A decorative graphic consisting of a thin yellow circle on the left side, partially overlapping a horizontal bar. The bar has a yellow-to-white gradient and is enclosed by large black and yellow square brackets.

CS257

Ant, JUnit, & Tigris

Presented by

Paul Nguyen & Denis Gefter

[Overview]

- Ant
- JUnit
- Project Build Files & Test Cases
- Tigris Project Site

[Ant]

- Apache Ant is a Java-based build tool. In theory, it is kind of like *make*, without *make's* wrinkles.
--Apache Software Foundation
- Why another make tool?
 - Ant is better than make
 - Java based, so is OS Independent, make relies on OS Shell commands
 - Ant uses XML format; no more ugly make tab problems!

Structure of an Ant Build File

■ My First Ant Build File (build1.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<project default="hello">
  <target name="hello" >
    <antstructure output="ant.dtd"/>
  </target>
</project>
```

[Run It!]

- `ant -file build1.xml`

```
C:\>ant -file build1.xml
```

```
Buildfile: build1.xml
```

```
hello:
```

```
BUILD SUCCESSFUL
```

```
Total time: 1 second
```

- A file named “ant.dtd” is produced in the current directory.
Let's look at it!

[Ant Build File DTD Defines...]

- Elements for (amongst others...)
 - Project - document root element
 - Target - can have dependencies
 - Tasks - executed by Ant
 - Properties – a variable in the Ant environment
- Note: The DTD is useful to see what Ant currently recognizes, but can not be used to validate. Developers can extend ant with custom tasks.

[Ant Project Element]

```
<!ELEMENT project (target | property | taskdef | typedef | %types;)*>
```

```
<!ATTLIST project
```

```
    name CDATA #IMPLIED
```

```
    default CDATA #REQUIRED
```

```
    basedir CDATA #IMPLIED>
```

[Ant Target Element]

```
<!ELEMENT target (%tasks; | %types;)*>
```

```
<!ATTLIST target
```

```
    id      ID      #IMPLIED
```

```
    name    CDATA #REQUIRED
```

```
    if      CDATA #IMPLIED
```

```
    unless  CDATA #IMPLIED
```

```
    depends CDATA #IMPLIED
```

```
    description CDATA #IMPLIED>
```


Ant Tasks

```
<!ENTITY % tasks "propertyfile | vsscheckin | sql | cvspass | p4reopen | csc | dirname | wlrn |  
p4label | p4revert | replaceregexp | get | jjtree | sleep | jarlib-display | dependset | zip |  
patch | jspc | style | test | tstamp | unwar | vsshhistory | icontract | cvschangelog | p4submit  
| ccmcheckin | p4change | bzip2 | p4delete | vssadd | javadoc | translate | signjar | vajload  
| jarlib-available | WsdltToDotnet | buildnumber | jpcovmerge | ejbjar | war | rename |  
sequential | serverdeploy | property | move | copydir | cccheckin | wljspc | fixcrlf | sosget |  
pathconvert | record | p4sync | exec | p4edit | manifest | maudit | antlr | netrexxc |  
jpcovreport | execon | ccmcheckout | ant | xmlvalidate | xslt | iplanet-ejbc |  
ccmcheckintask | gzip | native2ascii | starteam | ear | input | rmic | checksum | mail |  
loadfile | vsscheckout | stylebook | soscheckin | mimemail | stlabel | gunzip | concat | cab |  
touch | parallel | splash | antcall | cccheckout | typedef | p4have | filter | xmlproperty | copy  
| antstructure | ccmcreatetask | rpm | delete | replace | mmetrics | waitfor | untar |  
loadproperties | available | echoproperties | vajexport | stcheckout | bunzip2 | copyfile |  
vsscreate | ejbc | unjar | wsdltodotnet | mkdir | condition | cvs | tempfile | junitreport |  
taskdef | echo | ccupdate | java | renameext | vsslabel | basename | javadoc2 | vsscp | tar  
| vajimport | setproxy | wlstop | p4counter | ilasm | soscheckout | apply | ccuncheckout |  
jarlib-resolve | cvstagdiff | jlink | javacc | jarlib-manifest | pvcs | chmod | jar | sound |  
mparse | blgenclient | uptodate | genkey | javah | ccmreconfigure | fail | unzip | javac |  
p4add | soslabel | jpcovage | depend | vssget | deltree | ddcreator">
```

- NOTE: Look at Ant's Users Manual for information on how to use each task. The generated DTD also defines each of these elements.

[Ant Properties]

```
<!ELEMENT property (classpath)*>
<!ATTLIST property
  id ID #IMPLIED
  refid IDREF #IMPLIED
  name CDATA #IMPLIED
  classpath CDATA #IMPLIED
  file CDATA #IMPLIED
  userproperty %boolean; #IMPLIED
  resource CDATA #IMPLIED
  environment CDATA #IMPLIED
  prefix CDATA #IMPLIED
  taskname CDATA #IMPLIED
  value CDATA #IMPLIED
  location CDATA #IMPLIED
  description CDATA #IMPLIED>
```

[Ant Properties (Cont.)]

- Is a variable with a “name” and “value”
- Can be set in build file using the <property> task.
- Can be set outside of Ant
- Can be a System Property from Java
i.e. System.getProperties()
- Referenced in task attributes using `${prop}` in build file.
Example:
`<javac srcdir="${src}" destdir="${build}"/>`

Built-in Ant Properties

- **basedir** the absolute path of the project's basedir (as set with the basedir attribute of <project>).
- **ant.file** the absolute path of the buildfile.
- **ant.version** the version of Ant
- **ant.project.name** the name of the project that is currently executing; it is set in the name attribute of <project>.
- **ant.java.version** the JVM version Ant detected; currently it can hold the values "1.1", "1.2", "1.3" and "1.4".

[Hello World Ant!]

- Hello Ant Build File (build.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<project default="build">
  <target name="build" >
    <echo message="Hello, world"/>
  </target>
</project>
```

[Run It!]

- build.xml file is the default
 - Just run command> ant

C:\ ant

Buildfile: build.xml

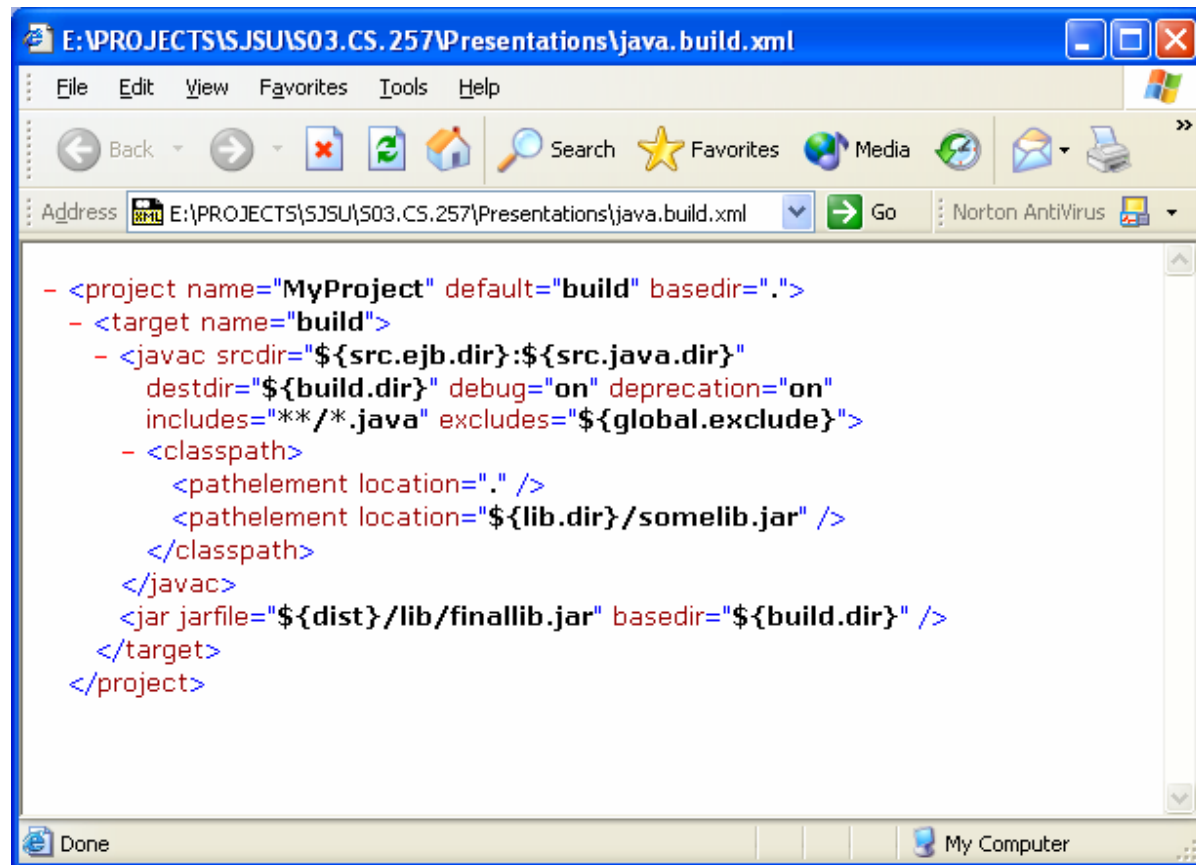
build:

[echo] Hello, world

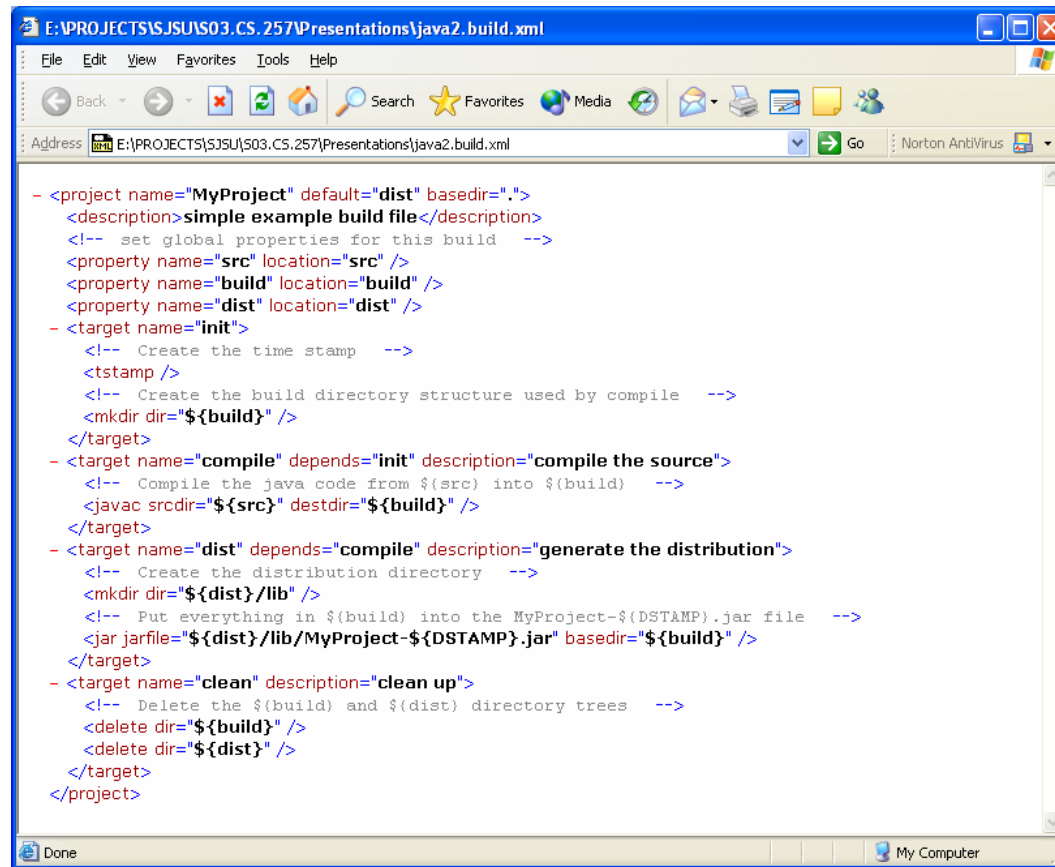
BUILD SUCCESSFUL

Total time: 0 seconds

A Simple Java Example



Example From Ant User's Manual

A screenshot of a web browser window displaying an Ant build.xml file. The browser's address bar shows the file path: E:\PROJECTS\JSJSU\503.CS.257\Presentations\java2.build.xml. The file content is XML code for an Ant project named 'MyProject'. It includes targets for 'init', 'compile', 'dist', and 'clean'. The 'init' target creates a build directory. The 'compile' target compiles Java source files. The 'dist' target creates a distribution directory and a jar file. The 'clean' target deletes the build and distribution directories. The browser's status bar at the bottom shows 'Done' and 'My Computer'.

```
- <project name="MyProject" default="dist" basedir=".">
  <description>simple example build file</description>
  <!-- set global properties for this build -->
  <property name="src" location="src" />
  <property name="build" location="build" />
  <property name="dist" location="dist" />
- <target name="init">
  <!-- Create the time stamp -->
  <tstamp />
  <!-- Create the build directory structure used by compile -->
  <mkdir dir="${build}" />
</target>
- <target name="compile" depends="init" description="compile the source">
  <!-- Compile the java code from ${src} into ${build} -->
  <javac srcdir="${src}" destdir="${build}" />
</target>
- <target name="dist" depends="compile" description="generate the distribution">
  <!-- Create the distribution directory -->
  <mkdir dir="${dist}/lib" />
  <!-- Put everything in ${build} into the MyProject-${DSTAMP}.jar file -->
  <jar jarfile="${dist}/lib/MyProject-${DSTAMP}.jar" basedir="${build}" />
</target>
- <target name="clean" description="clean up">
  <!-- Delete the ${build} and ${dist} directory trees -->
  <delete dir="${build}" />
  <delete dir="${dist}" />
</target>
</project>
```


[More on Ant... (not covered)]

- Data Types
- User Defined Tasks
- Master Build File (ant calling ant)

[JUnit]

- JUnit is a regression testing framework written by Erich Gamma and Kent Beck. -- *www.junit.org*

[JUnit Features]

- Assertions for testing expected results
- Test fixtures for sharing common test data
- Test suites for easily organizing and running tests
- Graphical and textual test runners

[JUnit Concepts]

- A TestCase is a subclass of the TestCase class.
- A TestCase can define any number of public testXXX() methods.
- Use assert() methods to check results (see: **junit.framework.Assert**)

[JUnit Concepts (Cont.)]

- Fixtures are common objects used by TestCase's public testXXX() methods
 - Initialize using setup()
 - Release using tearDown()
- TestCases can be combined into TestSuite to yield a single passed/failed status

[JUnit TestCase Example]

```
public class SimpleTest extends TestCase {
    protected int fValue1;
    protected int fValue2;

    protected void setUp() {
        fValue1= 2;
        fValue2= 3;
    }
    public static Test suite() {
        return new TestSuite(SimpleTest.class);
    }
    public void testAdd() {
        double result= fValue1 + fValue2;
        // forced failure result == 5
        assertTrue(result == 6);
    }
    public void testDivideByZero() {
        int zero= 0;
        int result= 8/zero;
    }
    public void testEquals() {
        assertEquals(12, 12);
        assertEquals(12L, 12L);
        assertEquals(new Long(12), new Long(12));

        assertEquals("Size", 12, 13);
        assertEquals("Capacity", 12.0, 11.99, 0.0);
    }
    public static void main (String[] args) {
        junit.textui.TestRunner.run(suite());
    }
}
```

[Web Resources]

- <http://ant.apache.org/>
- <http://www.junit.org>

[SM Build and Testing]

- Presentation by Denis Geftter
 - Directory Structure of CS257 Project
 - Ant Build Files
 - Running the Test Cases against the Reference Implementation

[Tigris.org]

- Tigris.org is a mid-sized open source community focused on building better tools for collaborative software development.
- Tigris.org has a policy of allowing university class projects to be hosted. These “student” projects are an exception to their typical rule of approving only open source projects.

[Student Project Category]

- “This category is for students to host their class projects. Projects can be open or proprietary based on the instructor's rules for the course. After the end of the course, projects will be archived and removed, or made into open source projects.” -- *tigris.org*

[Public vs Private Projects]

- Projects on Tigris.org can be Private or Public.
 - Private projects require membership to view and contribute.
 - Public projects are visible to the general public.

[Tigris.org for CS 257]

- Each team should create their own Private project.
- We are awaiting approval for the course shared project.
- To be named: sjsu-cs257
(<http://sjsu-cs257.tigris.org/>)

--still need to finalize a few details with Johnny, but should be coming soon!

[Naming Convention?]

- We currently have one team project on Tigris.org named:

sjsu-cs257-team1

- Could use this.name++ or make up your own.

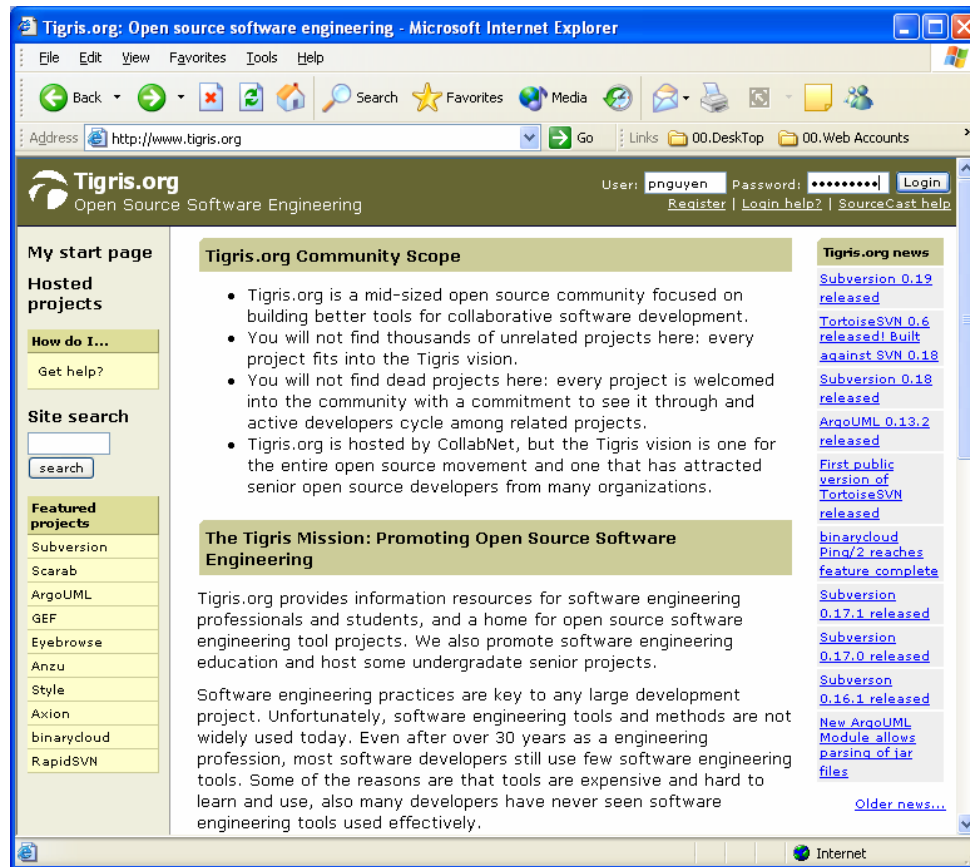
[Hosted Tools]

- Each Project has the following web based tools:
 - Membership
 - Mailing lists
 - Source code (CVSWeb)
 - Issue tracking
 - File sharing
 - News

[Private CVS Repository]

- Each Project also has a CVS repository only accessible to project members—you must login.
- Access is via “PSERVER”, SSH Tunnel recommended, but not required.

[Tigris.org]



Project Home Page

The screenshot shows a Microsoft Internet Explorer browser window displaying the Tigris.org project home page for 'sjsu-cs257-team1'. The browser's address bar shows 'http://sjsu-cs257-team1.tigris.org/'. The page header includes the Tigris.org logo and navigation links like 'Edit profile', 'Logout', and 'SourceCast help'. The main content area is titled 'Project home' and includes links for 'Edit this Project' and 'Request new project role'. A summary table lists project details: Summary (Storage Manager Implementation), Category (students), License (Apache License), Owner(s) (pnguyen), and Your role(s) (Project Owner). Below this is a 'Message from the owner(s)' section with a message from San Jose State University dated Spring 2003. A 'Mission' section describes the development of a generic storage manager. The page also features a sidebar with 'My start page', 'Hosted projects', 'How do I...', 'Site search', and 'Featured projects'.

sjsu-cs257-team1.tigris.org - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail News RSS Feeds

Address http://sjsu-cs257-team1.tigris.org/ Go Links 00.DeskTop 00.Web Accounts

Tigris.org
Open Source Software Engineering

Logged in: pnguyen
[Edit profile](#) | [Logout](#) | [SourceCast help](#)

My start page

Hosted projects

Project: sjsu-cs257-team1

- Membership
- Mailing lists
- Source code
- Issue tracking
- File sharing
- News

How do I...

Learn more about this project?
Join this project?
Get help?

Site search

Featured projects

- Subversion
- Scarab
- ArgoUML
- GEF
- Eyebrowse

Project home

[Edit this Project](#) | [Request new project role](#)

Project: sjsu-cs257-team1

Summary	Storage Manager Implementation
Category	students
License	Apache License
Owner(s)	pnguyen
Your role(s)	Project Owner

Message from the owner(s)

San Jose State University - Spring 2003 - CS257 Database Implementation. Prof. Johnny Martin (martin@cs.sjsu.edu)

Mission

We will be developing an implementation for a generic storage manager using a specific subset of the Shore APIs. The details of this subset will be provided by the professor; however, in order to effectively work, all team members must set up a development environment as detailed below.

Environment Requirements

Project Members

Membership - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail

Address http://sjsu-cs257-team1.tigris.org/servlets/ProjectMemberList Go Links 00.DeskTop 00.Web Accounts

Tigris.org
Open Source Software Engineering

Logged in: pnguyen
[Edit profile](#) | [Logout](#) | [SourceCast help](#)

My start page

Hosted projects

Project: sjsu-cs257-team1

- Membership
- Mailing lists
- Source code
- Issue tracking
- File sharing
- News

How do I...

- Check the members' roles
- View their issues?
- Process membership requests?
- Get help?

Site search

Featured projects

- Subversion
- Scarab
- ArgoUML
- GEF

Membership

[Invite new members](#) | [Request new project role](#) | [Add new member](#)

Project: sjsu-cs257-team1

User	Real name	Roles	Assigned issues
dgifter	Denis Gifter	<input type="checkbox"/> Developer	View issues
pdhupia	Pratik	<input type="checkbox"/> Developer	View issues
pnguyen	Paul Nguyen	<input type="checkbox"/> Project Owner	View issues
wachan	Wallun Chan	<input type="checkbox"/> Developer	View issues

Filter this list:

Requests pending approval

No users have outstanding role requests in this project.

Done Internet

Add new member - Find

NOTE:

To add a new member, the project owner finds and selects the tigris account to add.

So, team members must first create and account on Tigris and send the username to the project owner

Before you can add members, the project must be approved.

Send an email to: jrobbins@tigris.org

The screenshot shows a web browser window titled "Add member to project - Microsoft Internet Explorer". The address bar shows the URL "http://jsu-cs257-team1.tigris.org/servlets/ProjectMemberAdd". The page content includes a "Mass add" section with a text input field and a list of roles: Observer, Content Developer, Developer, Project Owner, Download Manager, and Release Manager. A checkbox labeled "Done granting roles" is checked. Below this, there is a "Filter this list" section with a text input field containing "pnguyen" and a "Filter" button. The "Filter" button and the text input field are circled in red.

Add New Member - Select

The screenshot shows the 'Add member to project' page on Tigris.org. The browser window is titled 'Add member to project - Microsoft Internet Explorer'. The address bar shows 'http://sjsu-cs257-team1.tigris.org/servlets/ProjectMemberAdd'. The page header includes the Tigris.org logo and 'Open Source Software Engineering'. The user is logged in as 'pnguyen'. The left sidebar contains navigation links for 'My start page', 'Hosted projects', and 'How do I...'. The main content area is titled 'Add member to project' and includes links for 'Invite new members', 'Request new project role', and 'Add new member'. A table with the heading 'Project: sjsu-cs257-team1' and 'Add User' is highlighted with a red circle. The table has two columns: 'User' and 'Real name'. The first row shows 'pnguyen' and 'Paul Nguyen'. Below the table is a 'Mass add' text input field. A paragraph explains that users can enter login names for a mass add. At the bottom, there is a section 'Grant these roles to all designated users:' with a list of roles: 'Observer', 'Content Developer', 'Developer', 'Project Owner' (highlighted), 'Download Manager', and 'Release Manager'. A checkbox labeled 'Done granting roles' is checked.

Project: sjsu-cs257-team1

Add User	Real name
<input checked="" type="checkbox"/> pnguyen	Paul Nguyen

Mass add

If you know the login names of the users to whom you want to assign roles in this project, you may enter them here to do a mass add (one login name per line, or multiple login names if a comma, colon, or semi-colon is used to separate them).

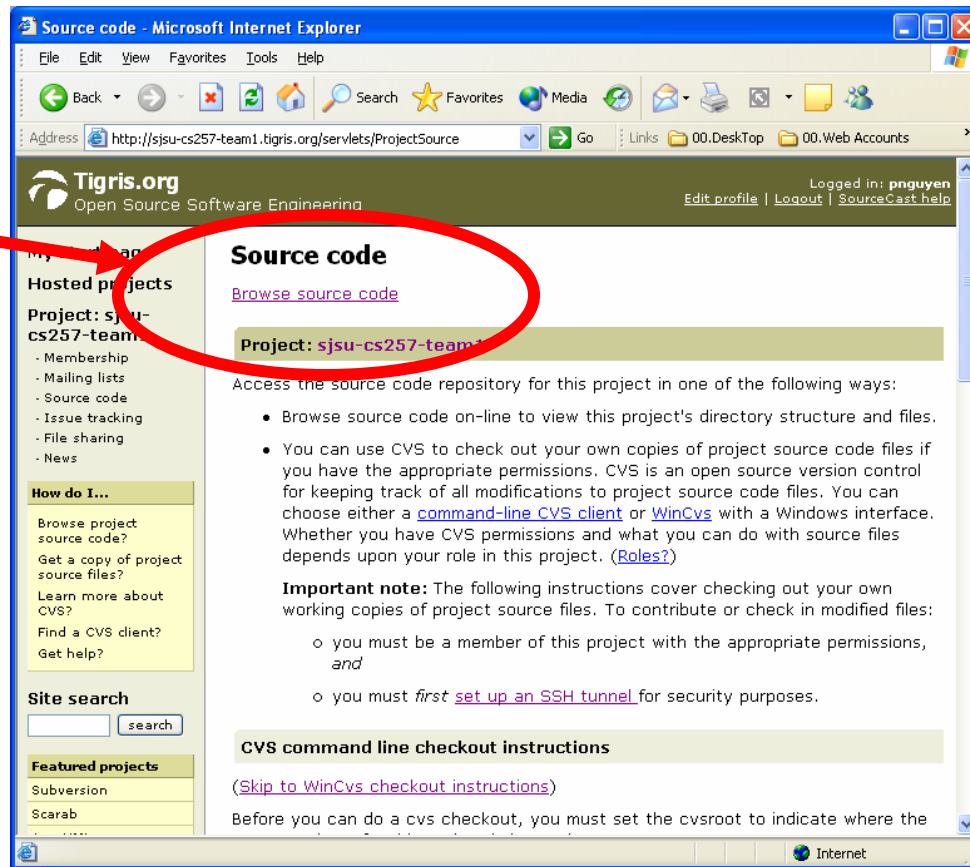
Grant these roles to all designated users:

- Observer
- Content Developer
- Developer
- Project Owner**
- Download Manager
- Release Manager

☒ Done granting roles

Browsing CVS Files

CLICK HERE



Browsing CVS Files

CVSWeb - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://sjsu-cs257-team1.tigris.org/source/browse/sjsu-cs257-team

Tigris.org
Open Source Software Engineering

Logged in: pnguyen
Edit profile | Logout | SourceCast help

My start page

Hosted projects

Project: sjsu-cs257-team1

- Membership
- Mailing lists
- Source code
- Issue tracking
- File sharing
- News

How do I...

Get help?

Site search

search

Featured projects

- Subversion
- Scarab
- ArgoUML
- GEF
- Eyebrowse
- Anzu
- Style
- Axion

CVSWeb

sjsu-cs257-team1 / cs257 / src/

Click on a directory to enter that directory. Click on a file to display its revision history and to get a chance to display diffs between revisions.

Current directory: [sjsu-cs257-team1](#) / [cs257](#) / [src](#)

Total File Count: 5

File	Rev.	Age	Author	Last log entry
Previous Directory				
Attic/ [show Attic files]				
edu/				
build.xml	1.2	5 days	pnguyen	Added Johnny's SM Interface Code
default.dfPackage	1.1	25 hours	pnguyen	Added trivial implementation.
models.tpr	1.1	25 hours	pnguyen	Added trivial implementation.
models.two	1.1	25	pnguyen	Added trivial implementation.

Internet

[Connecting to CVS]

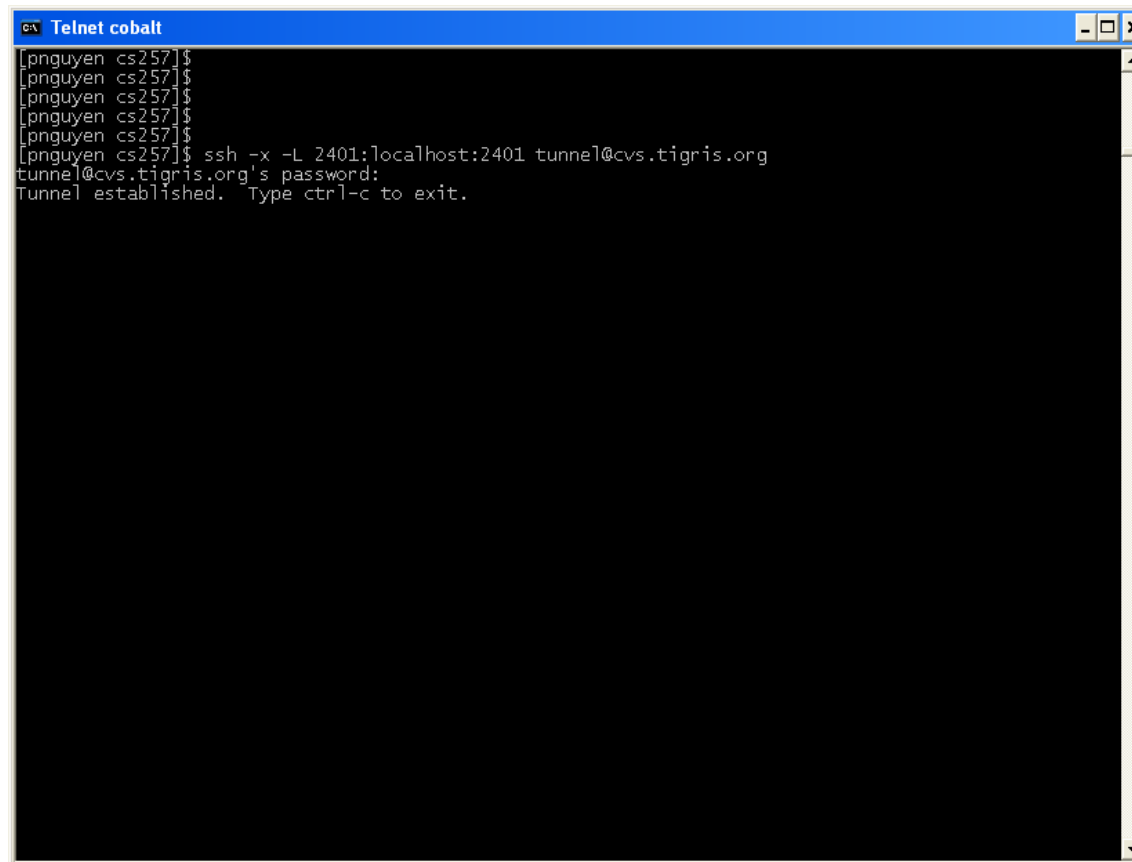
- Create a SSH Tunnel.

Use this command on Linux and Windows (Cygwin):

```
ssh -x -L 2401:localhost:2401 tunnel@cvs.tigris.org
```

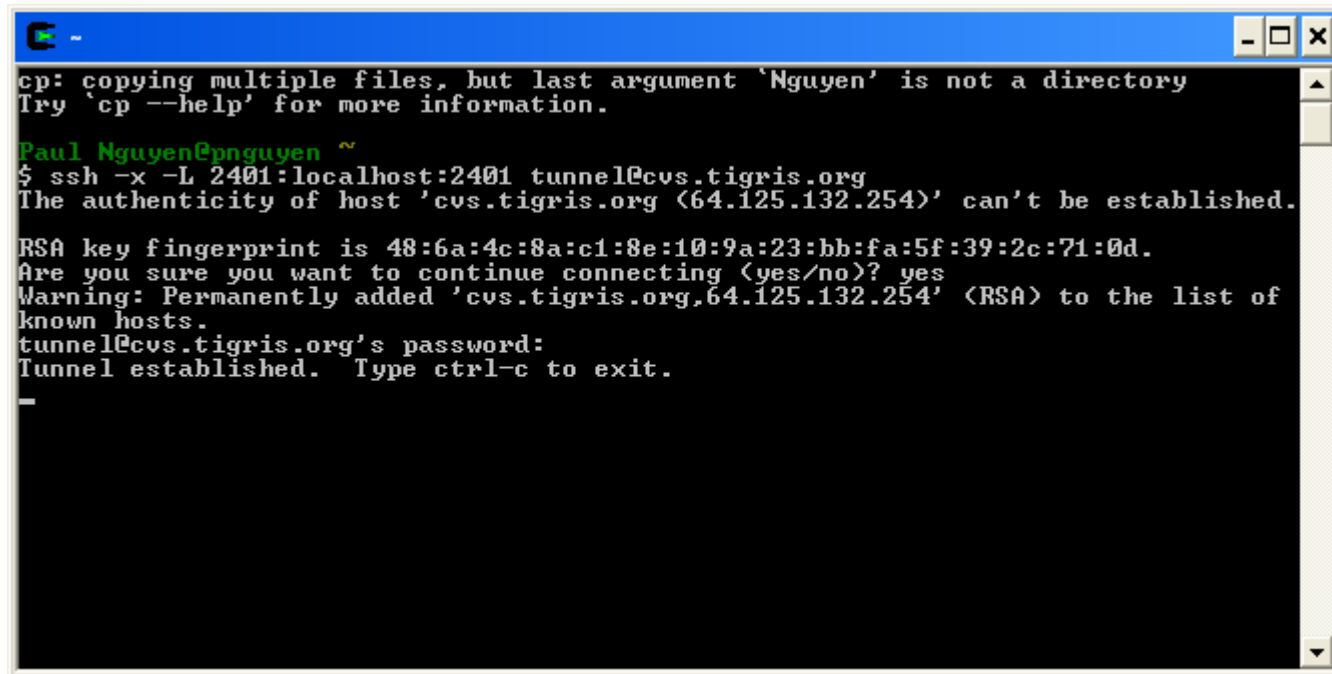
It will ask for a password. Type in: “tunnel”

[Linux SSH Tunnel]

A terminal window titled "Telnet cobalt" with standard window controls. The terminal shows a user named 'pnguyen' at 'cs257' entering a series of commands. The final command is 'ssh -x -L 2401:localhost:2401 tunnel@cvs.tigris.org', which results in a password prompt and a confirmation message: 'Tunnel established. Type ctrl-c to exit.'

```
Telnet cobalt
[pnguyen cs257]$ 
[pnguyen cs257]$ 
[pnguyen cs257]$ 
[pnguyen cs257]$ 
[pnguyen cs257]$ 
[pnguyen cs257]$ ssh -x -L 2401:localhost:2401 tunnel@cvs.tigris.org
tunnel@cvs.tigris.org's password:
Tunnel established. Type ctrl-c to exit.
```


Windows SSH Tunnel

A screenshot of a Windows command prompt window with a blue title bar. The window shows the output of a failed 'cp' command, followed by an 'ssh' command to establish a tunnel. The terminal text includes a warning about the host's authenticity and a confirmation to add it to the known hosts list. The tunnel is successfully established.

```
C -
cp: copying multiple files, but last argument 'Nguyen' is not a directory
Try 'cp --help' for more information.

Paul Nguyen@pnguyen ~
$ ssh -x -L 2401:localhost:2401 tunnel@cvs.tigris.org
The authenticity of host 'cvs.tigris.org (64.125.132.254)' can't be established.

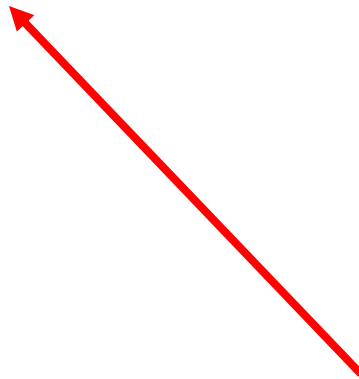
RSA key fingerprint is 48:6a:4c:8a:c1:8e:10:9a:23:bb:fa:5f:39:2c:71:0d.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'cvs.tigris.org,64.125.132.254' (RSA) to the list of
known hosts.
tunnel@cvs.tigris.org's password:
Tunnel established. Type ctrl-c to exit.
-
```

[Login to CVS & Check Out]

```
export CVSROOT=:pserver:pnguyen@localhost:/cvs
```

```
cvs login
```

```
cvs co sjsu-cs257-team1
```

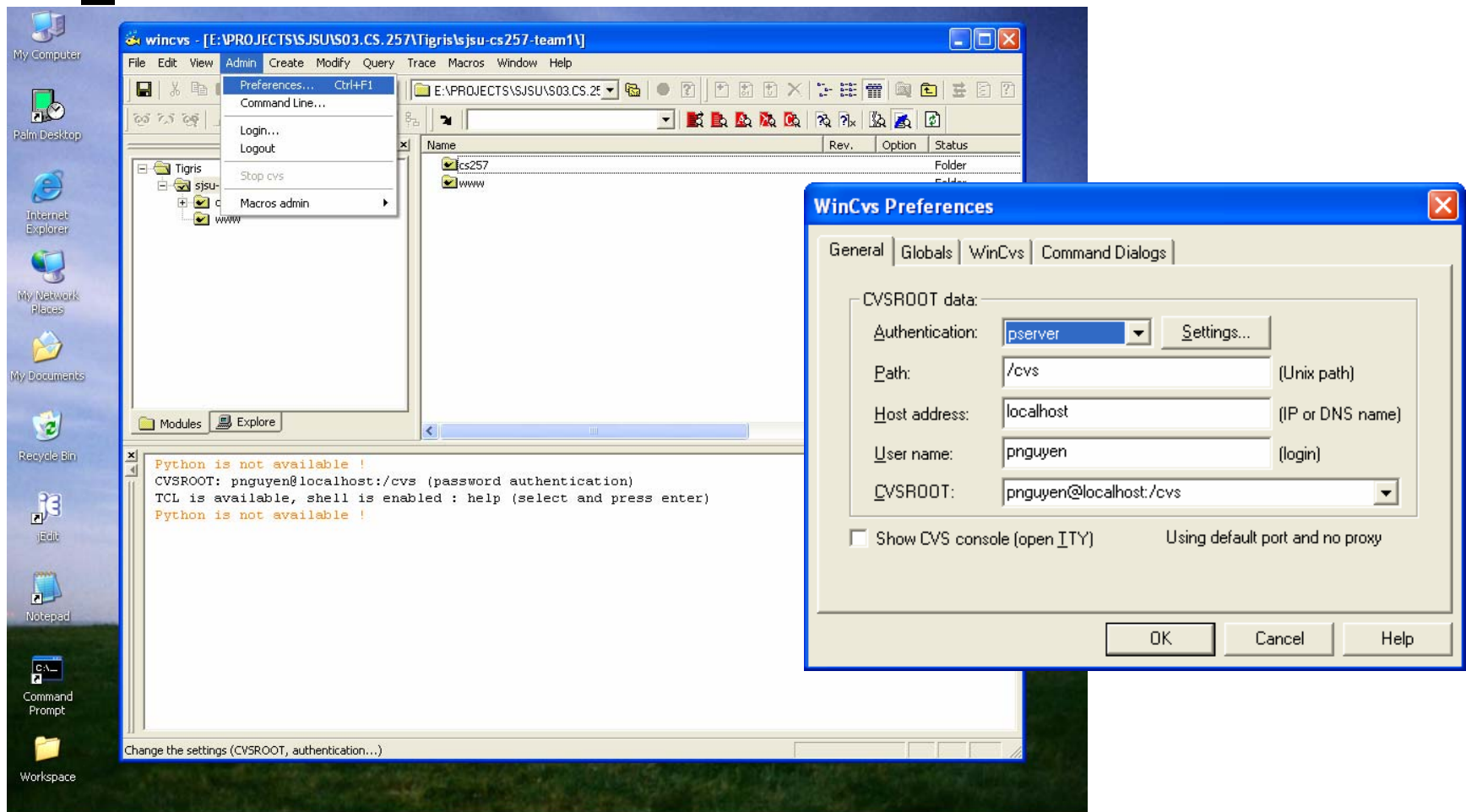


Use the name of your project!

Linux CVS Login & Check Out

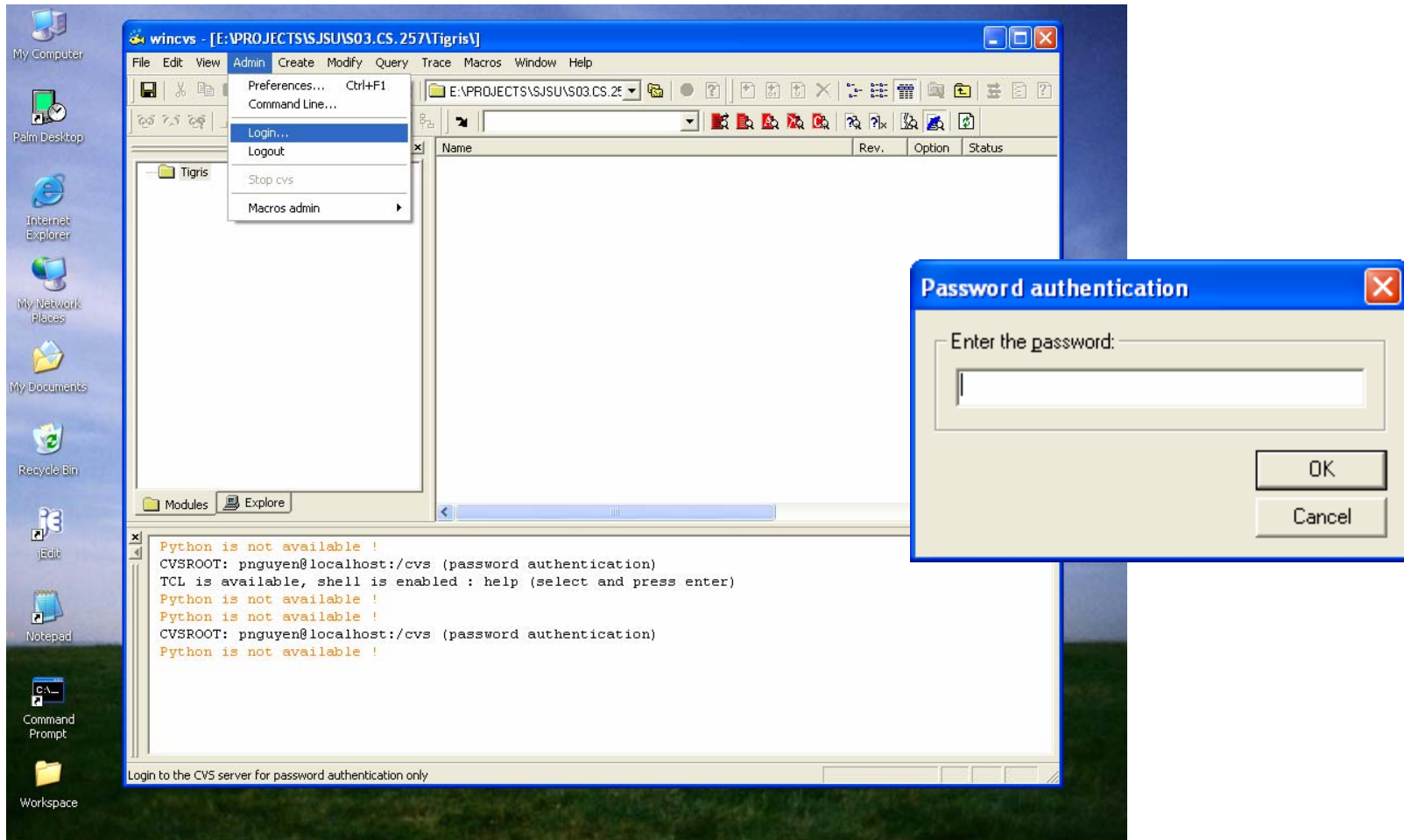
```
Telnet cobalt
[nguyen nguyen]$ cat .bashrc
export CVSROOT=:pserver:nguyen@localhost:/cvs
export PATH=$PATH:/home/jakarta-ant-1.5/bin
[nguyen nguyen]$ . .bashrc
[nguyen nguyen]$
[nguyen nguyen]$
[nguyen nguyen]$
[nguyen nguyen]$ cvs login
(Logging in to nguyen@localhost)
CVS password:
[nguyen nguyen]$
[nguyen nguyen]$
[nguyen nguyen]$ pwd
/home/users/nguyen
[nguyen nguyen]$ cd cvs
[nguyen cvs]$ cvs co sj-su-cs257-team1
? sj-su-cs257-team1/cs257/build/dist
? sj-su-cs257-team1/cs257/build/classes
? sj-su-cs257-team1/cs257/build/test
cvs server: Updating sj-su-cs257-team1
cvs server: Updating sj-su-cs257-team1/cs257
cvs server: Updating sj-su-cs257-team1/cs257/build
cvs server: Updating sj-su-cs257-team1/cs257/config
cvs server: Updating sj-su-cs257-team1/cs257/lib
cvs server: Updating sj-su-cs257-team1/cs257/local
cvs server: Updating sj-su-cs257-team1/cs257/src
cvs server: Updating sj-su-cs257-team1/cs257/src/edu
cvs server: Updating sj-su-cs257-team1/cs257/src/edu/sjsu
cvs server: Updating sj-su-cs257-team1/cs257/src/edu/sjsu/cs
cvs server: Updating sj-su-cs257-team1/cs257/src/edu/sjsu/cs/db
cvs server: Updating sj-su-cs257-team1/cs257/src/edu/sjsu/cs/db/sm
cvs server: Updating sj-su-cs257-team1/cs257/src/edu/sjsu/cs/db/sm/impl
cvs server: Updating sj-su-cs257-team1/cs257/test
cvs server: Updating sj-su-cs257-team1/cs257/test/edu
cvs server: Updating sj-su-cs257-team1/cs257/test/edu/sjsu
cvs server: Updating sj-su-cs257-team1/cs257/test/edu/sjsu/cs
cvs server: Updating sj-su-cs257-team1/cs257/test/edu/sjsu/cs/db
cvs server: Updating sj-su-cs257-team1/cs257/test/edu/sjsu/cs/db/sm
cvs server: Updating sj-su-cs257-team1/www
[nguyen cvs]$
```

Windows CVS – Set CVSROOT

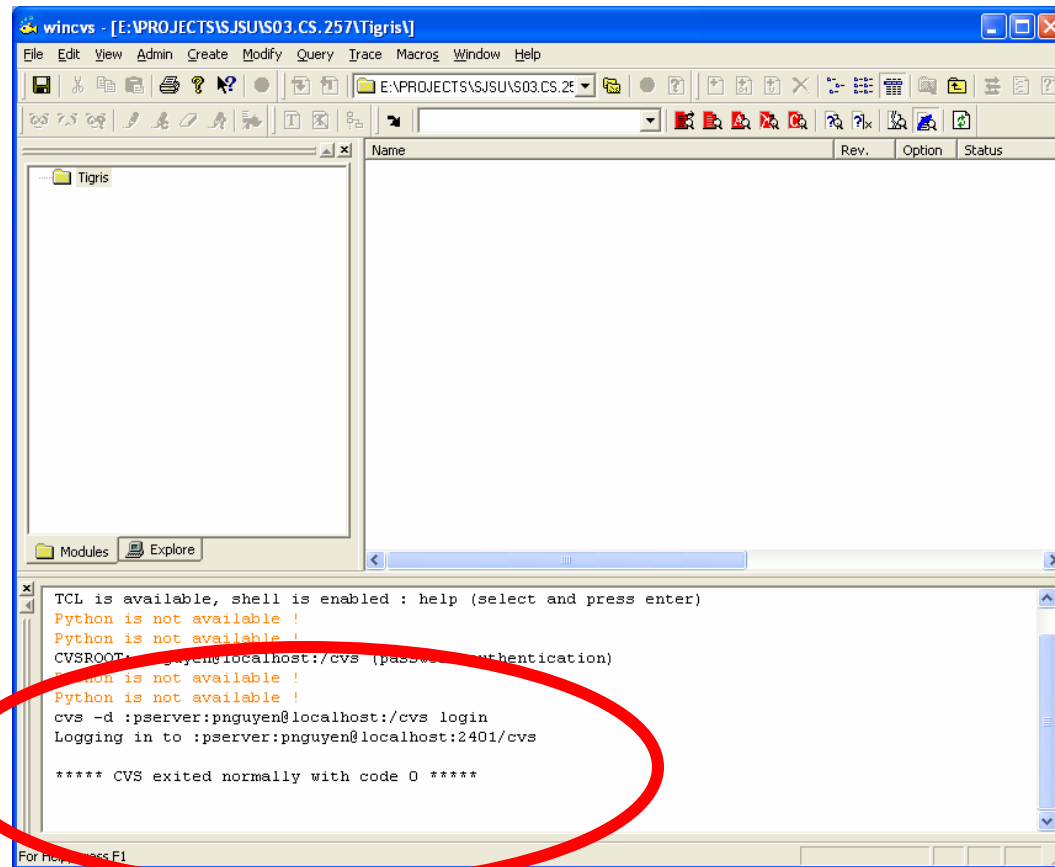


Using WinCVS (a GUI for Windows)

Windows CVS – Login



Windows CVS – Login OK



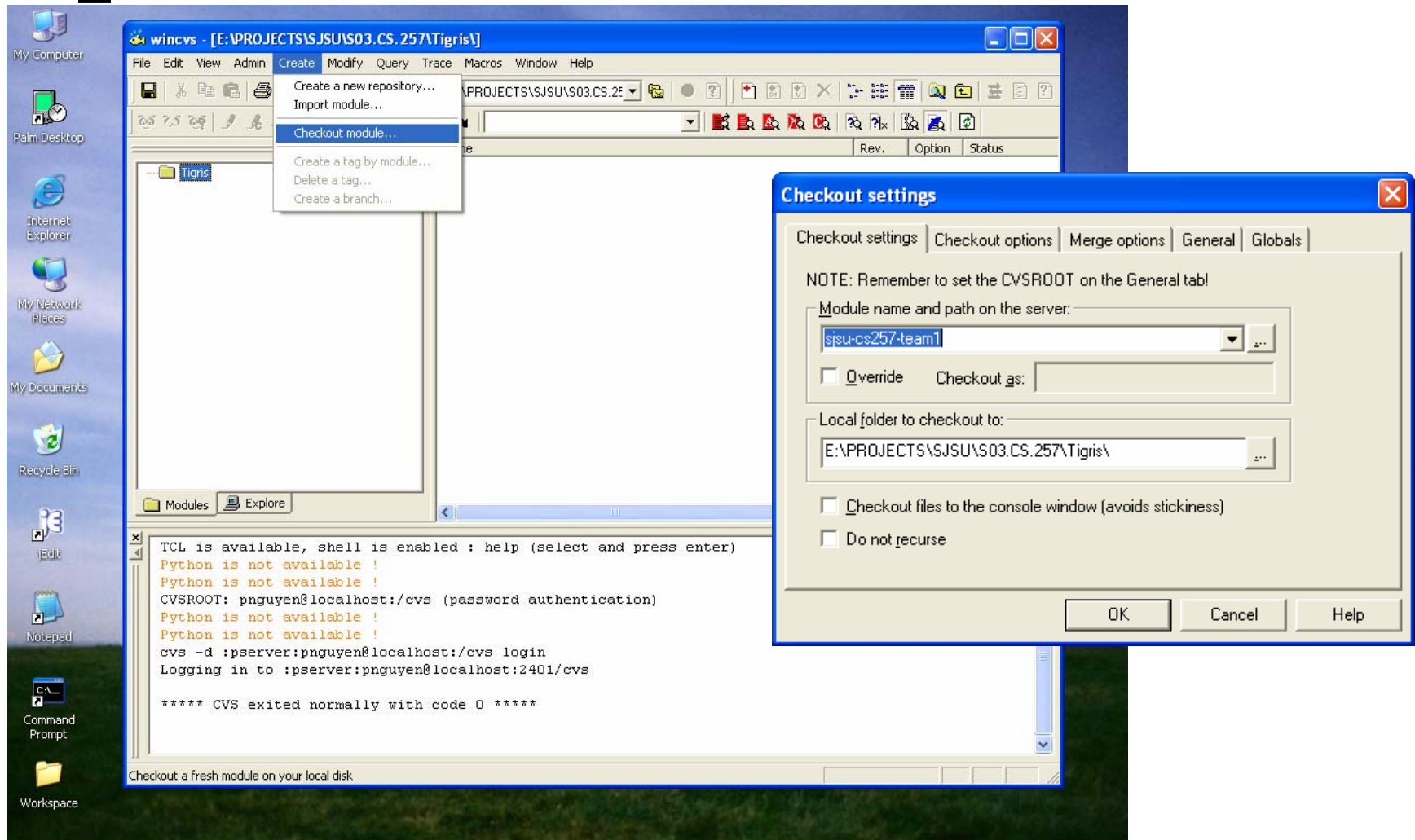
The screenshot shows the WinCVS application window titled "wincvs - [E:\PROJECTS\JSU\S03.CS.257\Tigris\]". The window has a menu bar (File, Edit, View, Admin, Create, Modify, Query, Trace, Macro, Window, Help) and a toolbar. The main area is split into two panes. The left pane shows a file explorer view of the "Tigris" directory. The right pane is empty. Below the panes is a "Modules" and "Explore" section. At the bottom is a command window with the following text:

```
TCL is available, shell is enabled : help (select and press enter)
Python is not available !
Python is not available !
CVSROOT: :pserver:pnguyen@localhost:/cvs (password authentication)
Python is not available !
Python is not available !
cvs -d :pserver:pnguyen@localhost:/cvs login
Logging in to :pserver:pnguyen@localhost:2401/cvs

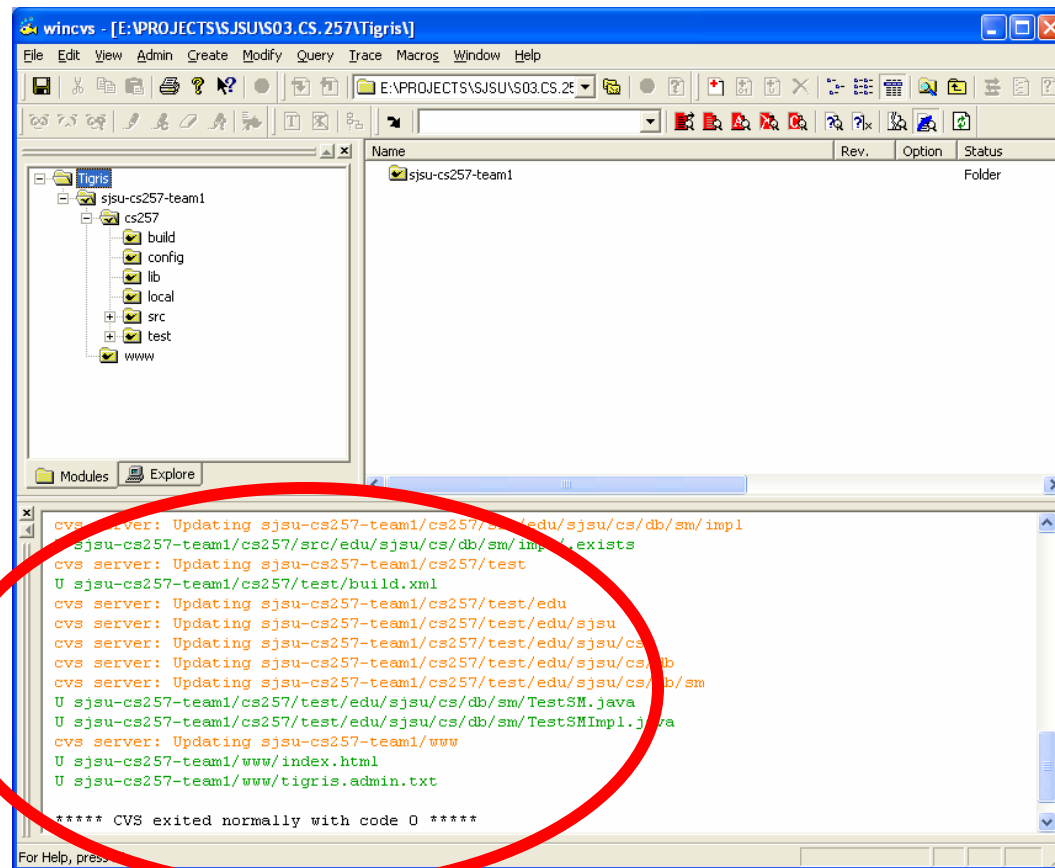
***** CVS exited normally with code 0 *****
```

A red circle is drawn around the login output in the command window.

Windows CVS – Check Out



Windows CVS – Check Out Messages



The screenshot shows a Windows CVS client window titled "wincvs - [E:\PROJECTS\JSU\S03.CS.257\Tigris\]". The window has a menu bar (File, Edit, View, Admin, Create, Modify, Query, Trace, Macro, Window, Help) and a toolbar. The left pane shows a tree view of the project structure: Tigris > sjsu-cs257-team1 > cs257 > build, config, lib, local, src, test, www. The right pane shows a table with columns Name, Rev., Option, and Status, containing a single entry: sjsu-cs257-team1 (Folder). The bottom pane shows the command window with the following output:

```
cvs server: Updating sjsu-cs257-team1/cs257/src/edu/sjsu/cs/db/sm/impl
sjsu-cs257-team1/cs257/src/edu/sjsu/cs/db/sm/impl exists
cvs server: Updating sjsu-cs257-team1/cs257/test
U sjsu-cs257-team1/cs257/test/build.xml
cvs server: Updating sjsu-cs257-team1/cs257/test/edu
cvs server: Updating sjsu-cs257-team1/cs257/test/edu/sjsu
cvs server: Updating sjsu-cs257-team1/cs257/test/edu/sjsu/cs
cvs server: Updating sjsu-cs257-team1/cs257/test/edu/sjsu/cs/db
cvs server: Updating sjsu-cs257-team1/cs257/test/edu/sjsu/cs/db/sm
U sjsu-cs257-team1/cs257/test/edu/sjsu/cs/db/sm/TestSM.java
U sjsu-cs257-team1/cs257/test/edu/sjsu/cs/db/sm/TestSMImpl.java
cvs server: Updating sjsu-cs257-team1/www
U sjsu-cs257-team1/www/index.html
U sjsu-cs257-team1/www/tigris.admin.txt

***** CVS exited normally with code 0 *****
```

A red circle is drawn around the checkout messages in the command window.

[Web Resources]

- <http://www.cygwin.com/>
- <http://www.wincvs.org/>
- <http://www.cvshome.org/project/www/docs/ddSSHGuideCygwin.html>