Code Repository Tool (Version Management)

To develop app. we write different types of files in projects (like java, xml, jsp etc..). All these files are stored finally in a central server i.e. Code Repository. It maintains the details of code. like

- ✓ How many times code is modified?
- ✓ Who modified what code(+/-   added/removed)
- ✓ Changes done on what date and time? Maintaining these details is known as Version Management/Version Control.

a. In developement mostly used tool is GIT(gitHub).
b. It maintains two level repository process for cross verify and commit.
c. For this we need to create an account in github.com(using email id and verify email after creation).
d. It maintains staging area to select /verify/comment the file while adding to local repository.

Git Process:

Git Repository Types:

a. Public : Free for every one
b. Private : Paid version.

Remote Repository(Steps):

a. goto https://github.com/
b. SignUp(Register) with details
c. Goto Email -and verfiy Link
d. login with un and pwd
e. Create Repository Type Public (Companies uses- Private Type)

f. Copy Git Link: https://github.com/abcd/testVen.git it is secured with un,pwd.

Eclipse Workspace and Local Repository Steps:

In eclipse,

1. Go to window menu
2. Show Views search for "GIT" select GitRepository and GitStaging.
3. Click on Window
4. showView
5. History.

Creating local repository:

1. Right Click on Project
2. choose "Team" option
3. Share project
4. select checkbox "create or use repository"
5. select option shown below
6. click create repository button
7. finish

GIT Operations with Flow:

- add file to git staging (Click-drag-drop).
- Commit and Push
- Paste Git Link in URI input box
- enter userName and password.
- next/next/finish.
- right click on project
- Choose team
- Select pull, then ,rebase to update local from remote.

Some Interview Questions over GIT:

1) Can you provide GIT/Client/Apps URLs? No, Sorry.all details what you are asking is private and confidential. I cannot provide those.

2)Who created your git account ?

Git administrator name-"XYZ", in my company. He handles all permission and auth. details.

3)What type of Account Repository areyou using?

It is private account handled by company.

4)When you check-in or push your code?

After Implementation of complete module and UnitTesting code in my local, I'll push to remote.

5)How will you find about one file all modification??

We use history option in eclipse

->right click->team->shown in history.

we can also choose any two files to compare with each other ex:select

any history options->right click->compare with each other.

6)How do you identify code modifications in GIT?

Using Symbols (+) code is added (-) code is removed.

7)What is the difference between commit and push?

commit: update code from workspace to Local Repository

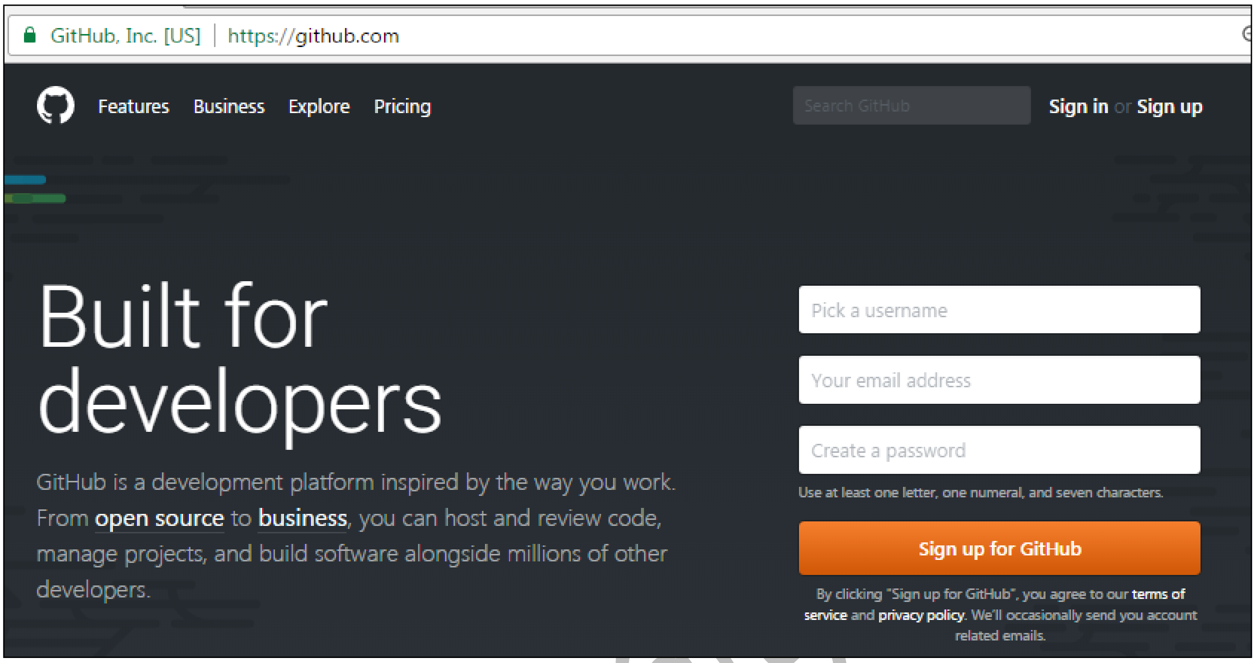
push :Update code from local to remote repository.

8)How can you update you workspace/local with remote?

By using pull and rebase operations.

Screens Help For Complete Process:

https://github.com/



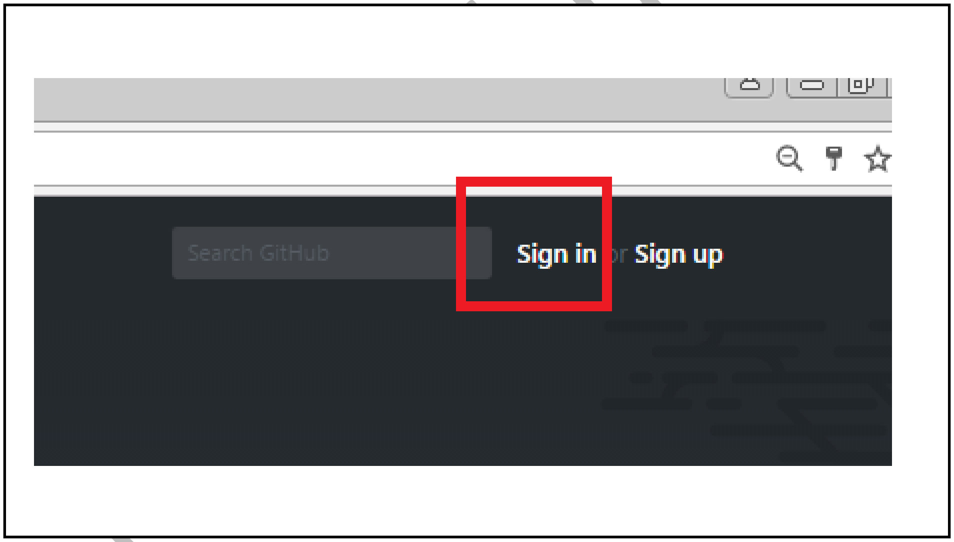
Built for developers

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside millions of other developers.

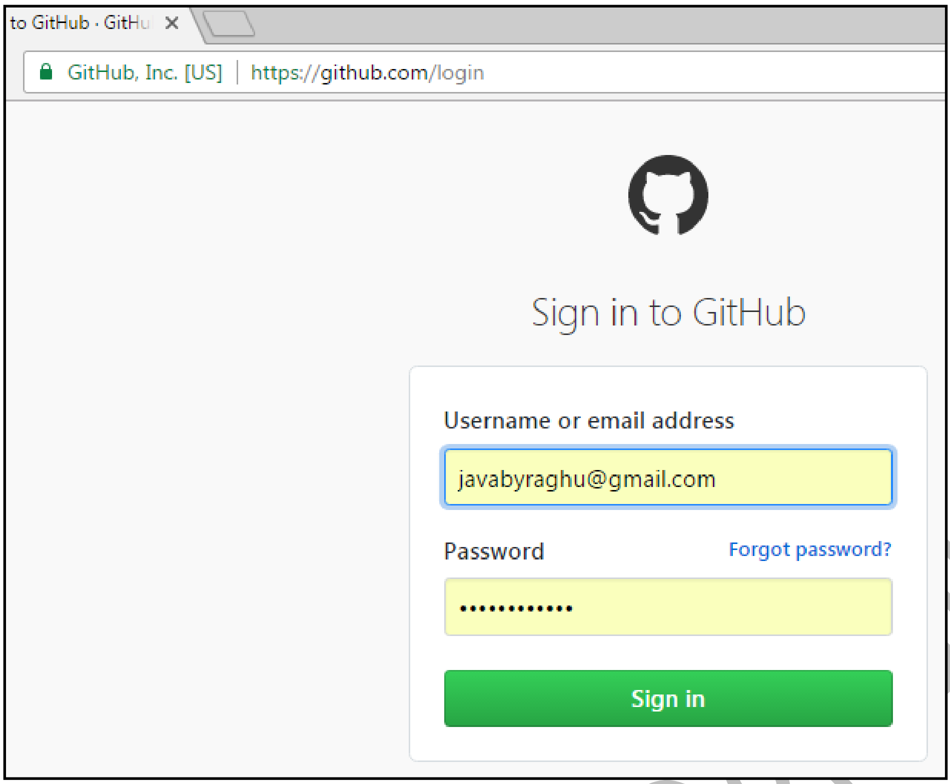
Use at least one letter, one numeral, and seven characters.

By clicking "Sign up for GitHub", you agree to our terms of service and privacy policy. We'll occasionally send you account related emails.

Goto email account and click on verify link. Click on signin



Enter user name and password.

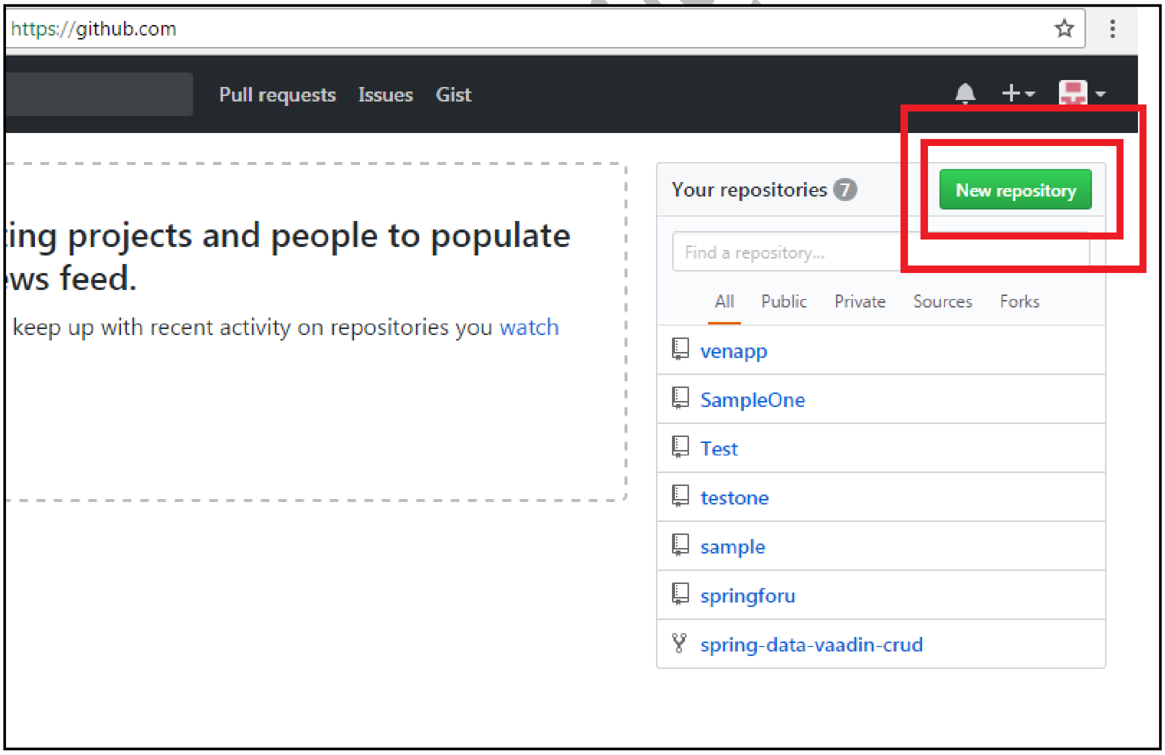


Sign in to GitHub

Username or email address: javabyraghu@gmail.com

Password: ••••••••••••

Click on new Repository



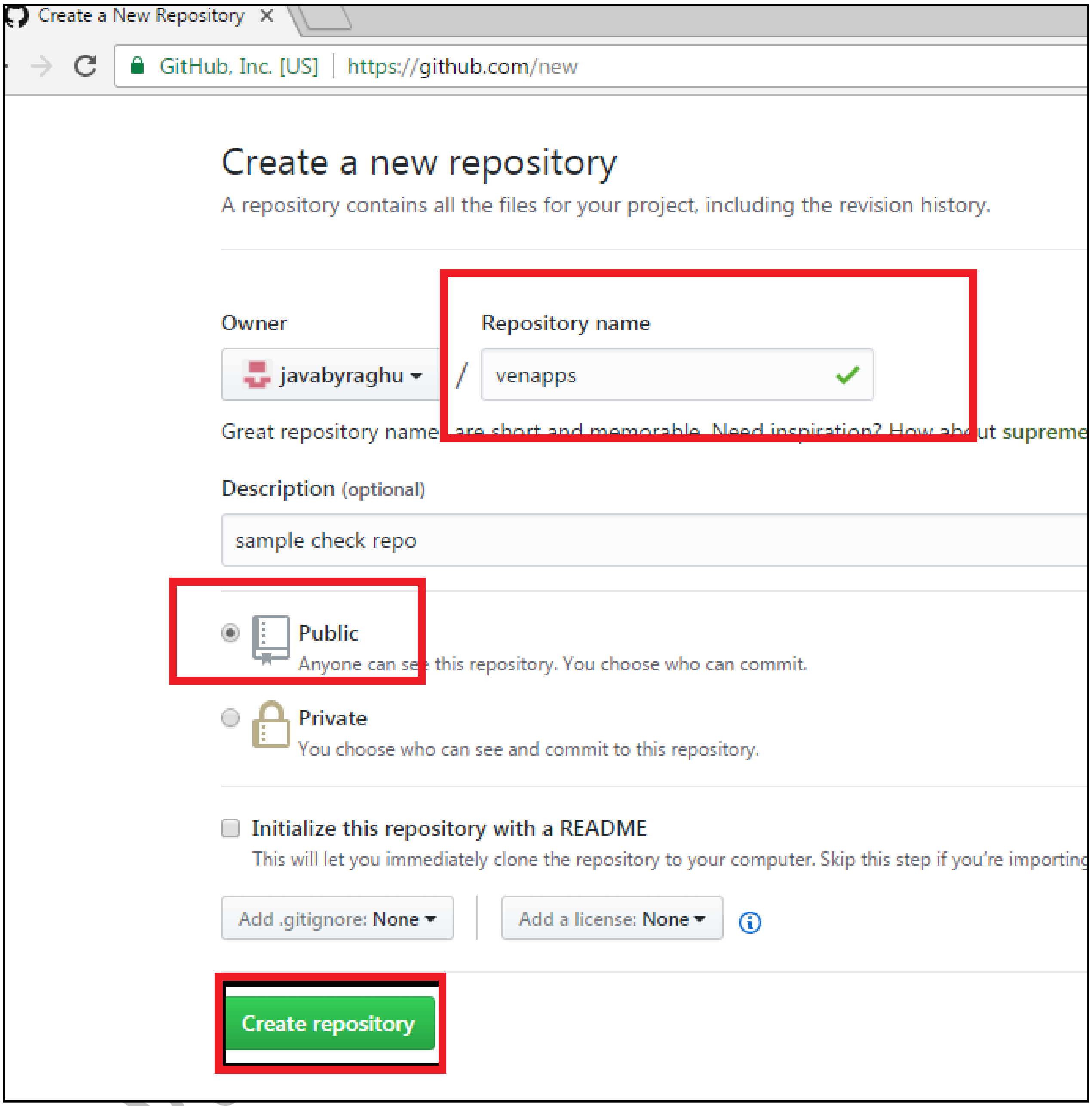
ing projects and people to populate
ws feed.

keep up with recent activity on repositories you watch

Your repositories 7

- venapp
- SampleOne
- Test
- testone
- sample
- springforu
- spring-data-vaadin-crud

Provide Repository Details



Create a New Repository

GitHub, Inc. [US] | https://github.com/new

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: javabyraghu

Repository name: venapps

Great repository names are short and memorable. Need inspiration? How about supreme

Description (optional): sample check repo

● Public
Anyone can see this repository. You choose who can commit.

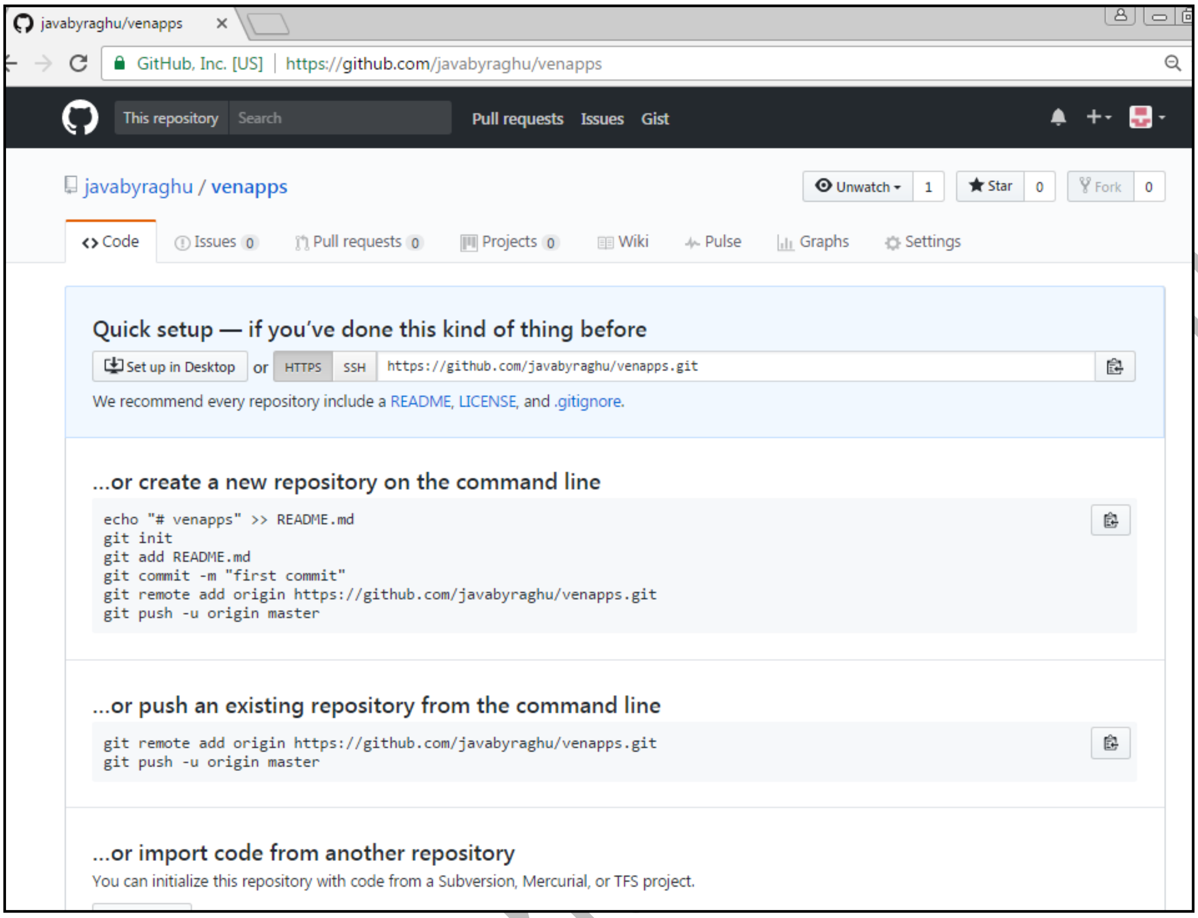
○ Private
You choose who can see and commit to this repository.

□ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing

Add .gitignore: None

Add a license: None

On click create repository



javabyraghu/venapps
GitHub, Inc. [US] | https://github.com/javabyraghu/venapps
This repository | Search
Pull requests Issues Gist
javabyraghu / venapps
Unwatch 1 | Star 0 | Fork 0
Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Pulse | Graphs | Settings
Set up in Desktop | or | HTTPS | SSH

Quick setup — if you've done this kind of thing before

https://github.com/javabyraghu/venapps.git

We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

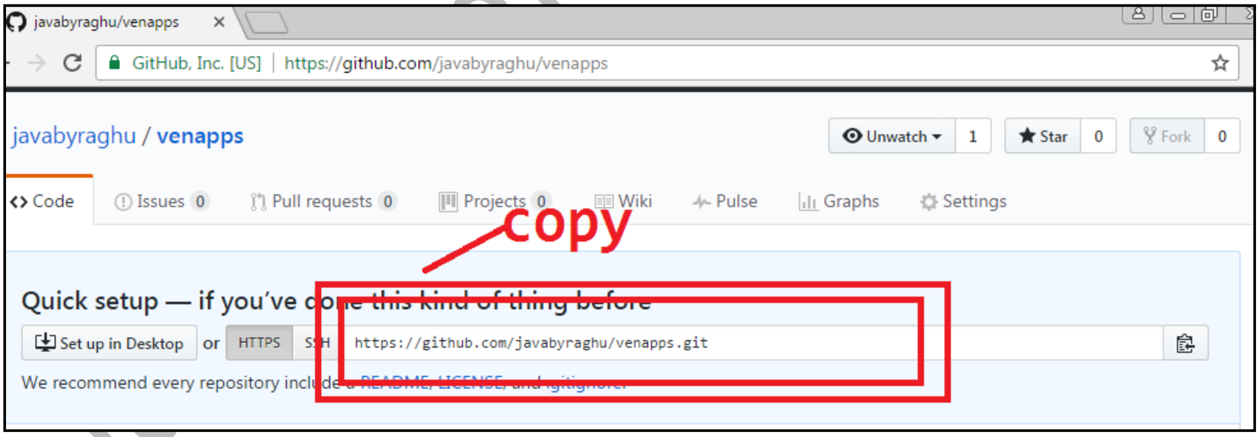
```
echo "# venapps" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/javabyraghu/venapps.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/javabyraghu/venapps.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.



javabyraghu/venapps
GitHub, Inc. [US] | https://github.com/javabyraghu/venapps
javabyraghu / venapps
Unwatch 1 | Star 0 | Fork 0
Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Pulse | Graphs | Settings
copy
Set up in Desktop | or | HTTPS | SSH

Quick setup — if you've done this kind of thing before

https://github.com/javabyraghu/venapps.git

We recommend every repository include a README, LICENSE, and .gitignore.

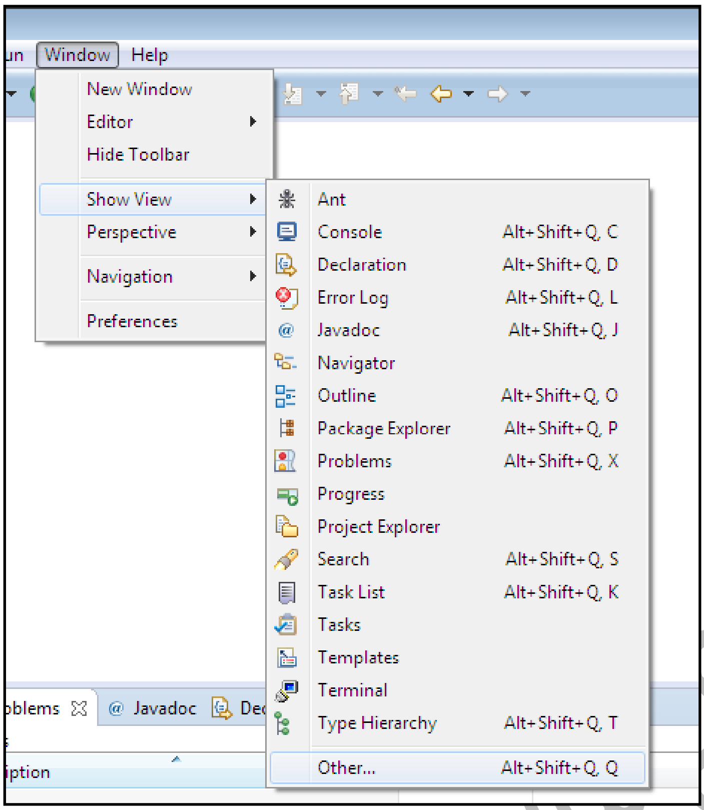
Ex : (Repository Link)

https://github.com/javabyraghu/venapps.git

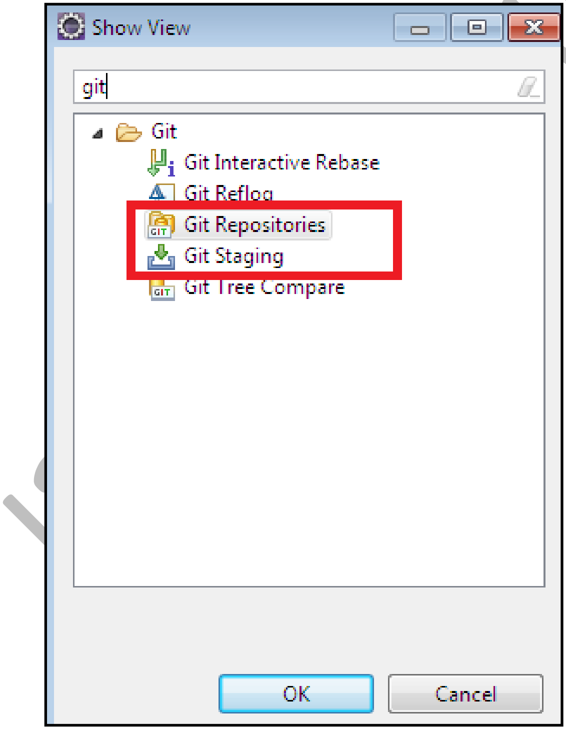
Got Eclipse:

- Create A new Java Project **(project name and repository name Should be same)**

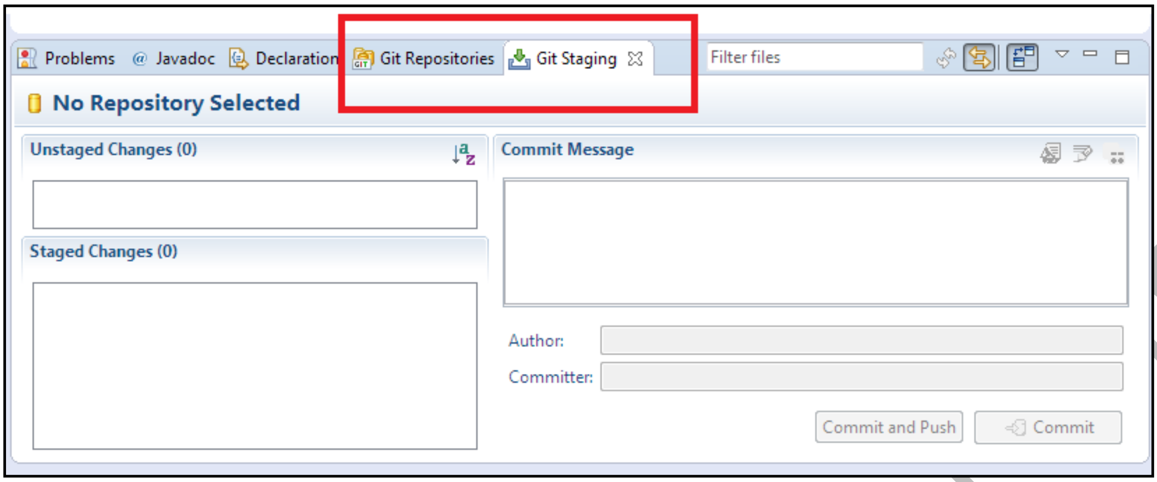
- Add Git Views



Serach with Git, select Git Repository, Staging

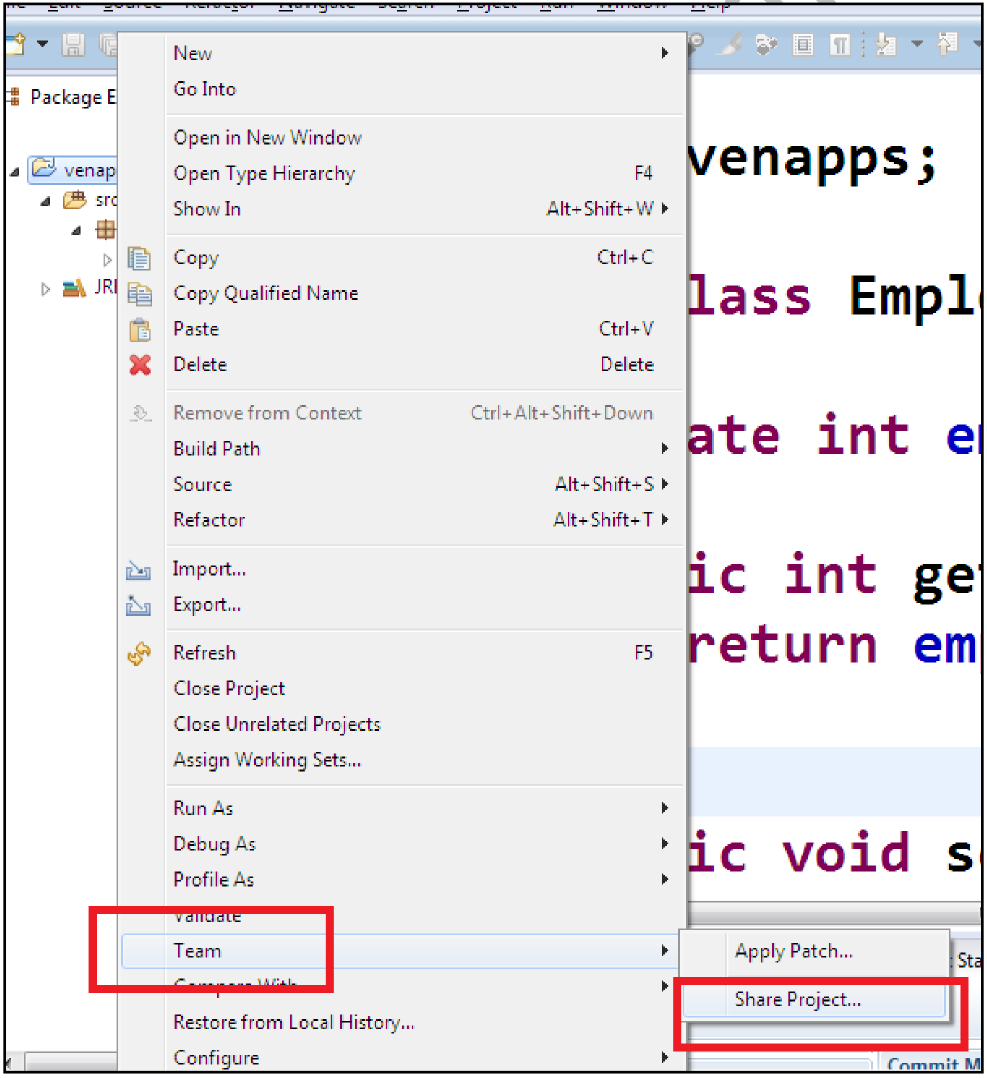


Observer options like



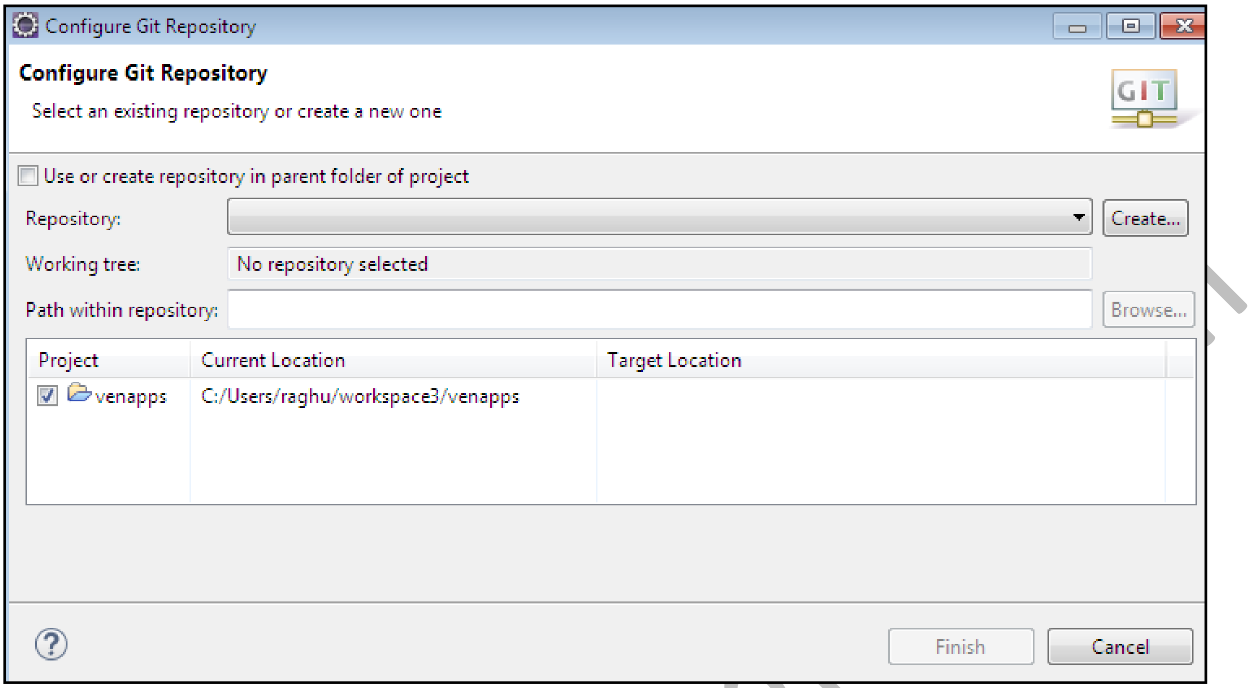
After writing of some code ex: Employee.java

>right click on project > tem > share project



```
venapps;
lass Empl
ate int e
ic int ge
return em
ic void s
```

looks as below:



Configure Git Repository

Configure Git Repository

Select an existing repository or create a new one

□ Use or create repository in parent folder of project

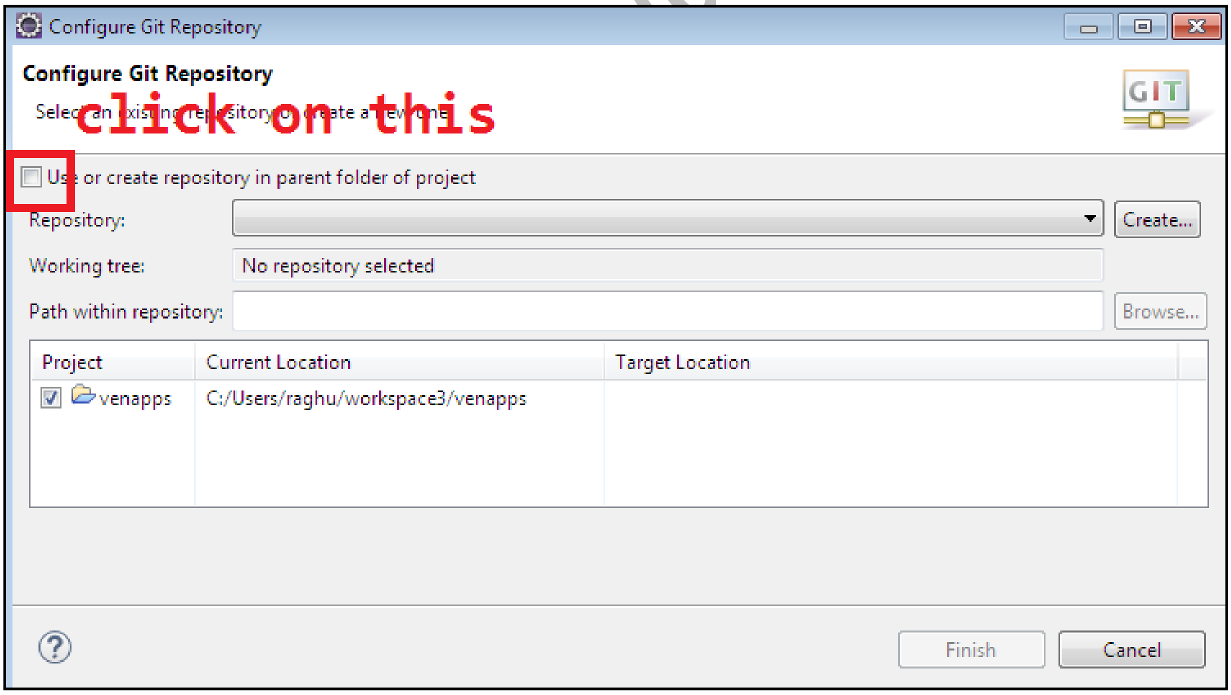
Repository: [ ] Create...

Working tree: No repository selected

Path within repository: [ ] Browse...

Project	Current Location	Target Location
☑ venapps	C:/Users/raghu/workspace3/venapps	

Click on check box:



Configure Git Repository

Configure Git Repository

Select an existing repository or create a new one

□ Use or create repository in parent folder of project

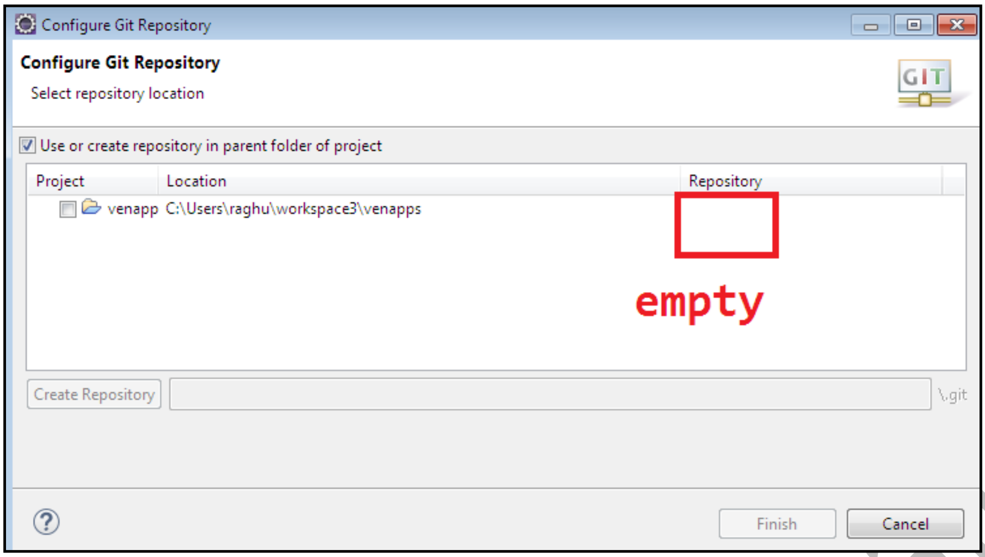
Repository: [ ] Create...

Working tree: No repository selected

Path within repository: [ ] Browse...

Project	Current Location	Target Location
☑ venapps	C:/Users/raghu/workspace3/venapps	

looks as below:



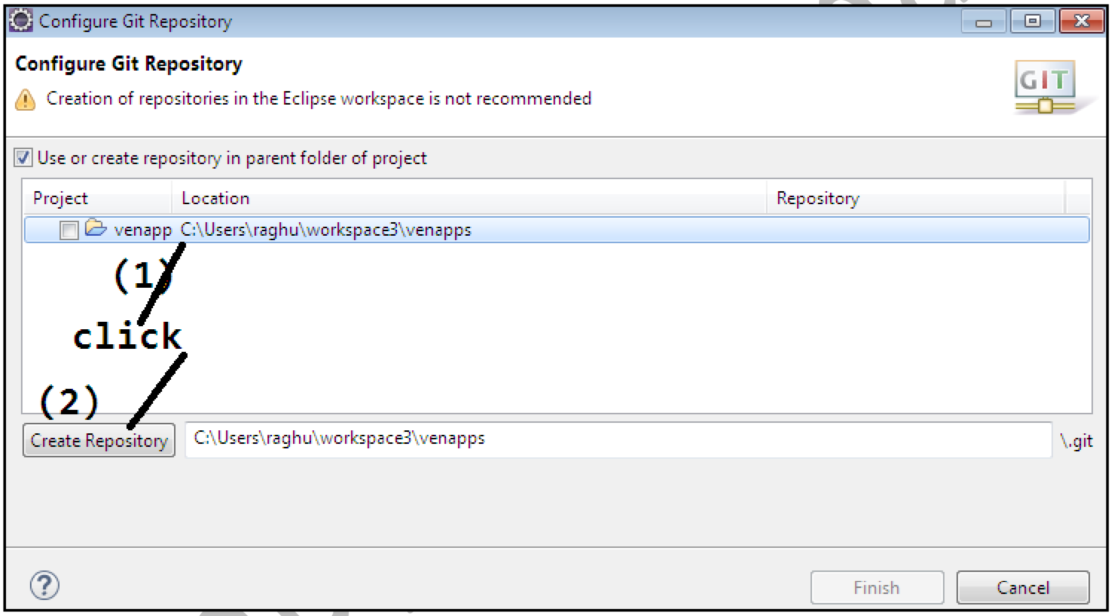
Configure Git Repository

Configure Git Repository

Select repository location

☑ Use or create repository in parent folder of project

Project	Location	Repository
□ venapp	C:\Users\raghu\workspace3\venapps	



Configure Git Repository

Configure Git Repository

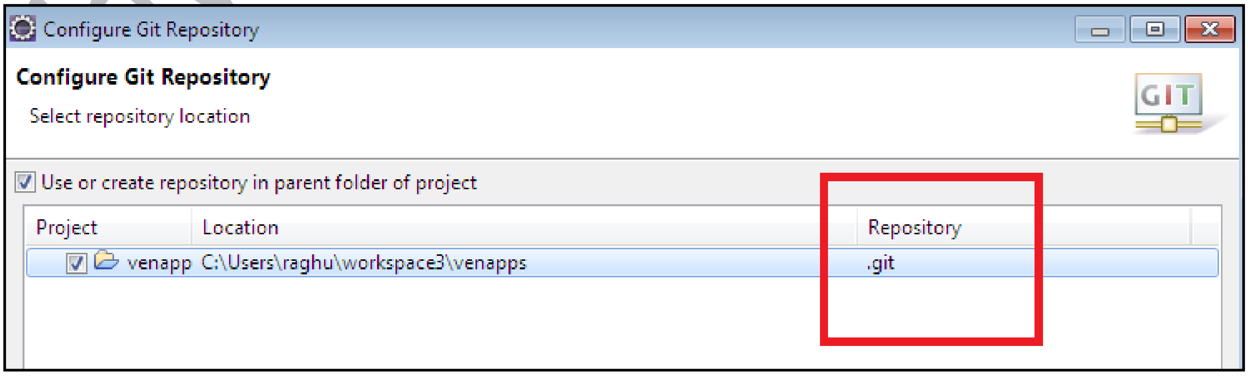
⚠ Creation of repositories in the Eclipse workspace is not recommended

☑ Use or create repository in parent folder of project

Project	Location	Repository
□ venapp	C:\Users\raghu\workspace3\venapps	

C:\Users\raghu\workspace3\venapps \.git

then observe repository name:



Configure Git Repository

Configure Git Repository

Select repository location

☑ Use or create repository in parent folder of project

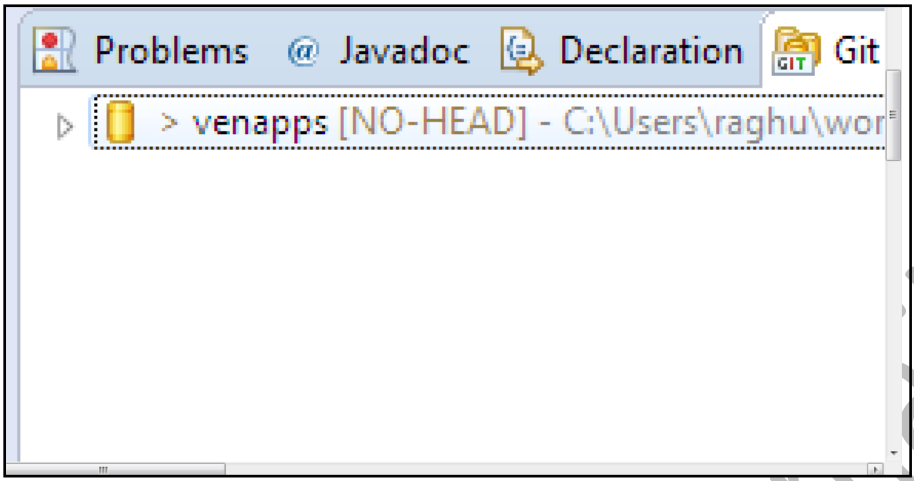
Project	Location	Repository
☑ venapp	C:\Users\raghu\workspace3\venapps	.git

click on finish.

By this we created local and remote repositories.

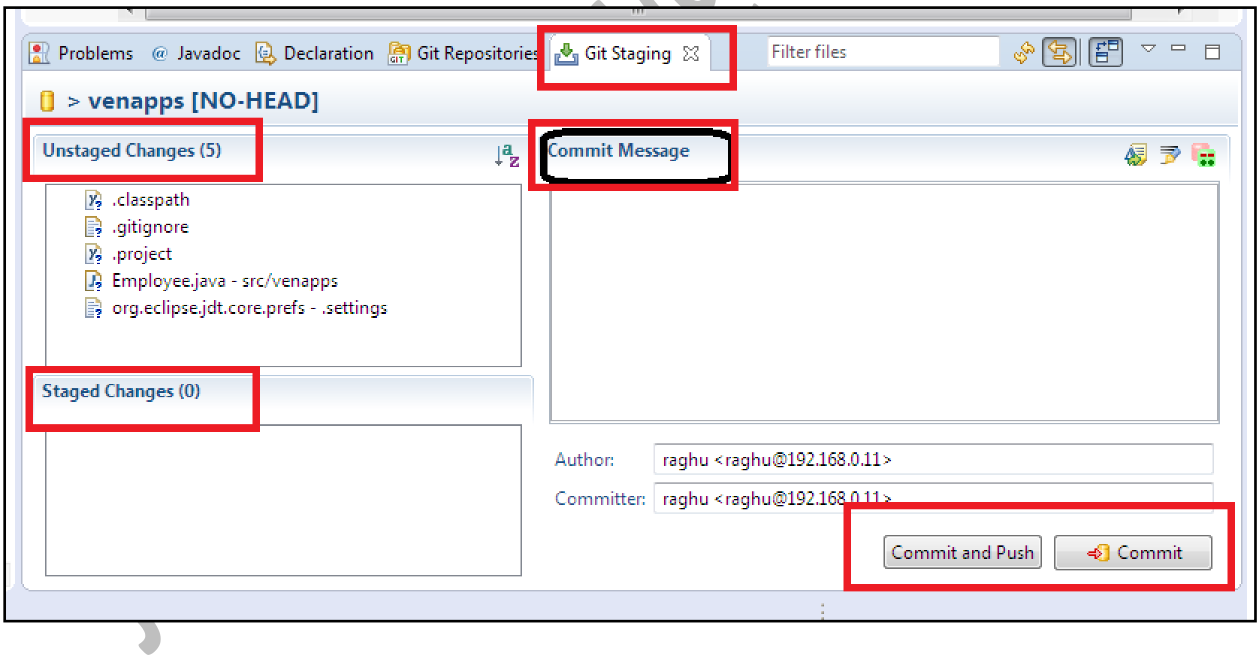
Now link local and remote, using firsh commit and push.

Come to git repositoes



> venapps [NO-HEAD] - C:\Users\raghu\wor

In staging are:



> venapps [NO-HEAD]

Unstaged Changes (5)

- .classpath
- .gitignore
- .project
- Employee.java - src/venapps
- org.eclipse.jdt.core.prefs - .settings

Staged Changes (0)

Author: raghu <raghu@192.168.0.11>

Committer: raghu <raghu@192.168.0.11>

1)Unstaged changes : files which are not ready to send to local

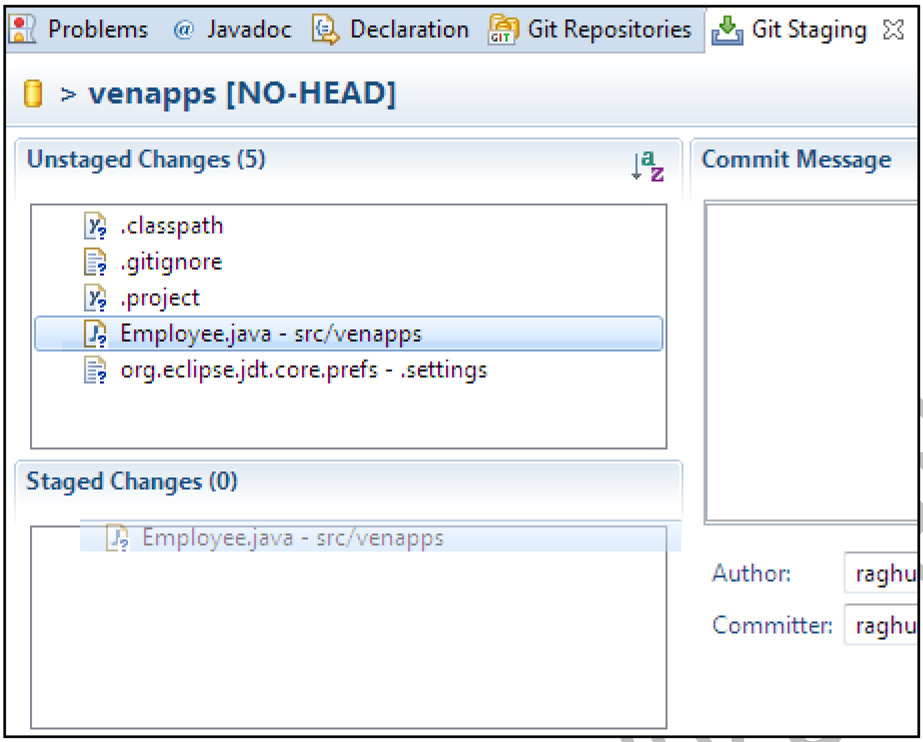
2)staged chages: file which needs to move to local/remote

3)commit message: what is the purpose specify here as message for sending of this code.

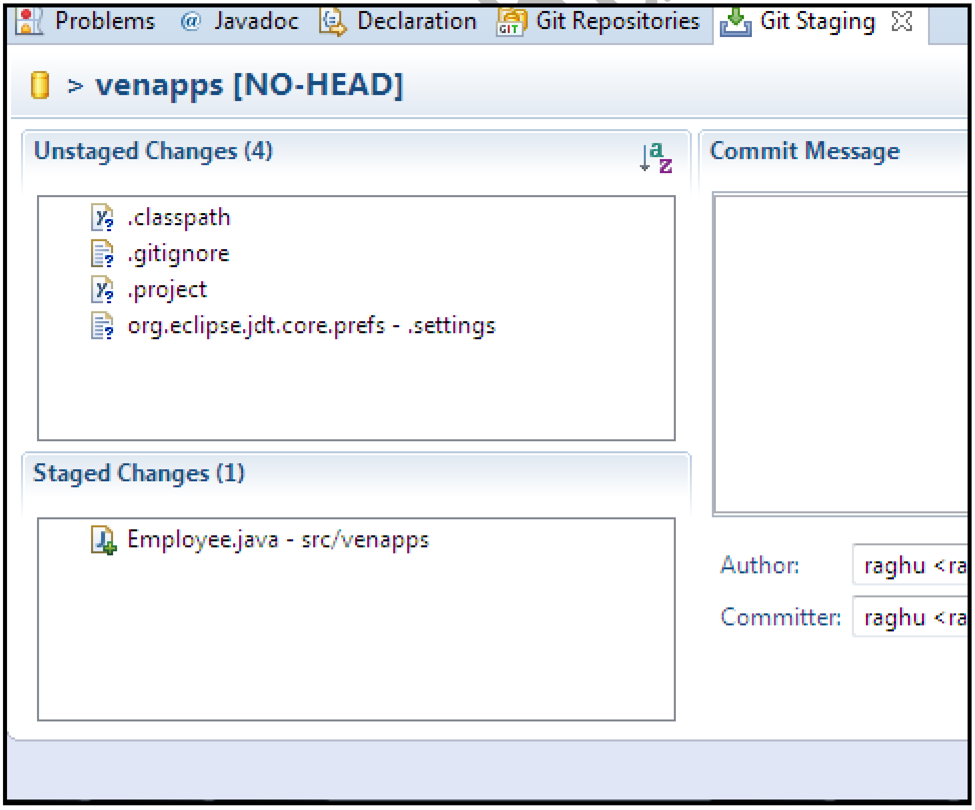
4) click commit/ commit & push to send the code.

*)Click on a file in unstage drag and drop into stage.

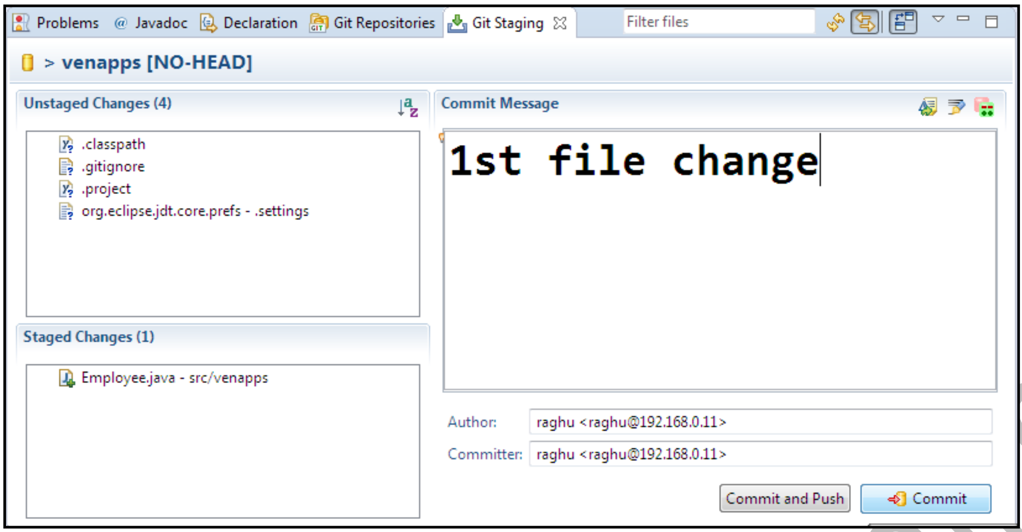
Click and Drag:



After drop



Enter commit message



.classpath
.gitignore
.project
org.eclipse.jdt.core.prefs - .settings

Employee.java - src/venapps

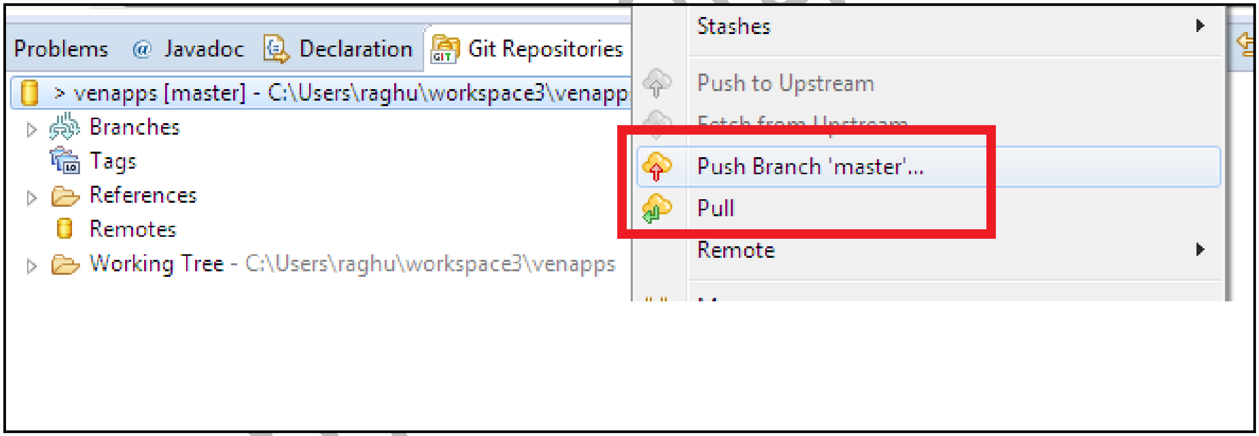
```
1st file change
```

Author: raghu <raghu@192.168.0.11>

Committer: raghu <raghu@192.168.0.11>

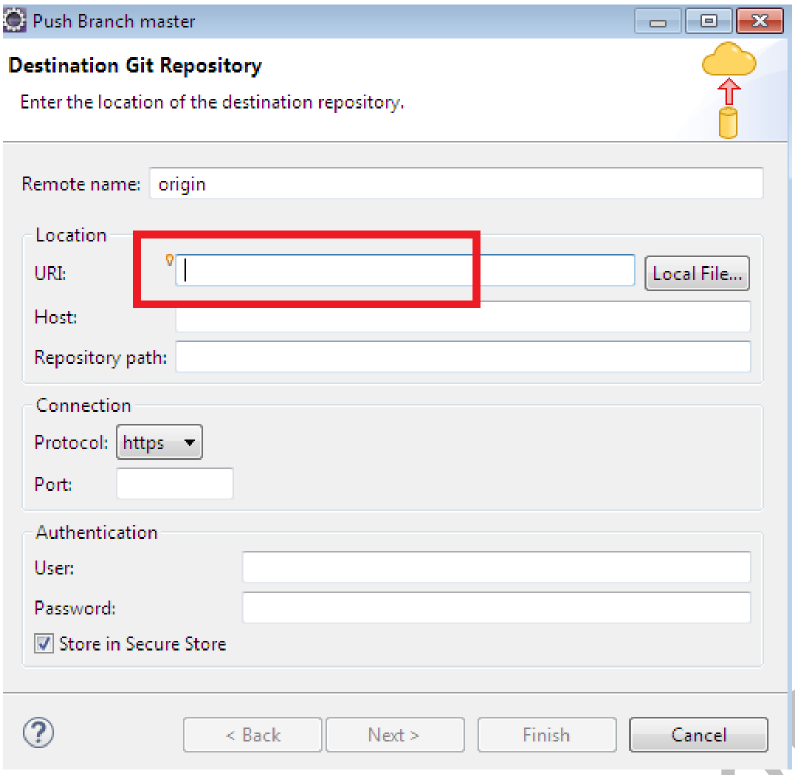
And click on commit.

In GitRepository , right click on projectName>choose Push Branch master



After Push option one screen looks below, there enter URI (Repository link)

Ex: https://github.com/javabyraghu/venapps.git



Push Branch master

Destination Git Repository

Enter the location of the destination repository.

Remote name: origin

Location

URI:

Host:

Repository path:

Connection

Protocol: https

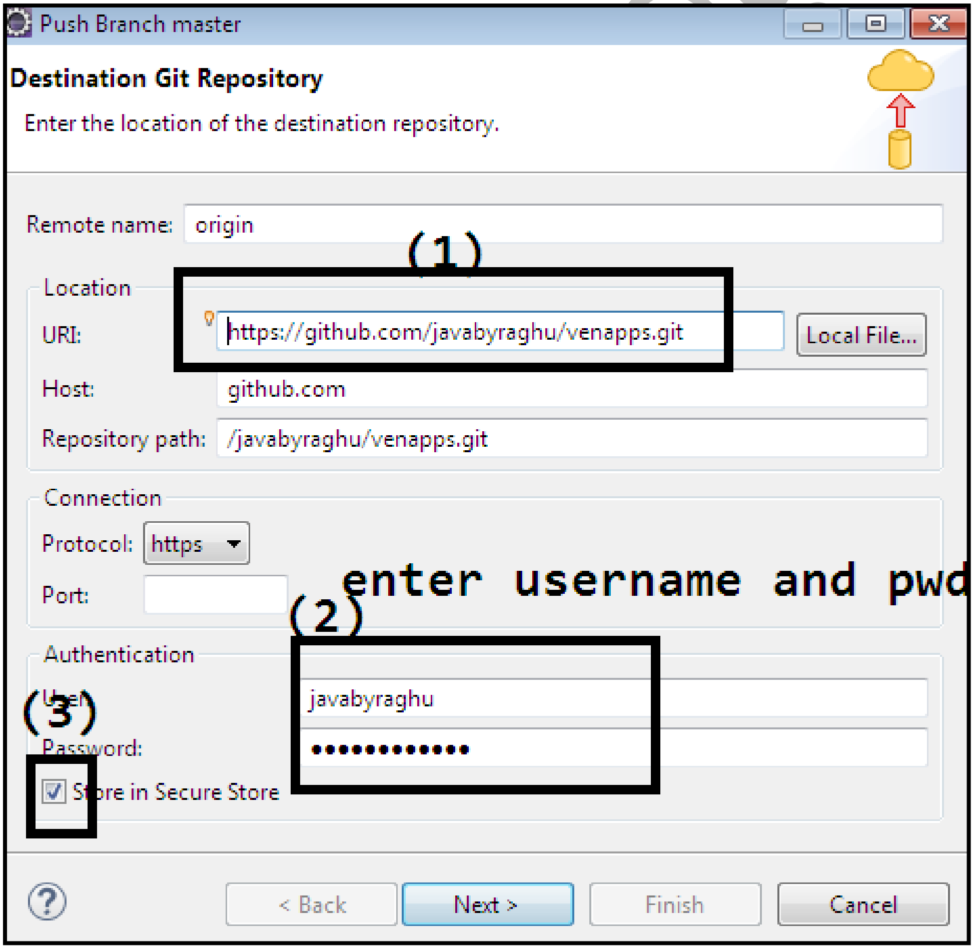
Port:

Authentication

User:

Password:

☑ Store in Secure Store



Push Branch master

Destination Git Repository

Enter the location of the destination repository.

Remote name: origin

(1)

Location

URI: https://github.com/javabyraghu/venapps.git

Host: github.com

Repository path: /javabyraghu/venapps.git

Connection

Protocol: https

Port:

enter username and pwd

(2)

Authentication

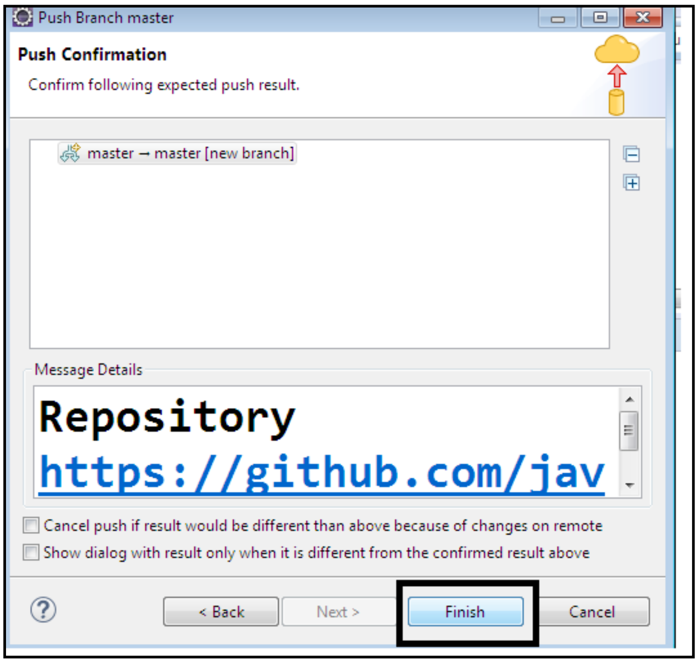
(3)

User: javabyraghu

Password: ●●●●●●●●●●●●

☑ Store in Secure Store

Click on next>next>finish



Push Branch master

Push Confirmation

Confirm following expected push result.

master → master [new branch]

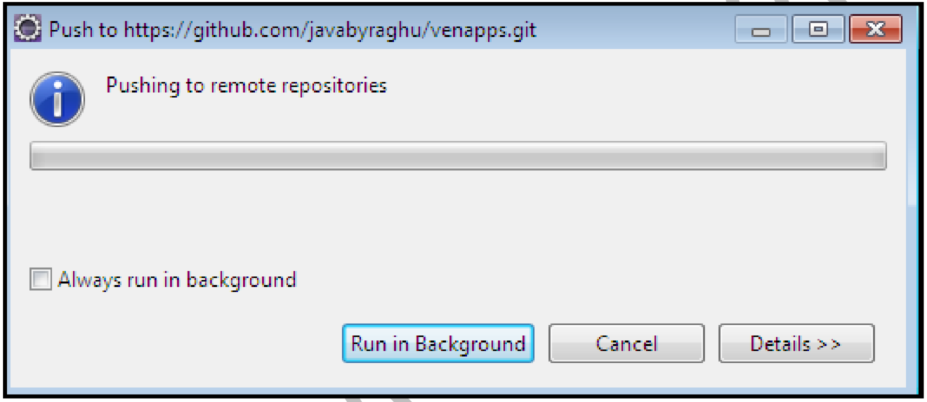
Message Details

Repository
https://github.com/jav

□ Cancel push if result would be different than above because of changes on remote

□ Show dialog with result only when it is different from the confirmed result above

Shows as below:

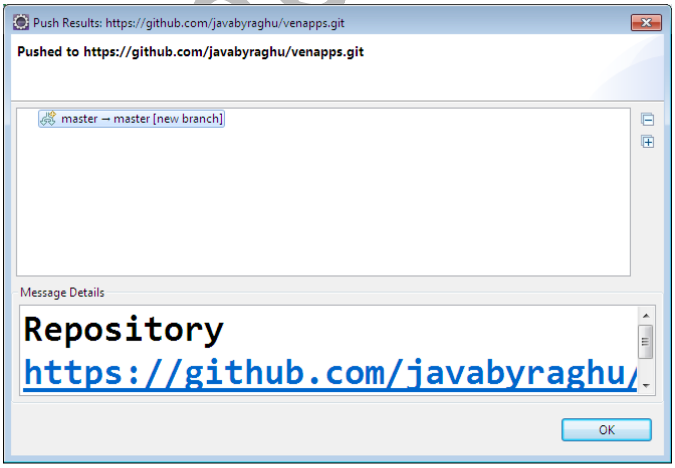


Push to https://github.com/javabyraghu/venapps.git

Pushing to remote repositories

□ Always run in background

On success:



Push Results: https://github.com/javabyraghu/venapps.git

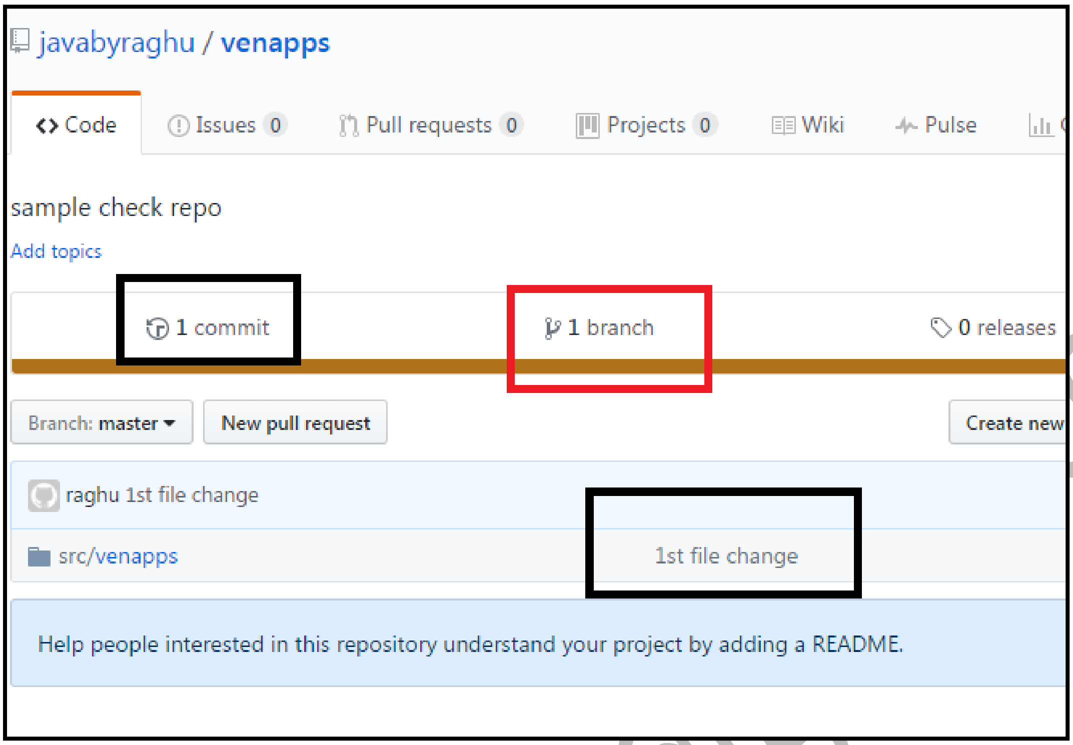
Pushed to https://github.com/javabyraghu/venapps.git

master → master [new branch]

Message Details

Repository
https://github.com/javabyraghu/

Come back to browser and refresh:



sample check repo

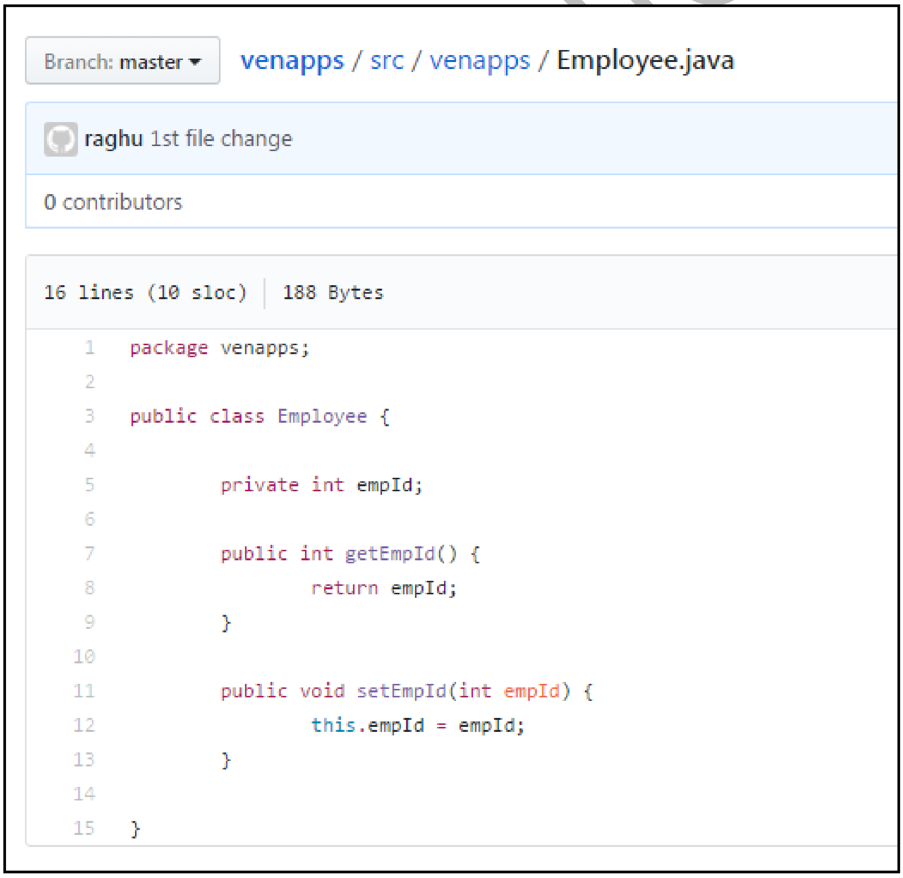
Add topics

raghu 1st file change

src/venapps — 1st file change

Help people interested in this repository understand your project by adding a README.

Click on src/venapp>Employee.java



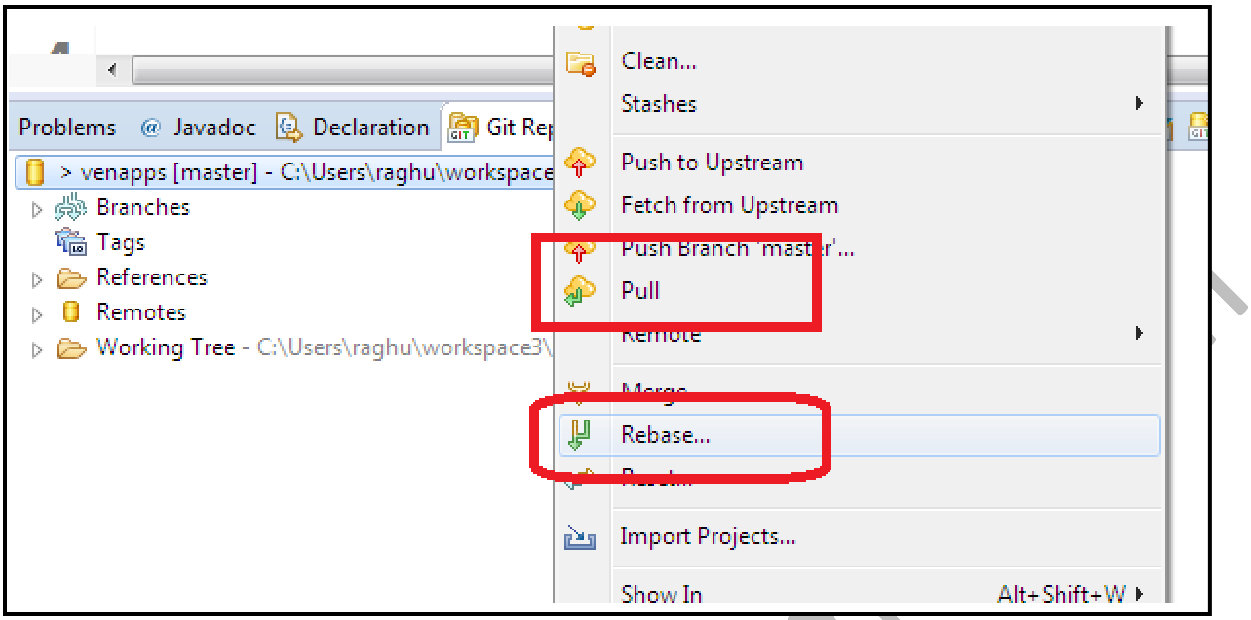
raghu 1st file change

0 contributors

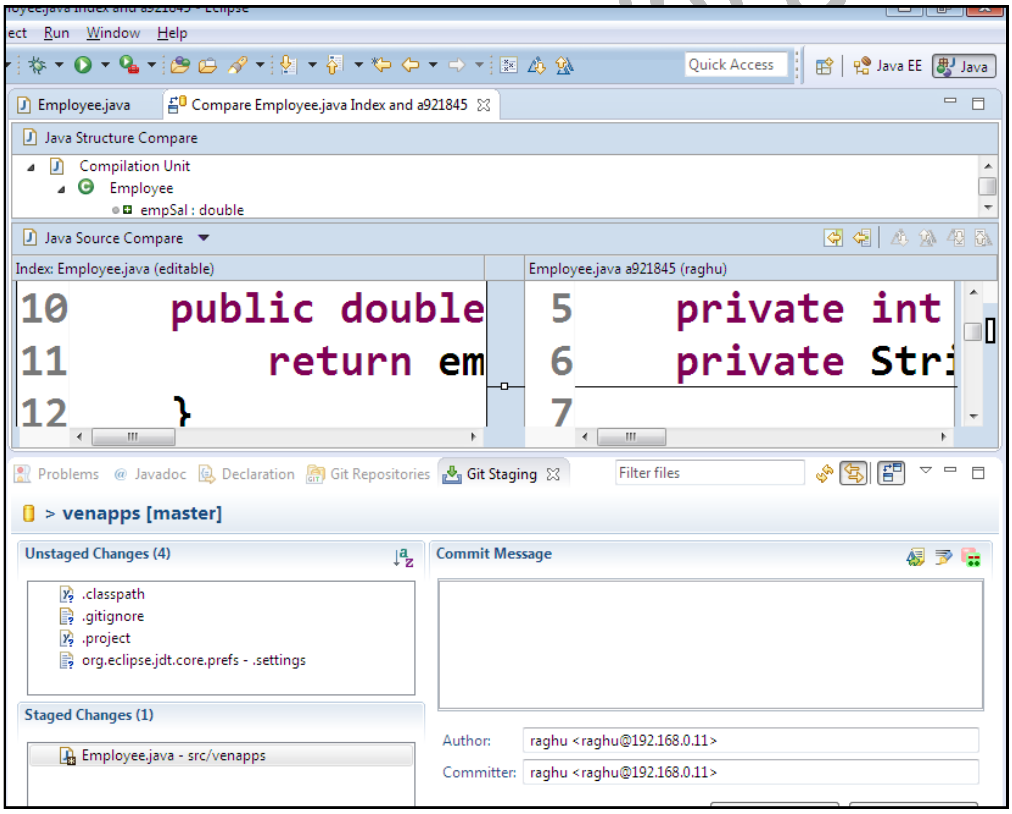
16 lines (10 sloc) | 188 Bytes

```java
package venapps;

public class Employee {

        private int empId;

        public int getEmpId() {
                return empId;
        }

        public void setEmpId(int empId) {
                this.empId = empId;
        }

}
```

Right click on project: pull and then rebase for other update:



After Ading file to staging ,double click on that to comapre with Old one (previous one)looks as :



```
10      public double
11          return em
12      }
```

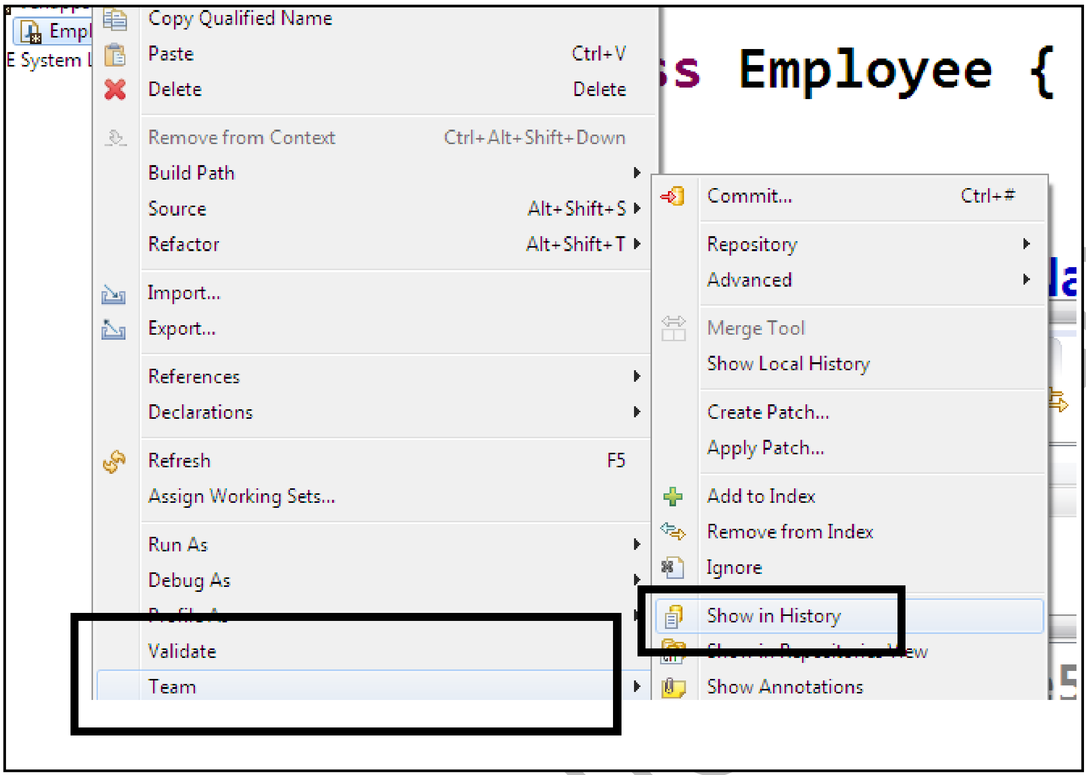
```
5       private int
6       private Stri
7
```

Author: raghu <raghu@192.168.0.11>

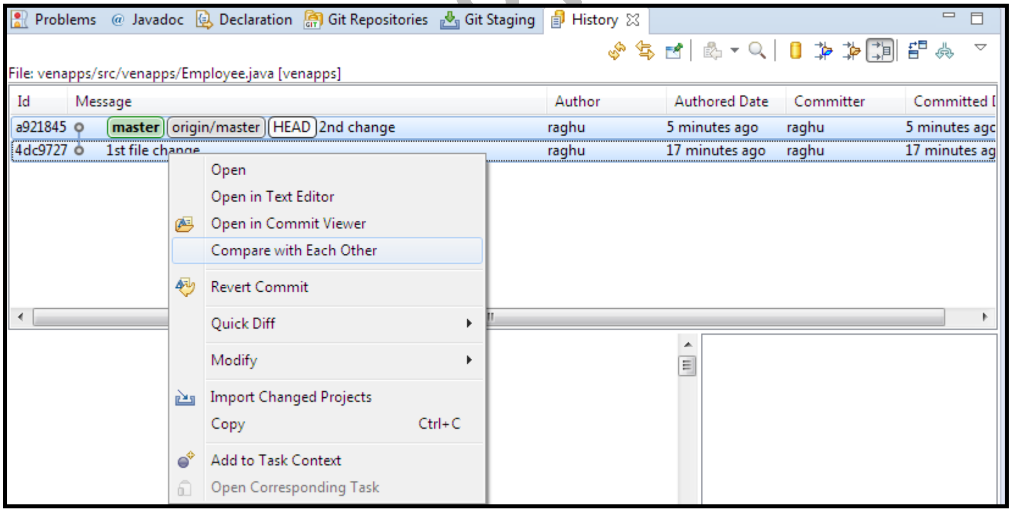
Committer: raghu <raghu@192.168.0.11>

To see history:

Right click on File (ex: Employee.java)> Team>Showin History



View history:



File: venapps/src/venapps/Employee.java [venapps]

Id	Message	Author	Authored Date	Committer	Committed
a921845	master origin/master HEAD 2nd change	raghu	5 minutes ago	raghu	5 minutes agc
4dc9727	1st file change	raghu	17 minutes ago	raghu	17 minutes ag

** GIT Merge Conflicts **

If any file code is modified(added/removed) then file version will be changed.

ex: Employee.java = Version#1

1. class Employee
2. {
3.
4. }

Employee.java = Version#2

1. class Employee
2. {

3.+ int empId;

4.+ String empName; - }

5.+ }

V#2 = V#1 [+3,-1].

Git manages version management automatically Here programmer do not need to remember any version number.

** On git pull & rebase, local repository and remote repository versions will become equal.

1. Repositories
2. right click on project
3. "pull" option
4. again right click
5. "rebase" option.

Merge conflict example:

Step#1: Dev#1 has done git pull & rebase

Step#2: Dev#2 has done git pull & rebase

Step#3: Dev#1 modified Employee.java

(that is changed to Version#2) and

did add/commit/push.

Now git has Employee.java Version#2

Step#4: Dev#2 Still working with old code

of Employee.java (Version#1).

File modified and did add/commit/

push, then git shows

Git - Version Conflict: Dev#2

Employee.java(version#1) not

matched with Git Repo Employee.java

(Version#2). Please update before

commit/push.

To Resolve merge conflict steps are.

- project -> right click -> reset
- project -> right click -> pull & rebase
- project -> right click -> merge

Now code will be updated to V#2,

write your code, then >>> add/commit/push.

facebook group Id:

https://www.facebook.com/groups/thejavatemple/

email :

javabyraghu@gmail.com