James Quach

Elizabeth O'neil

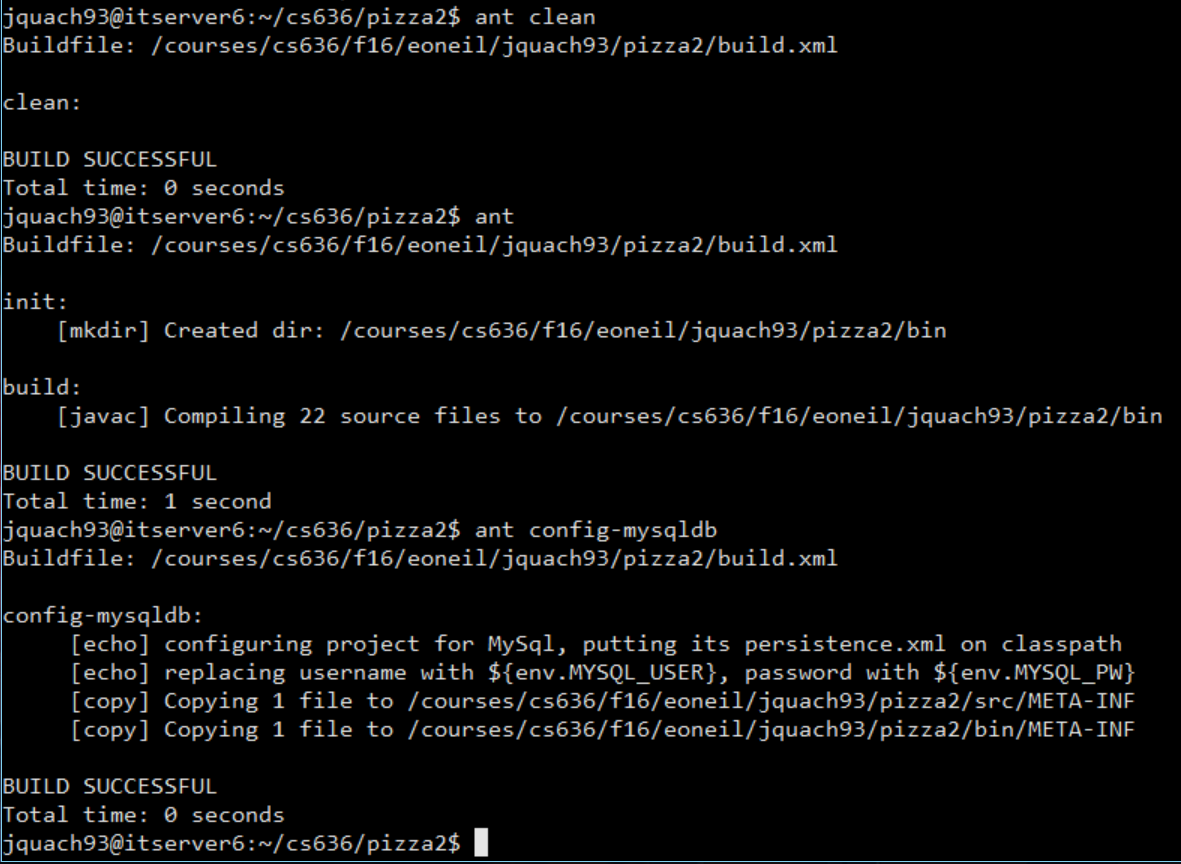
CS436

11/28/2016

Homework 4

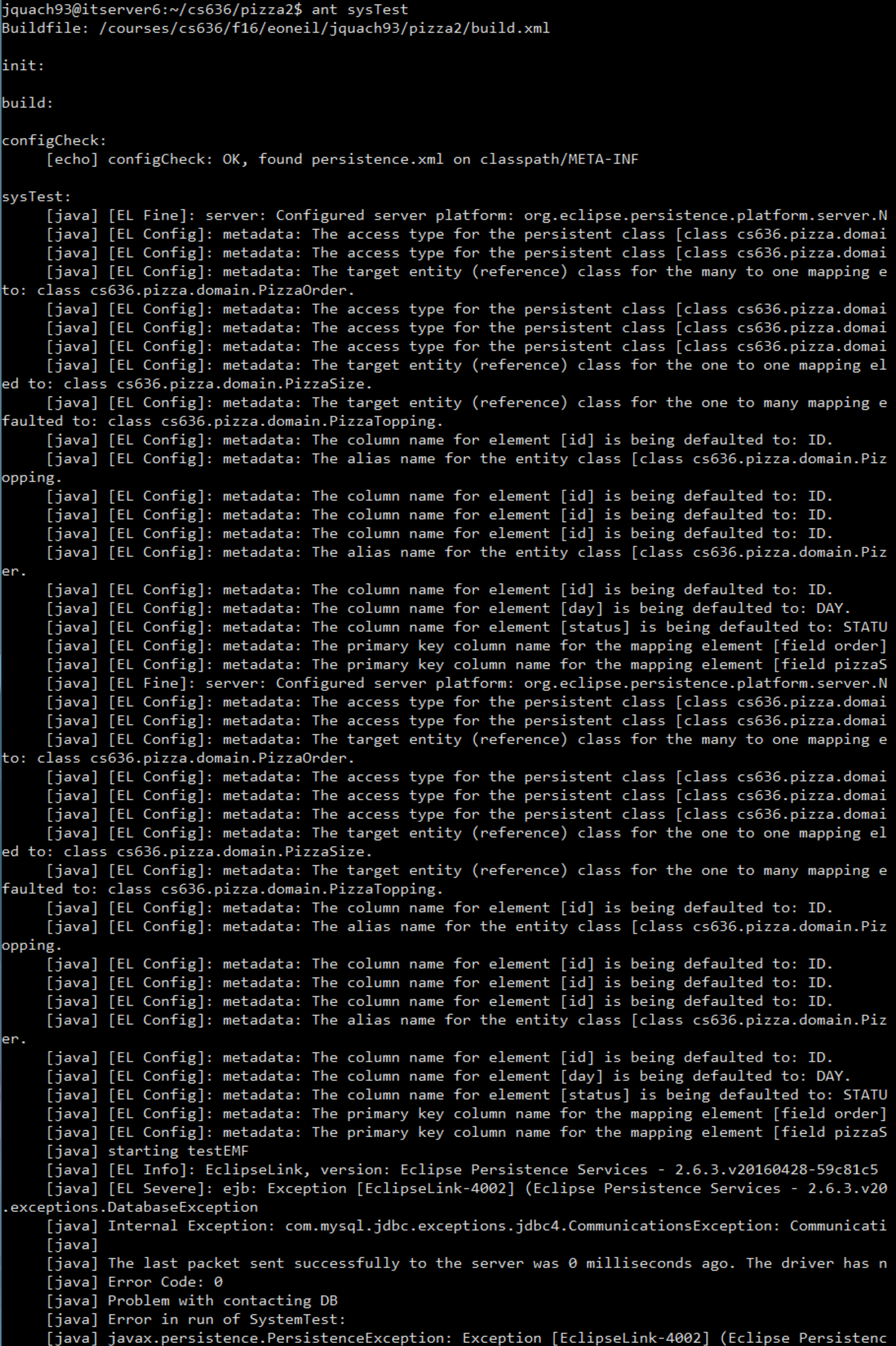
1.Learn about JPA and Object-relational Mapping. Read Chapter 13 of Murach, See [JPA2 Notes](http://www.cs.umb.edu/cs636/JPA2Notes.html) for a guide to this chapter.

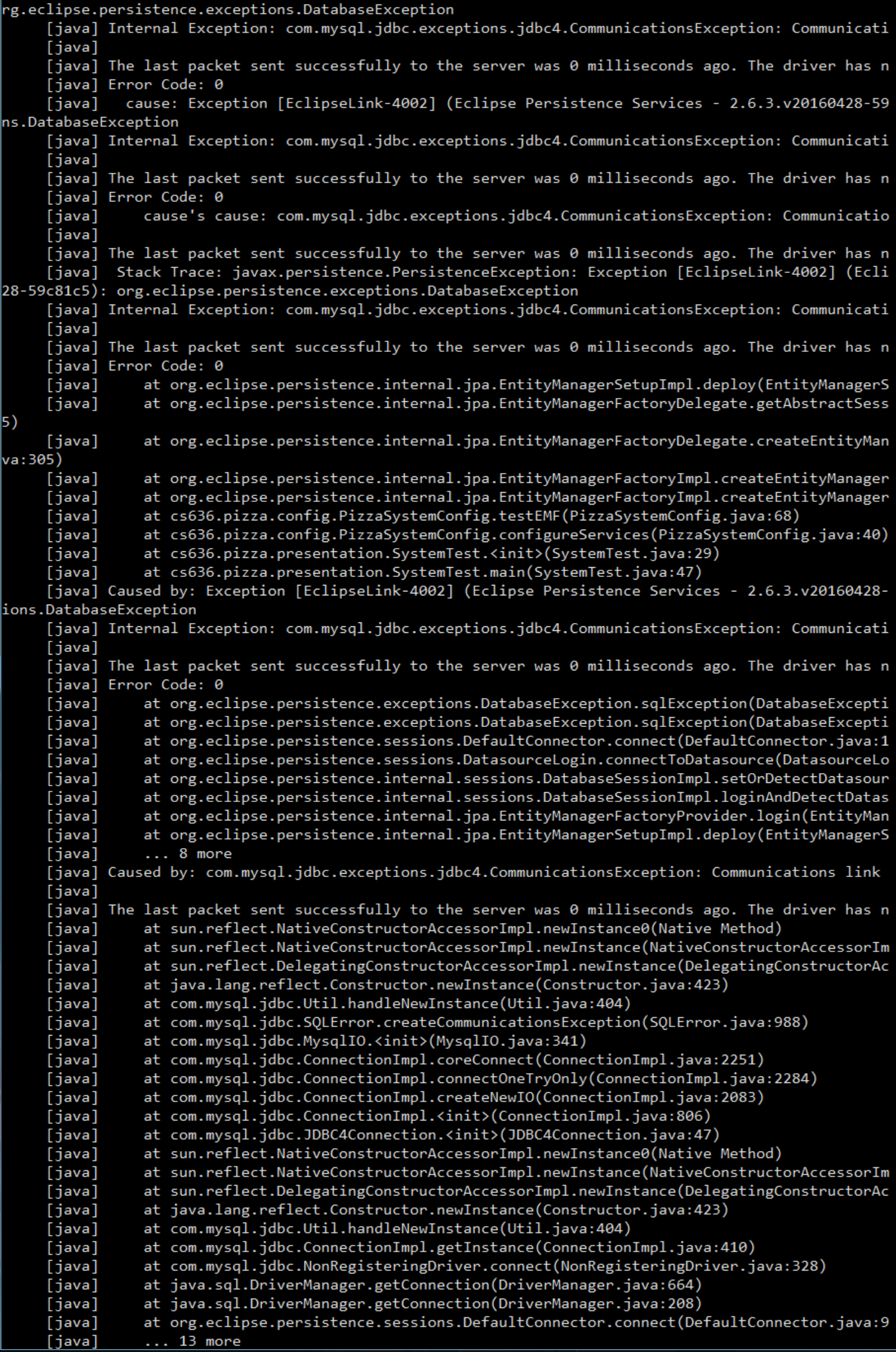
a. Download and use ant to try out pizza2. See [JPA2 Notes](http://www.cs.umb.edu/cs636/JPA2Notes.html) for details. Explain what "ant config-mysqldb" followed by "ant sysTest" does and how it uses ant to do it. Report on any problems. Note that "ant sysTest" expects a newly-loaded database, and database loads are handled in the database directory, as in music1-setup and pizza1.

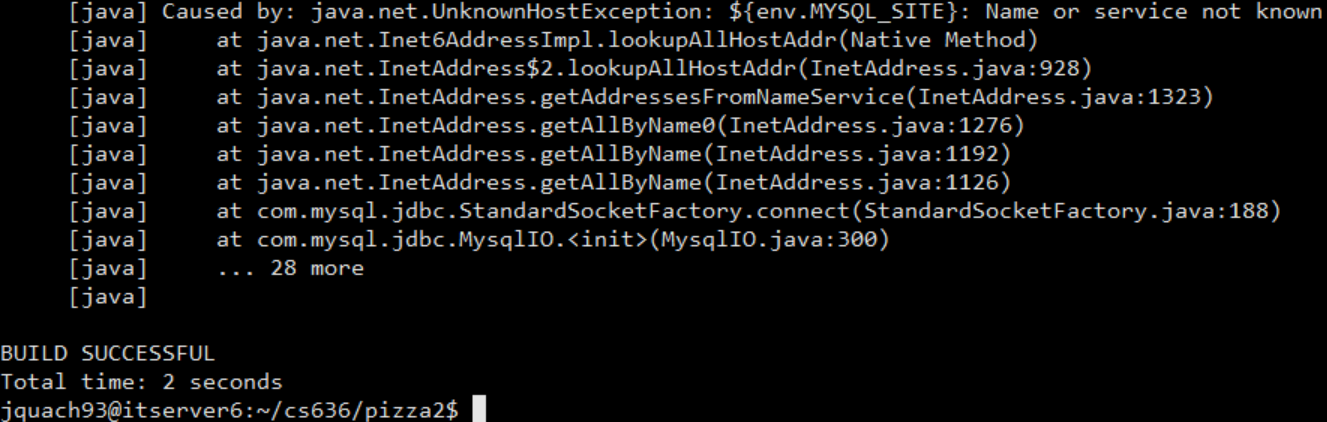


What "ant config-mysqldb" is that it simply configures the database with persistence.xml into the classpath. The respective usernames and passwords are replaced with a placeholder username and password from above. The user metadata is copied into the source and bin folder.

"ant sysTest" below then checks for the "persistence.xml" in the classpath folders, and then runs a system test on the newly built/compiled database and it's metadata and tries to connect to the server. Of course below it does throw exceptions below as it expects the newly loaded database to be handled in the database directory.



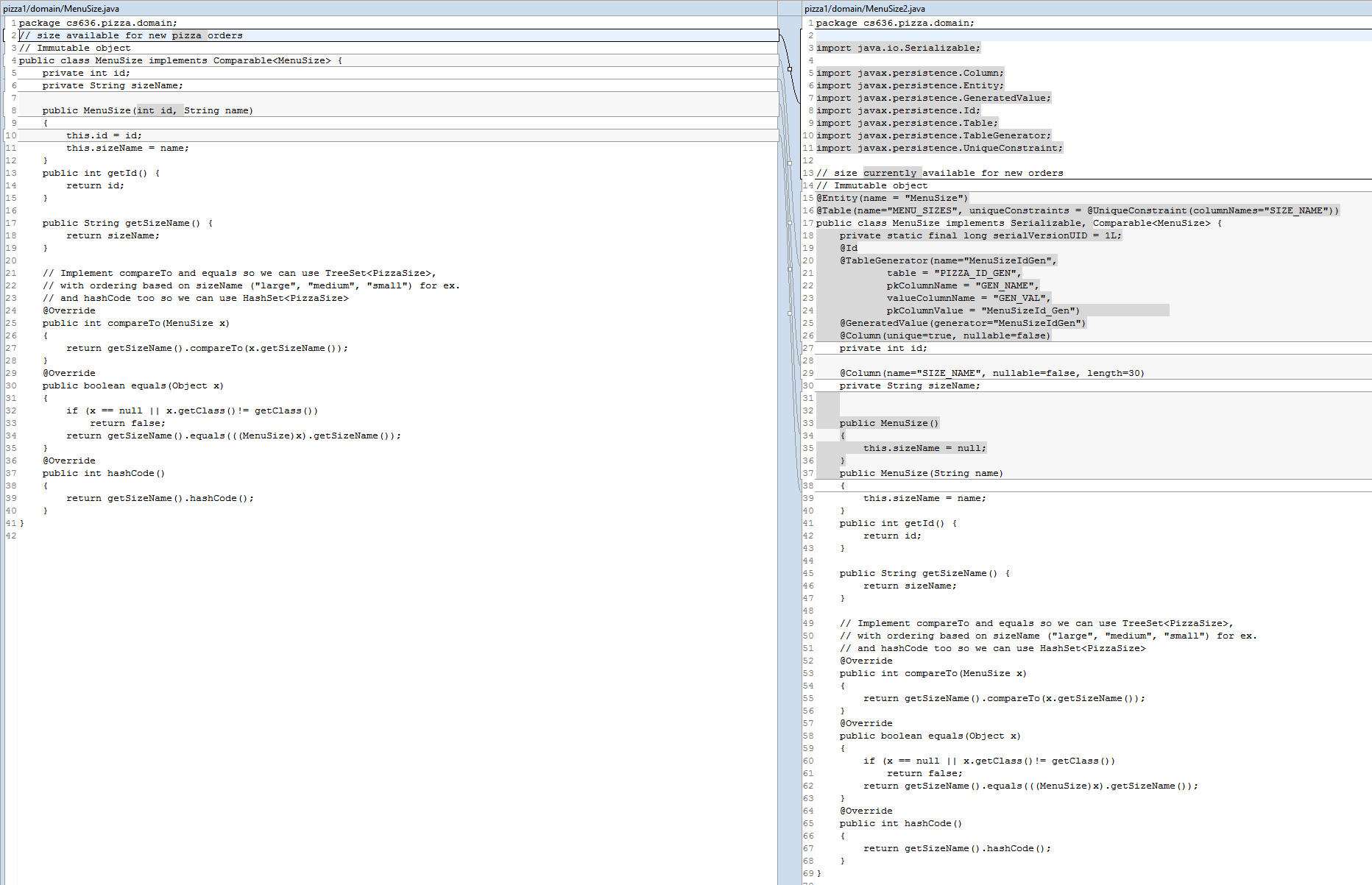


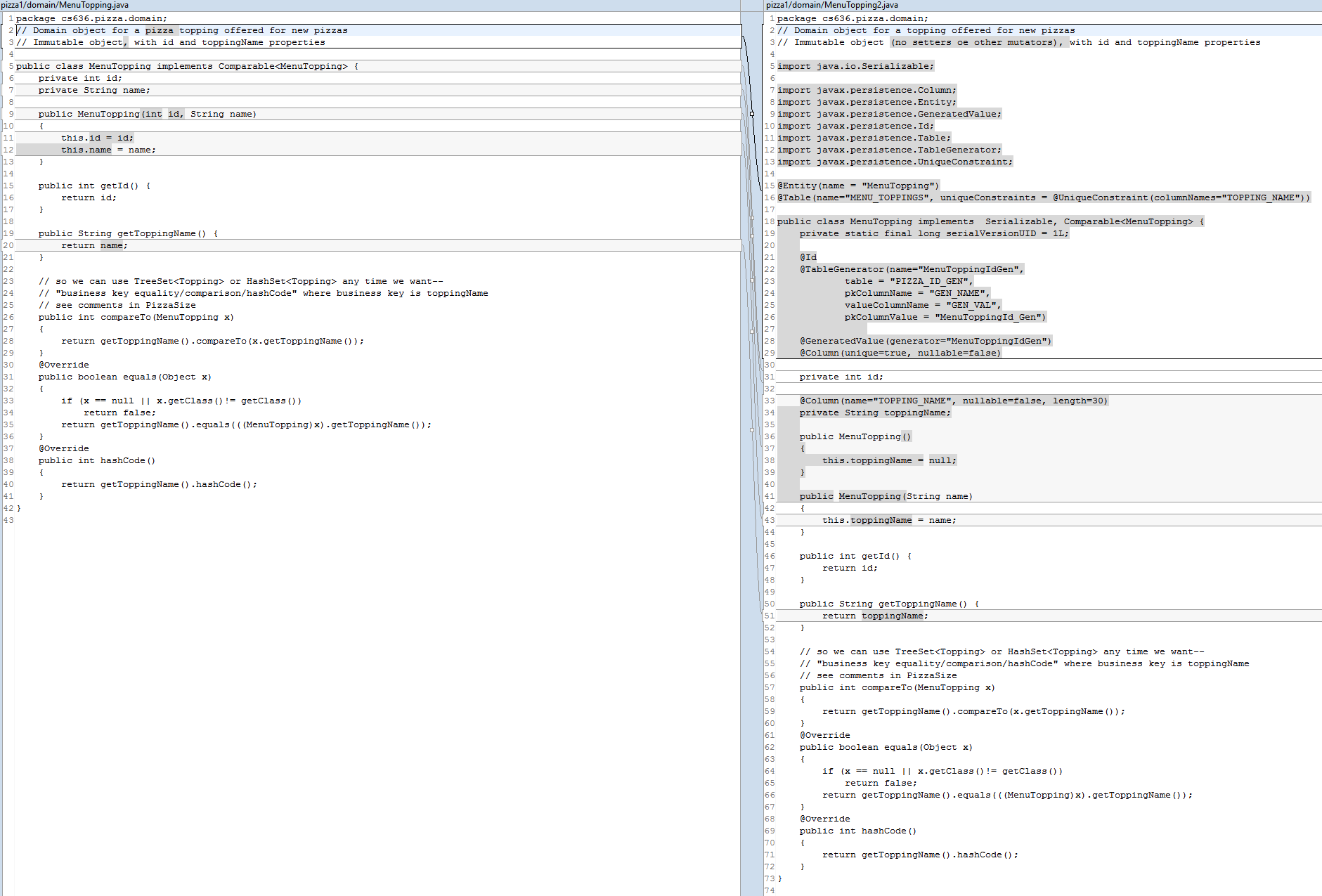


b. Set up the pizza2 project as a Java project in eclipse. Use eclipse to compare the domain packages of pizza1 and pizza2 and report on the differences, grouping similar differences. To compare two packages, select one in the Package Explorer, use control-click to select the other as well, right-click and Compare With>Each Other.

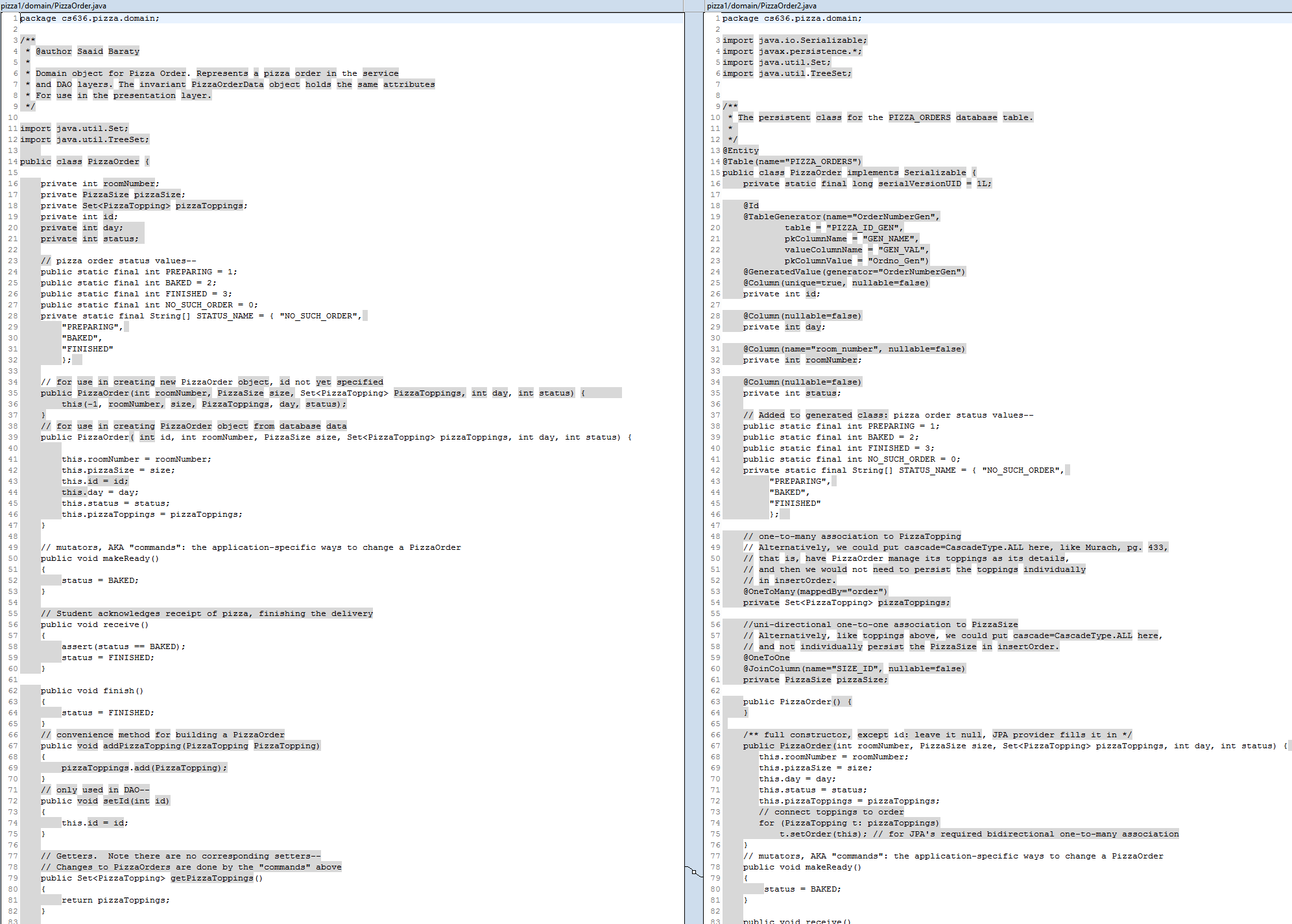
Due to the difficulties of getting eclipse of comparing two similar files in different folders, I renamed the respective java files with a '2' at the end to indicate the file was from pizza2. In this case MenuSize.java with Menusize2.java

The difference below here indicates an inclusion of an id and table generator for pizza2, and a MenuSize() method.

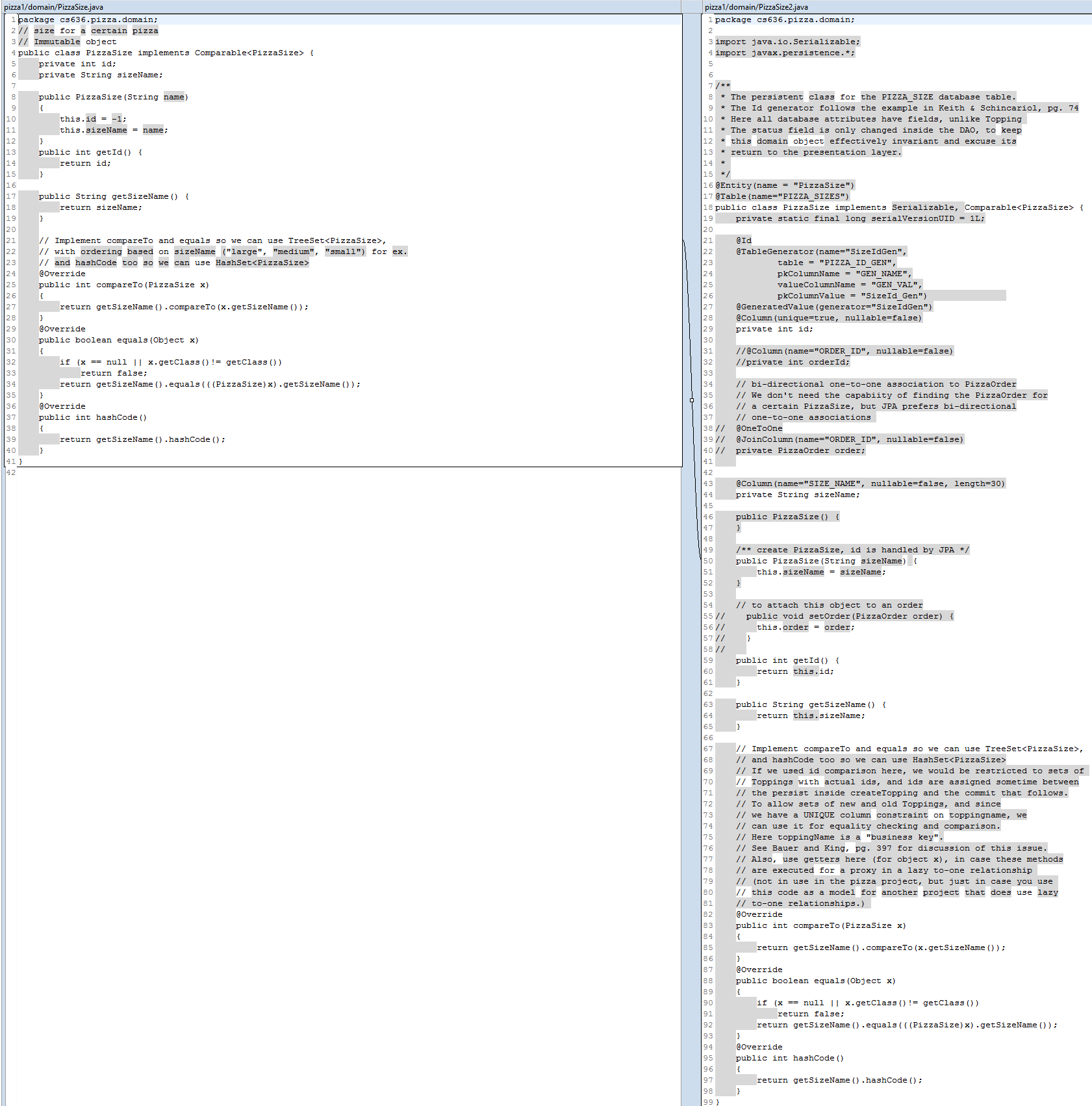


The difference here for the respective MenuTopping.java is the inclusion of import of javax. Libraries, MenuToppings implements Serializable in the pizza2 version. Pizza2 version has toppingName() method.

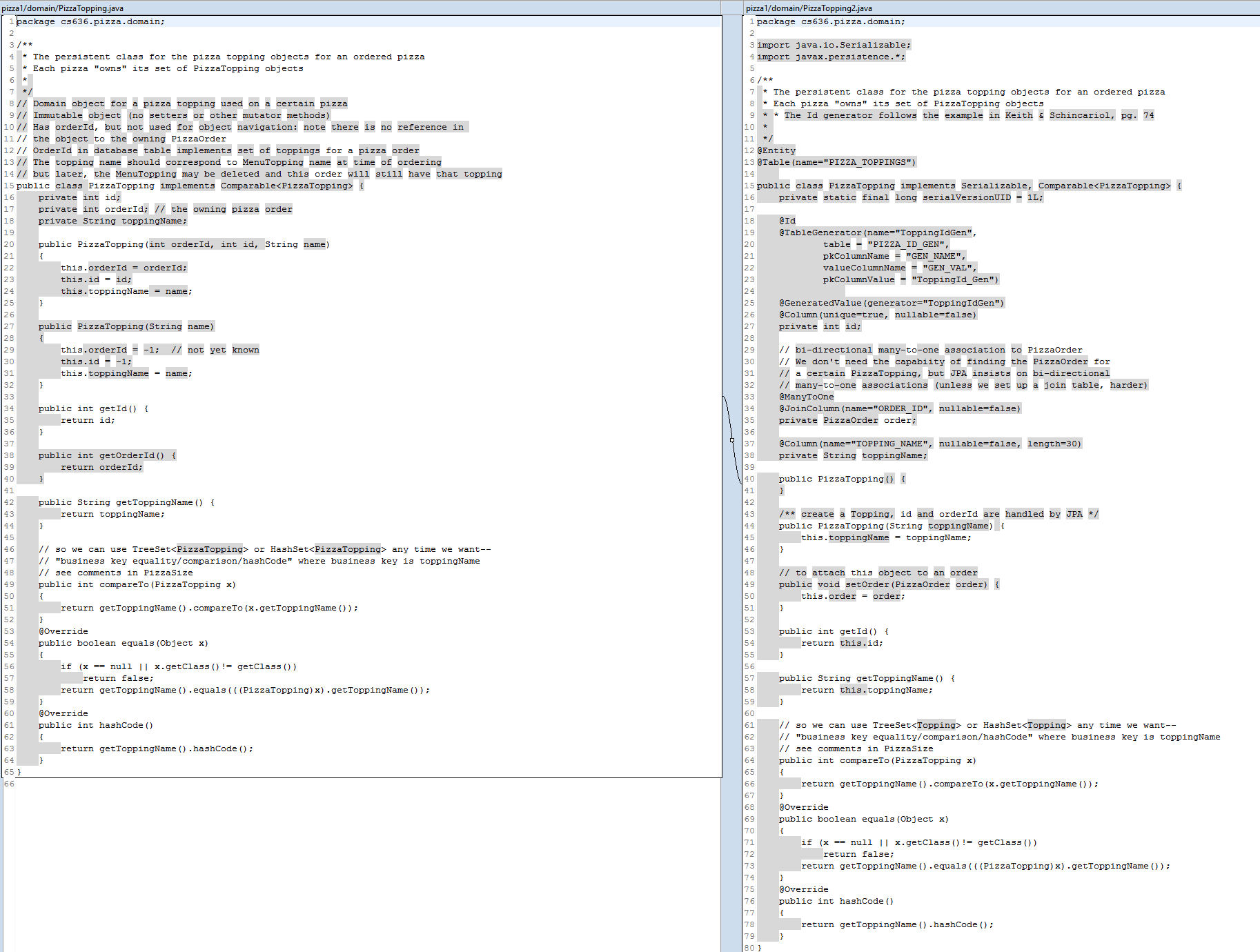
In the respective PizzaOrder.java classes, the Pizza2 version has a table generator, and a full constructor known as PizzaOrder().



In the PizzaSize.java class, in the pizza2 version, an setOrder() method is implemented, in getSizeName() a return call is implemented.



In the respective PizzaTopping.java classes, the pizza2 version implements a Table generator.



c. Try out the supplied unit tests in at least 4 different ways, including ant commands and eclipse: select project in Package Explorer, right-click, Run As>JUnit Test. Give the ant commands you used.

2.Try out tomcat, accessing just HTML pages to start. Tomcat is the servlet-capable web server that we will be using to execute web applications. I'll leave my installation of tomcat running on topcat.cs.umb.edu (on port 11600) for your experimentation. Note that you need a tunnel from localhost:11600 to topcat.cs.umb.edu:11600 to use this server, as you did earlier for the pizza page flow. You can look at this tomcat's files at ~eoneil/cs636/tomcat-8.5 in the UNIX/Linux filesystem. My directory ~eoneil/cs636 only has the tomcat installation, made following the student instructions. In other words, I'm pretending to be a student in the class, but letting you see the results, whereas when you put subdirectories under your own cs636 directory, noone else in the class may see them.

Had difficulty getting tomcat to work

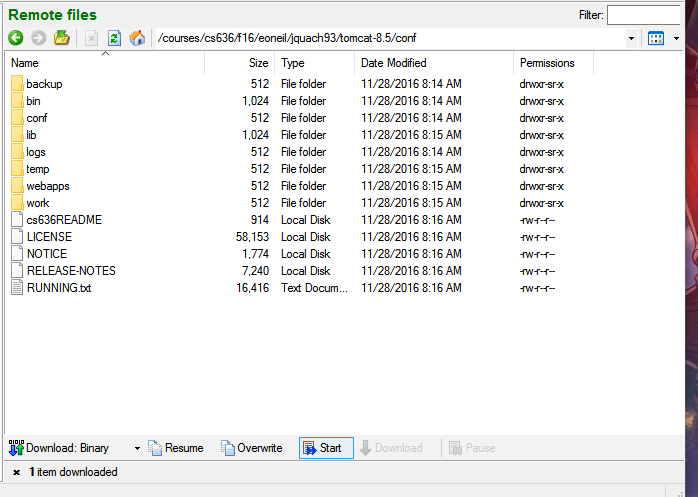
First browse to http:// topcat.cs.umb.edu:11600 to see the "root" page. From home, use URL http://localhost:11600/basicjsp/date.jsp, with a tunnel from localhost:11600 to topcat.cs.umb.edu:11600. You are using TCP port 11600, on which tomcat is listening on topcat. You should see a picture of a tomcat and some text about the Apache Tomcat project. Also links, including to some JSP examples of interest, but unfortunately written in old JSP.

Had trouble using tunnel to work

Then browse to http:// topcat.cs.umb.edu:11600/cs636/index.html to see my little index.html page--you will be making a similar one for yourself. This file index.html is situated at file path ~eoneil/cs636/tomcat-8.5/webapps/cs636/index.html in the UNIX filesystem. You are welcome to look at it there. The webapps directory is the root directory of this website served by my tomcat. That's why the URL, http:// topcat.cs.umb.edu:11600/cs636/index.html uses the part of the file path after webapps, the local path. Report on what this page says. Follow its link and see another little page. The first page was served by tomcat. What server served the second page?

Had trouble using tunnel to work

3.**Install Tomcat 8.5 on Linux system topcat.cs.umb.edu**, following the instructions linked from the class web page. As instructed there, make a little HTML page accessible at *http:// topcat.cs.umb.edu:NNNNN/cs636/index.html, (via the tunnel)* where NNNNN is the first port assigned to you, when your tomcat is running. This installation for your own tomcat will be in your cs636 directory, protected from other student's access. Don't worry, tomcat will be running under your user id, and will be able to read everything it needs. IMPORTANT: *Don't ever change permissions on your cs636 directory*. Report on success or problems.

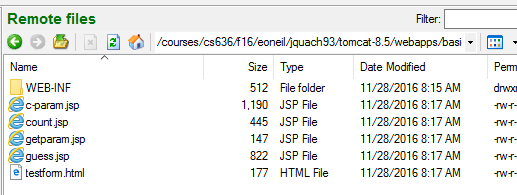


4.**Tomcat and simple JSPs and servlets**

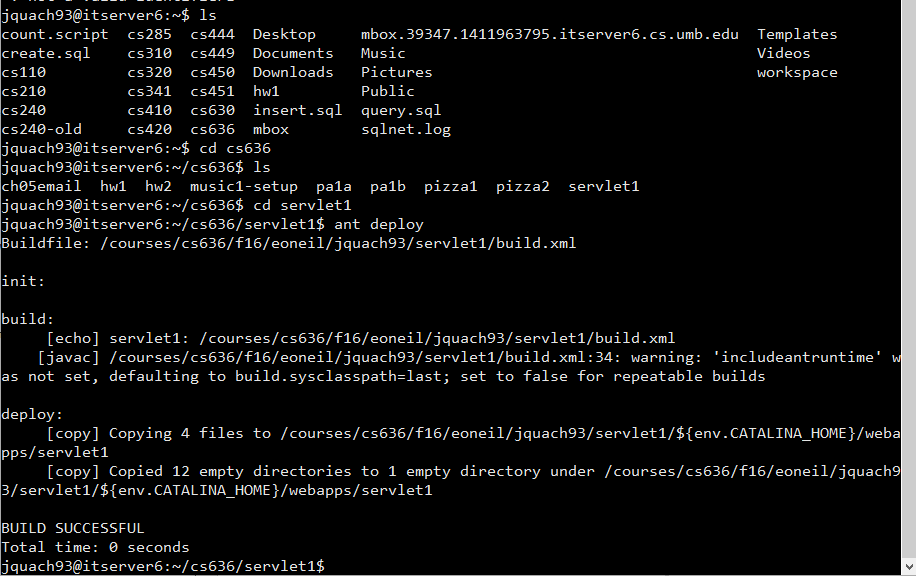
a.After you install tomcat on Linux, install the webapp named basicjsp from my Linux tomcat installation and use your ordinary browser to execute /basicjsp/date.jsp. From home, use URL http://localhost:xxxxx/basicjsp/date.jsp, with a tunnel from localhost:xxxxx to topcat.cs.umb.edu:xxxxx.



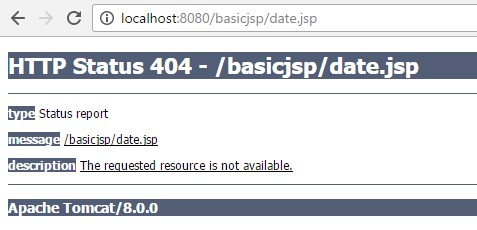
b.Similarly install the basicjstl web app from my tomcat, with its count.jsp and guess.jsp.



c.The servlet1 project: use "ant deploy" on your copy of this project to install in your tomcat.



Run these and report on what you see, and what HTML is sent to your browser. This shows the minimal setup for JSP and JSP with JSTL. For JSP alone, all we have to have is an empty WEB-INF directory in the webapp. For JSP with JSTL (e.g. with <@ taglib... prefix="c"%> and <c: in use), all we have to do is put jstl.jar and standard.jar in WEB-INF/lib, as you see for /basicjstl.

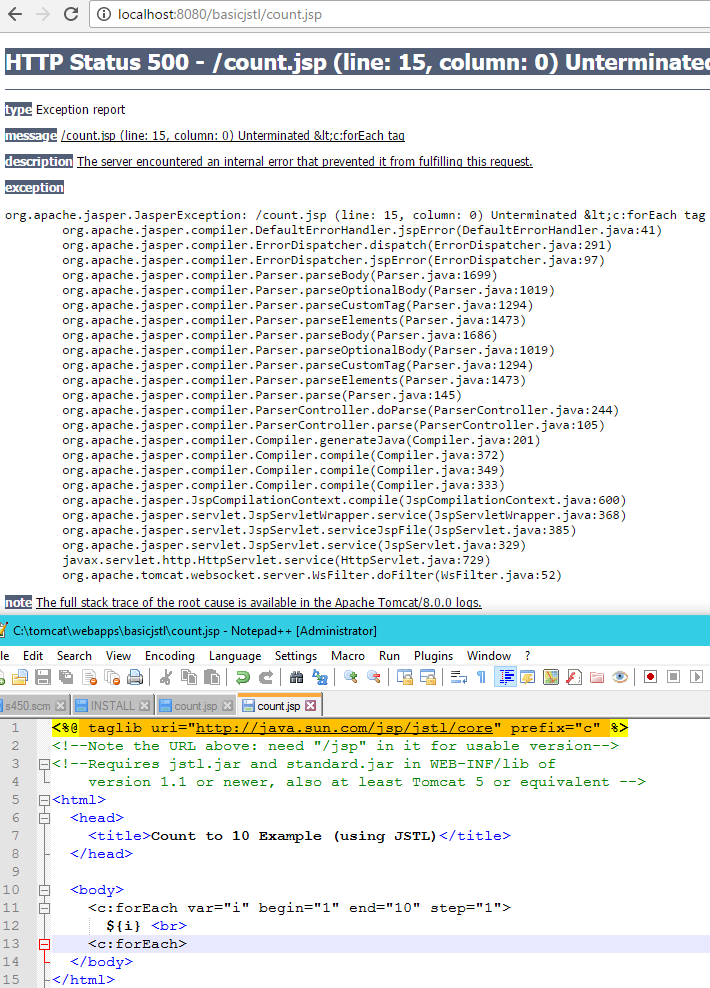


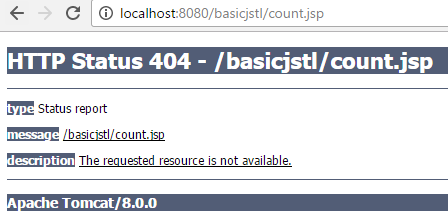
All I see is requested resource not available.

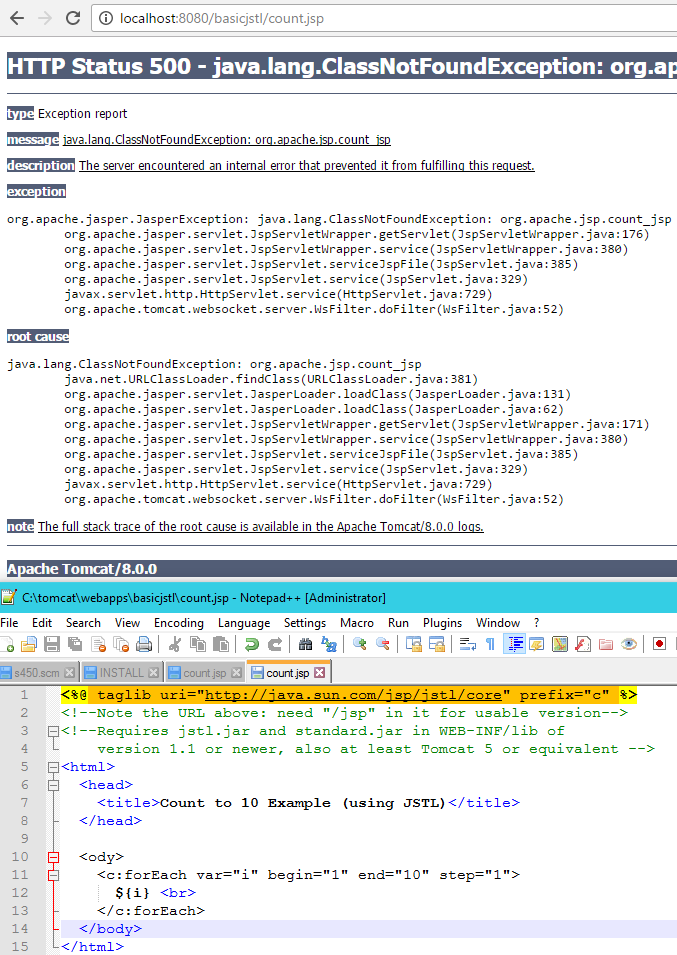
5.**JSP execution and errors.** Try the following on topcat, or your home tomcat if available.

a.Try breaking count.jsp by removing the / from </c:forEach>. Report on the error. Explain how it points to the bad line.

Tomcat points to a exception at line 15 ,column 0 where the </c:forEach> once was.



b.Try breaking count.jsp by misspelling body in <body>, say by deleting the b. What happens? Make sure the edit was effective by viewing the page source in your browser.  


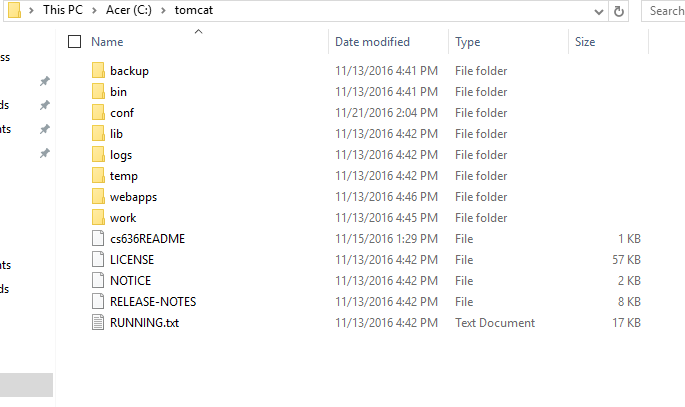
The results seem to indicate that the requested resource is not found.

c.Find the generated java files for date.jsp and count.jsp under tomcat's work directory. Report on their locations.

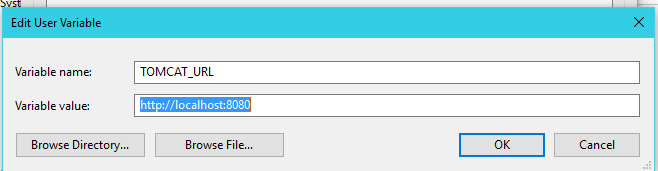
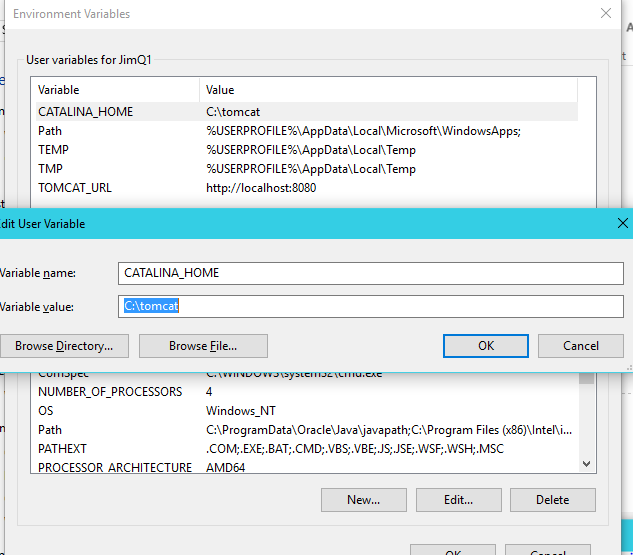
6.**Tomcat at Home:** Report on success or problems for these steps

1. a.Try out the installations in [Home System tomcat installation](http://www.cs.umb.edu/cs636/TomcatSetupForHome.html) described in steps 1-6.

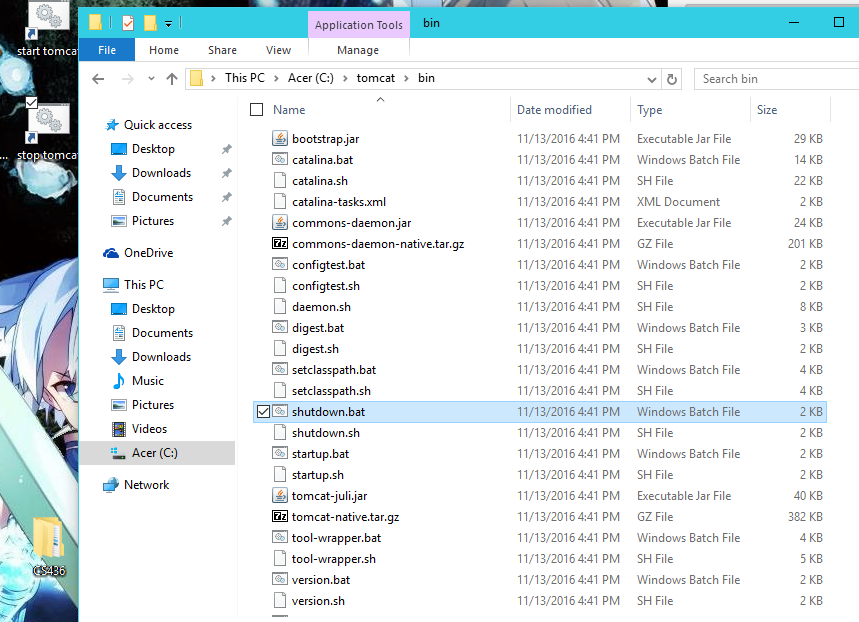
Part1 of setting up Tomcat



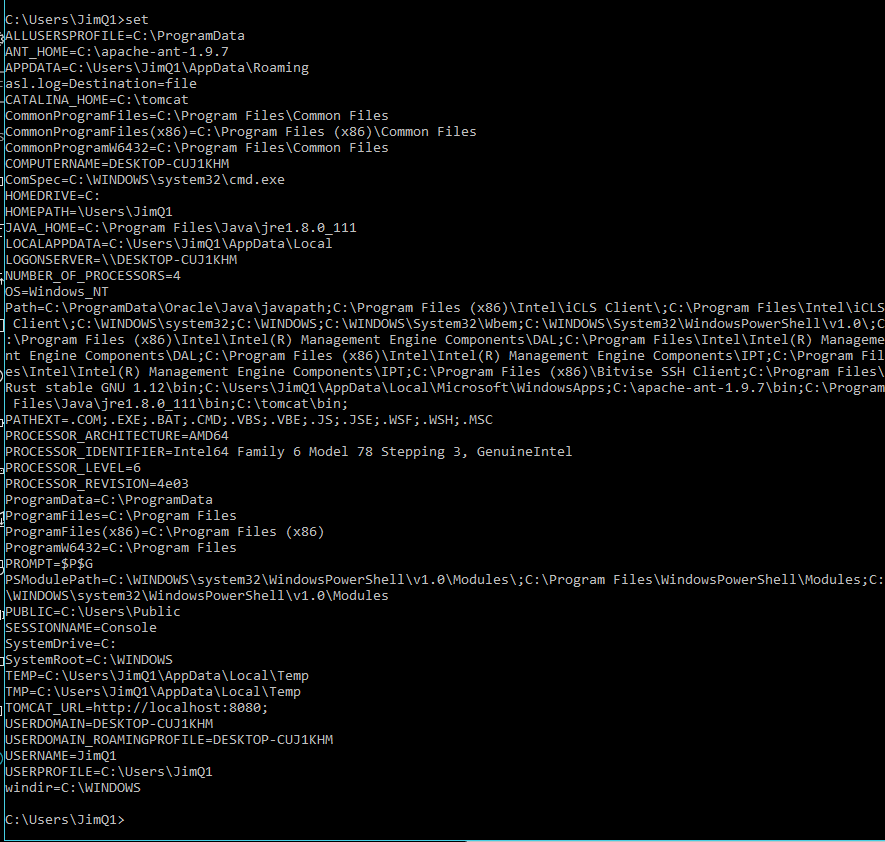
Part2 setting up CATALINA\_HOME



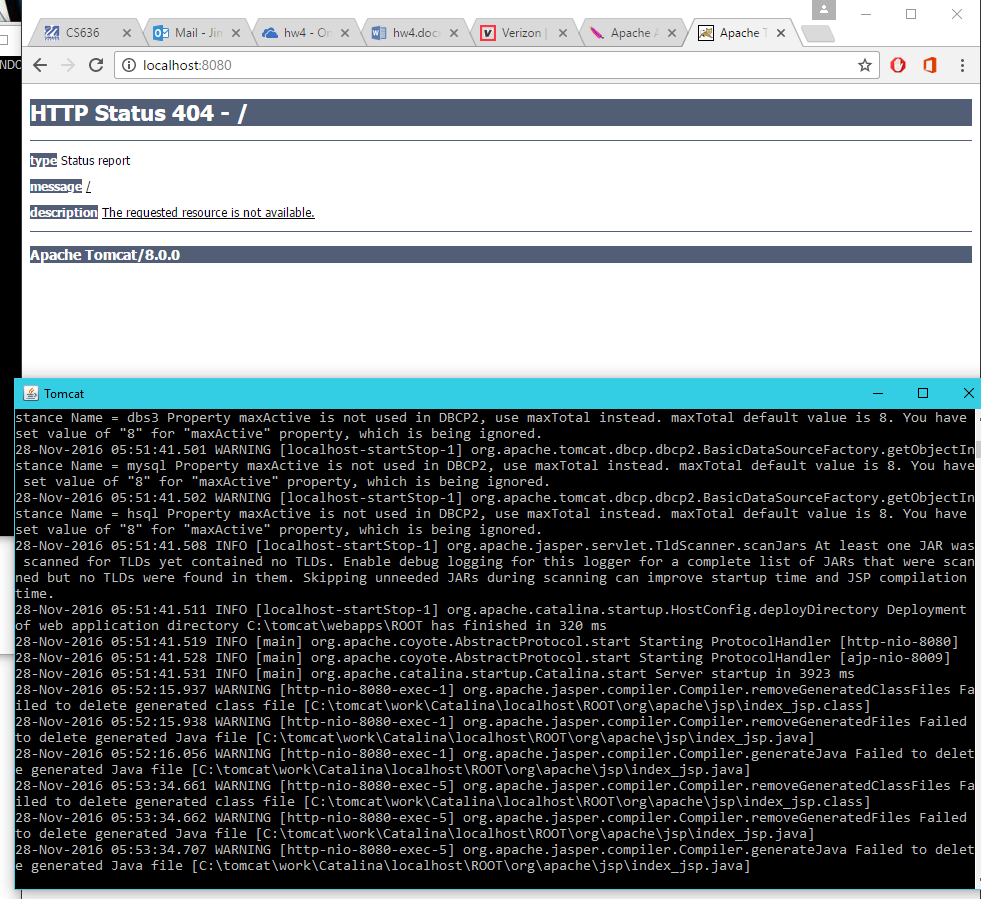
Part3 Shortcut Icons



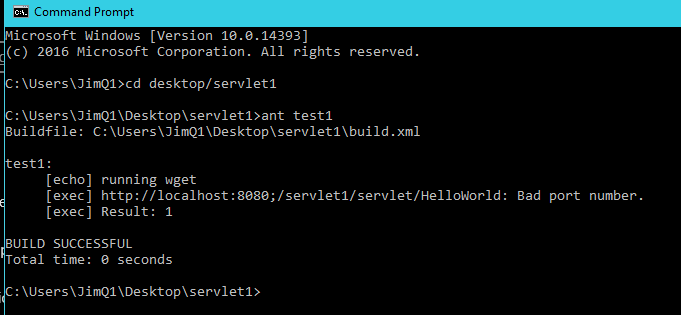
Part4 'set' to check environment variables



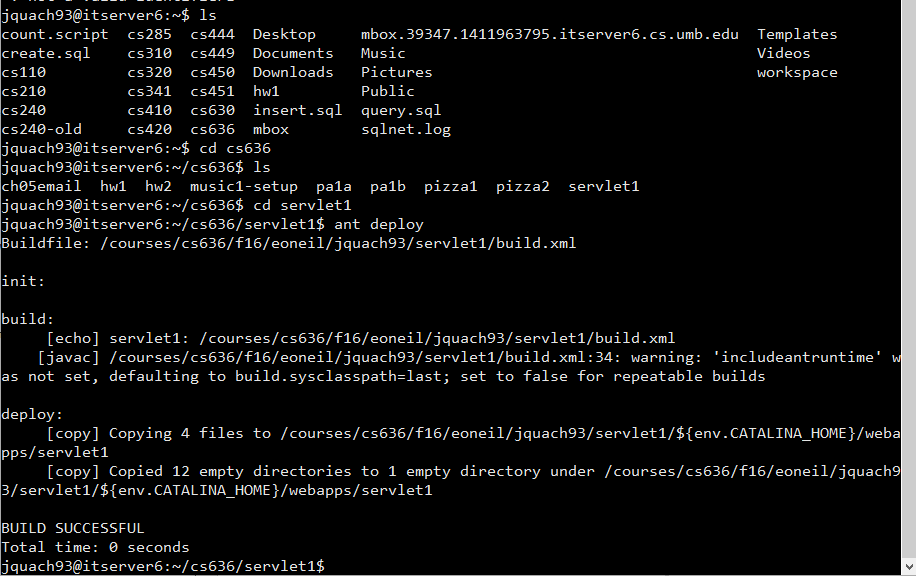
Part5 running http



b.Download the servlet1 project and try “ant test1” to use curl or wget to do the HTTP GET to its URL. If you are using Windows and wget is unavailable, it means your executables path doesn’t include tomcat’s bin directory, where it resides (to fix this, see step 2 of [Home PC tomcat installation](http://www.cs.umb.edu/cs636/TomcatSetupForHome.html)). If you are using Linux or Mac, curl should be available on the system.

.

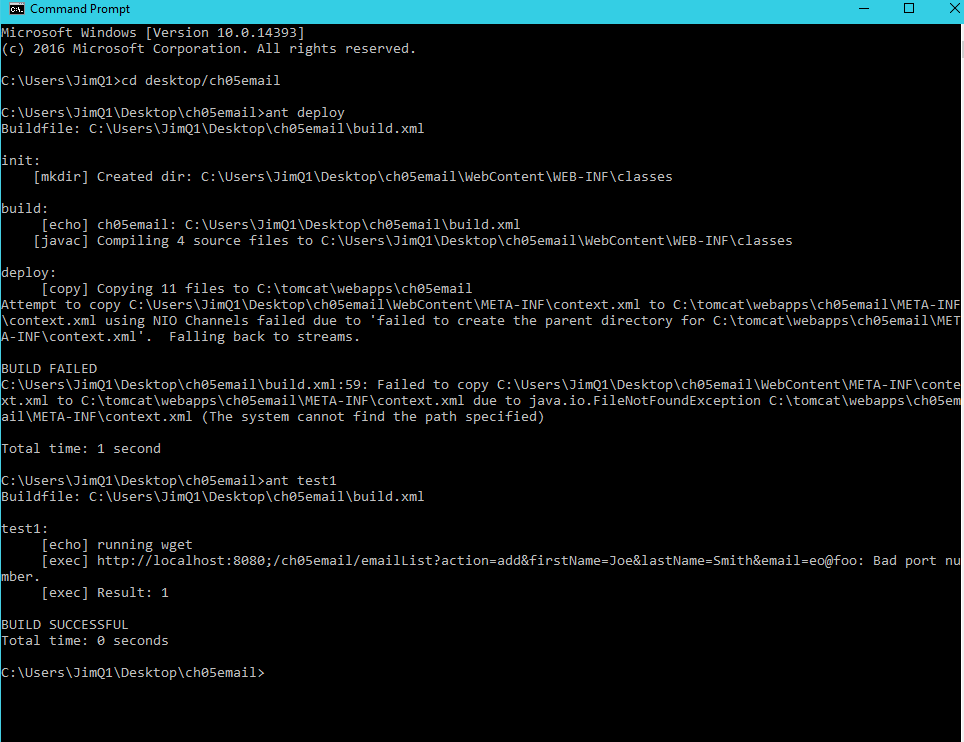
c.Also redeploy servlet1 in your Linux tomcat (i.e. using "ant deploy") and try it out the two ways. Report on your results.



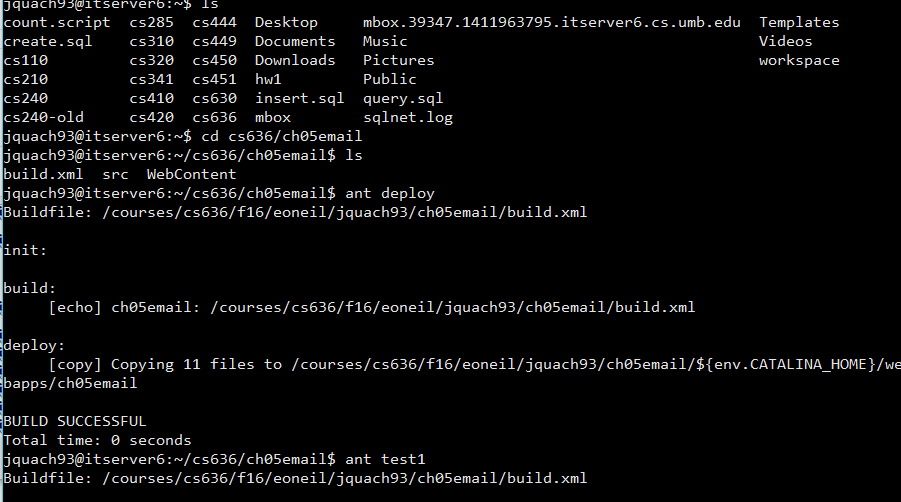
Servlet1 works with "ant redeploy"

7.**JSP and Servlets**: second servlet example, EmailListServlet

a.Unzip the provided [ch05email.zip](http://www.cs.umb.edu/cs636/ch05email.zip) project and use “ant deploy” to deploy it in your home tomcat and try it out the two ways (browse to its known URL and use “ant test1”). If your home tomcat is not working yet, say so and continue.

My tomcat doesn't seem to be working yet.

b.Also deploy ch05email in your cs.umb tomcat (i.e. using "ant deploy" on Linux at cs.umb.edu) and try it out the two ways, from topcat and also from users.cs.umb.edu. Report on your results. Note that curl is available on users.cs.umb.edu and our other Linux systems at cs.umb.edu, as well as on topcat.

From cs.umb.edu it works.