Object Oriented Programming

Week 1 Part 2
Git and eGit

Lecture

- Review of Git
- Local Repository
- Remote Repository
- Using Git from Eclipse

Review of Git

What is Git?

- Software Configuration Management (SCM)
- Supports *rollback*, returning software to an earlier state
  - So errors can be removed
  - So requirements can be dropped
  - So experiments can be done

SCM and Backup

- SCM is not backup.
  - Unless the copies are also copied to a remote disk, all versions can be lost.
  - But copying to remote disks make produce incompatible versions
- Git supports both SCM and Backup
  - Remote versions are copies of local versions
  - Versions are named according to their contents
    - Files with the same content have the same name

Local Repository

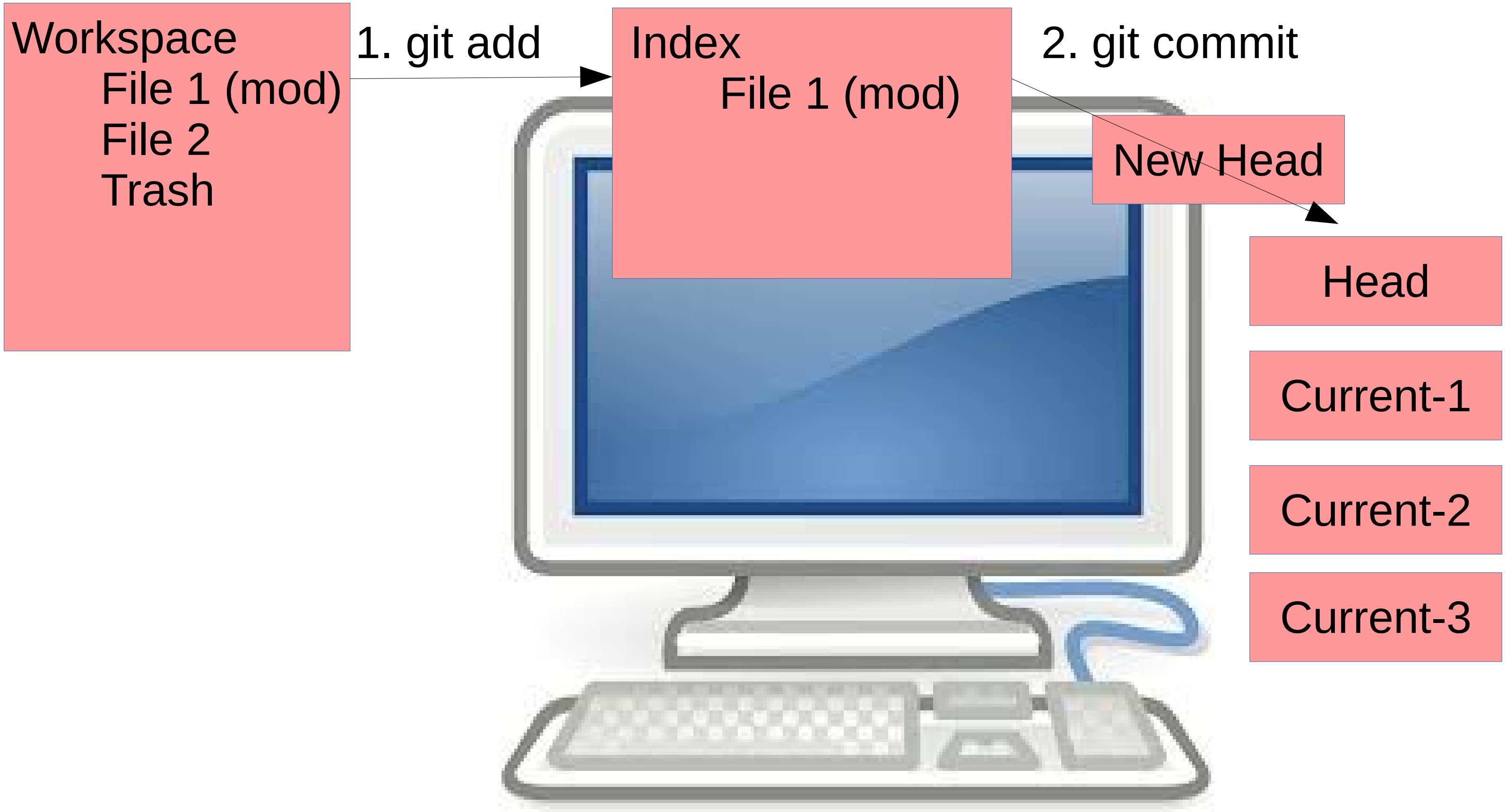
Local Repositories: SCM only

- Each repository contains all previous versions of the program.
- Losing the repository means losing all previous versions
- Local Repositories are useful when:
  - You accidentally overwrite a file
  - You cannot figure out why your program no longer works
  - You want to know who broke the program

Checking in to a Local Repository

- Create a repository (git init)
  - One time per program.
- Each time you change your program, *check in* to create a new version
  1) Tell Git what you added to the program (git add)
  2) Tell Git that to create a new versions (git commit)

Git check in



Workspace
- File 1 (mod)
- File 2
- Trash

1. git add

Index
- File 1 (mod)

2. git commit

New Head

Head

Current-1

Current-2

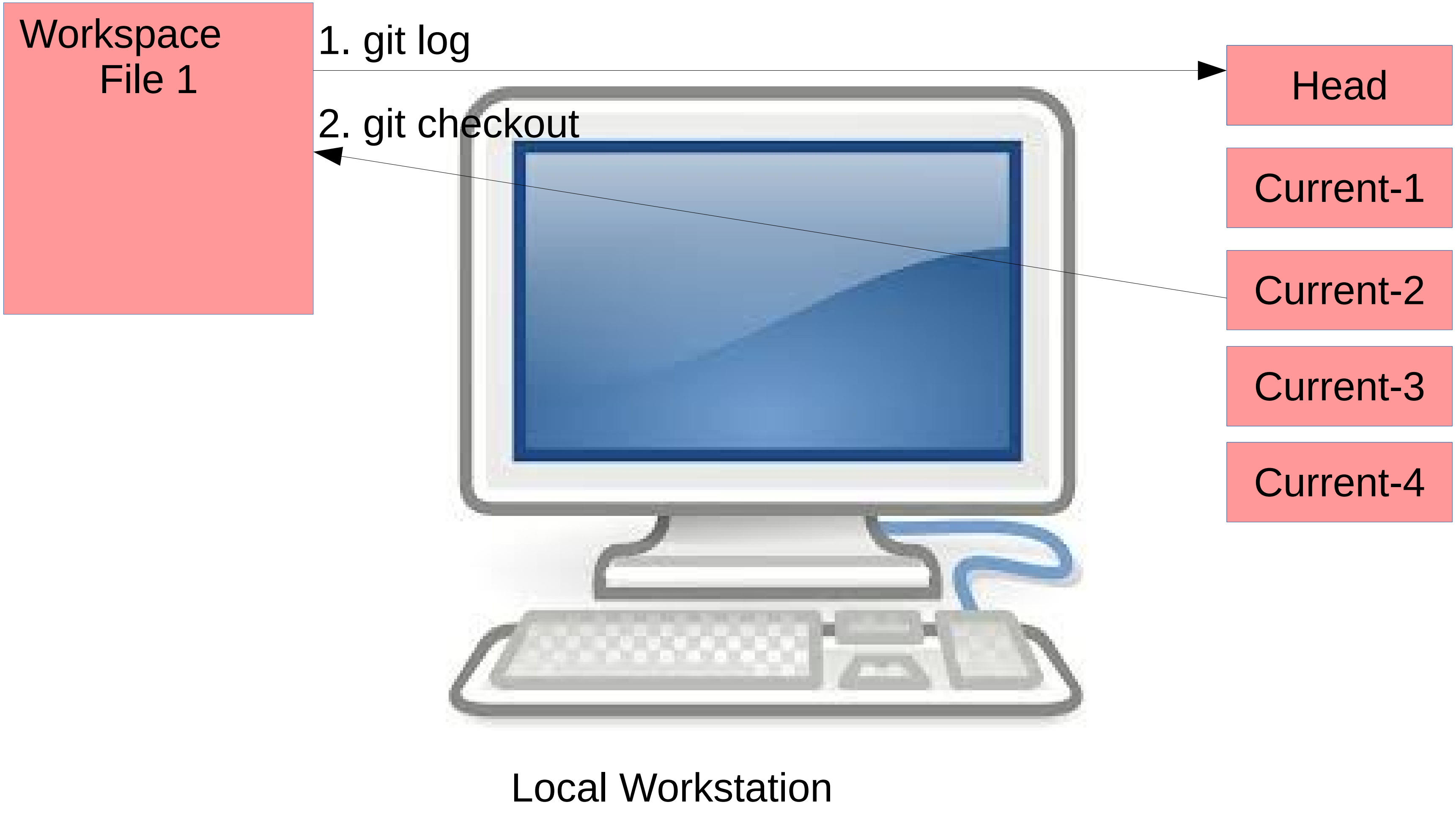
Current-3

Local Workstation

a Local Repository

- Find the version you want (git log)
- Retrieve the version you want (git checkout)
  - Puts the specified version in a working directory
    - You can look at the old version, copy from the old version, even check in the old version as the newest version.
    - Usually if you want to save the old version, check it out into a new branch (git checkout -b new_branch_name)
  - To get the most recent version (git checkout master)

Git check out



Workspace
File 1

1. git log

2. git checkout

Head

Current-1

Current-2

Current-3

Current-4

Local Workstation

Remote Repository

What is a remote repository?

- A remote repository is a copy of versions of a program stored on a remote *host*.
  - A host computer runs a program such as an SCM repository
  - If the local host fails, the repository survives on the remote host
- Users *pull* a version from the remote repository
- Users *push* a version to the repository to keep it safe

Git repository strategy

- Git differs from most repository in that all Git repositories hold the same content
  - Most SCM system have a central repository. Only one version is stored locally at a time.
  - Git stores all versions both locally and remotely
    - Checking in and checking out merges differences

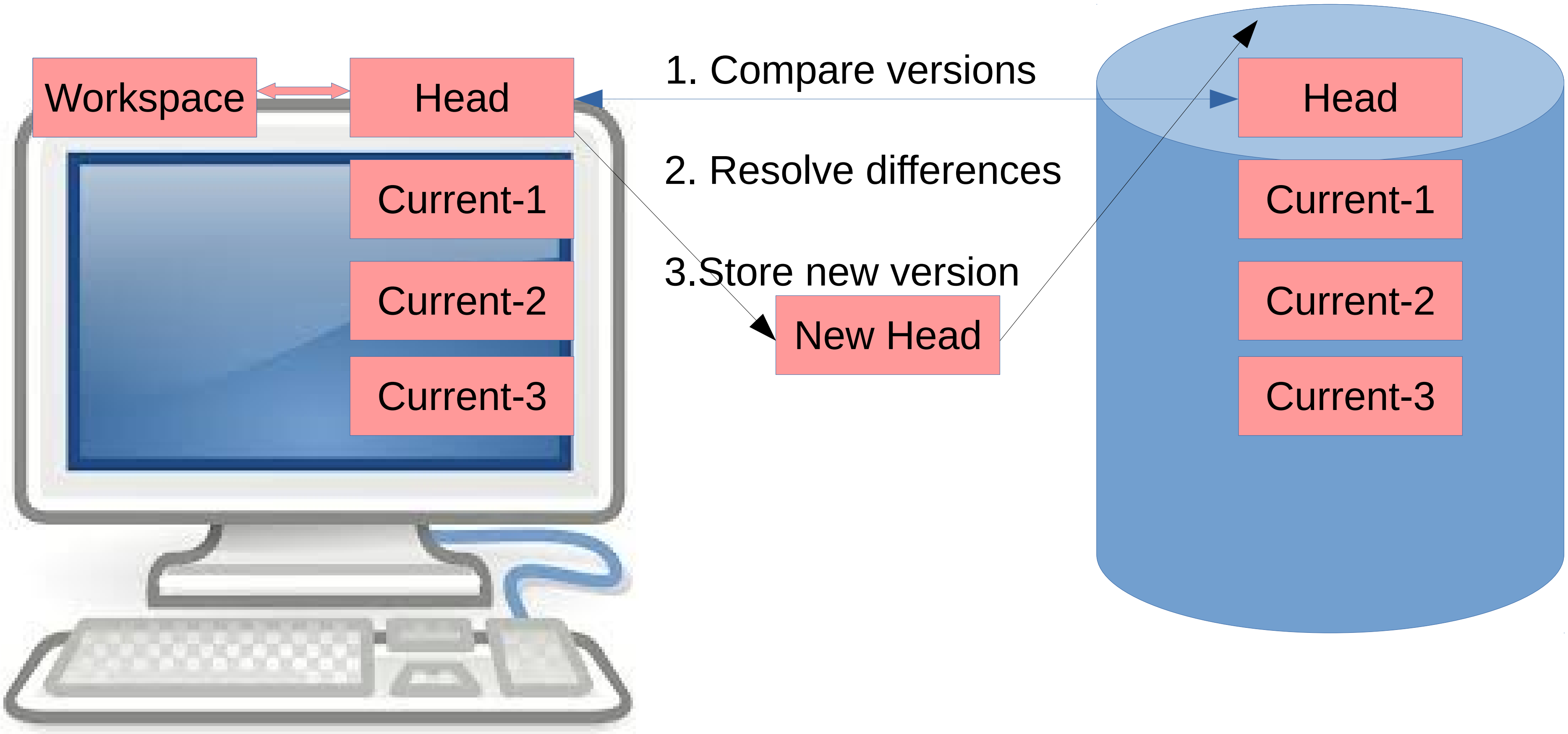
Remote Repository Set Up

- To set up a remote repository, you need access to both the remote computer and the local computer.
- On the remote computer you create a Git repository with nothing in it (git init –bare)
- On the local computer you tell git where the remote computer is (git remote add origin <user>@<remote.host>:<git repo>)

Push

- To use the remote repository, you push your files to the remote repository (git push origin master)
  - Copies all changes to the remote repository
- When pushing, Git compares the files in the local and remote repositories and makes you reconcile them
  - Only when they are the same can you push them

Git push



Workspace

Head

Current-1

Current-2

Current-3

1. Compare versions

2. Resolve differences

3.Store new version

New Head

Head

Current-1

Current-2

Current-3

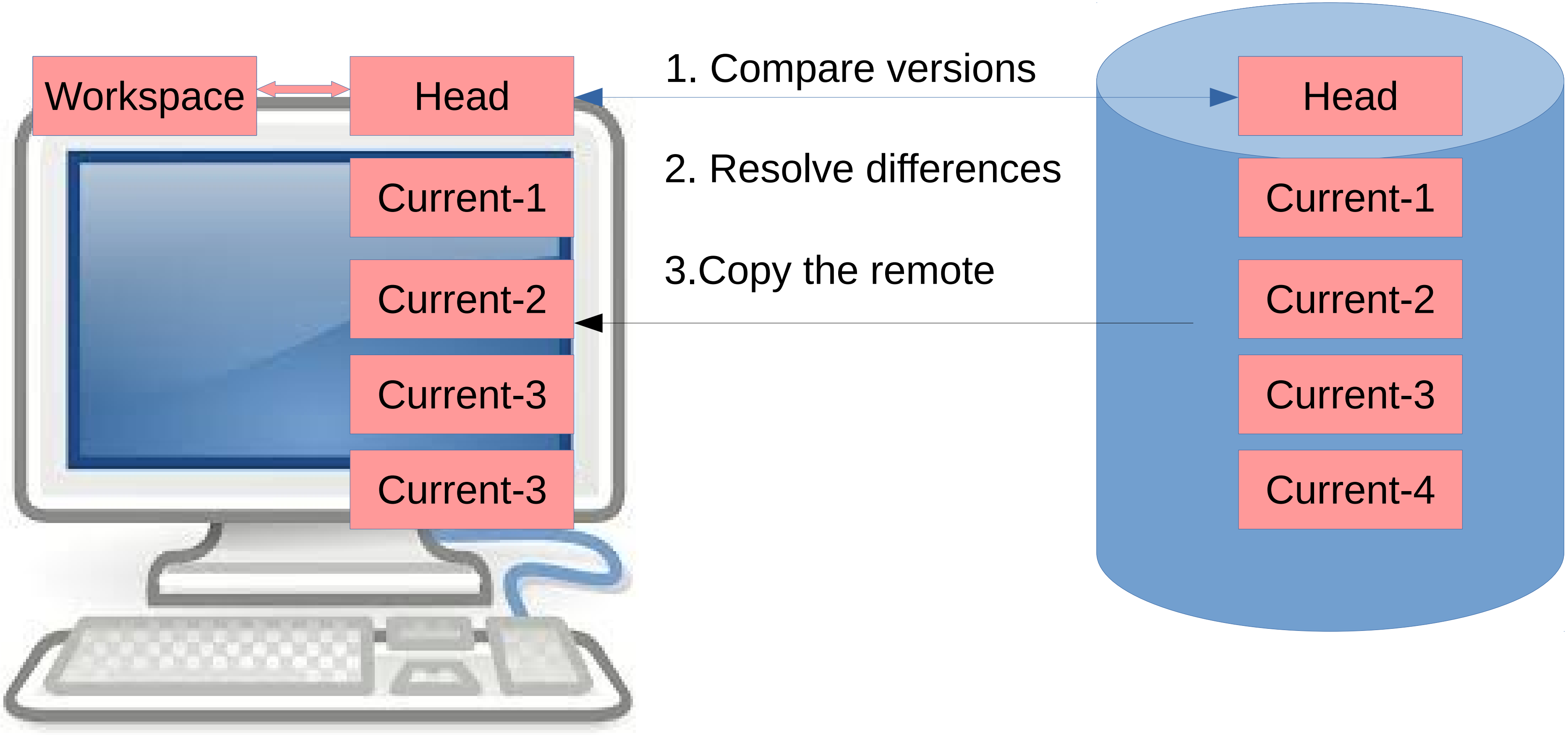
Local Workstation

Remote Host

Pull

- To retrieve the files you pull from the repository
  - You can also clone a remote repository which will create a new version with all existing versions
- There are two ways to pull
  - You can create a new local repository by cloning the remote repository (git clone <user>@<remote host>:<directory on remote host>
  - You can copy the directory into an existing local git repository after the repository is set up (git pull origin master)
    - If you pull to an existing repository, git will make you reconcile the files.

Git pull



Workspace

Head

Current-1

Current-2

Current-3

Current-3

1. Compare versions

2. Resolve differences

3.Copy the remote

Head

Current-1

Current-2

Current-3

Current-4

Local Workstation

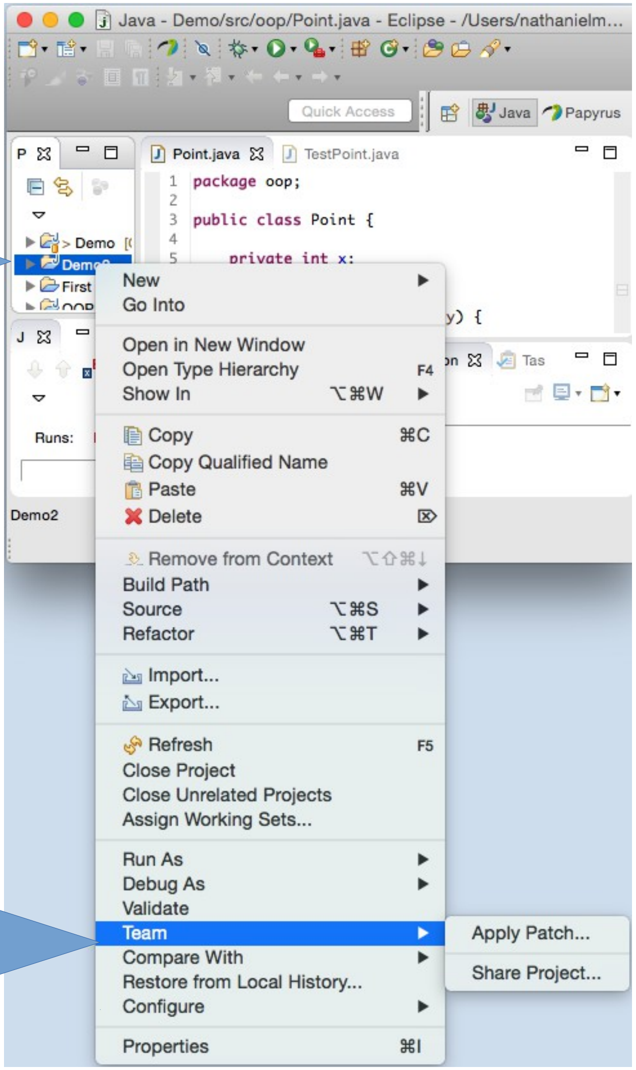
Remote Host

Git Local Repository from Eclipse

Set up Local Repository

1. Right Click on project

2. Select Team > Share Project

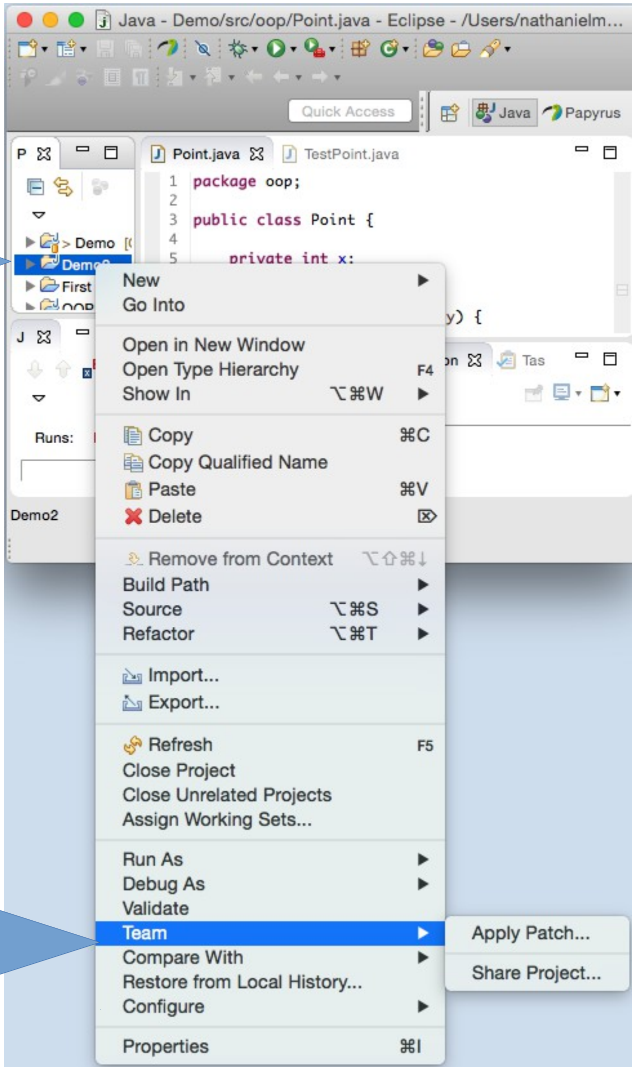


```
package oop;

public class Point {

    private int x;
```

Set up Local Repository

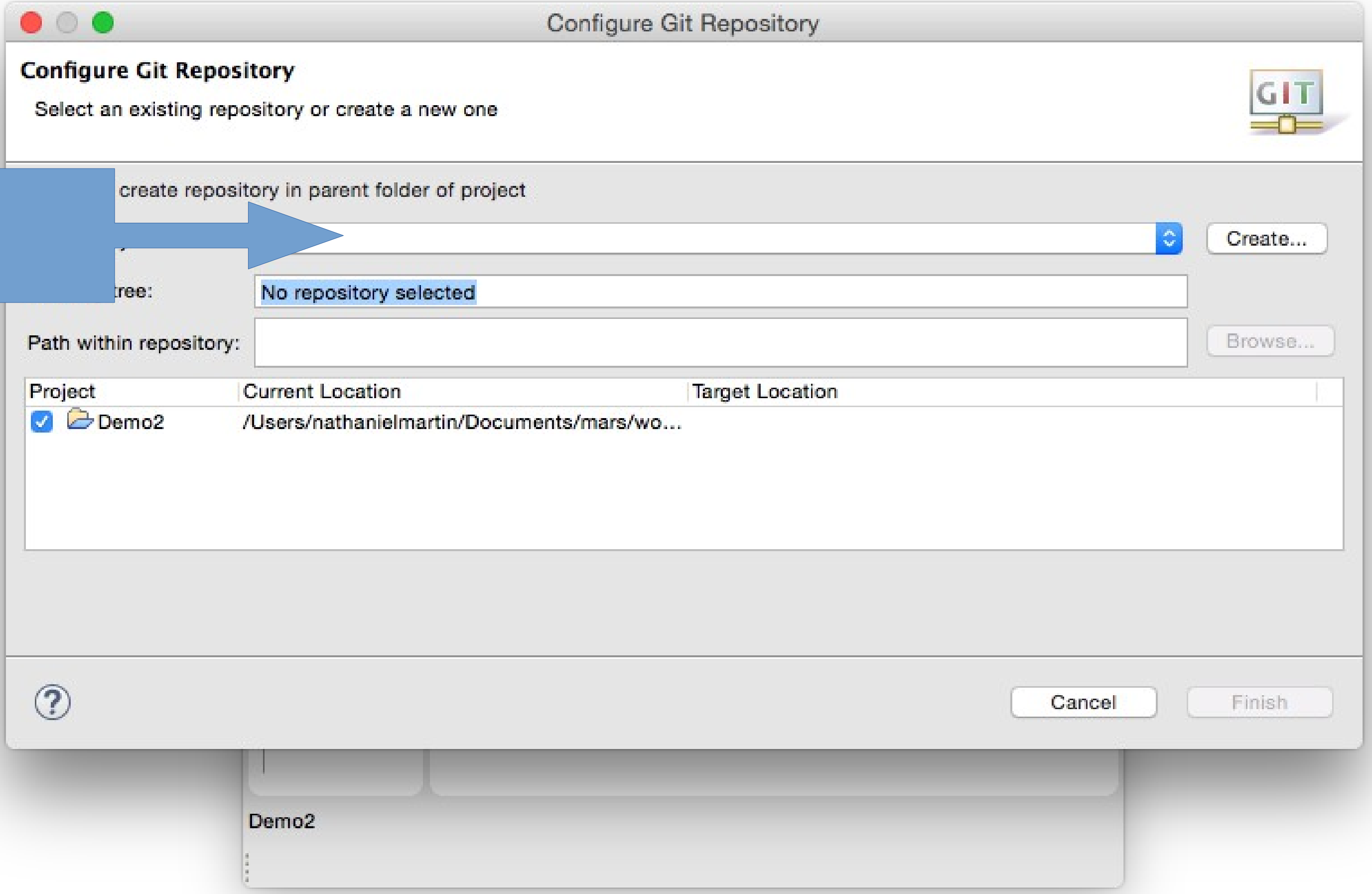
1. Right Click on project

2. Select Team > Share Project



```
package oop;

public class Point {

    private int x;
```

Specify Repository location



Configure Git Repository

Select an existing repository or create a new one

create repository in parent folder of project

ree: No repository selected

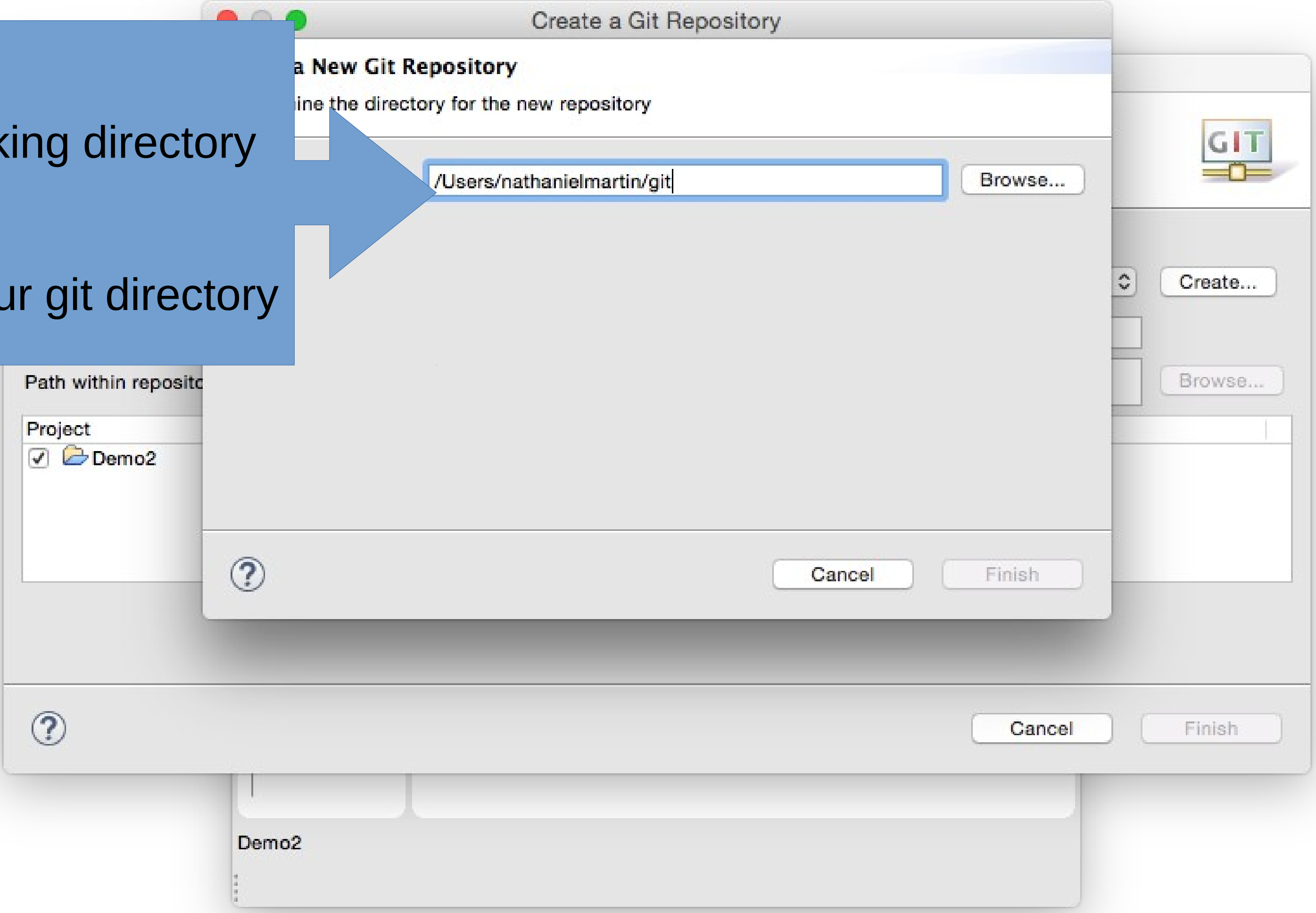
Path within repository:

Project	Current Location	Target Location
☑ Demo2	/Users/nathanielmartin/Documents/mars/wo...	

1.Click Create

Choose directory

1. Choose a directory
   Should not be in working directory
a. Select Browse
b. Create a new folder
c. Select that folder as your git directory



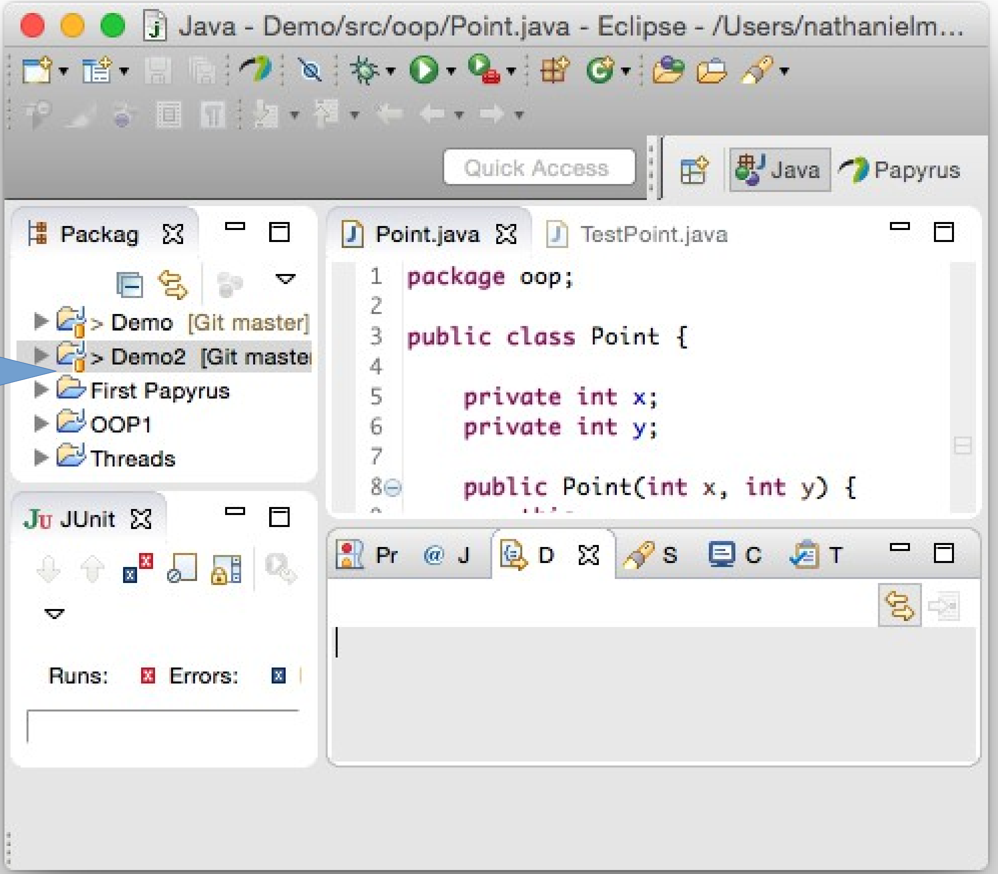
/Users/nathanielmartin/git

☑ Demo2

Demo2

Project now managed in Git

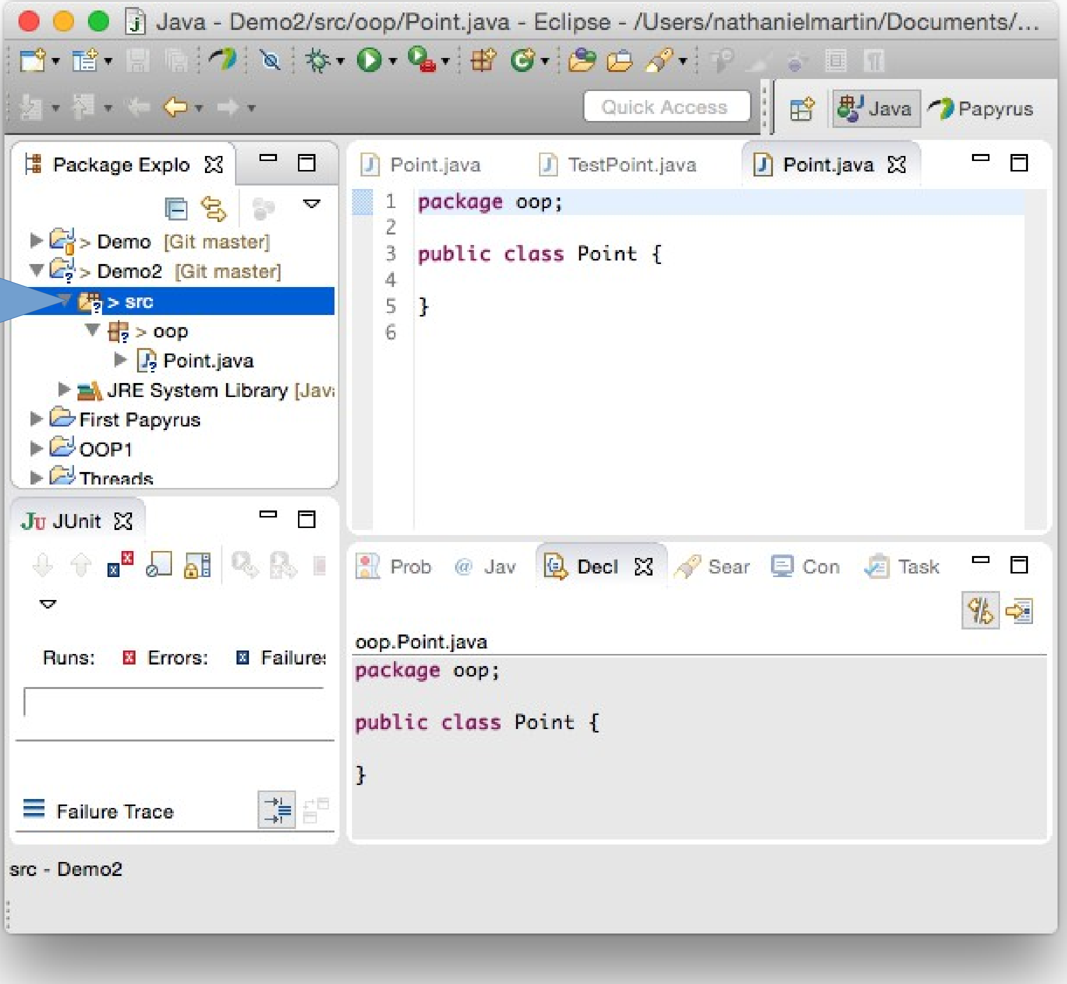
Project how marked with >



```
package oop;

public class Point {

    private int x;
    private int y;

    public Point(int x, int y) {
```

Add Content

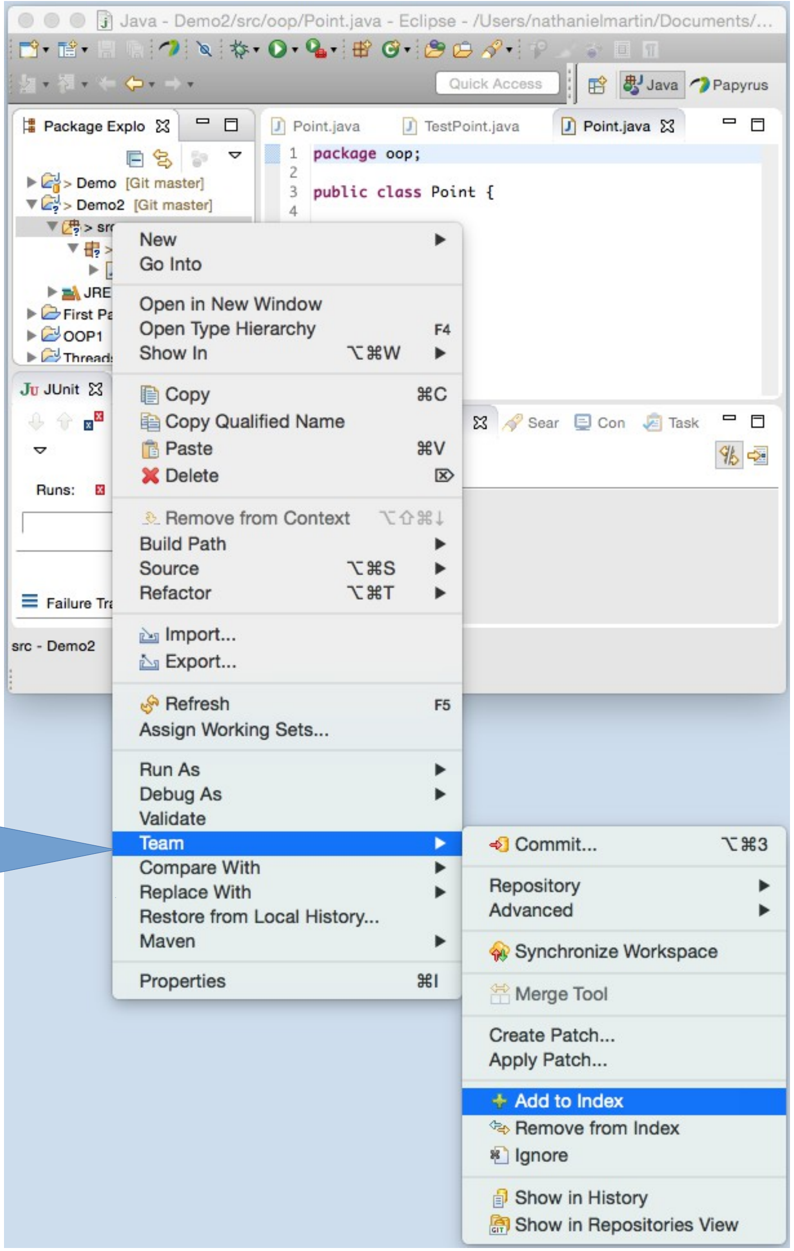
Now marked with ?
Files not checked in



```
package oop;

public class Point {

}
```

```
package oop;

public class Point {

}
```

Add to Index

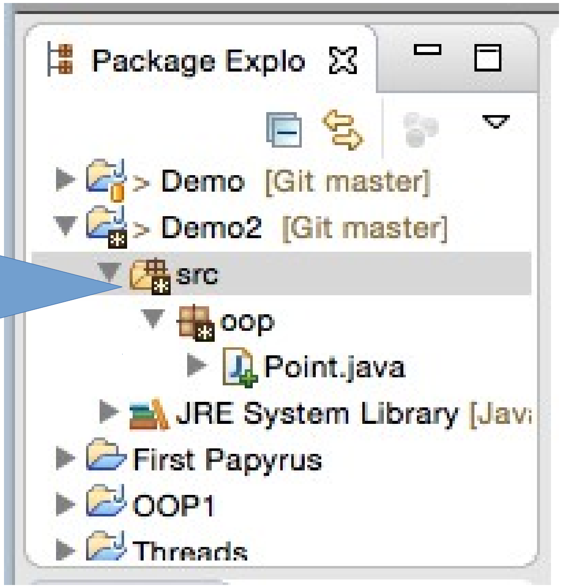


```
package oop;

public class Point {
```

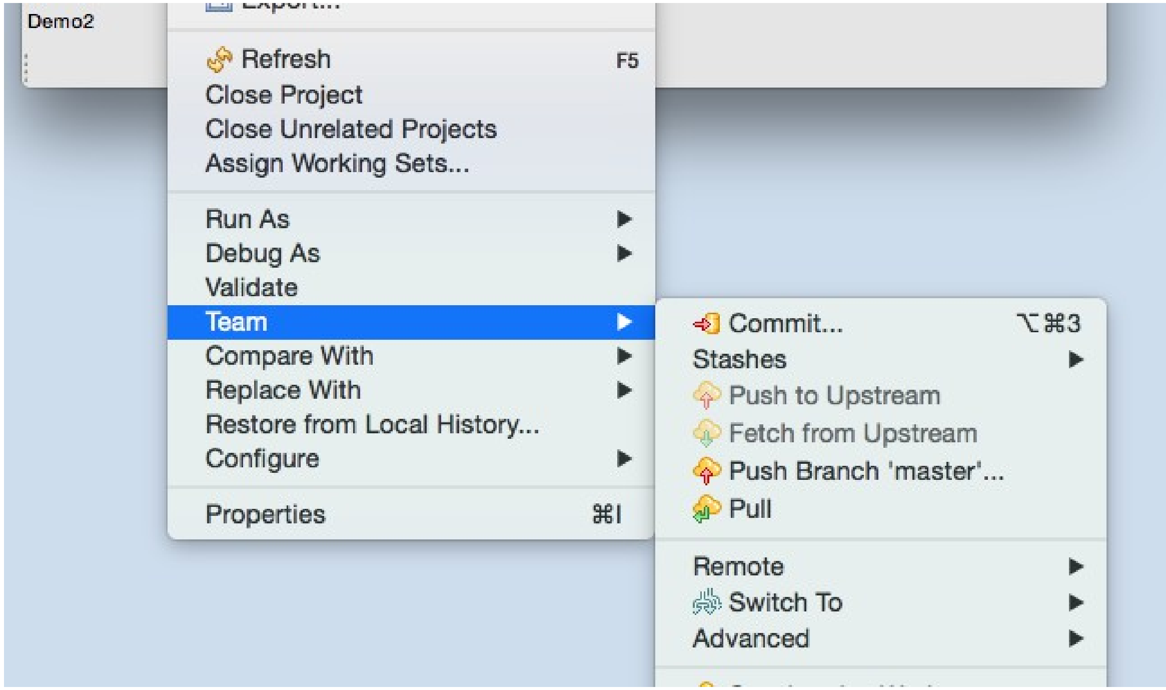
Team > Add to Index

Added to Index

Files are Stages

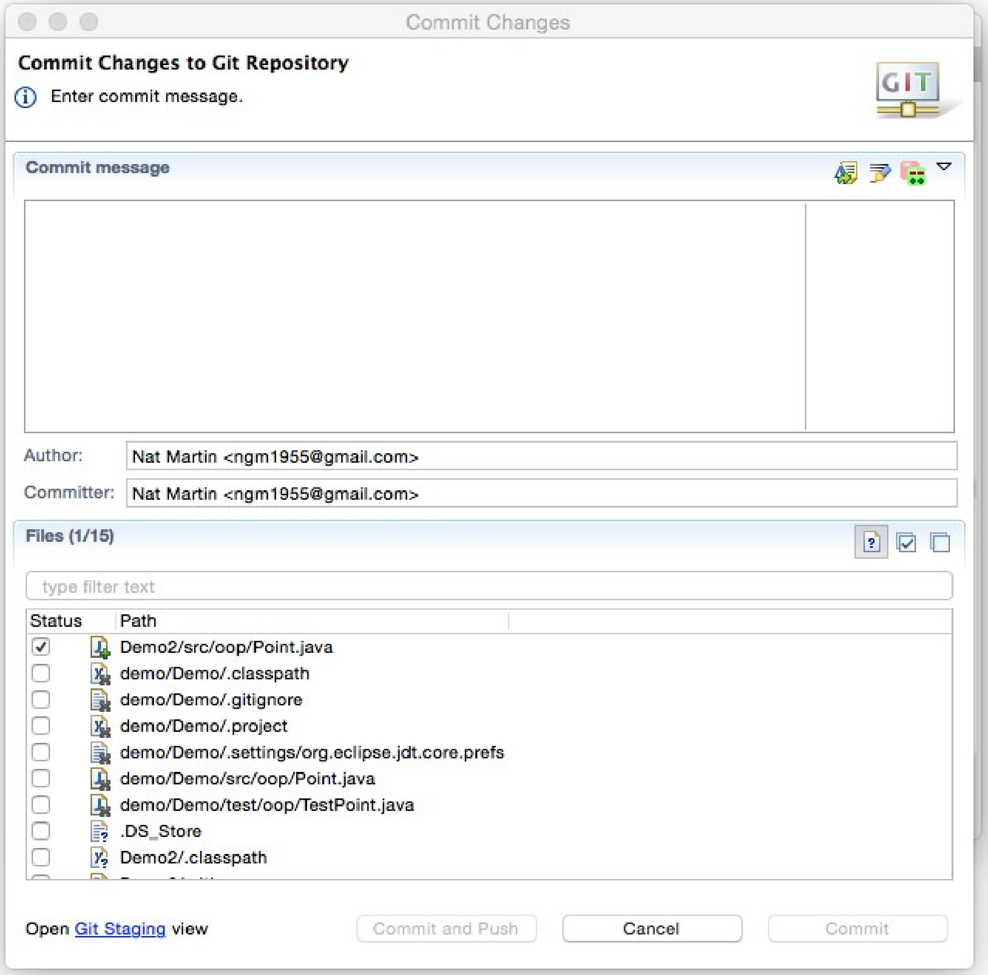


Commit



Demo2
Export...
Refresh F5
Close Project
Close Unrelated Projects
Assign Working Sets...
Run As ▶
Debug As ▶
Validate
Team ▶
Compare With ▶
Replace With ▶
Restore from Local History...
Configure ▶
Properties ⌘I
Commit... ⌥⌘3
Stashes ▶
Push to Upstream
Fetch from Upstream
Push Branch 'master'...
Pull
Remote ▶
Switch To ▶
Advanced ▶

Add Commit Message



Commit Changes to Git Repository

Enter commit message.

Commit message

Author: Nat Martin <ngm1955@gmail.com>

Committer: Nat Martin <ngm1955@gmail.com>

Files (1/15)

Status	Path
☑	Demo2/src/oop/Point.java
□	demo/Demo/.classpath
□	demo/Demo/.gitignore
□	demo/Demo/.project
□	demo/Demo/.settings/org.eclipse.jdt.core.prefs
□	demo/Demo/src/oop/Point.java
□	demo/Demo/test/oop/TestPoint.java
□	.DS_Store
□	Demo2/.classpath

Open Git Staging view

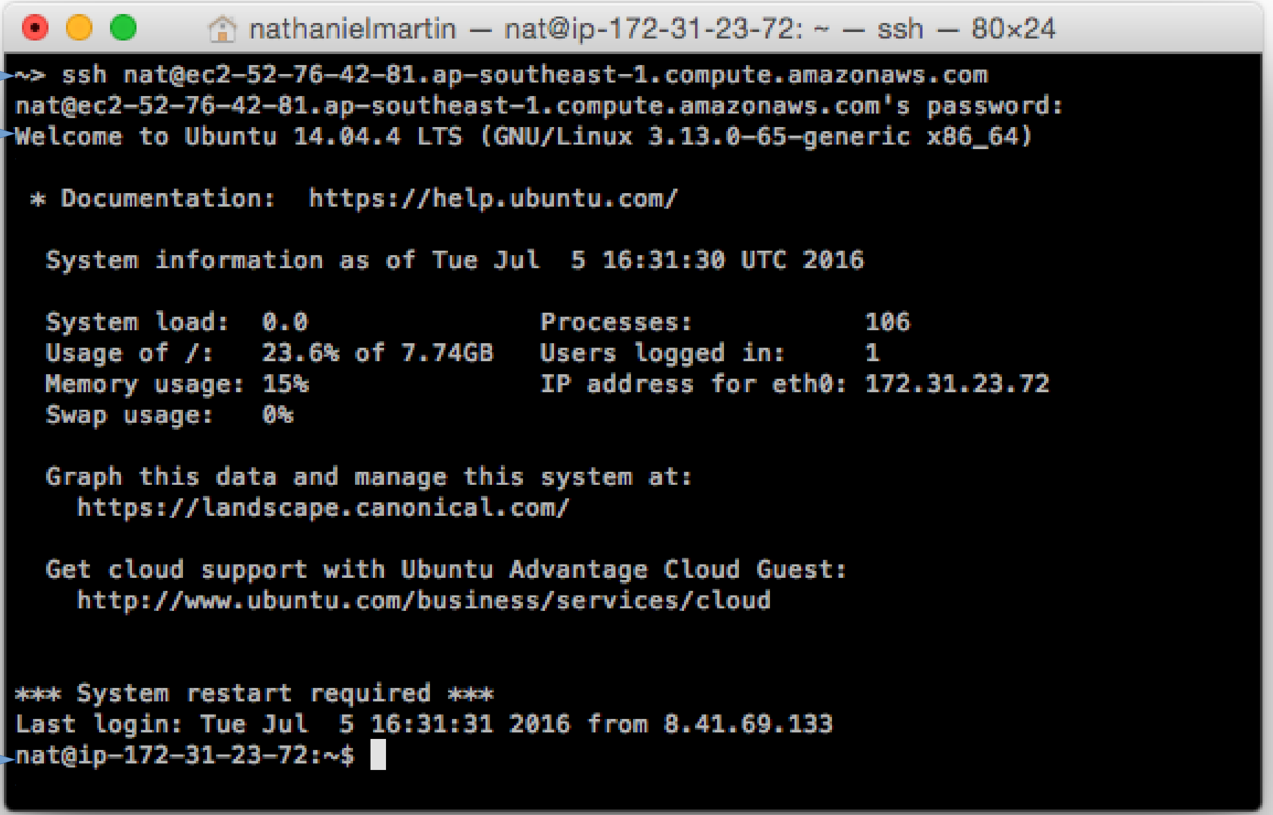
Git Remote Repository from Eclipse

Log in to the AWS server

ssh <username>@<host>

Enter password

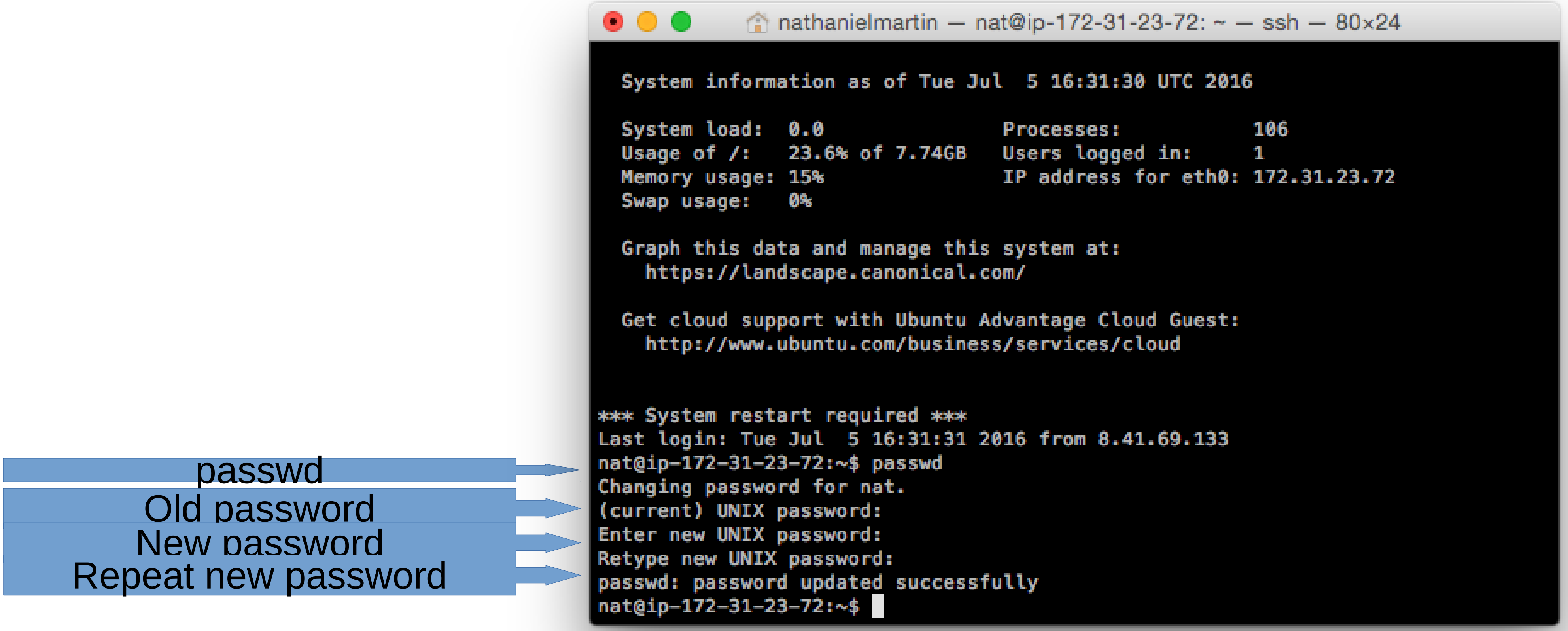
Prompt from remote host



```
~> ssh nat@ec2-52-76-42-81.ap-southeast-1.compute.amazonaws.com
nat@ec2-52-76-42-81.ap-southeast-1.compute.amazonaws.com's password:
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 3.13.0-65-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

  System information as of Tue Jul  5 16:31:30 UTC 2016

  System load:  0.0               Processes:           106
  Usage of /:   23.6% of 7.74GB   Users logged in:     1
  Memory usage: 15%               IP address for eth0: 172.31.23.72
  Swap usage:   0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud


*** System restart required ***
Last login: Tue Jul  5 16:31:31 2016 from 8.41.69.133
nat@ip-172-31-23-72:~$ 
```

Username: given to you in class
Hostname: ec2-52-76-42-81.ap-southeast-1.compute.amazonaws.com
Host IP: 52.76.42.81
Password: given to you in class

Change password if first login



passwd

Old password

New password

Repeat new password

```
  System information as of Tue Jul  5 16:31:30 UTC 2016

  System load:  0.0               Processes:           106
  Usage of /:   23.6% of 7.74GB   Users logged in:     1
  Memory usage: 15%               IP address for eth0: 172.31.23.72
  Swap usage:   0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud


*** System restart required ***
Last login: Tue Jul  5 16:31:31 2016 from 8.41.69.133
nat@ip-172-31-23-72:~$ passwd
Changing password for nat.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
nat@ip-172-31-23-72:~$ 
```

Write down your new password

Create a remote Git repository

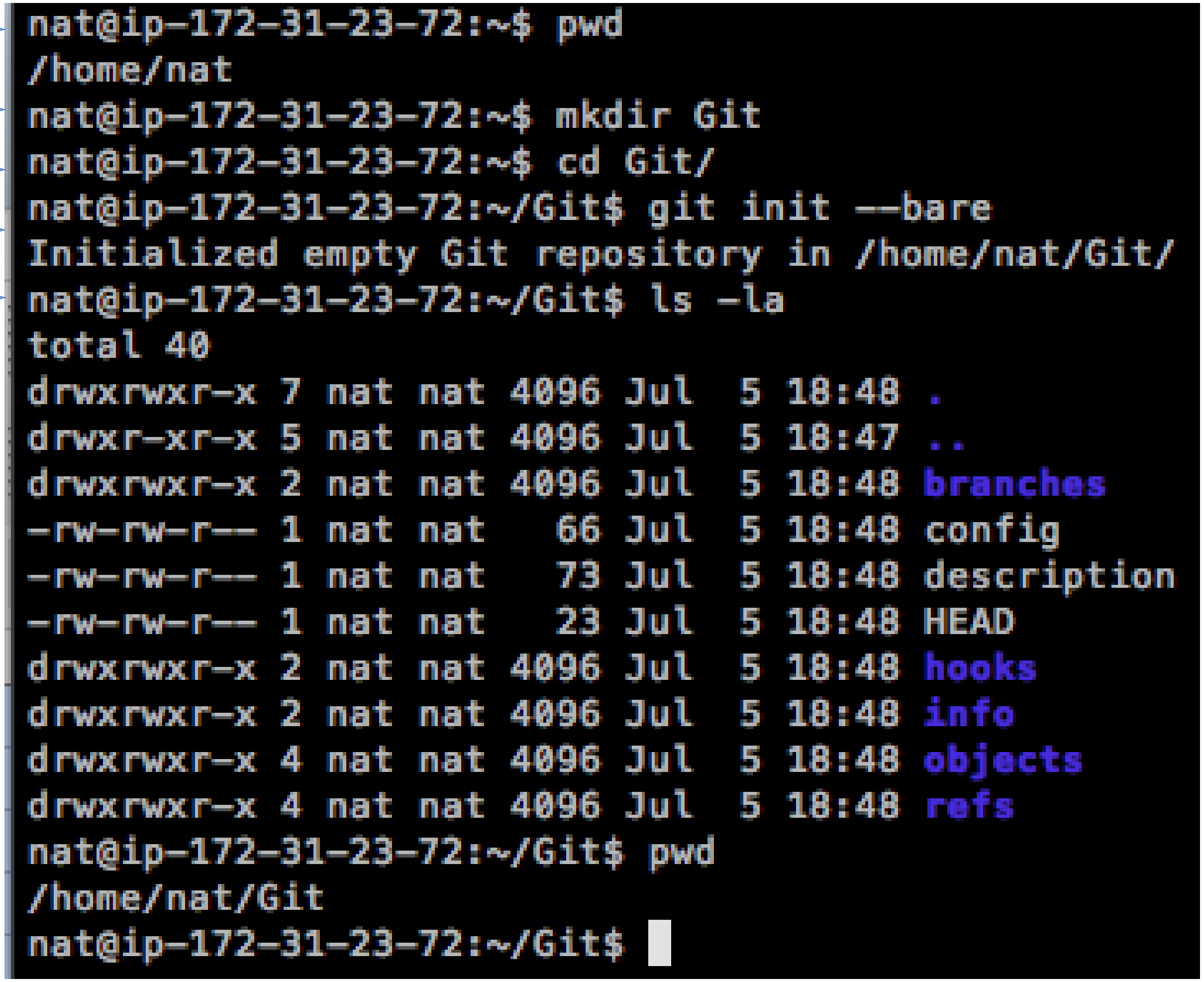
In home directory

Create a “Git” directory

Enter to the Git directory

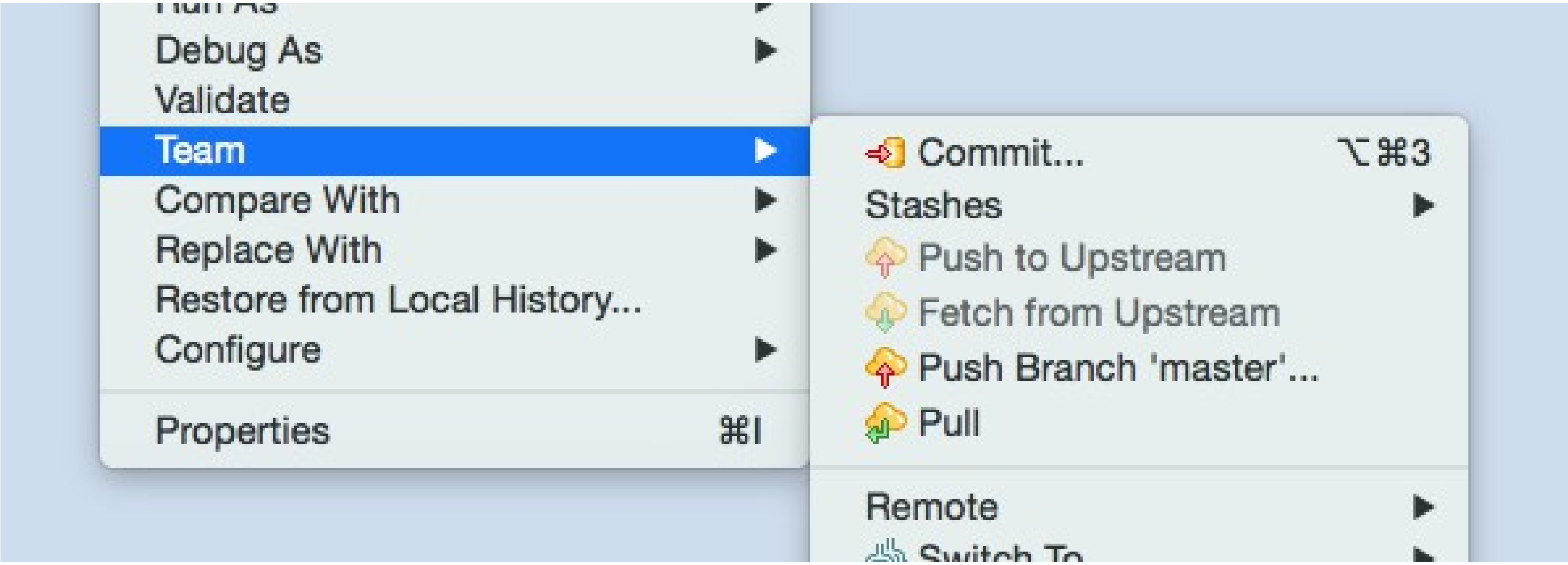
Initialize the remote repository

Git directory contains empty repo

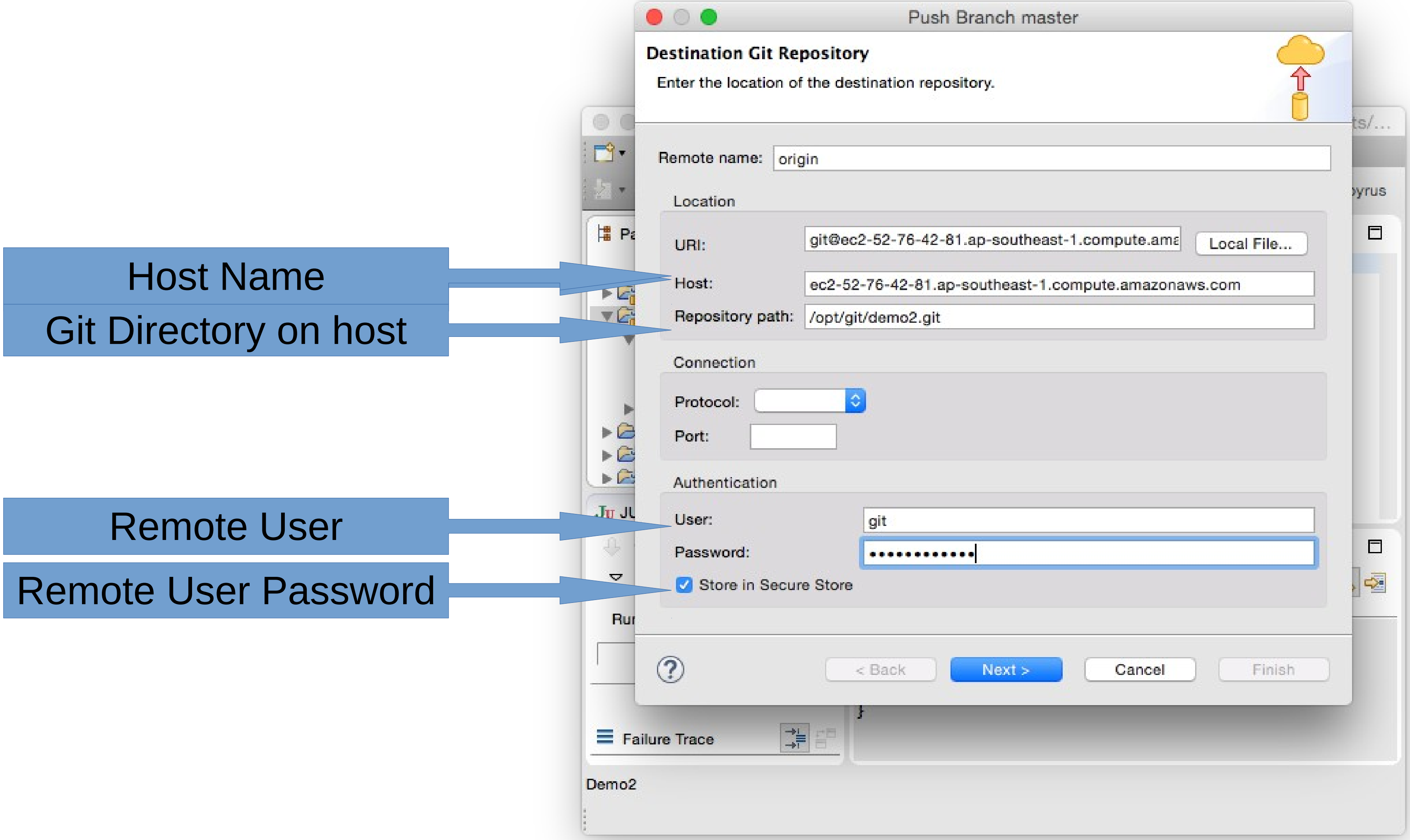


```
nat@ip-172-31-23-72:~$ pwd
/home/nat
nat@ip-172-31-23-72:~$ mkdir Git
nat@ip-172-31-23-72:~$ cd Git/
nat@ip-172-31-23-72:~/Git$ git init --bare
Initialized empty Git repository in /home/nat/Git/
nat@ip-172-31-23-72:~/Git$ ls -la
total 40
drwxrwxr-x 7 nat nat 4096 Jul  5 18:48 .
drwxr-xr-x 5 nat nat 4096 Jul  5 18:47 ..
drwxrwxr-x 2 nat nat 4096 Jul  5 18:48 branches
-rw-rw-r-- 1 nat nat   66 Jul  5 18:48 config
-rw-rw-r-- 1 nat nat   73 Jul  5 18:48 description
-rw-rw-r-- 1 nat nat   23 Jul  5 18:48 HEAD
drwxrwxr-x 2 nat nat 4096 Jul  5 18:48 hooks
drwxrwxr-x 2 nat nat 4096 Jul  5 18:48 info
drwxrwxr-x 4 nat nat 4096 Jul  5 18:48 objects
drwxrwxr-x 4 nat nat 4096 Jul  5 18:48 refs
nat@ip-172-31-23-72:~/Git$ pwd
/home/nat/Git
nat@ip-172-31-23-72:~/Git$ 
```

Choose Push Branch 'master'



Set Pointer to Remote Host



Host Name

Git Directory on host

Remote User

Remote User Password

Push Branch master

Destination Git Repository

Enter the location of the destination repository.

Remote name: origin

Location

URI: git@ec2-52-76-42-81.ap-southeast-1.compute.ama

Host: ec2-52-76-42-81.ap-southeast-1.compute.amazonaws.com

Repository path: /opt/git/demo2.git

Connection

Protocol:

Port:

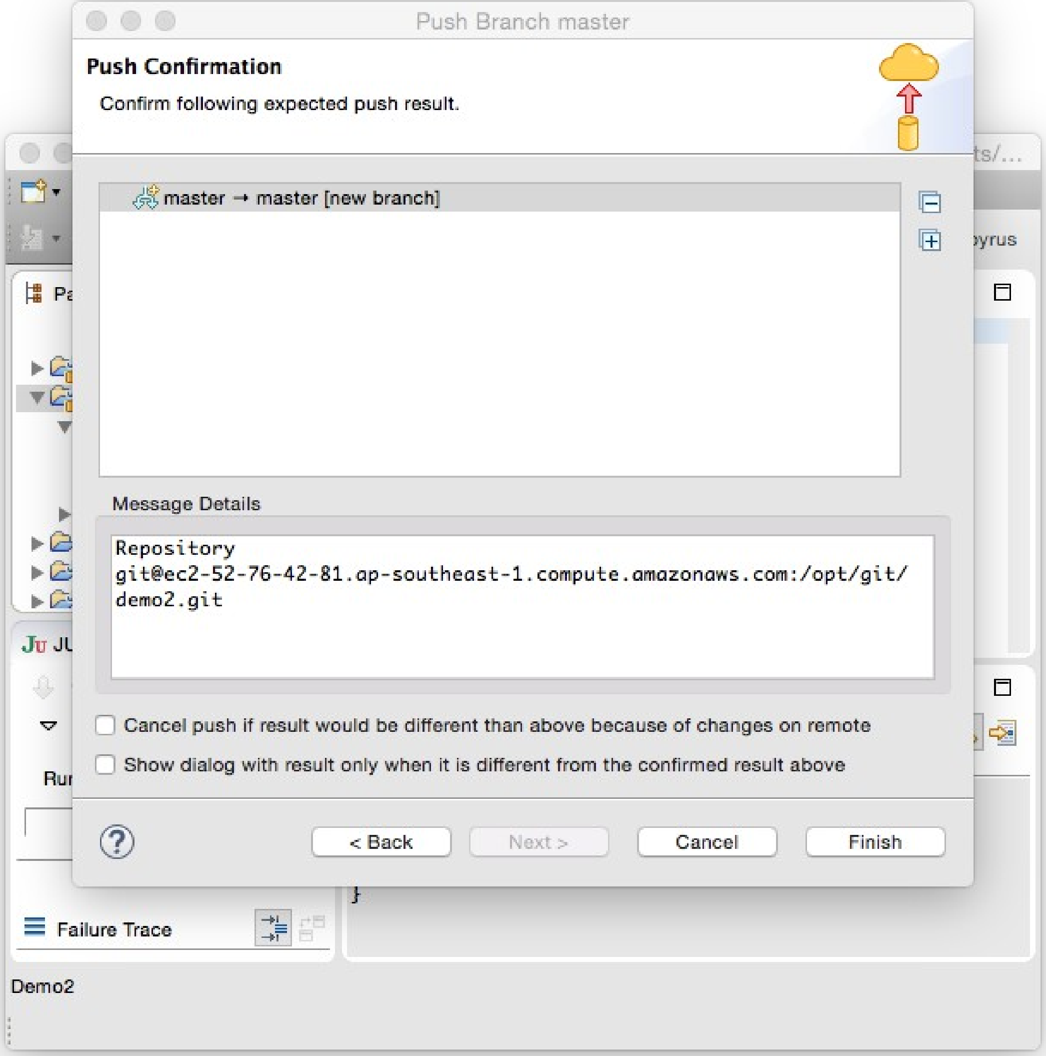
Authentication

User: git

Password: •••••••••••••

☑ Store in Secure Store

Push Acknowledged



Push Confirmation

Confirm following expected push result.

master → master [new branch]

Message Details

```
Repository
git@ec2-52-76-42-81.ap-southeast-1.compute.amazonaws.com:/opt/git/
demo2.git
```

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

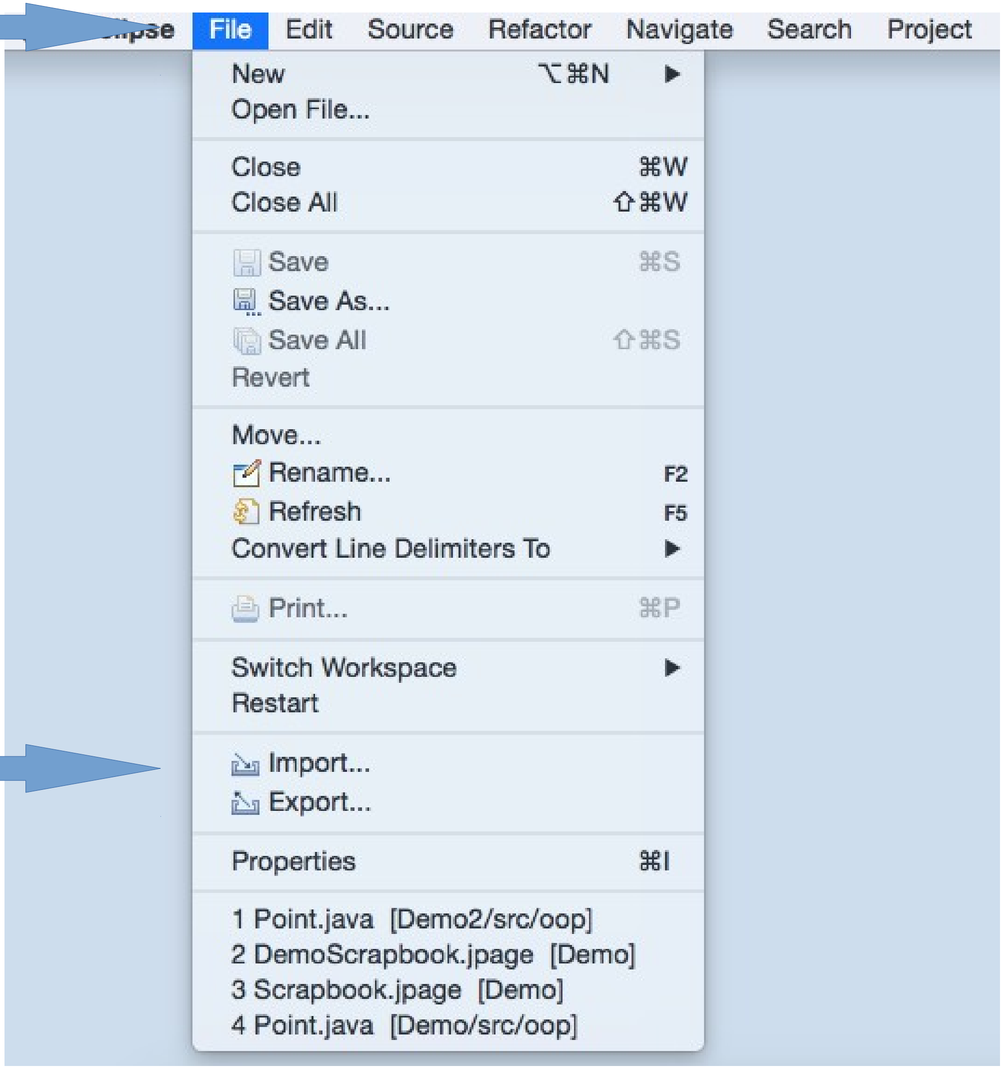
□ Show dialog with result only when it is different from the confirmed result above

Clone Remote Repository from Eclipse

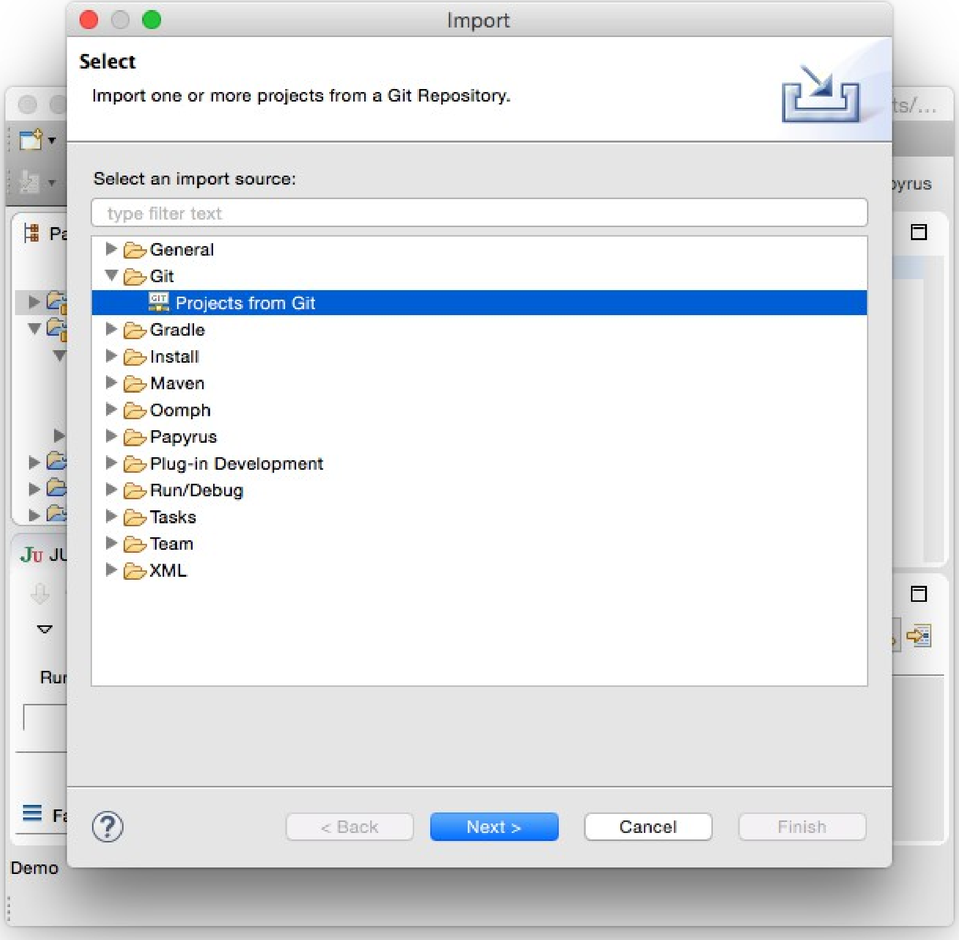
Import a project from Git

From the file menu

Choose Import...



Select Projects from Git



Import

Select

Import one or more projects from a Git Repository.

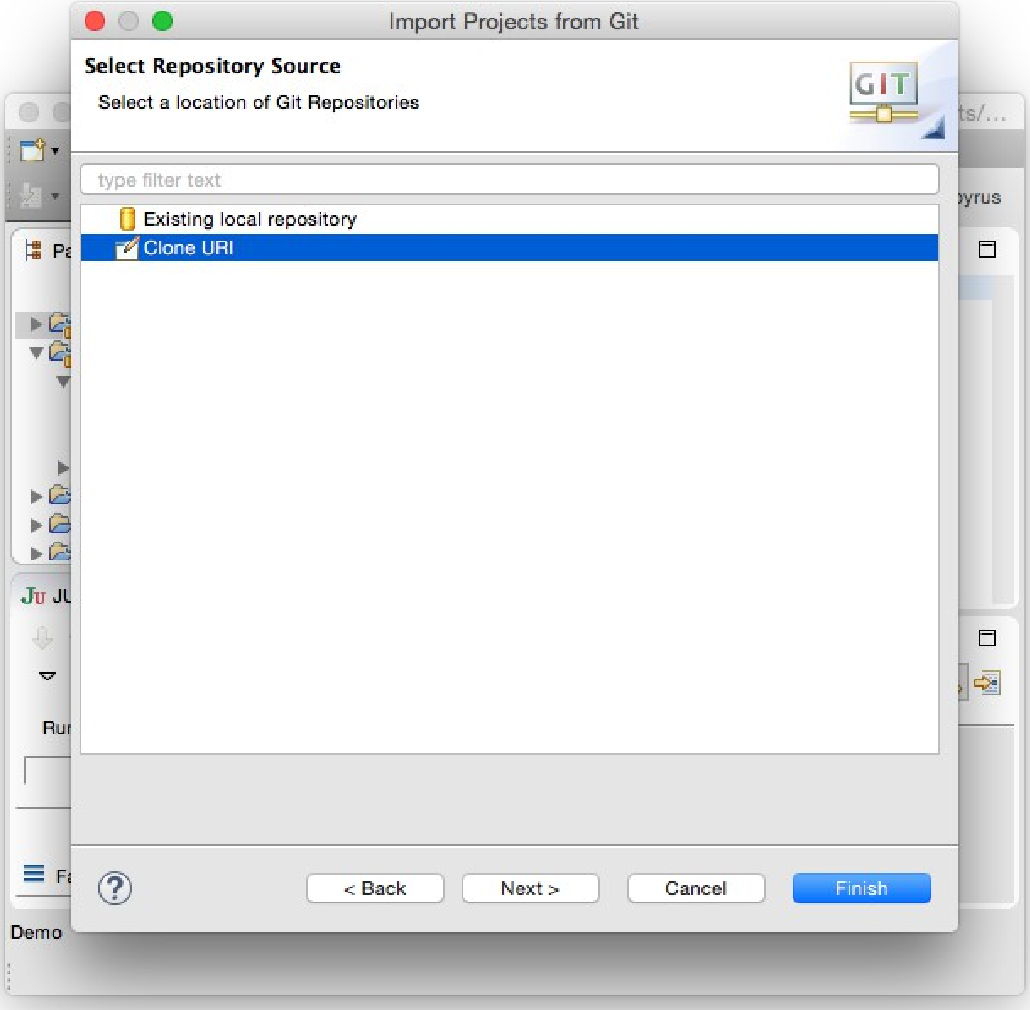
Select an import source:

type filter text

- General
- Git
  - Projects from Git
- Gradle
- Install
- Maven
- Oomph
- Papyrus
- Plug-in Development
- Run/Debug
- Tasks
- Team
- XML

Select Clone URI

Clone URI will let you download a project that is stored on a remote host. The URI represents the host and the directory that contains the Git repository. You will download it into a new Git repository.



Import Projects from Git

Select Repository Source

Select a location of Git Repositories

type filter text

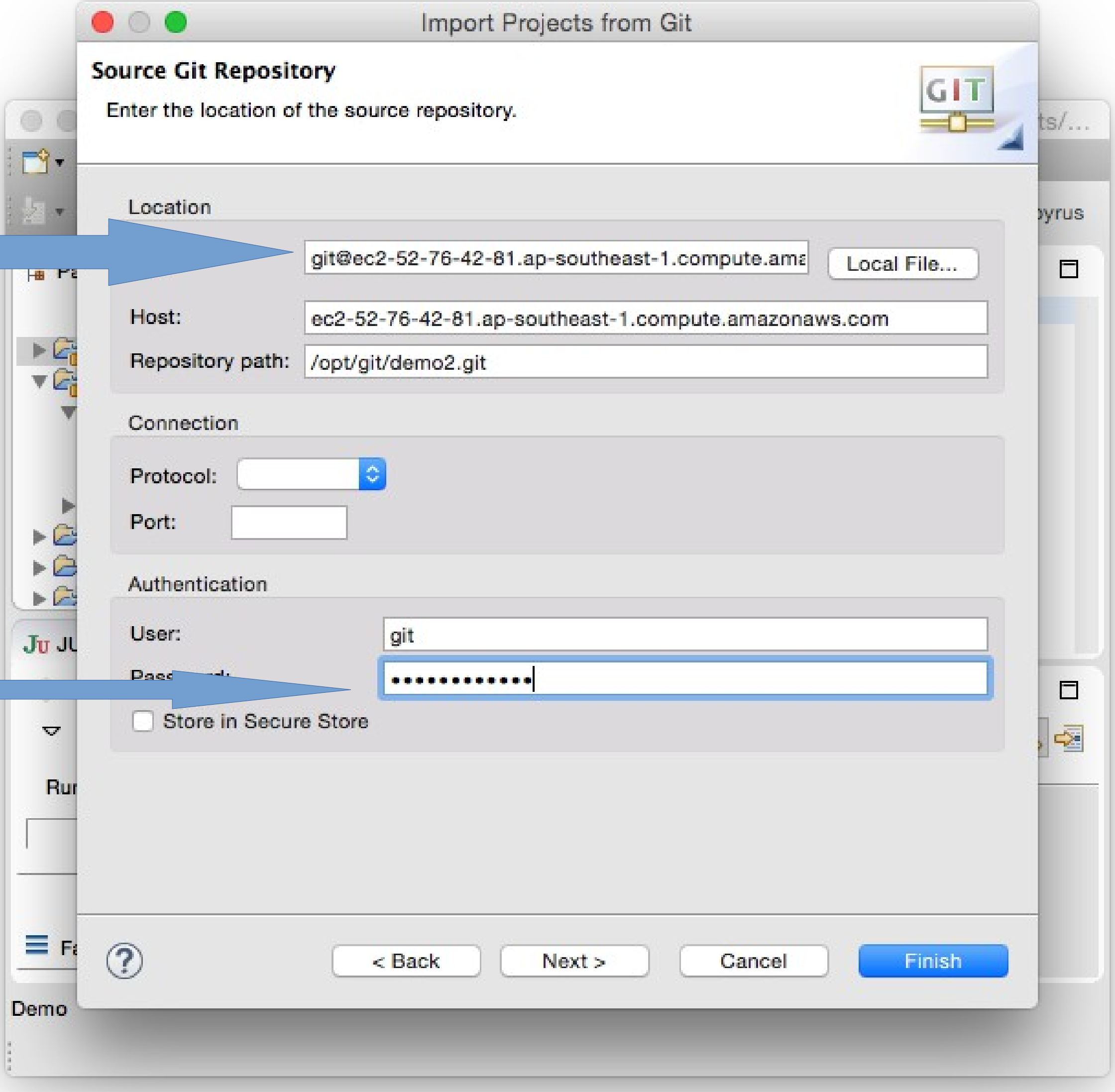
Existing local repository

Clone URI

Enter the URI

Enter the URI here:
It will be parsed into other fields

The password for "git" goes here



Source Git Repository

Enter the location of the source repository.

Location

git@ec2-52-76-42-81.ap-southeast-1.compute.ama

Host: ec2-52-76-42-81.ap-southeast-1.compute.amazonaws.com

Repository path: /opt/git/demo2.git

Connection

Protocol:

Port:

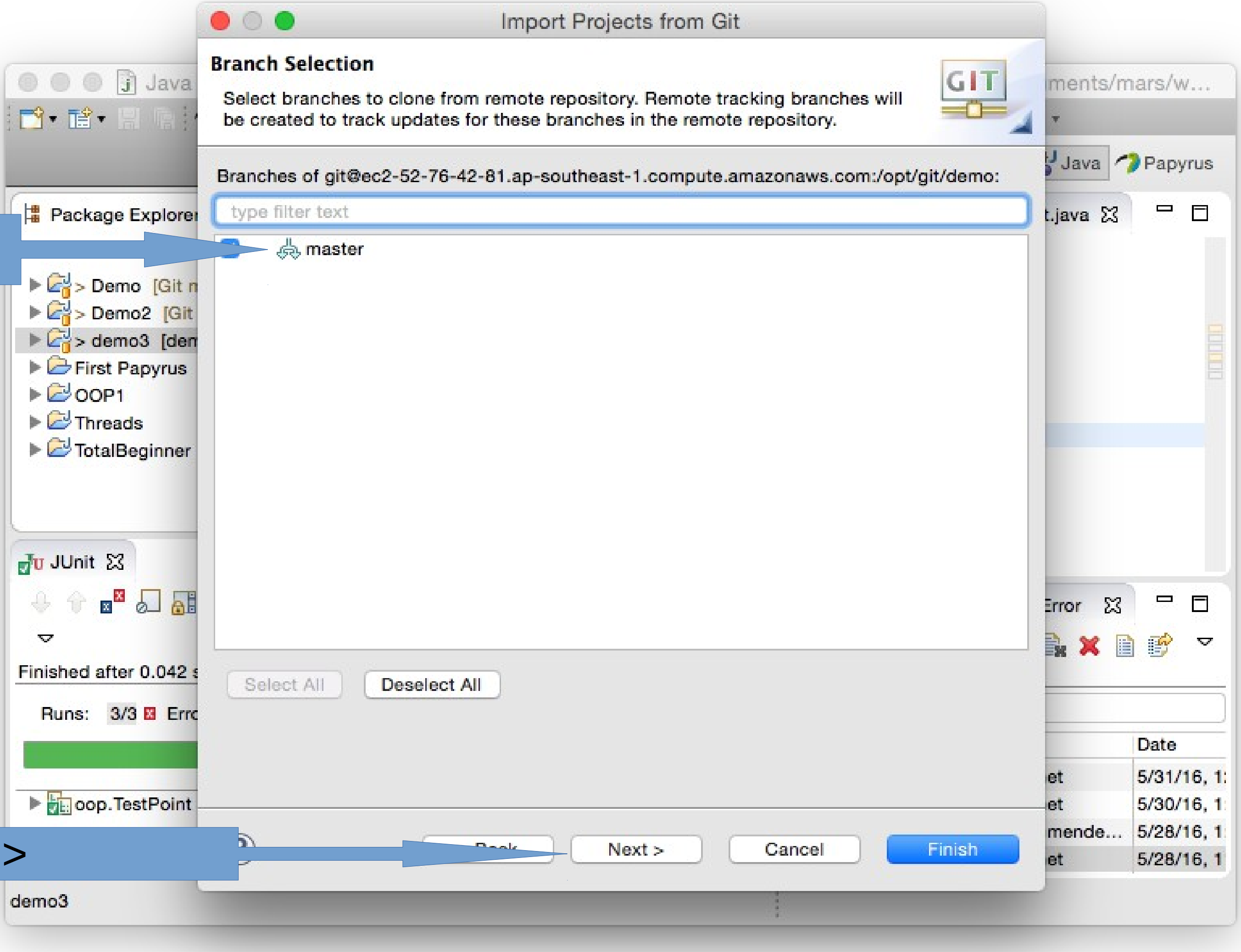
Authentication

User: git

Password: •••••••••••••

□ Store in Secure Store

Select Branch



Import Projects from Git

Branch Selection

Select branches to clone from remote repository. Remote tracking branches will be created to track updates for these branches in the remote repository.

Branches of git@ec2-52-76-42-81.ap-southeast-1.compute.amazonaws.com:/opt/git/demo:

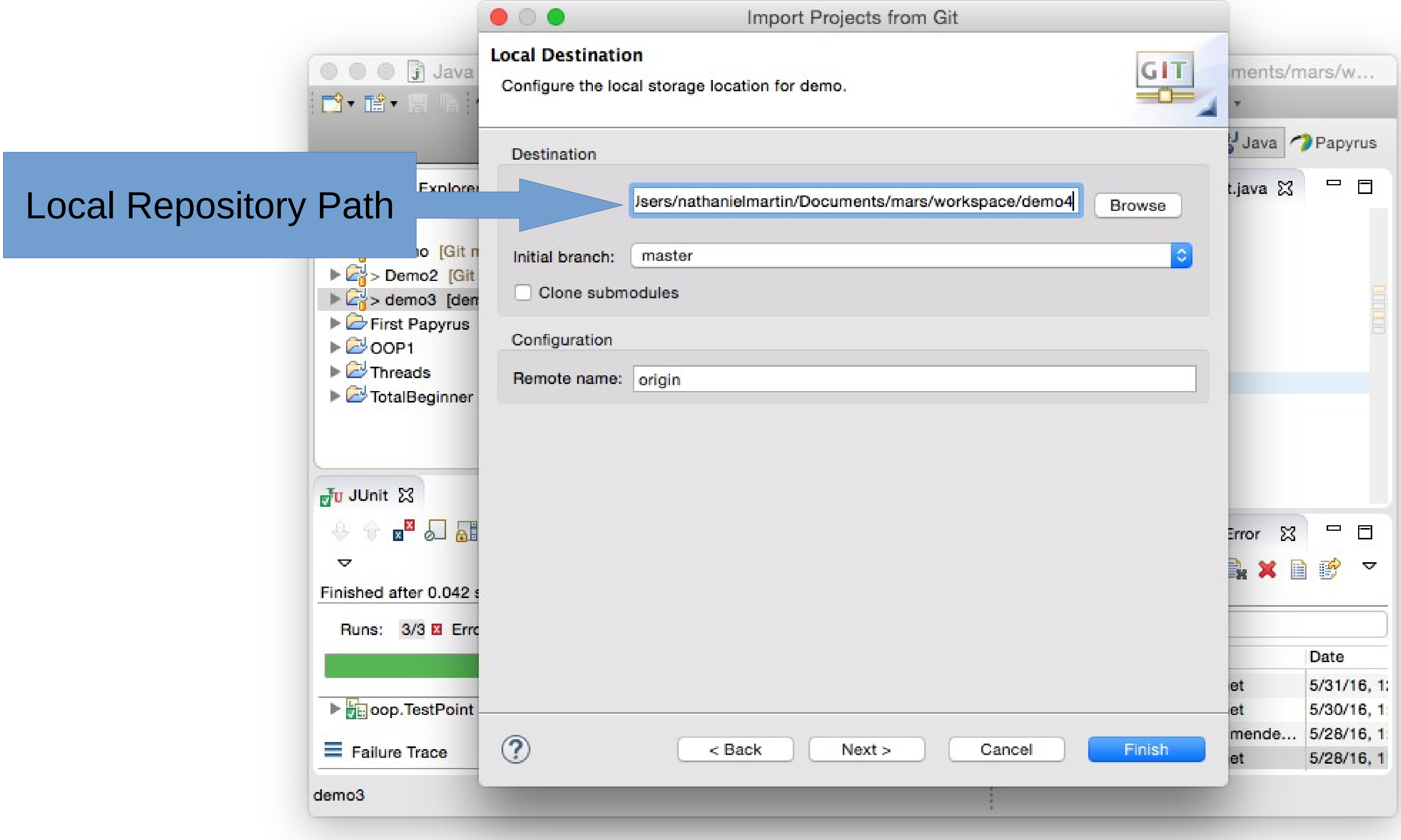
type filter text

☑ master

Select master

Click Next >

Indicate the local repository



Local Repository Path

Import Projects from Git

Local Destination

Configure the local storage location for demo.

Destination

/Users/nathanielmartin/Documents/mars/workspace/demo4

Initial branch: master

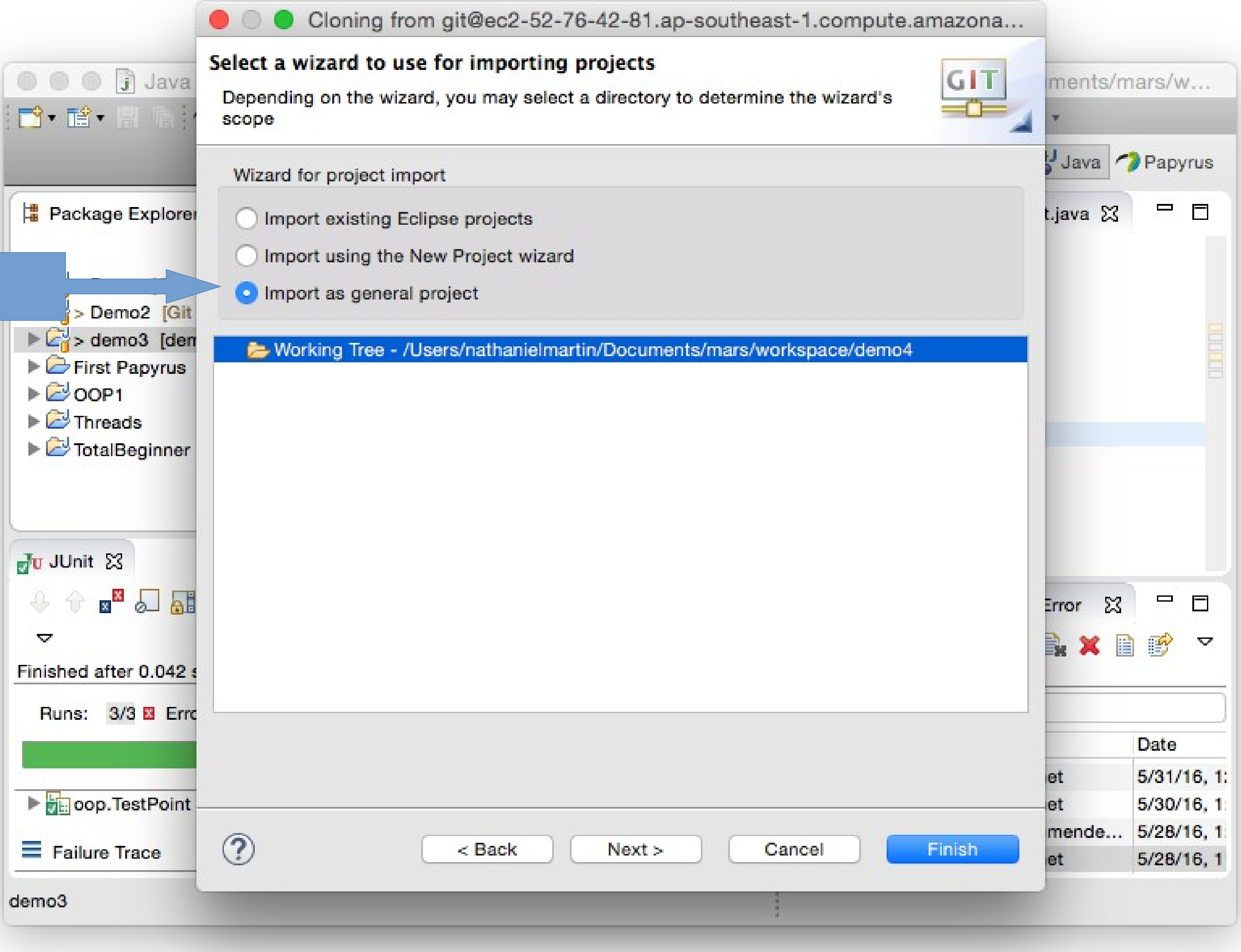
□ Clone submodules

Configuration

Remote name: origin

Select Project

Select general project



Cloning from git@ec2-52-76-42-81.ap-southeast-1.compute.amazona…

Select a wizard to use for importing projects

Depending on the wizard, you may select a directory to determine the wizard's scope

Wizard for project import

○ Import existing Eclipse projects

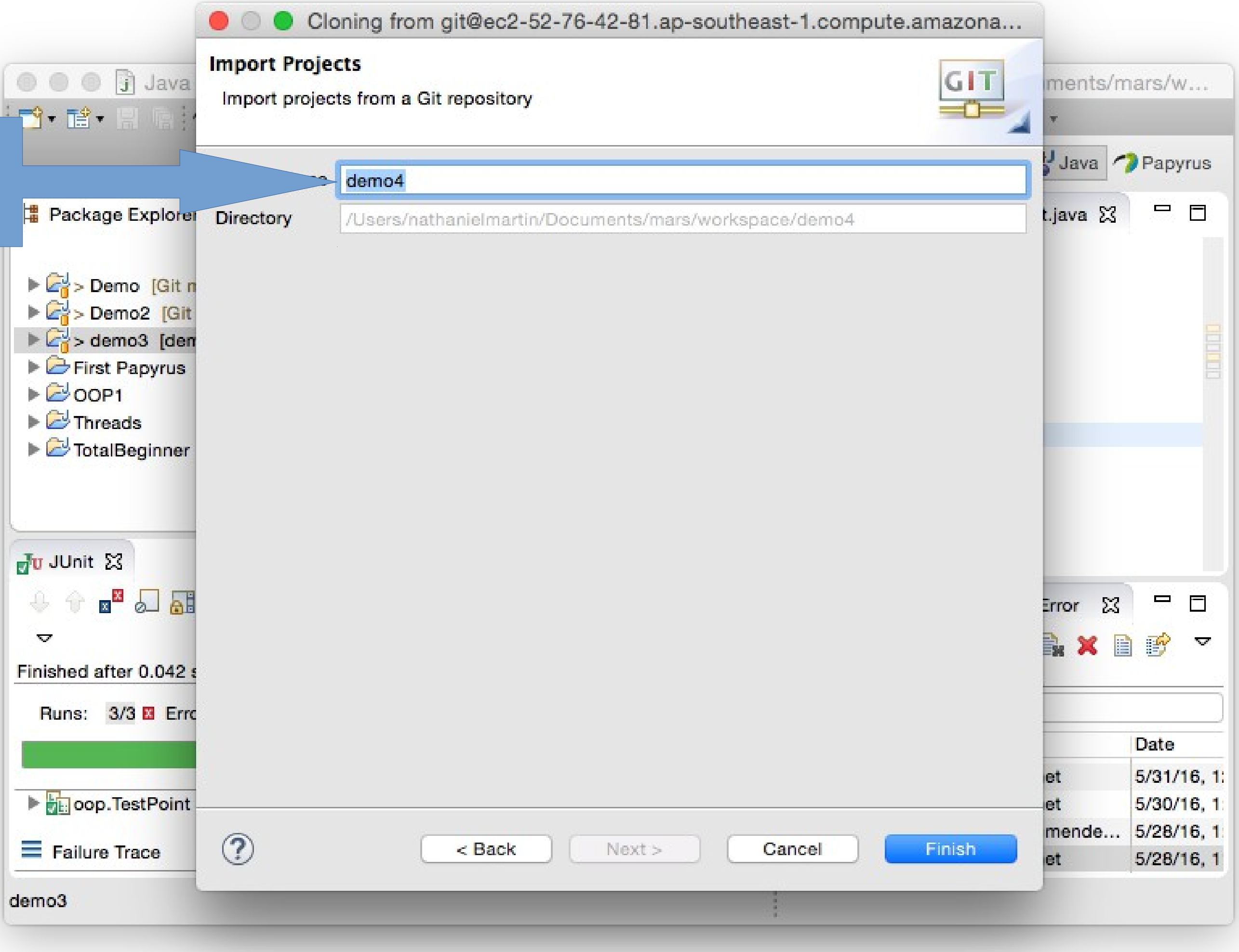
○ Import using the New Project wizard

● Import as general project

Working Tree - /Users/nathanielmartin/Documents/mars/workspace/demo4

Give the project a name

Demo4



Cloning from git@ec2-52-76-42-81.ap-southeast-1.compute.amazona…

Import Projects

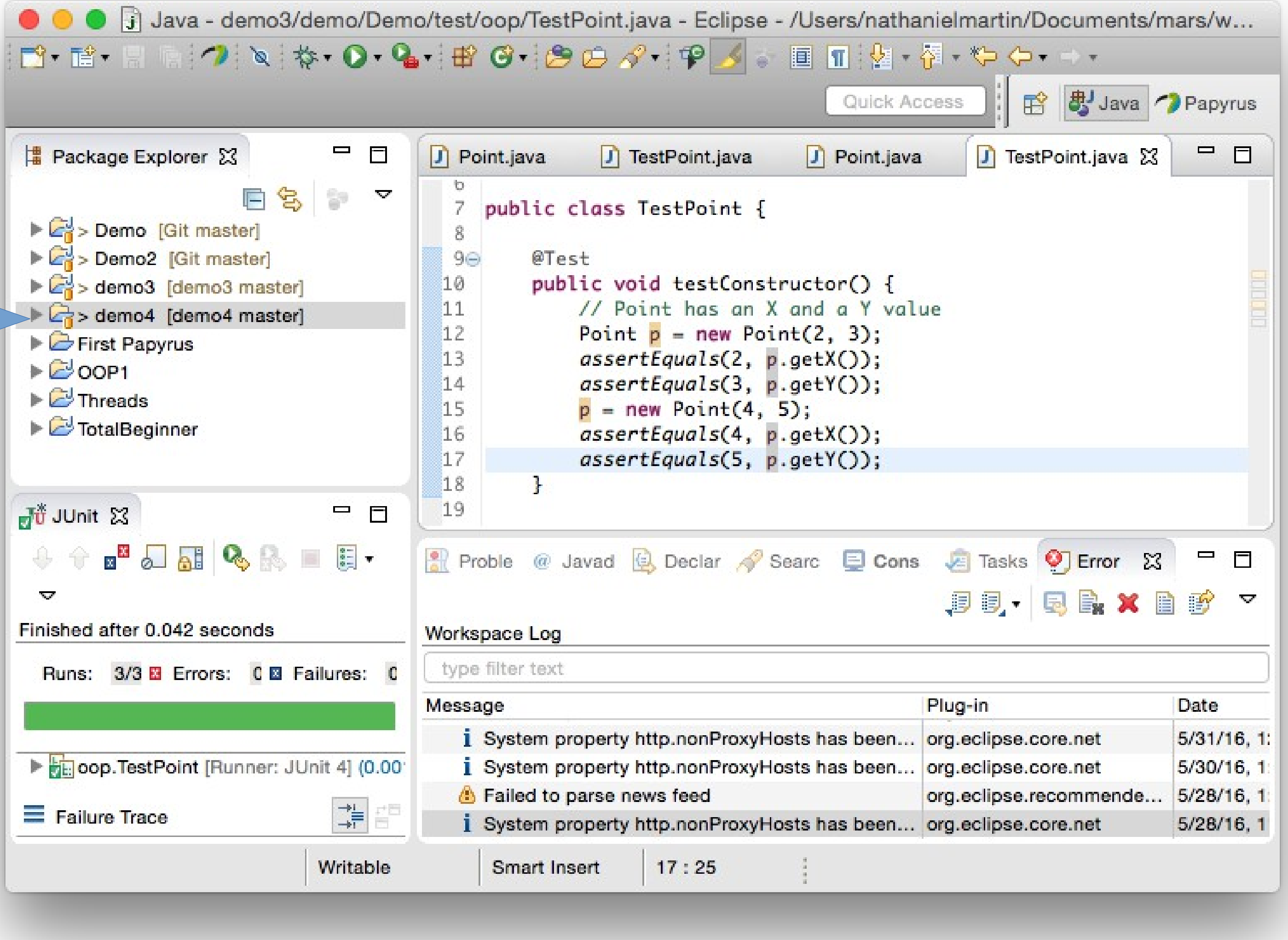
Import projects from a Git repository

demo4

Directory /Users/nathanielmartin/Documents/mars/workspace/demo4

Project Added

Demo4



Java - demo3/demo/Demo/test/oop/TestPoint.java - Eclipse - /Users/nathanielmartin/Documents/mars/w...

- > Demo [Git master]
- > Demo2 [Git master]
- > demo3 [demo3 master]
- > demo4 [demo4 master]
- First Papyrus
- OOP1
- Threads
- TotalBeginner

```
public class TestPoint {

    @Test
    public void testConstructor() {
        // Point has an X and a Y value
        Point p = new Point(2, 3);
        assertEquals(2, p.getX());
        assertEquals(3, p.getY());
        p = new Point(4, 5);
        assertEquals(4, p.getX());
        assertEquals(5, p.getY());
    }

```

Finished after 0.042 seconds

Runs: 3/3 Errors: 0 Failures: 0

oop.TestPoint [Runner: JUnit 4] (0.00

Failure Trace

Workspace Log

type filter text

Message	Plug-in	Date
System property http.nonProxyHosts has been...	org.eclipse.core.net	5/31/16, 1:
System property http.nonProxyHosts has been...	org.eclipse.core.net	5/30/16, 1
Failed to parse news feed	org.eclipse.recommende...	5/28/16, 1
System property http.nonProxyHosts has been...	org.eclipse.core.net	5/28/16, 1

Writable Smart Insert 17 : 25