
DatabaseServlet Web Application using Tomcat Server

AIM:

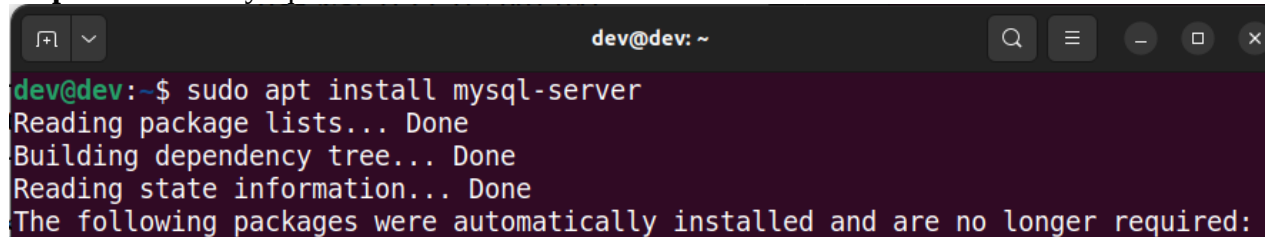
Develop a Simple Web application with MySQL database and Tomcat Server Authentication Application

Project Structure

- webapps
 - StudentDB
 - index.html
 - lib
 - mysql-connector-java-8.3.0.jar
 - WEB-INF
 - web.xml
 - classes
 - Main.java

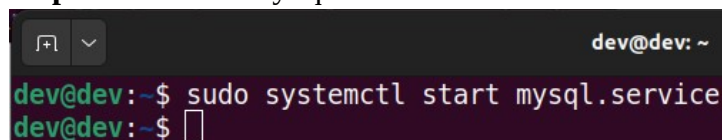
Procedure:

Step 01: Install MySql Server



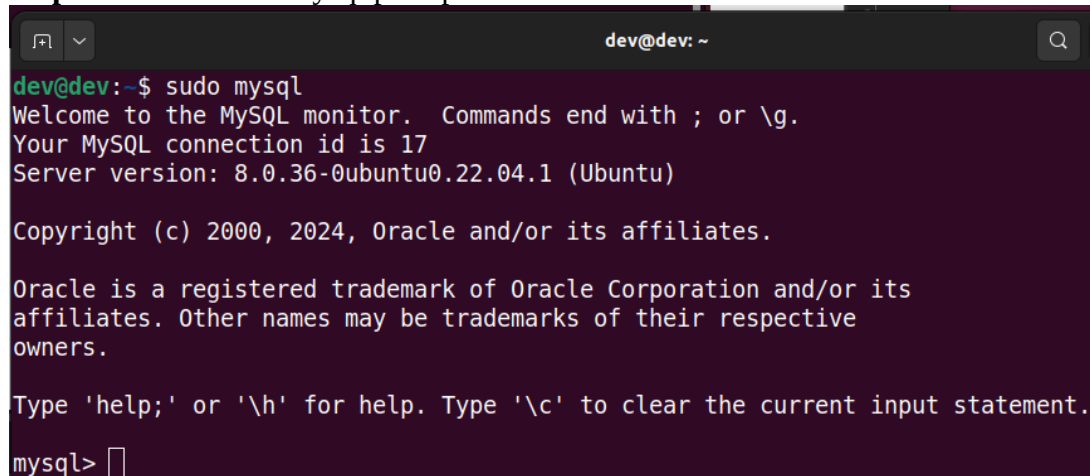
```
dev@dev: ~  
dev@dev:~$ sudo apt install mysql-server  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following packages were automatically installed and are no longer required:
```

Step 02: Start the MySql Service



```
dev@dev: ~  
dev@dev:~$ sudo systemctl start mysql.service  
dev@dev:~$
```

Step 03: Get into the mysql prompt



```
dev@dev:~$ sudo mysql  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 17  
Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu)  
  
Copyright (c) 2000, 2024, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql>
```

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'password';
Query OK, 0 rows affected (0.00 sec)

mysql> 
```

Step 05:

```
dev@dev: ~  
dev@dev:~$ mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 18  
Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu)  
  
Copyright (c) 2000, 2024, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql>  
mysql> 
```

Step 06: Create new User

```
dev@dev: ~  
mysql> CREATE USER 'user1'@'localhost' IDENTIFIED BY '10Red@pple';  
Query OK, 0 rows affected (0.02 sec)  
  
mysql> 
```

Step 07: Create Database and Table and provide privileges to the user1.

```
dev@dev: ~  
mysql> CREATE DATABASE authwebapp;  
Query OK, 1 row affected (0.01 sec)  
  
mysql> USE authwebapp;  
Database changed  
mysql> CREATE TABLE students(  
  -> name VARCHAR(255),  
  -> age INT  
  -> );  
Query OK, 0 rows affected (0.03 sec)  
  
mysql> GRANT ALL PRIVILEGES ON authwebapp.* TO 'user1'@'localhost';  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> 
```

Step 08: Download the JDBC Driver. <https://dev.mysql.com/downloads/connector/j/>

← → ↻ <https://dev.mysql.com/downloads/connector/j/>

IET G GH SSN java Cloud my CP APP Dev Clg-PRJ AI/ML IOT-Comm IP Rea

MySQL Community Downloads

Connector/J

General Availability (GA) Releases Archives ⓘ

Connector/J 8.3.0

Select Operating System:
Ubuntu Linux

Select OS Version:
All

Ubuntu Linux 23.10 (Architecture Independent), DEB Package (mysql-connector-j_8.3.0-1ubuntu23.10_all.deb)	8.3.0	2.3M	Download
Ubuntu Linux 22.04 (Architecture Independent), DEB Package (mysql-connector-j_8.3.0-1ubuntu22.04_all.deb)	8.3.0	2.3M	Download

MD5: 57f7804cb266b91d641dd66796446a37 | [Signature](#)

MD5: 986d56da97d002a50a0c67150a5b8d75 | [Signature](#)

! We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the

Step 09: Create a simple servlet project. (As per the Assignment-04)

```
dev@dev: /usr/local/apache-tomcat-9.0.85/webapps
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps$ sudo mkdir StudentDB
[sudo] password for dev:
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps$ cd StudentDB/
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB$ sudo touch index.html
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB$ sudo mkdir WEB-INF
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB$ cd WEB-INF/
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF$ sudo touch web.xml
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF$ sudo mkdir lib
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF$ sudo mkdir classes
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF$ cd classes/
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/classes$ sudo touch Main.java
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/classes$ cd /usr/local/apache-tomcat-9.0.85/webapps
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps$ sudo chmod -R 777 StudentDB
dev@dev:/usr/local/apache-tomcat-9.0.85/webapps$
```

Step 10: Extract the .deb file.

```
dev@dev: ~/Downloads
dev@dev:~/Downloads$ ls
mysql-connector-j_8.3.0-1ubuntu22.04_all.deb
dev@dev:~/Downloads$ dpkg-deb -x mysql-connector-j_8.3.0-1ubuntu22.04_all.deb mysql_connection
dev@dev:~/Downloads$
```

Step 11: Move the **mysql-connector-java-8.3.0.jar** file to the **WEB-INF/lib.** (if you don't have tomcat-servlet-api please follow the Assignment-04)

```
dev@dev: ~/Downloads
dev@dev:~/Downloads$ dpkg-deb -x mysql-connector-j_8.3.0-1ubuntu22.04_all.deb mysql_connection
dev@dev:~/Downloads$ sudo ln -s '/home/dev/Downloads/mysql_connection/usr/share/java/mysql-connector-java-8.3.0.jar' '/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/lib/mysql-connector-java-8.3.0.jar'
dev@dev:~/Downloads$ sudo cp '/usr/local/apache-tomcat-9.0.85/lib/tomcat-servlet-api-9.0.4.jar' '/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/lib/tomcat-servlet-api-9.0.4.jar'
dev@dev:~/Downloads$ sudo chmod 777 '/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/lib/tomcat-servlet-api-9.0.4.jar'
dev@dev:~/Downloads$ 
dev@dev:~$ systemctl restart mysql
dev@dev:~$
```

Step 12: Code

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Student Web App</title>
  <style>
    body{
      height: 100%;
      width: 100%;
      display: flex;
      align-items: center;
      justify-content: center;
      background-color: lightcoral;
    }
    .content{
      height: 70%;
      width: 400px;
      background-color: red;
      padding: 5%;
    }
    h1{
      text-align: center;
    }
    label{
      padding: 10px;
      display: block;
    }
    button{
      margin-left: 35%;
      padding: 4px 30px;
      margin-top: 20px;
    }
  </style>
```

```

</head>
<body>

    <section class="content">
        <h1>LOGIN</h1>
        <form action="login" method="get">
            <label for="name">Username: <input type="text" name="username"></label>
            <label for="age">Password: <input type="text" name="password"></label>
            <button type="submit">Login</button>
        </form>
    </section>
</body>
</html>

```

web.xml

```

<web-app>
    <servlet>
        <servlet-name>Main</servlet-name>
        <servlet-class>Main</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>Main</servlet-name>
        <url-pattern>/login</url-pattern>
    </servlet-mapping>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>
</web-app>

```

Main.java

```

import java.io.*;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import javax.servlet.*;
import javax.servlet.http.*;

public class Main extends HttpServlet {
    private static final String URL = "jdbc:mysql://localhost:3306/authwebapp";
    private static final String USERNAME = "user1";
    private static final String PASSWORD = "10Red@pple";

    static {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
        } catch (ClassNotFoundException e) {
            throw new RuntimeException("Error loading MySQL JDBC driver", e);
        }
    }
}

```

```

public void doGet(HttpServletRequest request, HttpServletResponse response)
    throws IOException, ServletException {
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();

    String username = request.getParameter("username");
    String password = request.getParameter("password");

    try (Connection connection = DriverManager.getConnection(URL, USERNAME,
PASSWORD)) {
        String sql = "SELECT * FROM myusers WHERE username = ? AND passwd = ?";
        try (PreparedStatement statement = connection.prepareStatement(sql)) {
            statement.setString(1, username);
            statement.setString(2, password);

            try (ResultSet resultSet = statement.executeQuery()) {
                if (resultSet.next()) {
                    out.println("<html><head><title>Login Success</title></head><body>");
                    out.println("<h2>Login Success</h2>");
                    out.println("<h2>Hello " + username + "!</h2>");
                    out.println("</body></html>");
                } else {
                    out.println("<html><head><title>User Not Found</title></head><body>");
                    out.println("<h2>User Not Found</h2>");
                    out.println("</body></html>");
                }
            }
        }
    } catch (SQLException e) {
        out.println("<html><head><title>Database Error</title></head><body>");
        out.println("<h2>Database Error</h2>");
        out.println("<p>" + e.getMessage() + "</p>");
        out.println("</body></html>");
    }
}

```

SQL Query:

```

CREATE TABLE myusers(
    username VARCHAR(255),
    passwd VARCHAR(255)
);
SELECT * FROM myusers;

INSERT INTO myusers VALUES('karthikeyan','password');
INSERT INTO myusers VALUES('Testing','testing');

SELECT * FROM myusers;

```

Step 13: restart the server

```
dev@dev:~/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/classes$ javac -cp "/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/lib/tomcat-servlet-api-9.0.4.jar" /usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/classes/Main.java && ${CATALINA_HOME}/bin/shutdown.sh && ${CATALINA_HOME}/bin/startup.sh
Using CATALINA_BASE:   /usr/local/apache-tomcat-9.0.85
Using CATALINA_HOME:   /usr/local/apache-tomcat-9.0.85
Using CATALINA_TMPDIR: /usr/local/apache-tomcat-9.0.85/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /usr/local/apache-tomcat-9.0.85/bin/bootstrap.jar:/usr/local/apache-tomcat-9.0.85/bin/tomcat-juli.jar
Using CATALINA_OPTS:
NOTE: Picked up JDK_JAVA_OPTIONS:  --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.util.concurrent=ALL-UNNAMED
Using CATALINA_BASE:   /usr/local/apache-tomcat-9.0.85
Using CATALINA_HOME:   /usr/local/apache-tomcat-9.0.85
Using CATALINA_TMPDIR: /usr/local/apache-tomcat-9.0.85/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /usr/local/apache-tomcat-9.0.85/bin/bootstrap.jar:/usr/local/apache-tomcat-9.0.85/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
dev@dev:~/usr/local/apache-tomcat-9.0.85/webapps/StudentDB/WEB-INF/classes$
```

Learning Outcome:

1. Understand the working the Server Client and Database communication in web application
2. Developed a simple Auth web application using Tomcat + JAVA + MySql database.

