

## FINAL PROJECT GUIDELINES

For the final project, you will need to be familiar with several tools that have been installed in your machine. Towards this end, the following document is a summary of the steps that we went through to create our first web application during Monday's lecture.

### 1. MySQL

MySQL is the database server that we will be using to store and retrieve information. The files used to run the MySQL server are in the folder **C:\mysql**. The two important locations that you should be aware of are **C:\mysql\bin**, which has executable files to start and interact with the server, and **C:\mysql\docs**, which has extensive documentation on the MySQL version that we have installed.

I will now give you some basic heads up to get you started with using MySQL. Details on this can be found in the documentation in your directory.

#### 1.1 Starting the MySQL server

In order to use MySQL in your applications, you need to start the MySQL server installed in your machines from the command prompt. Here are the steps you make:

- go to **START->command prompt**
- navigate to the folder **C:\mysql\bin** from the command prompt.
- Start the MySQL server by typing in **mysqld-nt**. If successful, typing in this command should return no output.

Once the server has started successfully, you can log in to the server and execute queries from the command prompt. The command used to log in to the server is **mysql -u root**. If successful, this command will display a welcome message and present the MySQL prompt, to which you will be entering your queries.

What follows is a brief overview of some of the basic SQL statements that were covered in class on Monday. For a more extensive review of that material, you should look through Chapter 3 in the manual.html page in your **C:\mysql\docs** folder.

#### 1.2 Viewing and Creating Databases

You can see the list of databases already created in MySQL by typing in the command:

```
show databases;
```

Creating a database is also simple. The command for doing this is:

```
create database databaseName;
```

You can also delete a database by using the command

```
drop database databaseName;
```

Be careful when you use this command because once you delete a database, there is no going back.

#### 1.3 Working on a database

Once you have created a database, you can add tables to it by using the **create table** keyword.

Other important SQL keywords that you should know about are: **Insert, select, update** and **alter**. These are described in great detail in the MySQL manual that is already installed in the **C:\mysql\docs** folder.

## 2. TOMCAT

Tomcat is a commercial web server that helps display the output of your JSP pages. This web server has been installed in separate locations in many of the machines. A good place to look would be **C:/tomcat**.

The important folders that you will be working with are the **webapps** folder and the **conf** folder (which contains the server.xml file). We have created a **tomcat folder in your desktop** with shortcuts to all the files that you will be using to write your web application. Besides shortcuts to webapps and server.xml, this folder also contains shortcuts to the two executables that **startup** and **shutdown** the tomcat server.

### 2.1 Creating a new Web Application in Tomcat

#### 2.1.1 Where to put your files

In order to use tomcat to run your web applications, **you must put all your files in the webapps folder of the tomcat installation**. For example, for the lecture application that we created on Monday, we created a new folder named **lecture** and put that folder in webapps.

After creating the application's folder (lecture in this case), you should arrange all other files in the way that I outlined during lecture. In general, you should put all the files in the presentation layer in the lecture folder directly (that includes all html, JSP and images). You should then create a new folder named **WEB-INF**, and in the WEB-INF folder, create a new folder called **classes** that will hold all the java files necessary to run your application.

#### 2.1.2 Editing the server.xml file (do it once for every new web module)

After creating all the files for the web application, or whenever you are ready to see the output of your work, you should inform tomcat about the existence of a new web module before starting the server. This is done by adding the context path to the application in the **server.xml** file.

Usually, the context path is added near the very bottom of the server.xml file. If you want to find the right place to add a new context path, find the string "context path" in the file (ctrl-f in JCreator). This will direct you to the section of the file that contains context paths. For the lecture application, these are the lines that we added:

```
<!-- This is a comment. Lecture Application context -->  
<Context path="/lecture" docBase="lecture"  
debug="0" reloadable="true" crossContext="true"/>
```

After adding these lines, save and close the server.xml file. You are now ready to use tomcat to view the output of your JSP pages.

### 2.1.3 Starting Tomcat

You are now ready to see the output of your work. To start tomcat, just **go to the tomcat folder on your desktop** and click on the **shortcut to startup** icon. This will open a command prompt window with information on the status of tomcat as it starts. **DO NOT CLOSE THIS WINDOW.** Closing this window amounts to shutting down tomcat without following the right procedure. The right way to shut down tomcat is by clicking on the **shortcut to shutdown** icon on the tomcat folder in your desktop.

Once tomcat is up and running, open Internet Explorer and click on the **localhost** shortcut in the **favorites** tab. If successful, this should display the default homepage of the Apache-tomcat web server.

To see the output of a web module, add the name of the web module at the end of the current URL on the browser. For example, if the homepage is running at <http://localhost>, go to <http://localhost/lecture> to see the output of the lecture application. This should display a listing of the files that are associated with the lecture web module.

Please note that you may need to restart the tomcat server from time to time to see changes that you make to files in your web module. The right way to do this is by using the shortcuts in the tomcat folder on your desktop. Do not close the tomcat window directly. Be sure to refresh Internet Explorer after restarting the server to see any changes.

## 3 Interfacing MySQL with your JSP Pages

Getting JSP to work with your MySQL pages may seem like a daunting task, but it is much simpler than it appears at first. This is because MySQL (and Samuel), have already provided an API to help interface it with java, and all you have to do is add the classes that have already been created to your own **WEB-INF/classes** folder.

I will therefore ask you to copy the following folders and files from the **D:\aiti\jsp** folder into your own web application's **WEB-INF/classes** folder:

- the folders **org** and **com**, which are provided for you by MYSQL
- The file **DbConnect.java**, which I wrote to make calls to classes in the org and com folder and also the javax.sql classes. Make sure you compile this file before running your web application

I have included a file **UsingDatabases.java** in the **D:\aiti\jsp** folder, which has a method that shows you how to connect to a MySQL database and run some queries on it.

Make sure you have the mysql server running before you ask tomcat to execute any jsp script that uses the database.

-Samuel, G-UNIT