Programming Environment Setup

Tecnologie Informatiche per il Web (TIW)
DEIB, Politecnico di Milano

prof. Piero Fraternali

AY 21/22

Last updated on March 2, 2022

Contents

1 Software Requirements **1**

2 Installation **3**
2.1 Installing the latest JDK . . . 3
2.2 Installing Eclipse . . . 4
2.3 Installing Tomcat . . . 4
2.4 Installing MySQL Server and Workbench . . . 6

3 Configuring Installed Environment **9**
3.1 Configuring MySQL Workbench . . . 9
3.1.1 Adding Connections, Schemas and Tables . . . 9
3.1.2 Importing data in MySQL Workbench . . . 9
3.2 Configuring Eclipse . . . 10
3.2.1 Setting JDK . . . 11
3.2.2 Adding Tomcat . . . 11
3.2.3 Configuring Tomcat . . . 12
3.2.4 Starting Tomcat . . . 13
3.2.5 Stopping Tomcat . . . 14

4 Publishing Content **16**
4.1 Creating a Dynamic Web Project . . . 16
4.2 Writing the WEB Component . . . 16
4.3 Testing Your Application . . . 18
4.4 Removing a Project from Eclipse . . . 19
4.5 Removing a Project from Tomcat . . . 19

5 Connecting with JDBC **20**
5.1 Creating a New Project . . . 20
5.2 Adding MySQL Connector to the Project . . . 21
5.3 Testing the Connection . . . 21

6 Importing and Exporting Projects **22**
6.1 Importing a ZIP file . . . 22
6.2 Fixing Build Path . . . 22
6.3 Exporting a ZIP file . . . 23
6.4 Exporting schemas from MySQL Workbench . . . 24

7 Troubleshooting **26**
7.1 Understanding the Error . . . 26
7.2 Checking for a Solution . . . 26
7.3 Asking for Help . . . 27
7.4 Specific Errors . . . 27
7.4.1 Tomcat is not Starting . . . 27
7.4.2 MySQL Access Denied . . . 27
7.4.3 404 . . . 27
7.4.4 MySQL Unrecognized Timezone . . . 28
7.4.5 Code Changes are not Built . . . 28

1 Software Requirements

Download the following software packages. **Please be sure to get the exact software version to ensure full compatibility. You can click on any software on the list to open the corresponding download page.**

The coding environment has been tested on:

- Windows 10
- MacOS BigSur, Monterey
- Linux Ubuntu 20.04

Windows

- Eclipse Installer 2021-12R (top of the page)
- Apache Tomcat 9, core 64-bit Windows zip
- Java JDK 17, Windows x64 Installer
- Microsoft Visual C++ 2019 Redistributable Packages
- MySQL installer 8.0.28
- MySQL connector/j 8.0.28, Platform Independent (Architecture Independent), ZIP or TAR Archive.

MacOS

- Eclipse Installer 2021-12R (top of the page)
- Apache Tomcat 9, core zip
- Java JDK 17, MacOS DMG Installer
- MySQL connector/j 8.0.28, Platform Independent (Architecture Independent), ZIP or TAR Archive.
- MySQL Community server 8.0.28, DMG Archive
- MySQL Workbench 8.0.28, DMG Archive

Linux

- Eclipse Installer 2021-12R (top of the page)
- Apache Tomcat 9, core tar.gz
- Java JDK 17, to install from command line
- MySQL connector/j 8.0.28, Platform Independent (Architecture Independent), ZIP or TAR Archive.

Virtual Machine

Instead of installing all the packages in your environment, you can download a pre-configured Virtual Machine from here. The virtual machine can be easily executed after downloading and installing VirtualBox from here. You will need at least 4GB of ram, 2 cores of cpu, and 20GB available disk space.

If you decide to use the provided Virtual Machine, you can skip the Installation Section and continue with the Configuring Installed Environment Section.

2 Installation

2.1 Installing the latest JDK

The JDK installation guide for any platform can be found here.

Windows

After installing the Java JDK, you need to set the environment variables as explained here. **Since Java SE 13, the *JRE_HOME* variable is not needed anymore.**

The environment variables to set are:

```
JAVA_HOME: C:\Program Files\Java\jdk-17.0.2
JDK_HOME: %JAVA_HOME%
CLASSPATH: .;%JAVA_HOME%\lib
PATH: %JAVA_HOME%\bin;your-unique-entries
```

your-unique-entries represent the already existing entries in the PATH variable. They should be something like "C:\windows\system32;C:\windows;etc..." You should not modify those entries.

Figure 1 shows an example of how your JAVA variables should look.

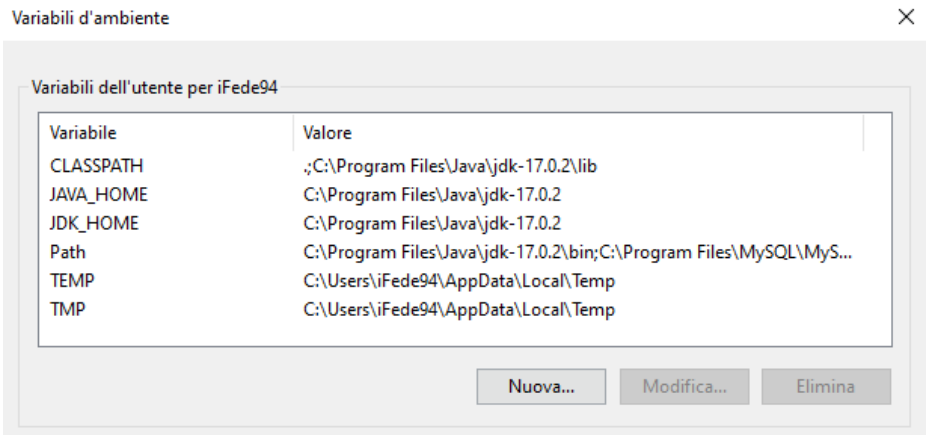

Figure 1: Windows environment variables required by JDK

MacOS

Just install the JDK, no additional step is required.

Linux

In the remainder of this guide, we will indicate with "Linux" any Debian-based distribution. Please notice that the provided instructions have been tested on the Linux Ubuntu 20.04 release.

You need to add the linuxuprising/java third-party repository. Open a terminal and execute the following commands:

```
sudo add-apt-repository ppa:linuxuprising/java
sudo apt update
```

Then, to install the Oracle JDK, execute:

```
sudo apt install oracle-java17-installer --install-recommends
```

You will be prompted a message to confirm the Oracle license terms & agreement, accept them (use the tab button to navigate the options), and continue.

Windows, MacOS, Linux

At the end of the Java installation process, open a terminal (Linux, MacOS) or a command prompt (Windows) and type these two commands:

```
java -version
javac -version
```

If everything went fine, the reported JDK versions should match the ones you just installed. See figure 2 for an example.

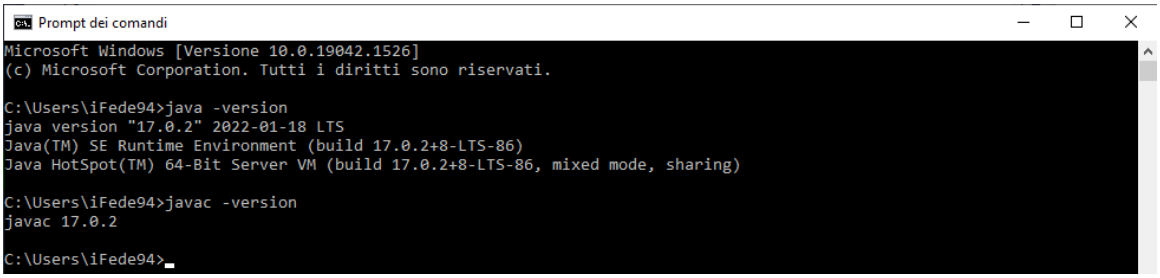

Figure 2: JDK versions after executing the terminal commands

2.2 Installing Eclipse

For all platforms, execute the Eclipse Installer you should have downloaded from the link in the Software Requirements Section. You will find the installer at the top of the page. When asked, install the **Eclipse IDE for Java Enterprise and Web Developers**. Figure 3 shows an example of the selection step. In the next window, select the correct JVM version. If a message appears asking to install an unsigned jar in the Eclipse folder, proceed to accept.

2.3 Installing Tomcat

Windows

After unzipping, rename the folder to "Tomcat 9.0" and copy it into your favorite drive (e.g., C: or D:). An example of installation path can be

```
C:\Program Files\Apache Software Foundation\Tomcat 9.0
```

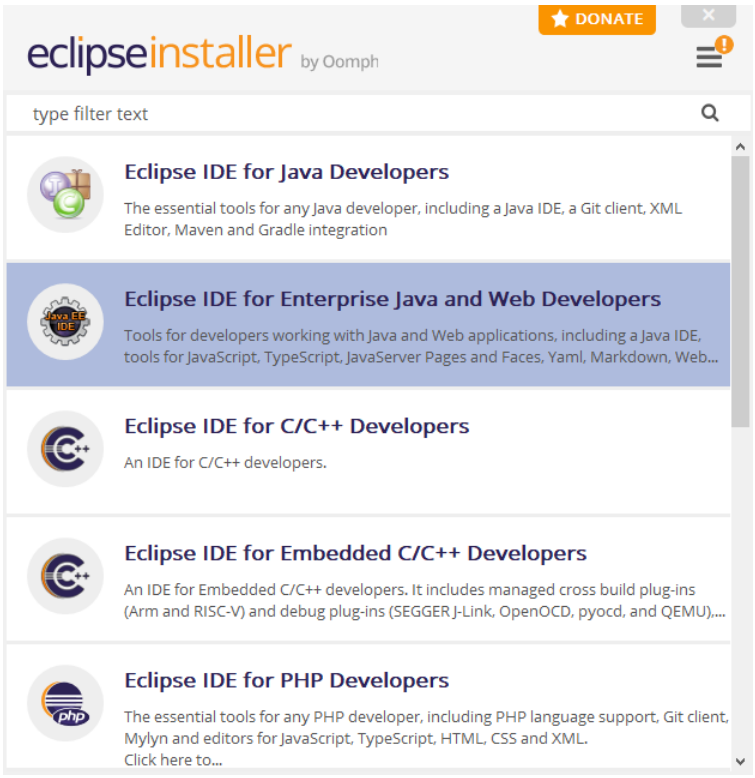

Figure 3: Installation of Eclipse IDE for Java Enterprise and Web Developers

MacOS

After unzipping, rename the folder to "Tomcat" and copy it into the Applications folder. Some scripts may not be executable so you need to gave them the right permissions. To do so, open a terminal and execute the following commands

```
sudo chmod +x /Applications/Tomcat/bin/*.sh
```

Linux

Extract the content of the downloaded file by opening a Terminal in your download folder and then typing:

```
tar xzf apache-tomcat-{version_number}.tar.gz
```

where {*version_number*} must be replaced by the tomcat version you downloaded.

Windows, MacOS, Linux

Inside the Tomcat folder, create a folder named "backup". It will be used by Eclipse.

2.4 Installing MySQL Server and Workbench

Windows

For the following steps you will need to have the "Microsoft Visual C++ 2019 Redistributable Package" installed in your system.

Execute MySQL Installer and choose the Custom installation. Here select MySQL Server, MySQL Workbench and MySQL Connector/J as shown in Figure 4. Accept all the default configurations and, when asked, you need to provide a **ROOT** password. Please, be sure to store that password because you will need it later. By default, MySQL is launched as a service at startup.

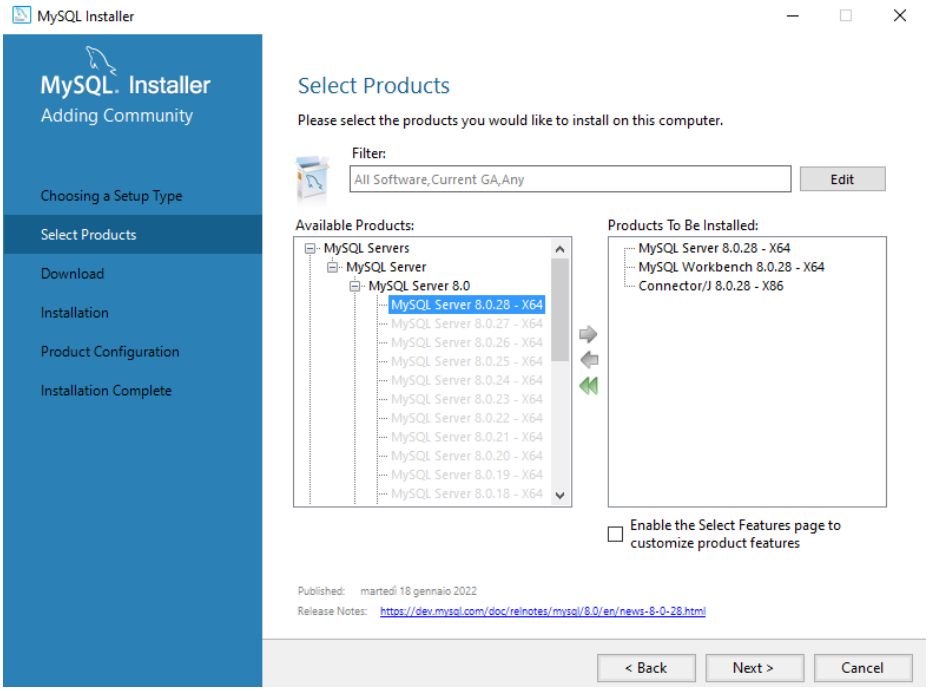

Figure 4: Selection of MySQL Server, MySQL Workbench and MySQL Connector/J

MacOS

Install MySQL Server with all the default configurations. When asked for the root password, insert a new password and be sure to remember it. You will need it later.

You can then start and stop your MySQL Server by going to the MySQL option in your MacOS settings (Figure 5).

Then, install MySQL Workbench. After the installation, open a terminal and input the following command.

```
sudo ln -s /Applications/MySQLWorkbench.app/Contents/MacOS/mysqldump /usr/local/bin/mysqldump
```

If when you open MySQL Workbench for the first time you encounter a "security issue", open your MacOS settings and select "Security and Privacy → General → Open anyway".

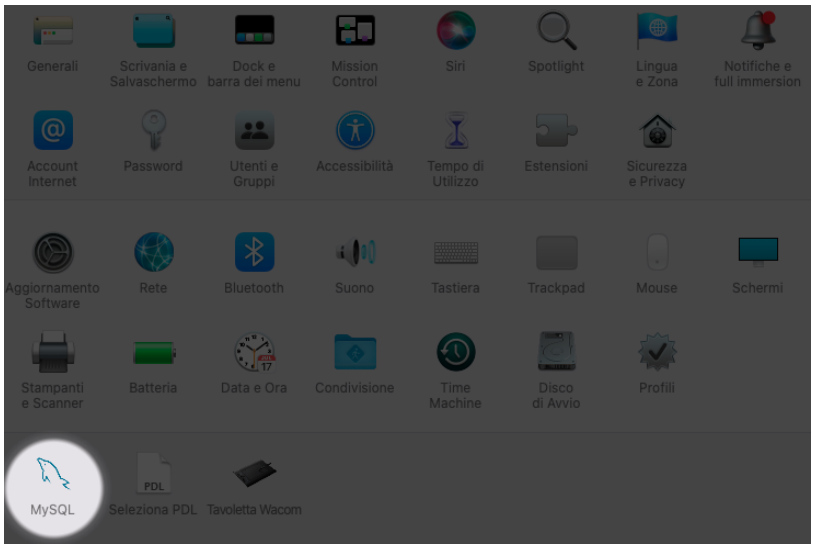

Figure 5: MySQL option in MacOS settings

Linux

Open a terminal and type:

```
sudo apt-get install mysql-server
```

During the installation you will be prompted to provide a password for the root user of the MySQL Server, be sure to remember the password. You will need it later. If the installation does not ask you to set the root password, you can follow this guide to set it. You will need to replace "my_new_pass" with the password of your choice.

Once installation is completed, the service will start automatically. To verify the status of the service, execute the following command:

```
sudo systemctl status mysql
```

To install MySQL Workbench type this in the terminal:

```
sudo apt-get install mysql-workbench
```

If mysql-workbench cannot be found, try following these steps:

1. Download the deb file from here.
2. navigate to your download directory and execute

   ```
   sudo dpkg -i mysql-apt-config_0.8.22-1_all.deb
   ```

3. In the new windows that appears, make sure that "MySQL Tools and Connectors" is enabled and you can disable "MySQL Server and Cluster" by pressing enter and then selecting "none". Next navigate to "ok" and press enter (Figure 6).
4. execute the following command

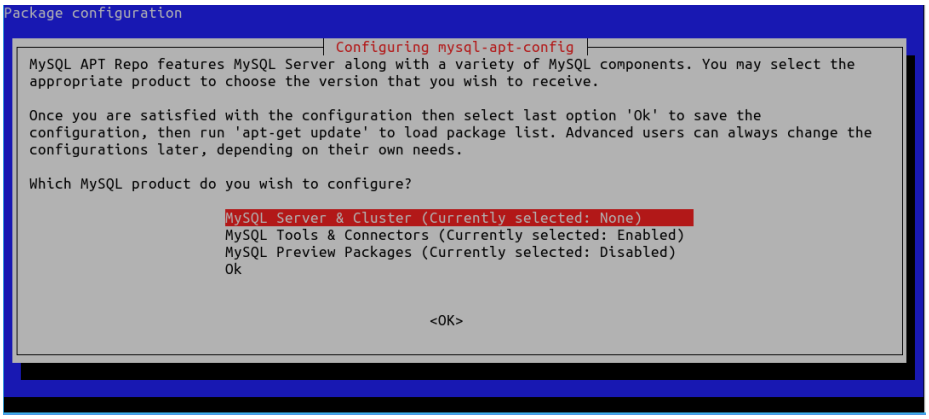

Figure 6: "MySQL Tools and Connectors" is enabled and "MySQL Server and Cluster" is disabled

```
sudo apt-get update
```

5. install MySQL Workbench with

```
sudo apt-get install mysql-workbench-community
```

If some error occurs, follow the suggestions in the terminal. Usually another update is required or the `--fix-missing` option must be put at the end of the last command. MySQL Workbench application will be available in the app drawer.

By default the server starts at boot time. If you wish to remove mysql-server from the startup, open a terminal and type

```
sudo update-rc.d mysql disable
```

You can start and stop the server by using, respectively:

```
sudo service mysql start
sudo service mysql stop
```

3 Configuring Installed Environment

3.1 Configuring MySQL Workbench

3.1.1 Adding Connections, Schemas and Tables

Open MySQL Workbench. In the homepage, press the "+" button near "MySQL Connections". In the new popup you will need to put a connection name (e.g. "TIW"), the username (usually "root"), and then press on "Test Connection" (Figure 7). You will be asked to insert your MySQL Server password. If everything is working correctly, you can press "OK" to save your connection. You will now find it in the homepage.

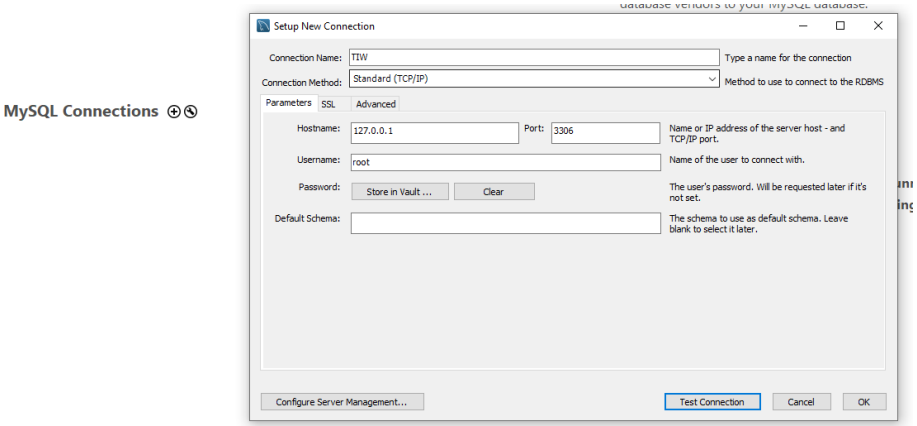

Figure 7: Adding a connection to MySQL Workbench

Click on the connection to open it. If asked, provide the MySQL server password. The MySQL editor will open. You will need to select the "Schemas" tab. If you cannot see this tab, enlarge the left sidebar in the window that appeared, and you will see it close to the "Administration" tab.

Right-click on the schemas list and select "Create schema" (Figure 8). Name the schema "dbtest" and press "apply" on the bottom-right. Now press the little refresh button near "Schemas". Your new schema should appear in the list.

To create a new table you have to select a schema and then right-click on "tables" and select "Create table". Here you can input the table name, create new columns and set their properties (e.g., data type, primary key, not null, auto incremented, etc..) (Figure 9). If you need to set foreign keys, you can do it by selecting the "Foreign Keys" tab on the bottom.

3.1.2 Importing data in MySQL Workbench

To import database dumps into MySQL Workbench you have to open the "Administration" tab and select "Data Import/Restore". In the new window, select the "Import from Self-Contained file" and browse to the desired .sql dump file. Then press "Start Import" (Figure 10).

This operation will be necessary to import the data provided with the course coding exercises.

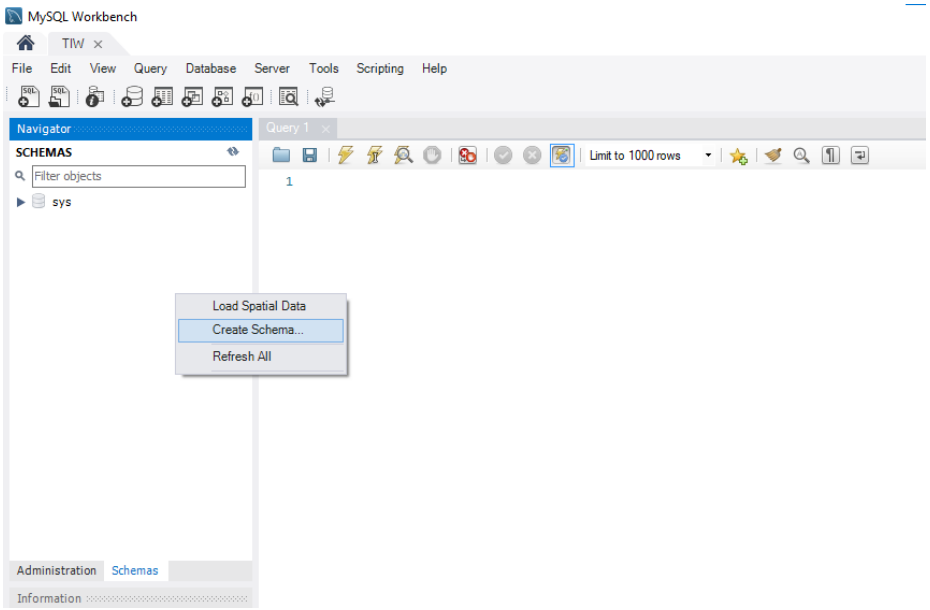

Figure 8: Creating a new schema

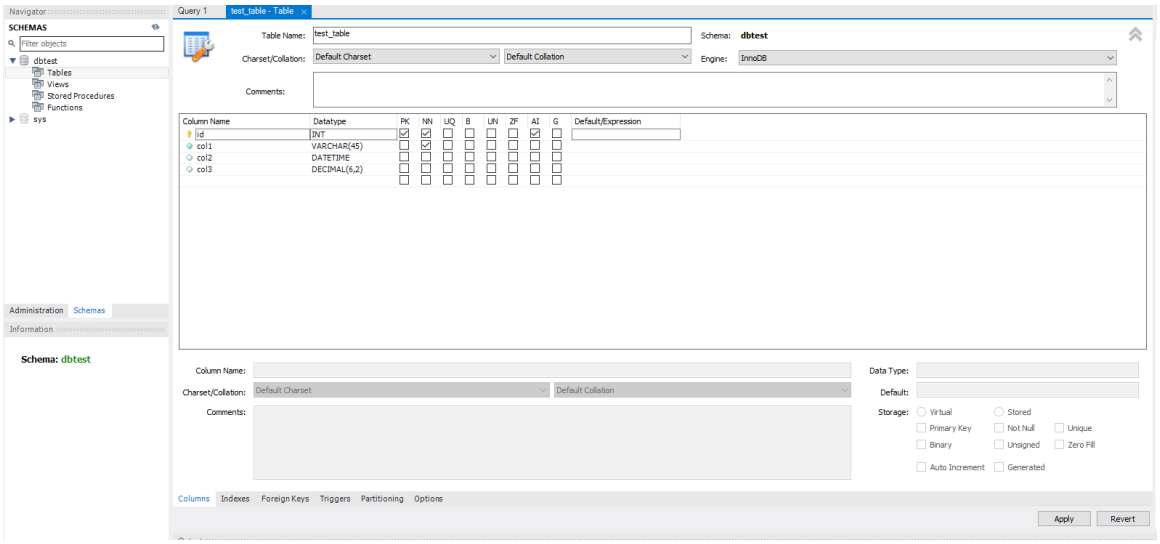

Figure 9: Creating a new table

3.2 Configuring Eclipse

If you are using Windows, be sure to run Eclipse as administrator since it will need write access to the Tomcat folder.

When opening Eclipse for the first time, you will be asked to choose a workspace folder. Just

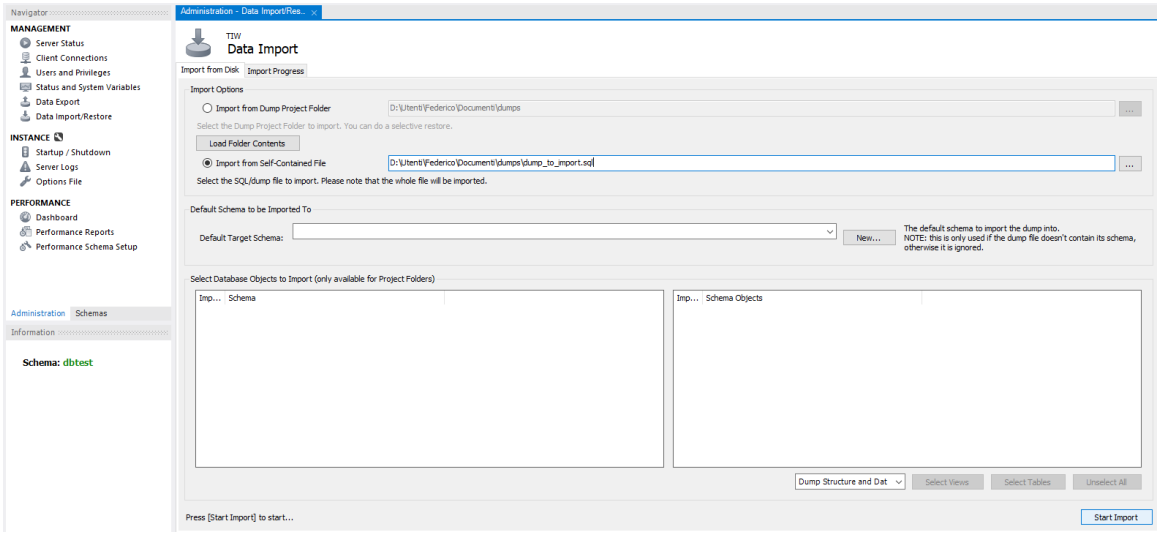

Figure 10: Importing a .sql file into MySQL Workbench

leave the default folder, then check "Use this as default and do not ask again" and continue.

3.2.1 Setting JDK

Open Eclipse. Go to Preferences:
Windows & Linux: Window → Preferences
MacOS: Eclipse → Preferences
then select the entry "Java → Installed JREs" and check if the installed JDK has been automatically detected by Eclipse (Figure 11). If Eclipse fails to automatically detect the presence of the installed JDK, proceed as follows:

Press the "Add" button. Then, select "MacOS X VM" (MacOS) or "Standard VM" (Windows, Linux). Finally, specify the location of the JDK installed in your PC and choose a name that you want (e.g., jdk-17.0.2).

Example path for JDK in MacOS:

```
/Library/Java/JavaVirtualMachines/jdk-17.0.2.jdk/Contents/Home
```

Then select "Java → Compiler" and be sure that the "Compiler compliance level" value is set to 17.

3.2.2 Adding Tomcat

Open Eclipse. Go to Preferences:
Windows & Linux: Window → Preferences
MacOS: Eclipse → Preferences
then select the entry "Server → Runtime Environments" and press the "Add" button. Select "Apache Tomcat 9.0" (Figure 12) and press "Next". Now, select the path of the Tomcat folder on your system and press Finish (Figure 13).

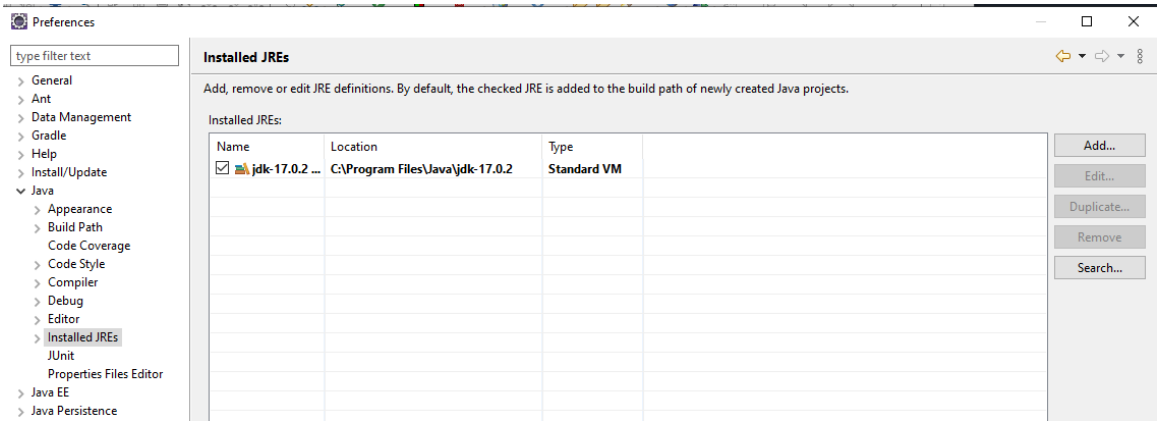

Figure 11: Set default JDK into Eclipse

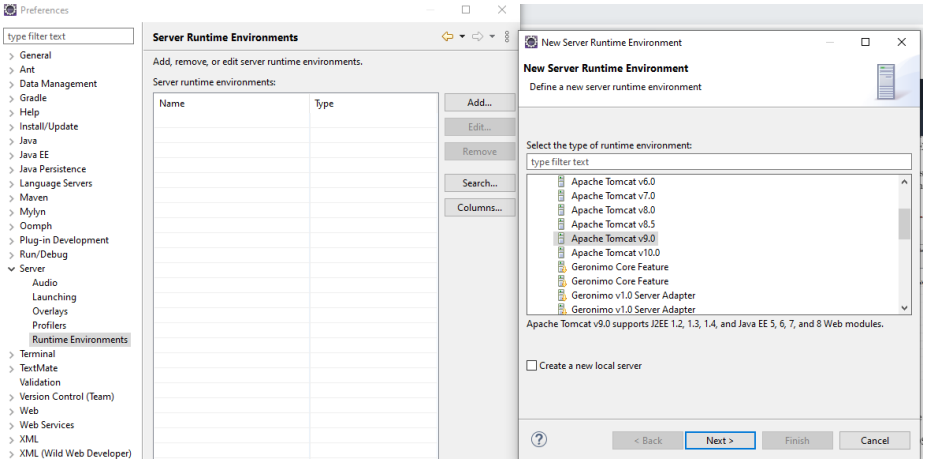

Figure 12: Adding a server runtime to Eclipse

Now, press “Apply and close” to save and close the preferences. Go to the “Servers” tab (Figure 14). If it is not visible, close the “Welcome” tab and go to “Window → Select View (Show View in Linux) → Other → Servers”.

Click on “No servers are available...”. Now, you have to select “Tomcat v9.0 Server” (Figure 15) and then click “Finish”. At this point, the Tomcat server is connected to Eclipse and one can interact with it through the Eclipse interface.

3.2.3 Configuring Tomcat

In the “Servers” tab double click on the server you just created. The server properties will open. Under the “Server Locations” section, select the option “Use Tomcat installation (takes control of Tomcat installation)” (Figure 16). Then modify the “Server ports” as shown in Figure 16. If you have other services running on the 8080 port, change the “HTTP” port to a value different than

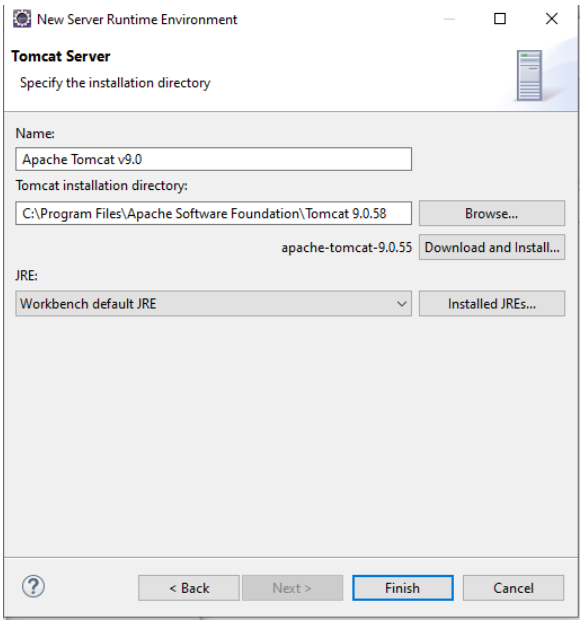

Figure 13: Adding a server runtime to Eclipse, step 2

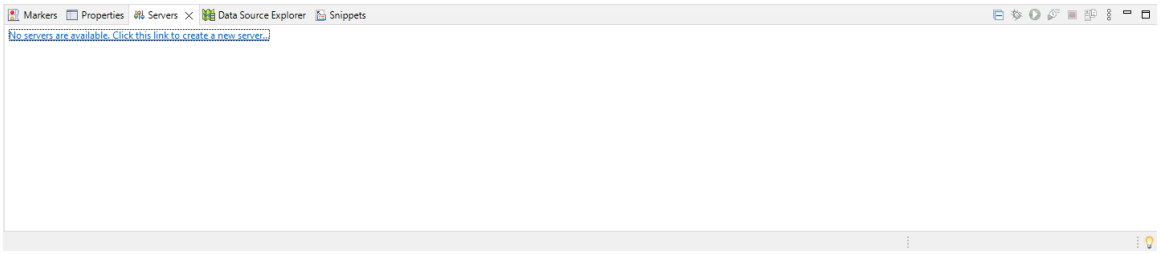

Figure 14: Servers tab

8080 (e.g., 8081). Save (ctrl-s or cmd-s) and close the server properties.

3.2.4 Starting Tomcat

In the "Servers" tab, right click on the Tomcat server and click on "Start". The console shows the server status. When the message "INFO: Server startup in XXX ms" is shown, the server is up and running (Figure 17). **If any problem occurs and the server is not starting, please refer to the Eclipse console to see the exact error**.

To check that Tomcat is running correctly, open your preferred Web Browser and navigate to http://localhost:8080.

Before continuing, go to "Window → Web Browsers" and select "1 Default web browser". This way your projects will automatically run on your preferred web browser.

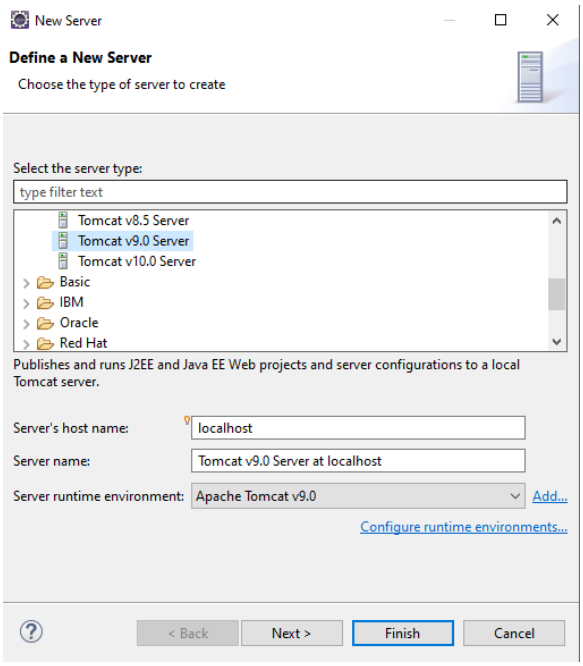

Figure 15: Adding Tomcat to Eclipse

3.2.5 Stopping Tomcat

When you are not using Tomcat anymore, you can stop the server by right-clicking on the server name and then selecting “Stop”.

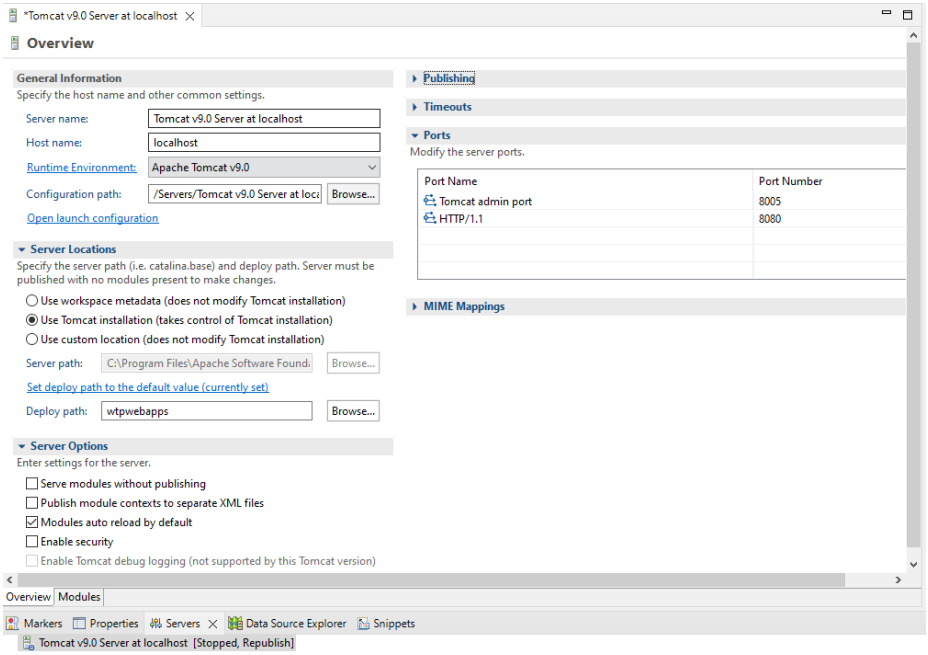

Figure 16: Configuring Tomcat deploy path and ports

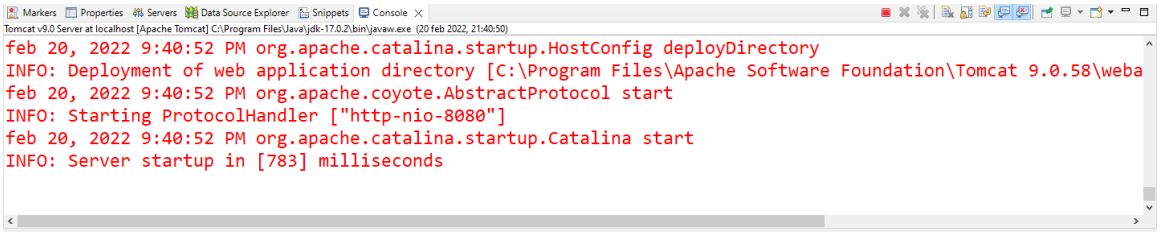

Figure 17: Tomcat console log after starting successfully

4 Publishing Content

4.1 Creating a Dynamic Web Project

Inside Eclipse, press "File → New → Project → Web → Dynamic Web Project" (Figure 18).

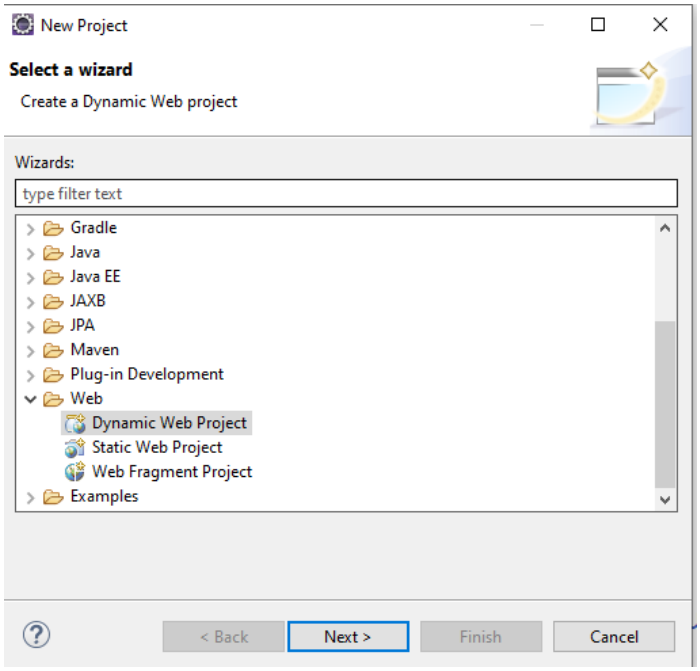

Figure 18: Creating a new WEB project

In the project creation wizard, name the project "TestProject" and press "Finish" (Figure 19).

4.2 Writing the WEB Component

Using Eclipse, double click on the newly created project in the left sidebar. Then, right click on the project source folder, i.e., "src/main/java", inside the created application, then select "New → Servlet". Give the name "TestServlet" to the servlet by writing it in the "Class name" field, and "it.polimi.tiw.test" in the "Java package" field. Then press "Finish" (Figure 20)

Now, double-click on "TestServlet.java" (inside "src/main/java *rightarrow* it.polimi.tiw.test"), and replace all the automatically-generated java code with the following code. You can use CRTL-SHIFT-F to indent the code automatically.

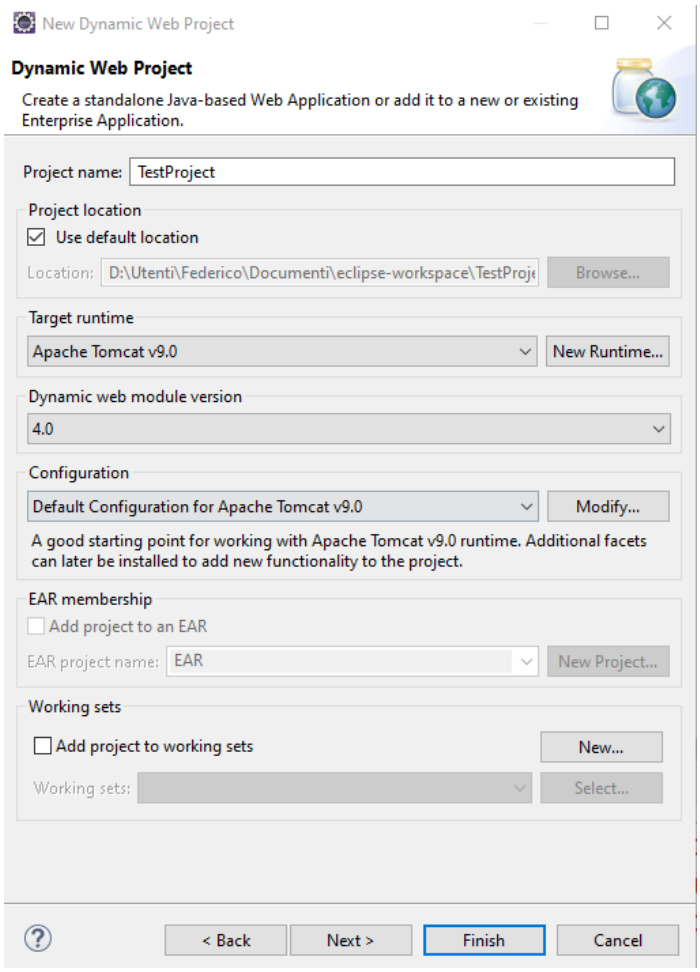

Figure 19: Creating a new WEB project

```
package it.polimi.tiw.test;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class TestServlet extends HttpServlet {
private static final long serialVersionUID = 1L;

  protected void doGet(HttpServletRequest request, HttpServletResponse response)
  throws ServletException, IOException {

    response.setContentType("text/plain");
    PrintWriter out = response.getWriter();
    out.println("Hello this is a test");
    out.close();

  }
}
```

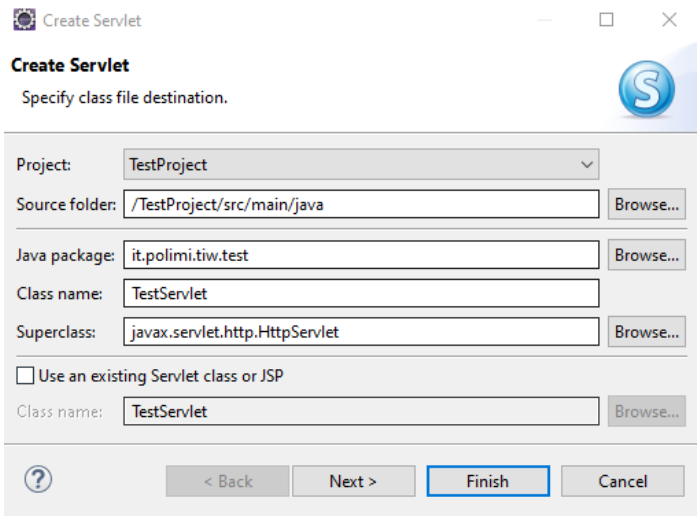

Figure 20: Creating a new servlet

Right click on the project and select "Java EE Tools → Generate Deployment Descriptor Stub". Now navigate to "src → main → webapp → WEB-INF". Right-click on the "web.xml" file and select "Open with → Generic Text Editor". Put the following content in it.

```
<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
version="4.0">
  <servlet>
    <servlet-name>TestServlet</servlet-name>
    <servlet-class>it.polimi.tiw.test.TestServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>TestServlet</servlet-name>
    <url-pattern>/</url-pattern>
  </servlet-mapping>
</web-app>
```

4.3 Testing Your Application

Right-click on "TestProject" project in the Project Explorer and select "Run As → Run on Server". Then, select the Tomcat server you created and press "Finish" (Figure 21) . If the server is already running, a popup will appear. Select the "Restart Server" option and press "continue".

Now your browser should open at the http://localhost:8080/TestProject address and a "Hello this is a test" message should appear (Figure 22). **If any problem occurs, please check the Eclipse console for any error message.**

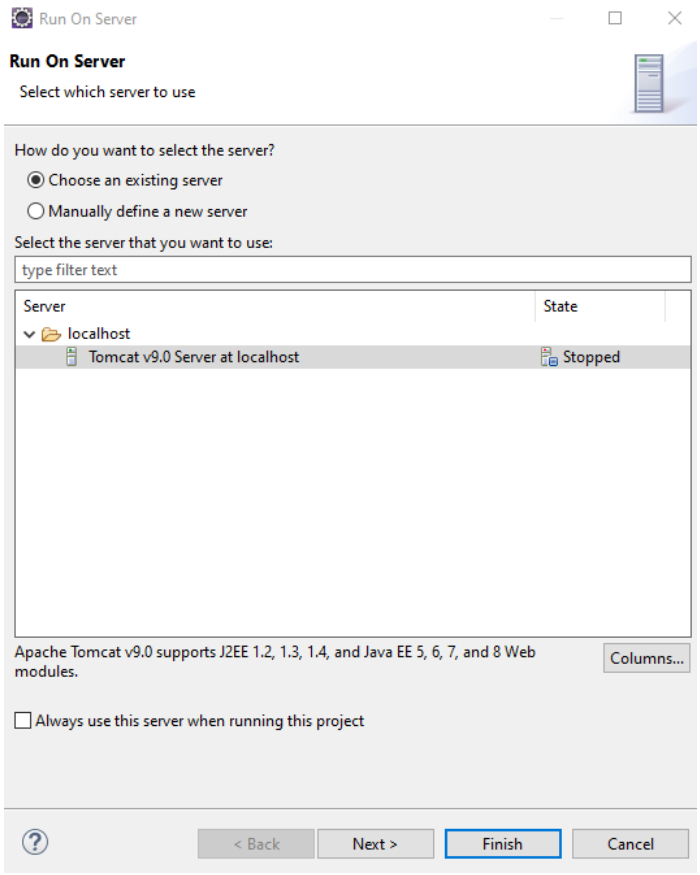

Figure 21: Running WEB project on Tomcat

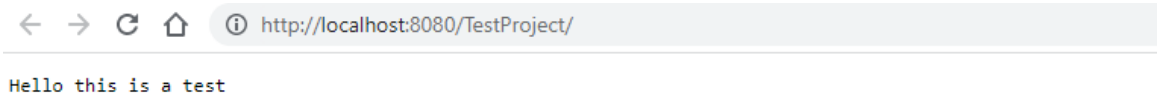

Figure 22: "Hello this is a test"

4.4 Removing a Project from Eclipse

If you want to delete a project from Eclipse, right-click on the project name from the Project Explorer and select "Delete". On the popup that appears, **make sure to check the "Delete project contents on disk (cannot be undone)"** and then press "OK".

4.5 Removing a Project from Tomcat

If you want to remove a deployed project from Tomcat, you can select the server and then right-click on the project you want to remove and press "Remove".

5 Connecting with JDBC

5.1 Creating a New Project

Create a new Dynamic Web Project and call it "TestJDBC". Then create a new servlet, the name should be "ConnectionTester" and the package "it.polimi.tiw.jdbctest". Then, open the servlet java file and replace all the code with the following code. Be sure to modify the *USER* and *PASS* variables with those of your MySQL installation. If you did not create a new MySQL user, just leave "root".

Remember to create the web.xml file and to declare the servlet mapping in the web.xml file as done in Publishing Content Section.

```
package it.polimi.tiw.jdbctest;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.DriverManager;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class ConnectionTester extends HttpServlet {
  private static final long serialVersionUID = 1L;

  protected void doGet(HttpServletRequest request, HttpServletResponse response)
      throws ServletException, IOException {

    final String DB_URL = "jdbc:mysql://localhost:3306/dbtest?serverTimezone=UTC";
    final String USER = "root";
    final String PASS = "difficult-root-pass";
    String result = "Connection worked";
    try {
      Class.forName("com.mysql.cj.jdbc.Driver");
      DriverManager.getConnection(DB_URL, USER, PASS);
    } catch (Exception e) {
      result = "Connection failed";
      e.printStackTrace();
    }

    response.setContentType("text/plain");
    PrintWriter out = response.getWriter();
    out.println(result);
    out.close();
  }
}
```

5.2 Adding MySQL Connector to the Project

Before continuing, check that you have downloaded the Connector/J archive from the Software Requirements Section and have extracted it somewhere. If you are using Windows and selected Connector/J in MySQL Installer, you will find it in a path similar to this.

```
C:\Program Files (x86)\MySQL\Connector J 8.0\mysql-connector-java-8.0.28.jar
```

To add the jar file to the project, you must copy it and then paste it in the "src → main → webapp → WEB-INF → lib" folder. Then, right-click on the jar file from the Project Explorer and select "Build Path → Add to Build Path" (Figure 23).

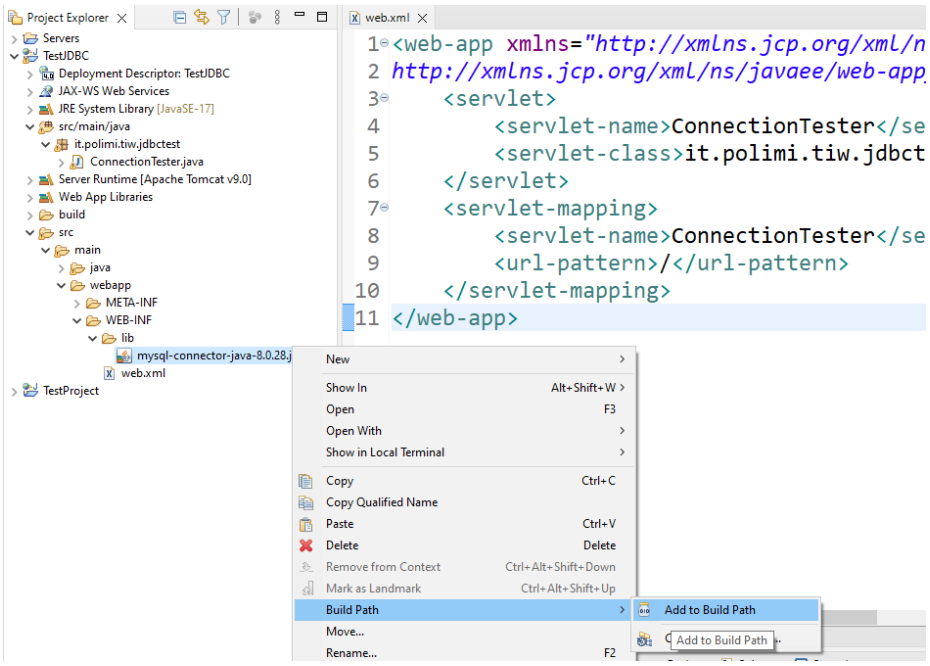

Figure 23: Adding the Connector/J jar to the project build path

5.3 Testing the Connection

To test the connection you need to have a "dbtest" database schema (refer to the Configuring MySQL Workbench Section).

Now, you can run the project ("right-click → Run As → Run on Server") and, if everything works correctly, you should see a "Connection worked" message. In case of errors, remember to check if the project was added properly to the Tomcat server. **If you get a "Connection Failed" message or any error, please check the Eclipse console to understand what is wrong.**

6 Importing and Exporting Projects

6.1 Importing a ZIP file

If you want to import an existing project into Eclipse, you must go to "File → Import → General → Projects from Folder or Archive". Then, you have to press on "Archive" and select the ZIP file of project you want to import. Two folders will appear, you have to select only the one that is indicated as "Import As → Eclipse Project" (Figure 24). Finally you can click "Finish" and the project will appear in the Eclipse Project Explorer. **You can continue with the Fixing Build Path Section to fix errors in the import project.**

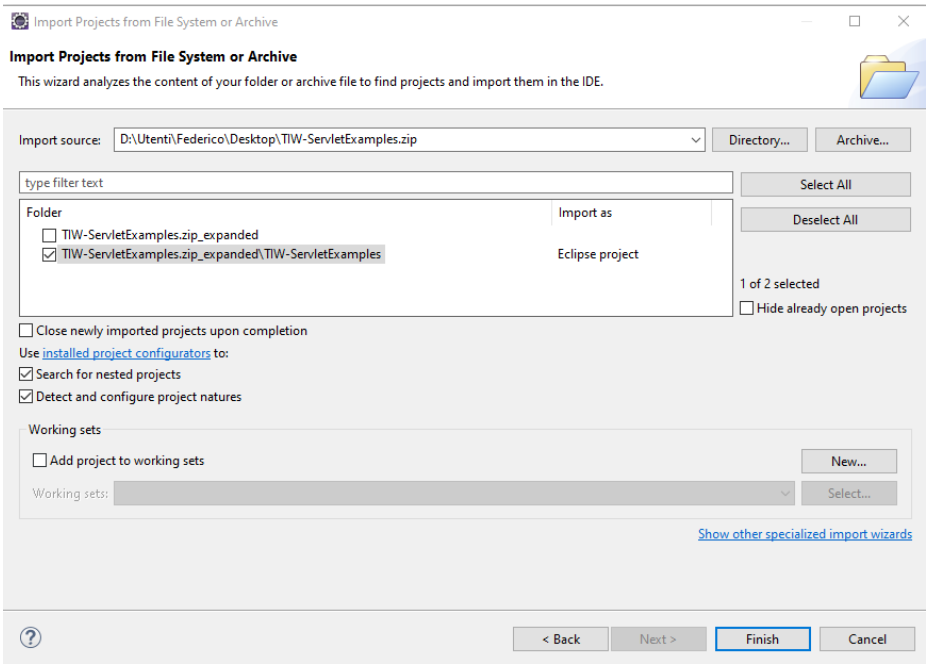

Figure 24: Importing a ZIP project

6.2 Fixing Build Path

After import a project you may note a little red "x" on the project folder. This means that there are some errors to solve before being able to run the project on Tomcat. You can see those errors on the "Markers" tab of Eclipse (Figure 25). For each error, you can see the corresponding project in the "Resource" column.

Most of the errors will be coming from the "Build Path" because the project may have been exported from an environment with a different JDK or libraries. You can check the "Build Path" by right-clicking on the project and then selecting "Properties → Java Build Path". If there are errors in the Build Path you will see that some libraries will have an "(unbound)" note near their name (Figure 26). You have to select them, one by one, and then press "Edit" and select the version installed in your environment. For example, when fixing the "JRE System Library" of Figure 26, you will need to select "Workspace Default JRE". The "(unbound)" note will disappear.

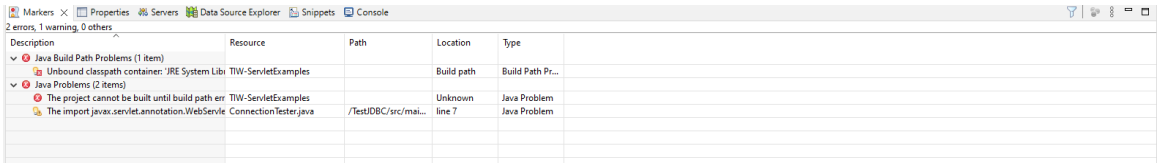

Figure 25: Markers tab with errors related to the "TIW-ServletExamples" project

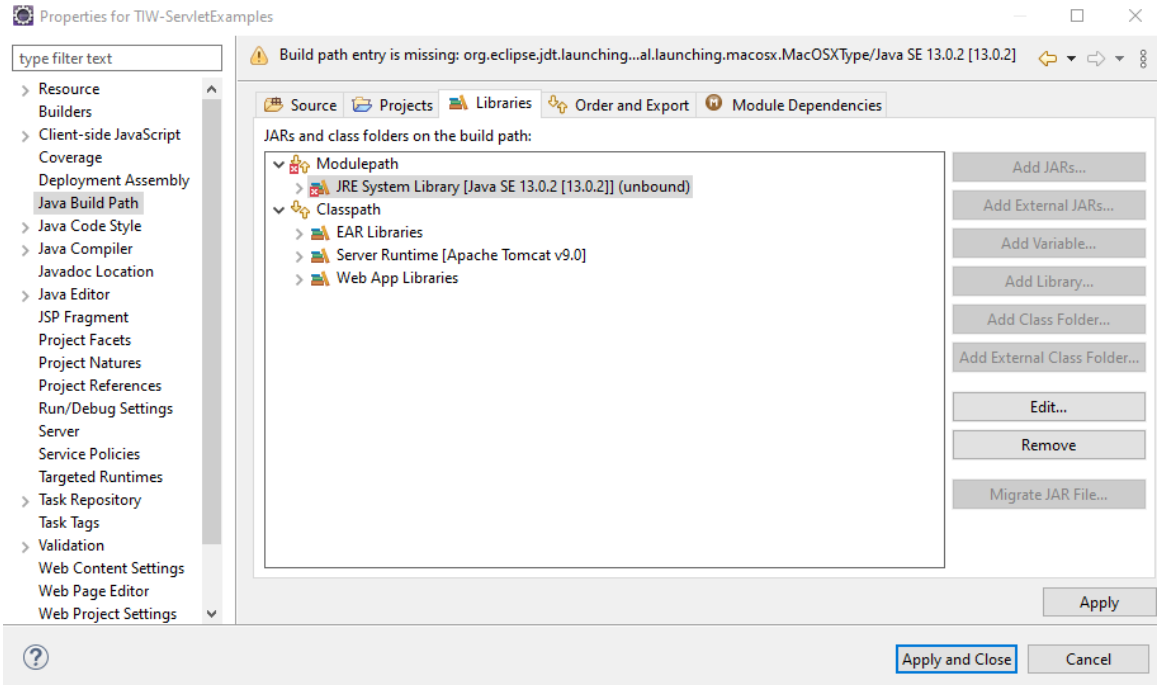

Figure 26: Build Path with unbound libraries

The same thing must be done on "Properties → Java Compiler", setting the "Compiler Compliance Level" to 17 and "Properties → Targeted Runtimes" by selecting only the Tomcat runtime configured in your system.**Please note that some errors, e.g. in JSP or HTML or JS, may remain and will not have any influence on your project runtime.**

Finally you must check the web.xml deployment descriptor or the java files to change some environment configurations (e.g., MySQL URL, username and password) to match yours. **If any readme file is provided with a project, read that carefully because it may contain instructions on how to configure or run a project.**

6.3 Exporting a ZIP file

To export an Eclipse project, you can select "File → Export → General → Archive File". In the new window, select the project you want to export and press on "Browse" to select the output folder

and the ZIP file name. Leave the default options selected (Figure 27) and press on “Finish”.

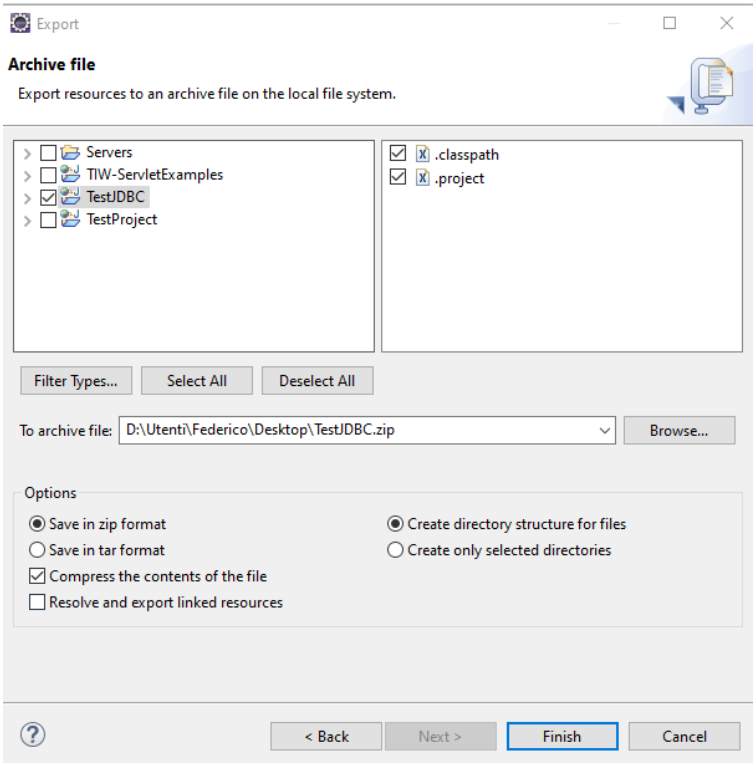

Figure 27: Exporting a ZIP Project

6.4 Exporting schemas from MySQL Workbench

To export a database dump from MySQL Workbench you have to open the “Administration” tab and select “Data Export”. In the new window, select the schema and then the tables you want to export. Check the “Dump Stored Procedures and Functions” and “Dump Triggers” options. Select “Export to self-contained file” and check the “Create Dump in a Single Transaction” and “Include Create Schema” options. Finally, press “Start Export” (Figure 28).

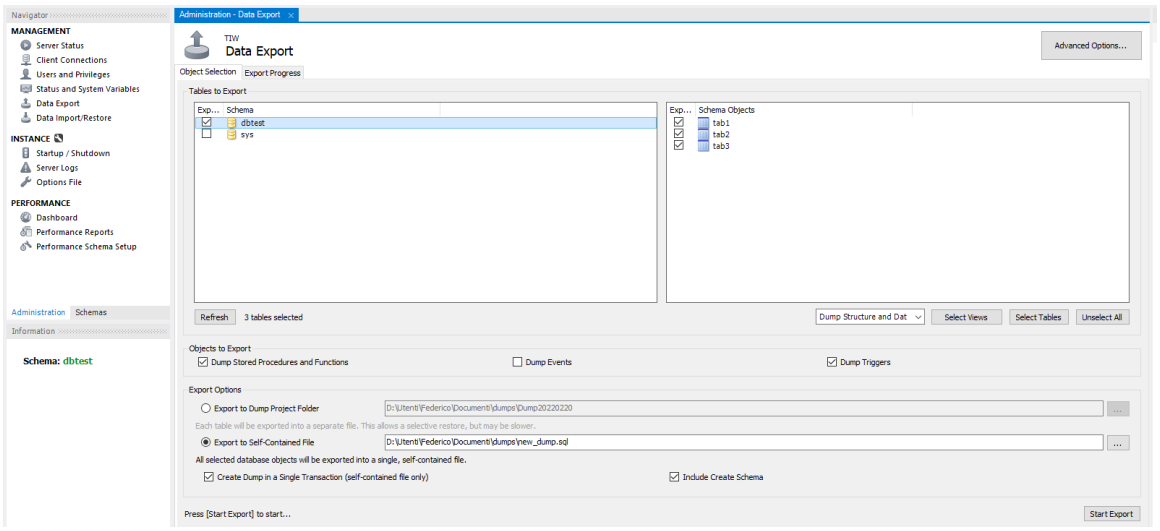

Figure 28: Exporting a .sql file from MySQL Workbench

7 Troubleshooting

7.1 Understanding the Error

When working with servlets or the database, the error must always be read in the Eclipse Console and not the web browser!

- In the Eclipse console you will find the exception containing: a name, a description and a stack-trace with the exact function (and row) in which the exception is thrown (Figure 29)
- Check if the message is a standard Java message (e.g. NullPointerException) or is a custom message written by us (e.g. Couldn't get DB connection). In the second case you must add "e.printStackTrace()" near our print to see the exact exception stack-trace
- Many times the Eclipse console shows more than one exception. You have to start from the top of the log and read everything to the end

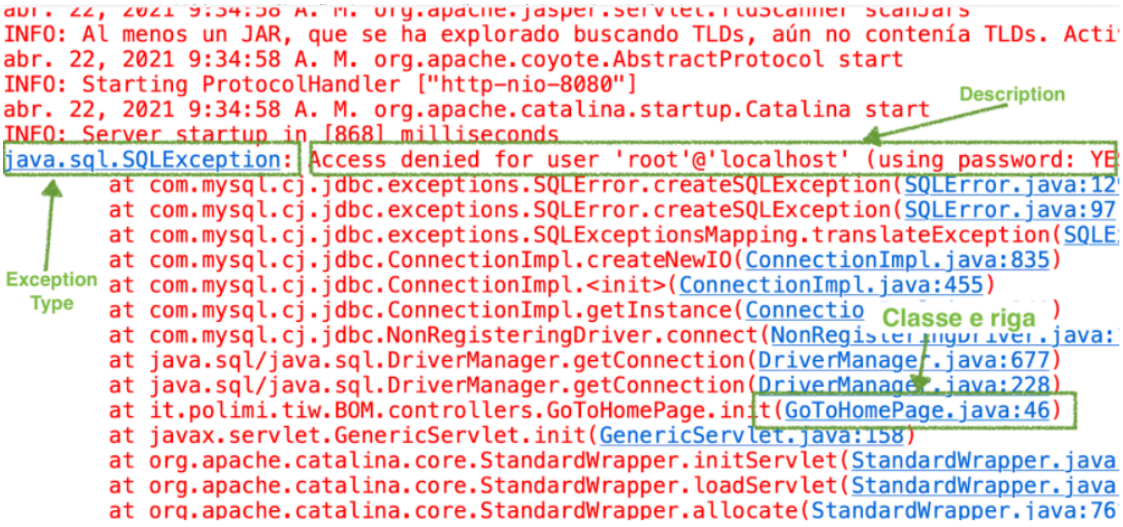

Figure 29: Details of an exception

7.2 Checking for a Solution

Re-installing all your programming environments is never the solution to an Eclipse exception!

After checking the Eclipse console, if the error is not clear you can copy the exception type and description on Google. Surely the same problem has already happened to someone else and you can find a solution on Stack Overflow.

Many errors will be explained during the exercise sessions!

If you encounter some errors when following this guide, please be sure to have followed all the steps.

If the error is inside your project, please check our coding examples uploaded on webeep to understand what you are doing differently and how we solved the problem.

7.3 Asking for Help

It is preferred that you ask questions on the webeep forum so the answers can help other students.

We will not debug errors in the exam projects! Debugging your code is part of the exam!

For any other error you will need to send an email to prof. Fraternali (piero.fraternali@polimi.it) by putting in cc (federico.milani@polimi.it and nicolooreste.pinciroli@polimi.it). In the email subject indicate "TIW" and a quick description of the error. **Be sure to hit reply-all when replying to our emails.**

In the mail body you need to put:

- Name of the project if it is a project downloaded from webeep.
- How to reproduce the error (e.g. go to the homepage, click on that button and make this action)
- The solutions that you already checked on StackOverflow and that did not work in your case
- A txt file with the **complete log** from the Eclipse console
- Screenshots only if really needed

Please put all these information in the email to help us solving your problem faster.

7.4 Specific Errors

7.4.1 Tomcat is not Starting

If Tomcat is not starting it can be caused by the projects that you deployed on the server. If the Eclipse console does not contain any error, remove all the deployed projects from the Tomcat server and run only the project you are working on. Now, if there is any error, it should appear in the Eclipse console.

7.4.2 MySQL Access Denied

If access to the database is denied when running a project in Eclipse, you must check that:

- MySQL server is active
- You changed the database credentials in the web.xml file or in the servlets where they are declared
- You are using the correct credentials. To check this, try connecting to MySQL Workbench by using the same credentials.

7.4.3 404

If a 404 error appears on your browser there may be two reasons:

- You are trying to access a resource that does not exist.
  Solution: check that the resource (servlet, html, jsp, etc..) that you are trying to access really exist, is correctly mapped and is in the correct deployment folder.

- There are some errors in the deployment of your project.
  Solution: check the Eclipse console to understand where the error comes from (**be sure to start from the top of the console log**) and make a quick search on the web on how to solve it.

7.4.4 MySQL Unrecognized Timezone

If a "MySQL Unrecognized Timezone Error" is logged in the Eclipse console, you can solve it by putting "?serverTimezone=UTC" at the end of your MySQL connection URL. This is an example.

```
jdbc:mysql://localhost:3306/dbtest?serverTimezone=UTC
```

7.4.5 Code Changes are not Built

If you are modifying the code and the changes are not reflected in the deployed project, you can do three steps:

- Save all the files you modified
- If the code changes are on a servlet, restart the tomcat server. If this does not solve the problem, remove the project from tomcat and re-deploy it.
- If the changes are on a static file (html, css, js, etc..), the problem resides on the browser cache. Usually you need to refresh the web page a couple of times to solve the issue. If needed you can also clean the browser cache before continuing.