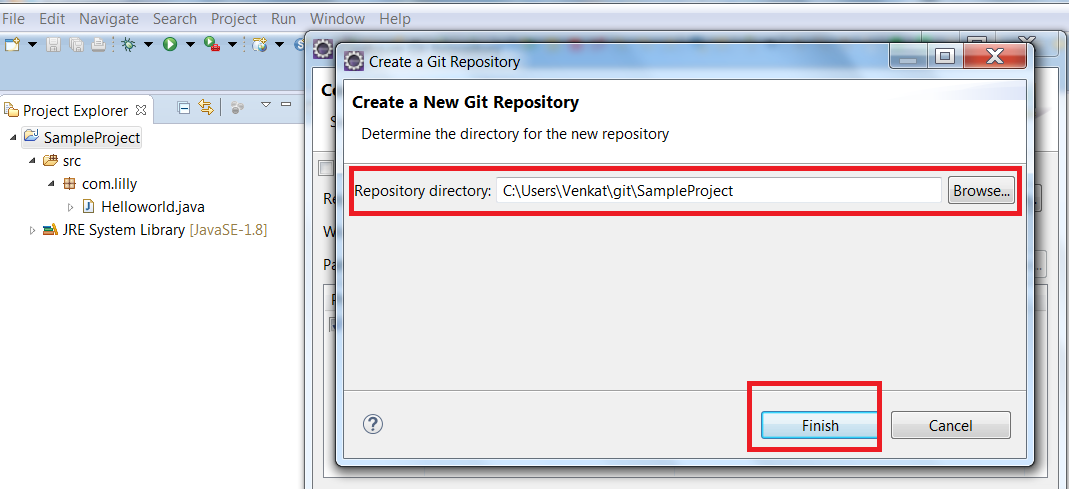
Step 3:

* Click on ***create button*** to provide local repository path.



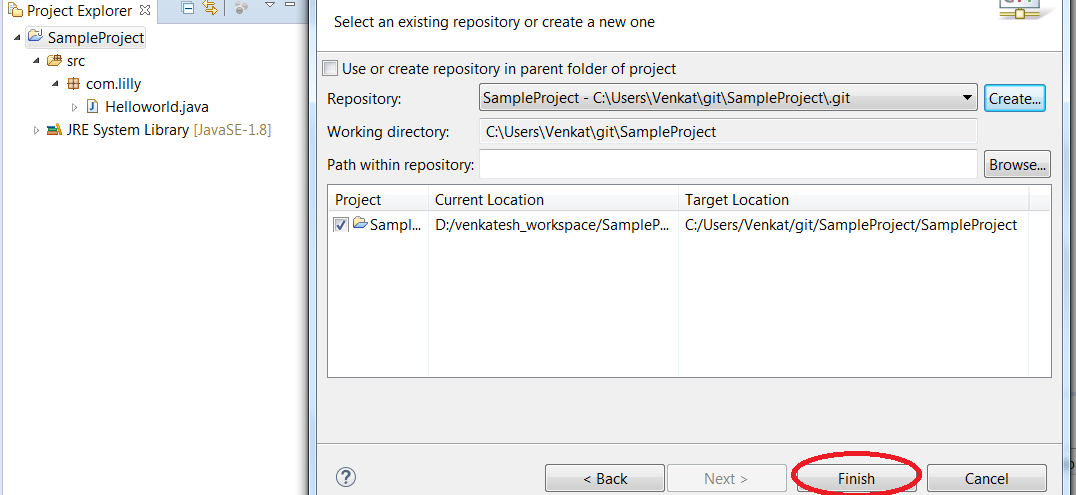
Step 4:

* Browse local repository location and click on ***finish***.
* Local repository can be any path in local system.



Step 5:

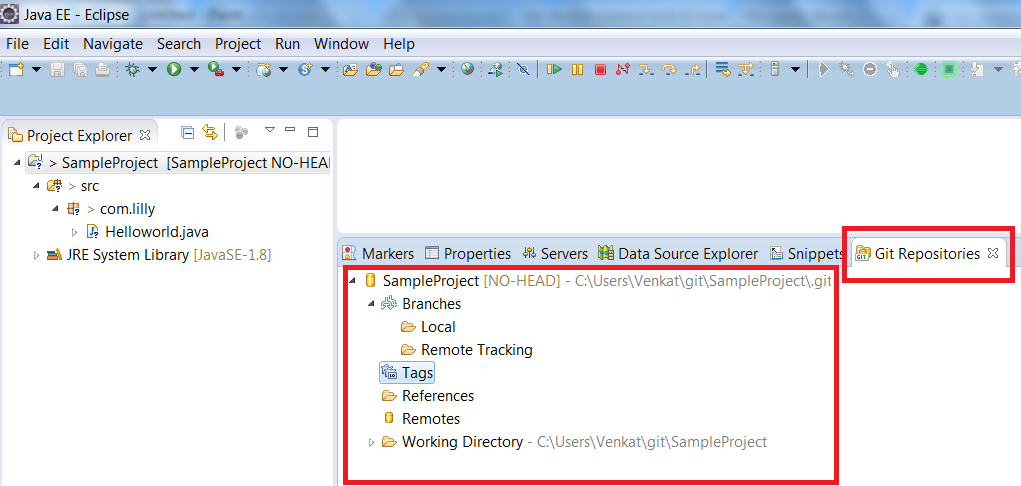
* Click on ***finish button*** to create local repository.



Step 6:

* **Open git repositories to view the project in local repository.**

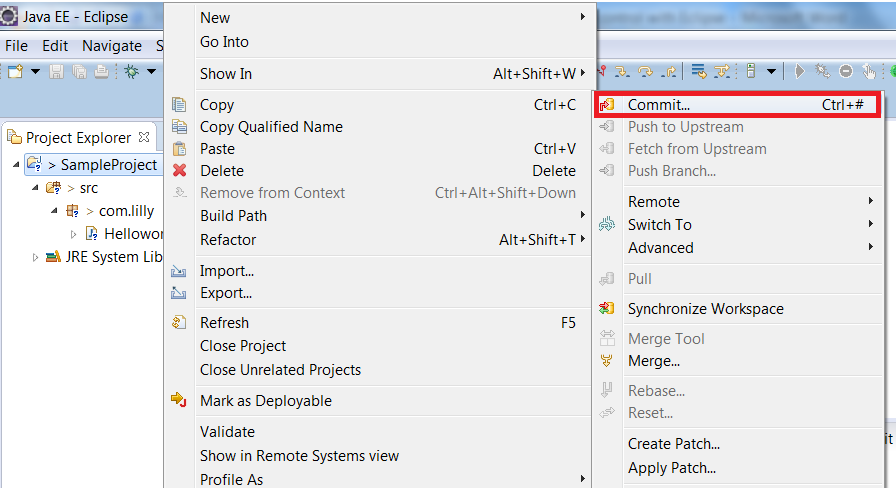
***Windows ▸ show view ▸ other... ▸ expand git ▸ select git repositories***



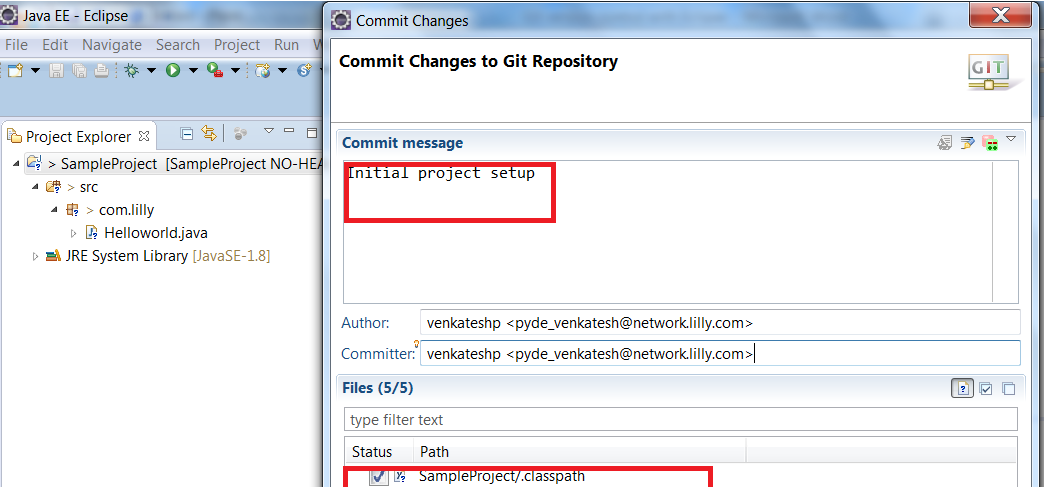
Step 7:

* Now move complete project from eclipse to local repository and to we need to follow below step.

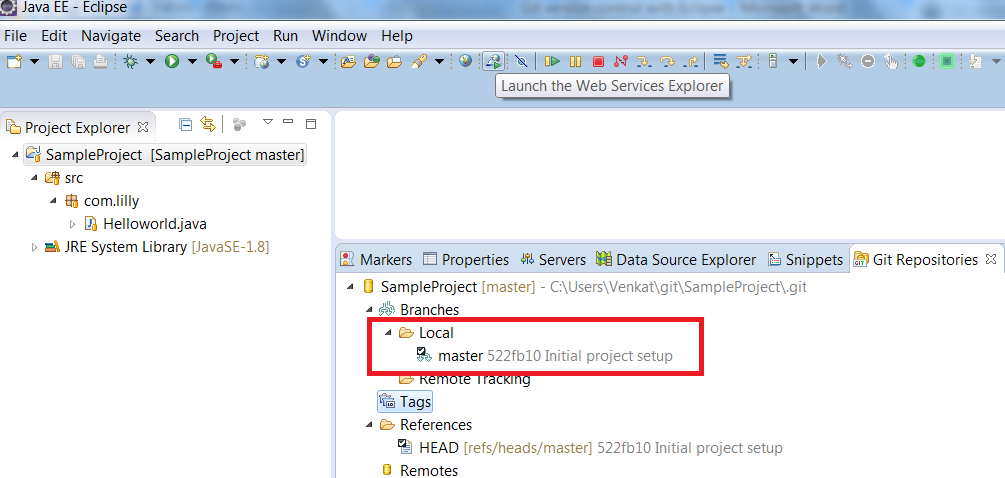
Right clickon ***Project ▸ Team ▸ Commit***



**Select all files, enter appropriate comments, and finally click on commit button.**



* Now verify in Git repositories view for local repo project.

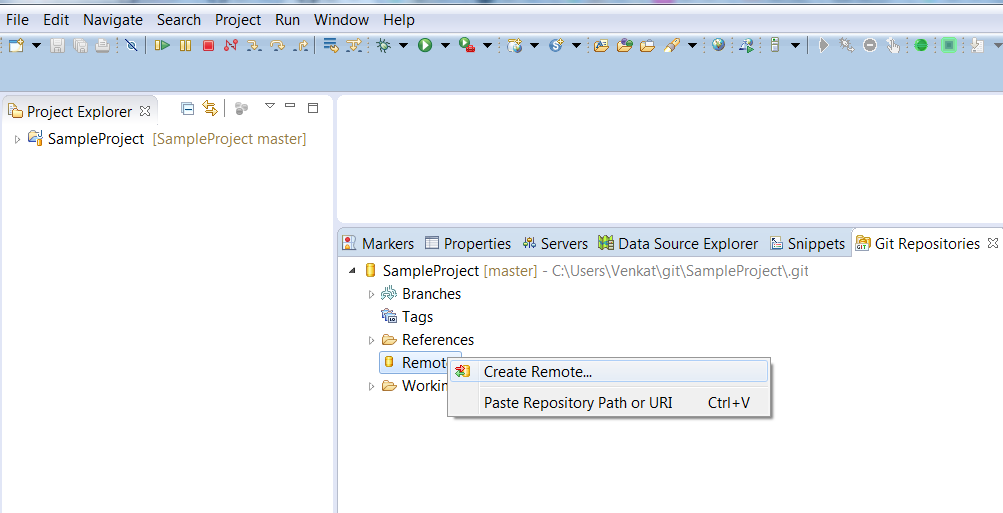


* 1. [**Adding a new project to a remote Git repository**](http://www.vogella.com/tutorials/EclipseGit/article.html#adding-a-new-project-to-a-git-repository)
* To add the above project into GIT remote repository we need to follow below steps,
* Before doing this we need remote repository location. Here is my remote repository location URL.

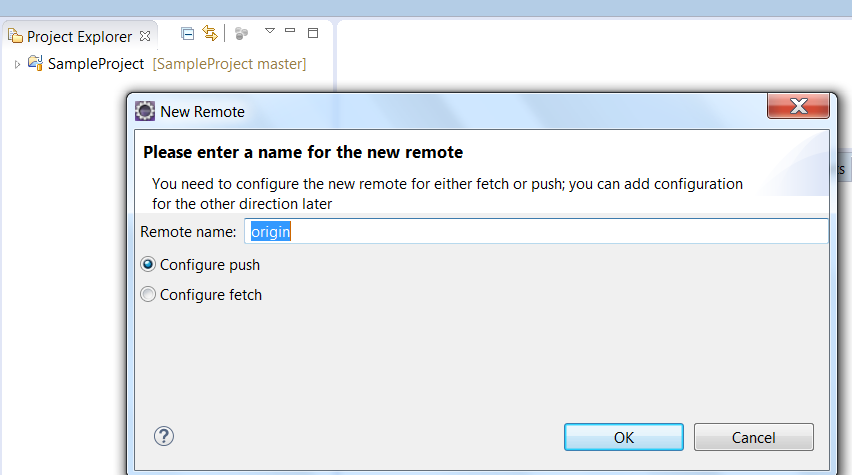
<https://github.com/venkateshpyde/SampleProject.git>

Step 1:

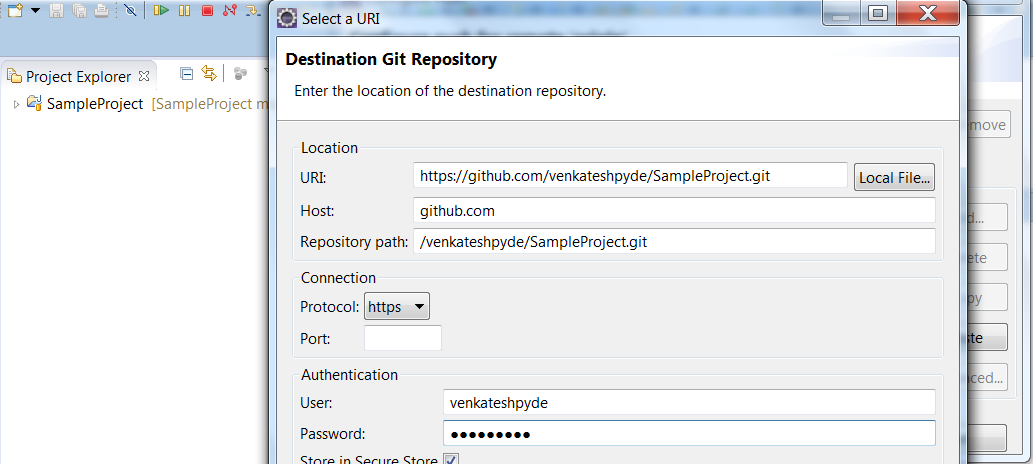
* Right click on ***remote*** option in Git Repositories view and select “***Create Remote”***



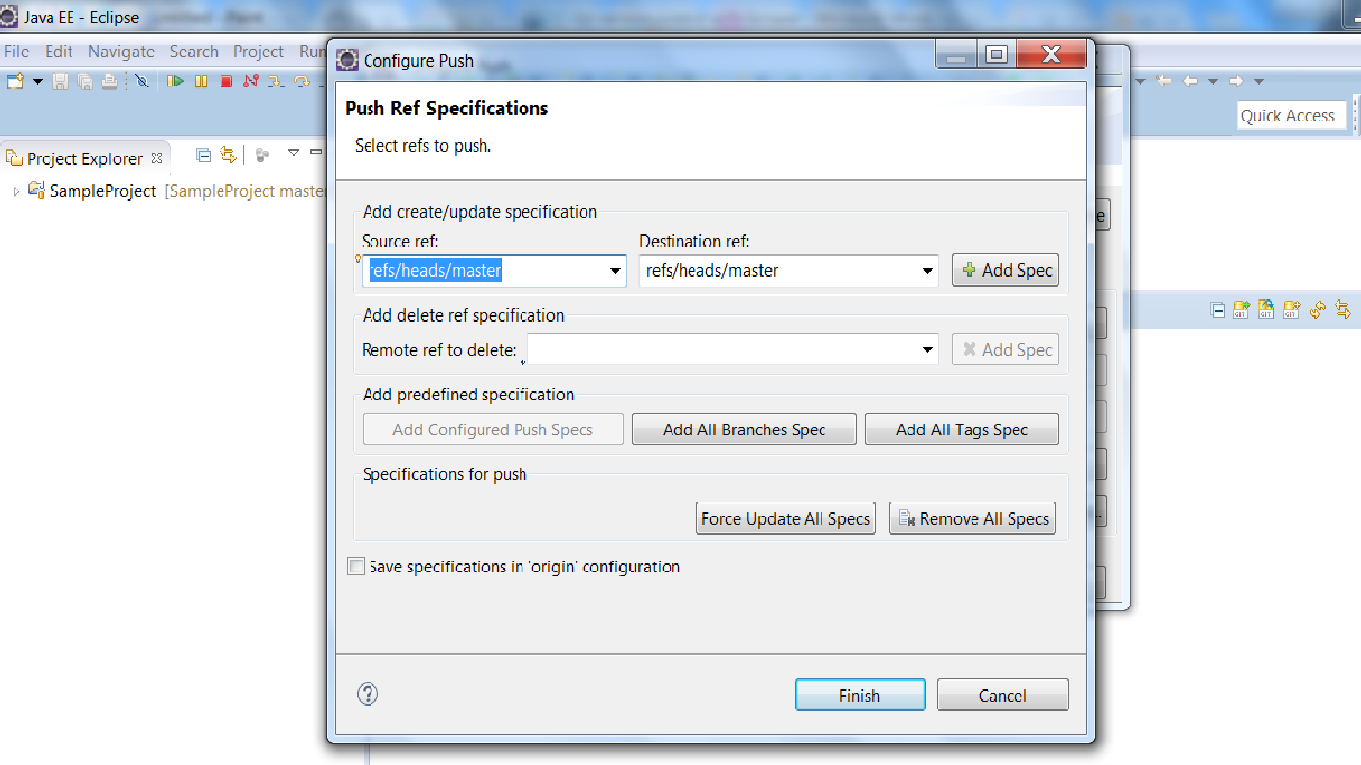
* Use default name as **origin** for remote repository and select “***Configure push***” radio button and click on ok.



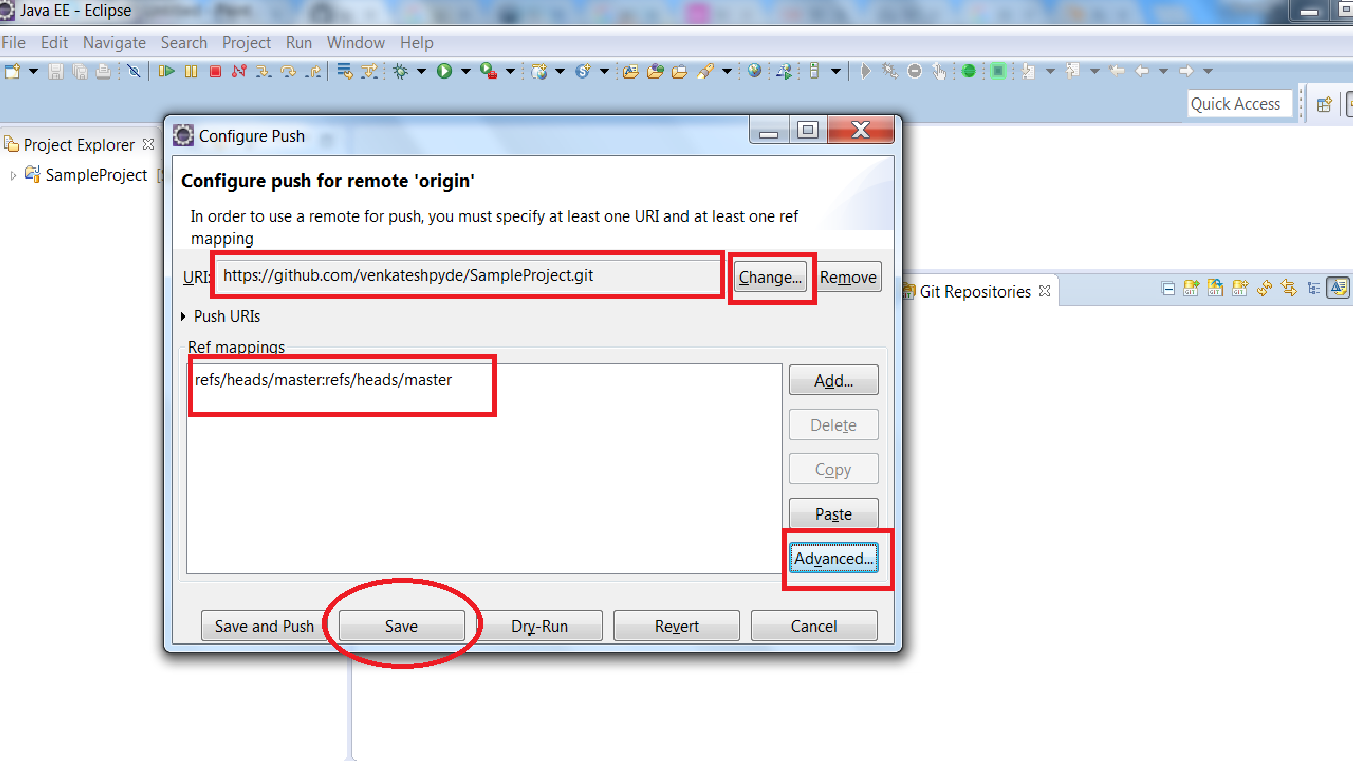
* Click on ***change button*** and provide ***GIT URL***, ***username and password***, ***enable Store in Secure store*** check box to save password in eclipse and click on finish button.



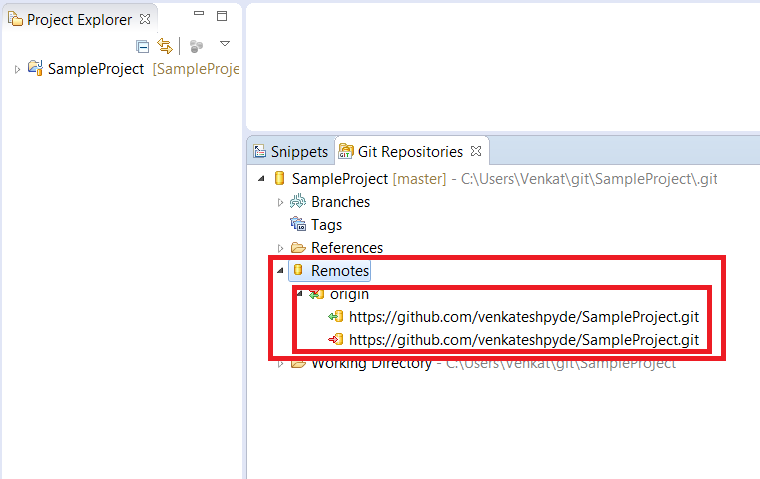
* Click on ***Advanced button***, **Select Source ref** as ***master [branch]*** and **Destination ref** as ***refs/heads/master*** and Click on ***AddSpec*** and finally click on ***finish*** button.



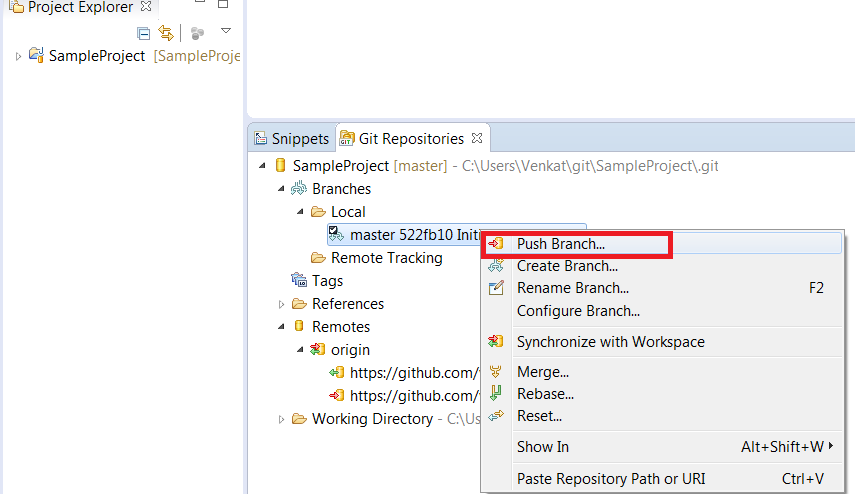
* Click on ***Save*** button to save configurations.



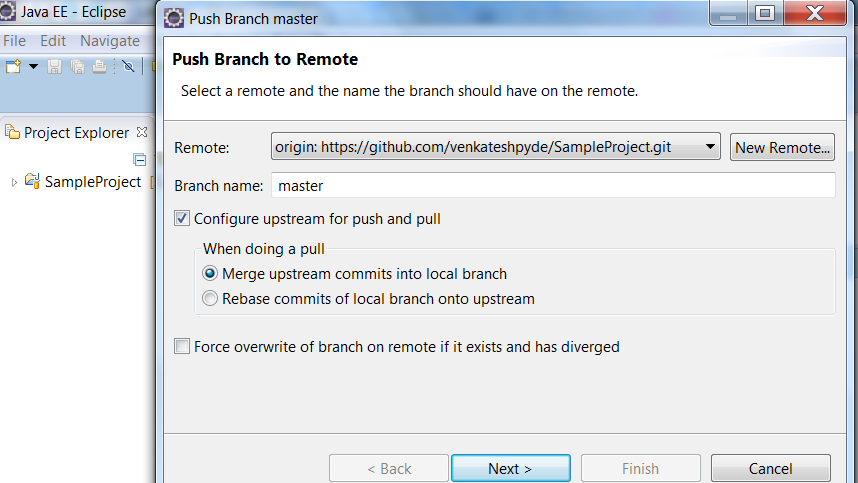
* Now check remote option in GIT repositories view for remote git repository location.



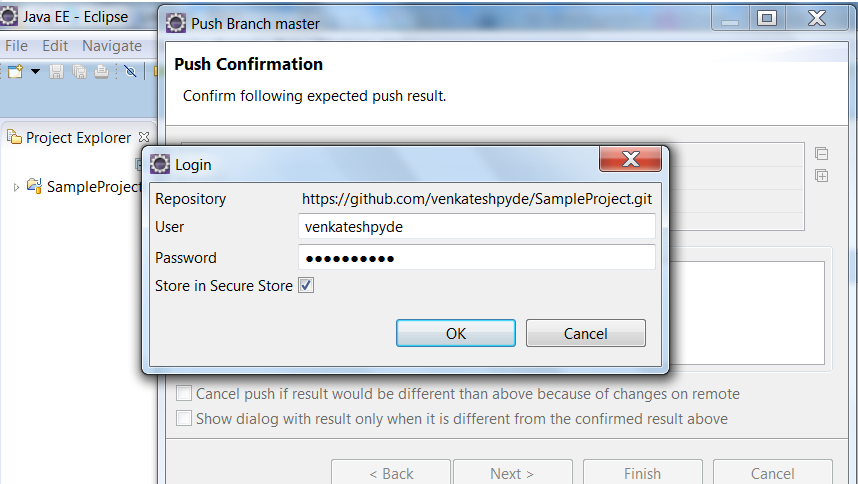
* Now push local repository project to remote repository.
* Right click on ***master*** under local and select ***Push branch***.



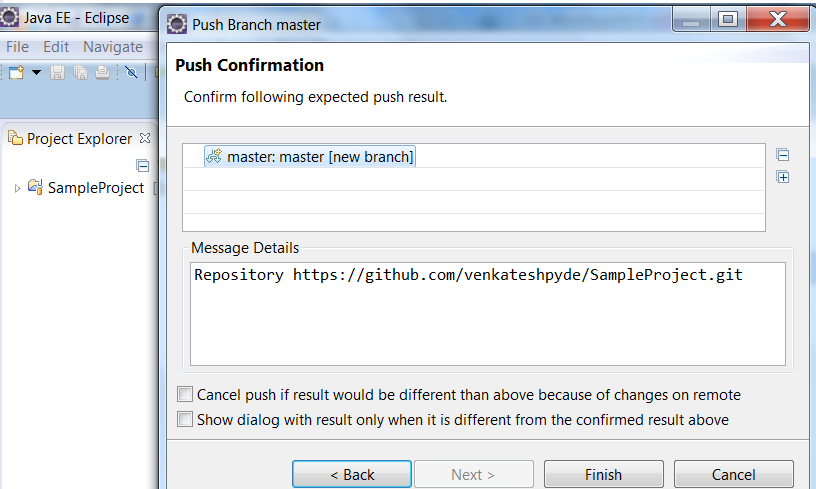
* Click on ***Next.***



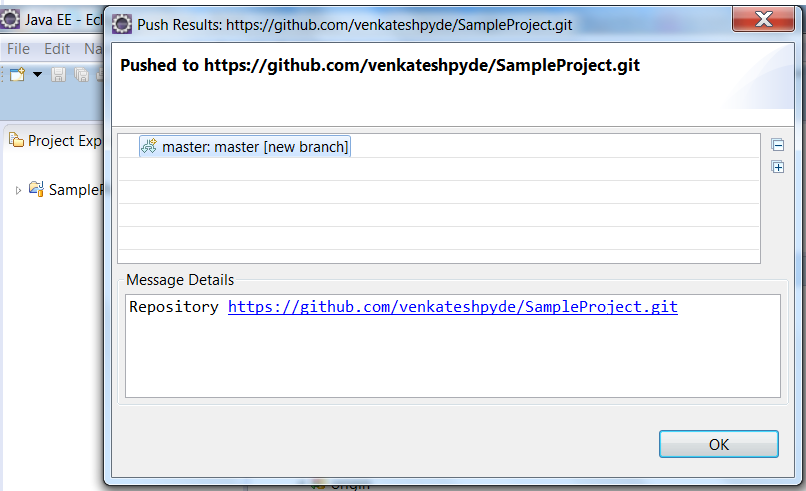
* Enter ***username and password***, enable ***Store in Secure store***, and click on ***Ok.***



* Click on ***Finish***.



* Click on ***OK.***

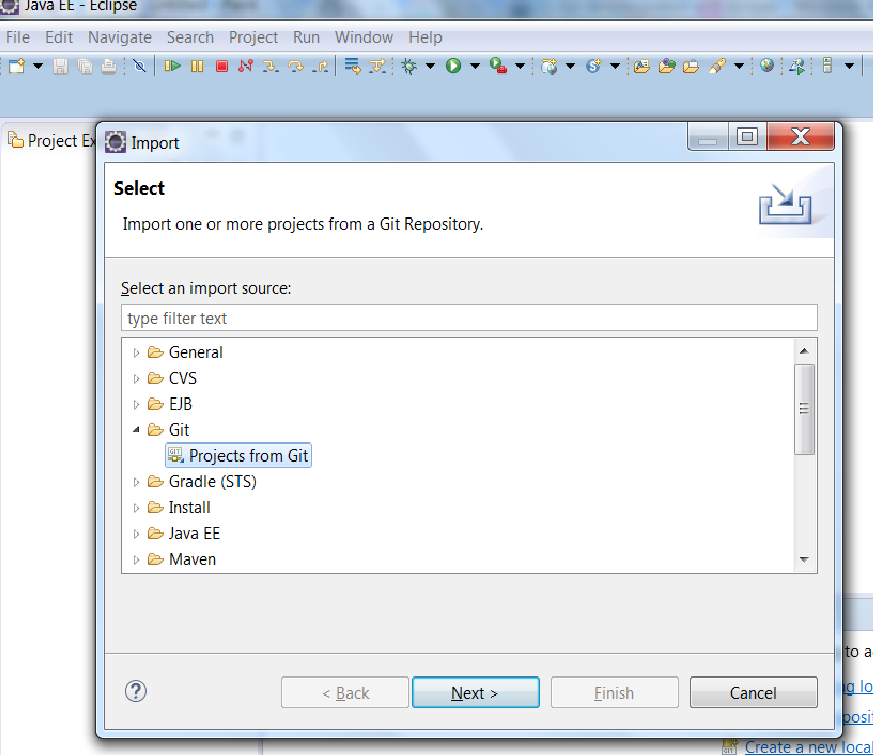


* Now verify your project in GIT remote repository.

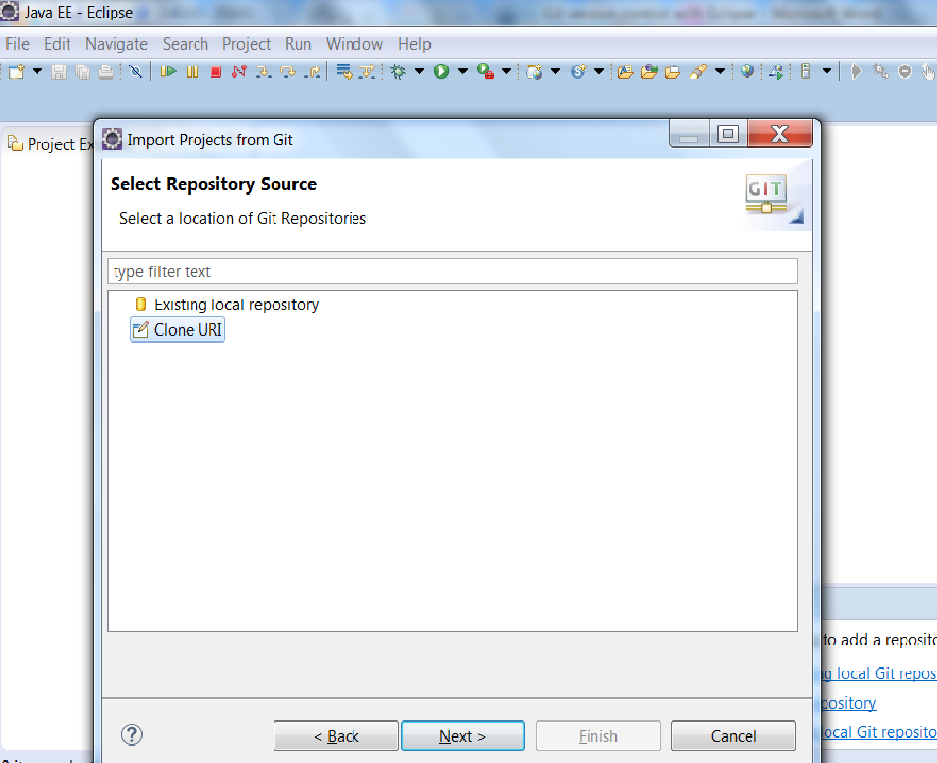
[**5. Import (clone) projects from a remote GIT repository**](http://www.vogella.com/tutorials/EclipseGit/article.html#import-projects-from-an-existing-repository)

* Before clone remote repository, we need project GIT remote repository URL and GIT username and password.
* Follow below steps to import projects from GIT remote repository.

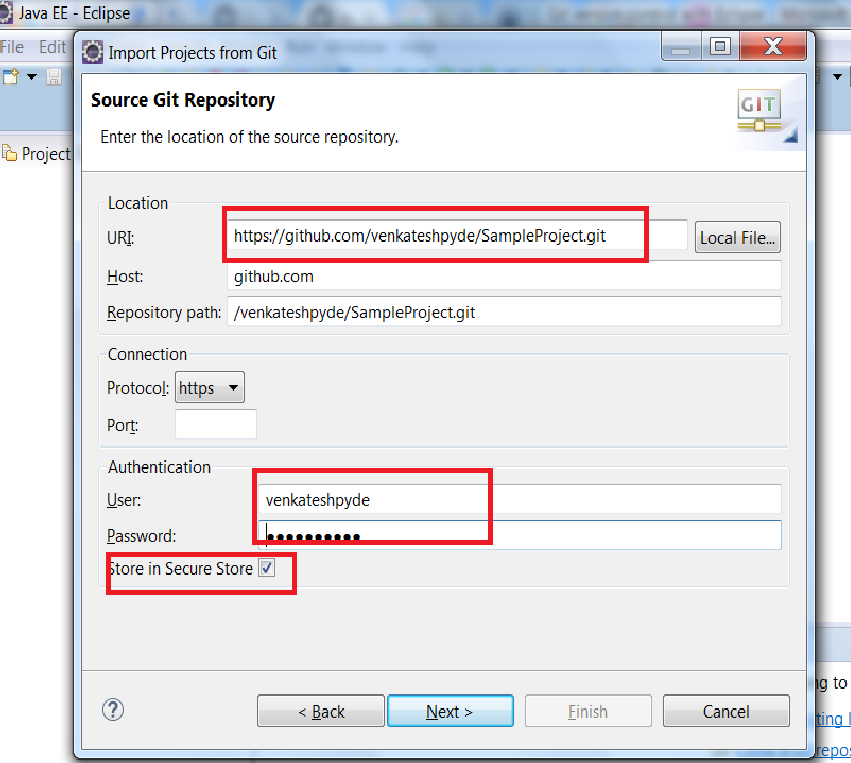
Go to ***File ▸ import ▸ expand git ▸ select Projects from Git ▸ Click on Next.***



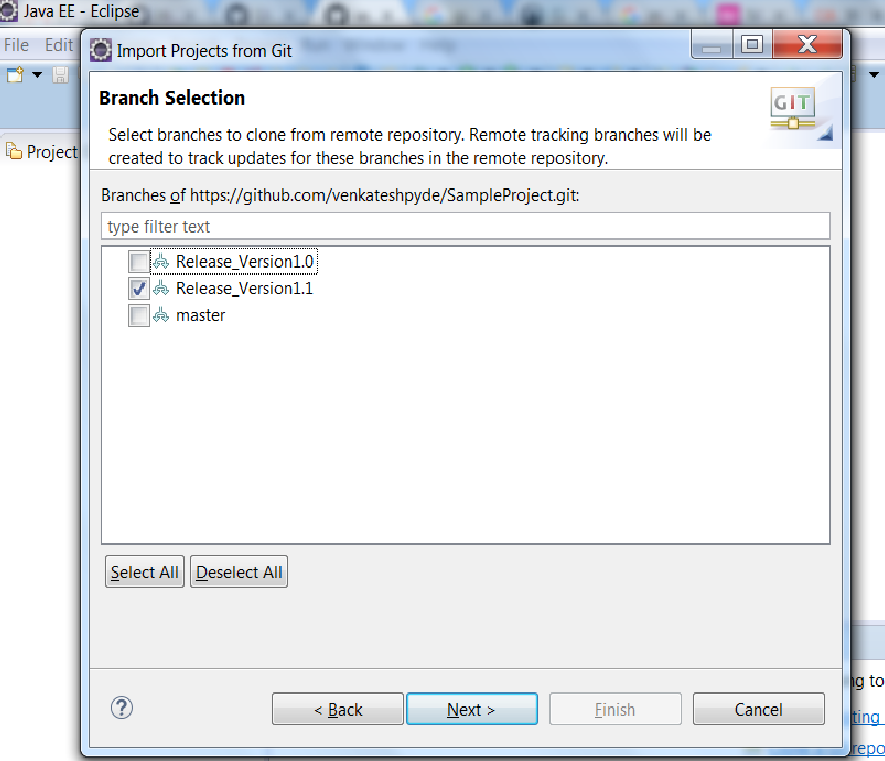
* Select ***Clone URI*** and click on ***Next button.***



* Enter project GIT remote repository ***URL*** and enter ***GIT username and password.*** Make sure ***enable Store in Secure store*** and click on ***Finish.***



* Select ***appropriate branch*** and click on ***Next***.



* Click on ***Browse button*** to choose local repository project location and click on ***Next ▸ Next▸ Finish***

