To kill the occupied port type the following @ command prompt:

```
netstat -ano | findstr 8080 -to know the pid 1676

taskkill /F /pid 1676- to close the port.
```

```
> create database strao;
> use strao;
> create table login (uname varchar(20) primary key, password varchar(20));
> insert into login values(‘cscorner’, ‘unknown’);
> select * from login;

> show databases;
> create database userlogin;
> use userlogin;
> show tables;
> create table user(id int(11) unsigned auto-increment primary key not null,
          username varchar(25) not nul,
          password varchar(25) not null,
          email varchar(40) not null);
> show tables;
> describe user;
```

Java Database Connectivity

Download and Install MYSQL

> Open browser and type mysql download

> click on MySQL Downloads

> Click on MySQL Community (GPL) Downloads

> Click on MySQL Installer for Windows

> it opens **MySQL Installer 8.0.29**

> Click on Download button of (MySQL Community Downloads)

Windows (x86, 32-bit), MSI Installer	8.0.29	439.6M	**Download**
(mysql-installer-community-8.0.29.0.msi)			

> click on **No thanks, just start my download.**

> after mysql-installer-community-8.0.29.0.msi downloaded click on it to start installation.

> allow permission to install

> Now choose the setup type as custom and click next

> We need to install three products for development purposes, 1. Workbench, 2. Shell(Command prompt) and 3. Server

> Select MySQL Servers/MySQL Server 8.0/ MySQL Server 8.0.29-X64;
Applications/MySQL Workbench/ MySQL Workbench 8.0/ MySQL Workbench 8.0.29-X64;
Applications/MySQL Shell/ MySQL Shell 8.0/ MySQL Shell 8.0.29-X64; into Products To Be Installed and click Next

> Now click Execute to start installation, after the status shows Complete, click on Next to Product configuration

> again click Next

>click Next and shows Authentication Method

> click Next to password settings root and root and click Next

> Shows Windows Service and click Next to show Apply Configuration and click Execute and Finish Button

> Shows Product Configuration and click on Next button, shows Installation Complete and click on Finish Button

> starts MySQL shell and MySQL Workbench

> in workbench click on root and enter password to work with database (it is GUI of MySQl Server) or

> Close the Shell also

> In windows search type mysql and select “ MySQL 8.0 Command Line Client”

> enter root password to access database

> mysql> prompt will appear

> from now on database commands will work

```
mysql> create database aiml;
mysql>use aiml;
mysql> create table students(rollno int, name varchar(20));
mysql> desc students; describes the table properties.
```

Download MySQL Connector and Configure JDBC in Eclipse IDE

> Open browser and type dev mysql download

> click on MySQL Community Downloads

> click on Connector/J to download JAVA JDBC Connector

> opens Connector/J 8.0.27 or latest, in select operating system: Select Platform Independent option

> now select ZIP Archive and click on Download

> Click on **No thanks, just start my download.**

> save it on desktop or any location

> Extract the ZIP file and find the mysql-connector-java-8.0.27.jar file in the folder, this jar file is used to connect java programs to mysql database.

> open the Eclipse IDE and how to add this jar file into Eclipse project

> start Eclipse IDE and go to Java perspective and create java project

> go to File/New/Java project

> Enter project name as JDBCDemo and click on Finish button, asks module and don't create

> Now expand JDBCDemo project in left pane

> Now we can add the connector jar as follows

> right click on JRE System Library[JavaSE]/select Build/ click on Configure Build Path

> select Modulepath and click on Add External JARs and browse the jar file and select and click on Apply button and Apply and Close button

> Now you can see on the left side below JRE System Library we get Referenced Library now expand mysql connector jar to various packages and we use com.mysql.cj.jdbc for developing JDBC programs.

Example:

Create database: aiml (> create database aiml;)

Create table : login (> create table login (uname varchar(20) primary key, password varchar(20));)

>insert into login values(‘cscorner’, ‘unknown’);

> select * from login;

Java Project with JDBC

Example.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;

public class Example {

        public static void main(String[] args) throws ClassNotFoundException {

                try {
                        Class.forName("com.mysql.cj.jdbc.Driver");
                        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/strao","root","root");
                        Statement st=con.createStatement();
                        System.out.println("Inserting records");
                        String sql="Insert into login values('cse','123')";
                        st.executeUpdate(sql);
                } catch (SQLException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                }
        }

}
```

Now run the Java program and check the data in table as follows:

> select * from login;

>now we get two records in the login table.

Java Web Project (LoginDemo)

Login Page using JSP+Servlet+JDBC+MySQL Step by Step Tutorial

> Start Eclipse File/New/Dynamic Web Project -provide LoginDemo, click next and check Generate web.xml deployment descriptor and click on Finish.

> create JSPs page as follows:

Login.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Login</title>
</head>
<body>
<div align=center>
<h1>User Login</h1>
<form action=LoginServlet method=post>
<table>
<tr><td>Enter Name:</td><td><input type=text name=txtName></td></tr>
<tr><td>Enter Password:</td><td><input type=password
name=txtPwd></td></tr>
<tr><td><input type=submit value=Login></td><td><input type=reset
value=Reset></td></tr>
</table>
</form>
```

```
</div>
</body>
</html>
```

Welcome.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<h1> Login Successful!!</h1>
</body>
</html>
```

LoginServlet.java

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

import jakarta.servlet.RequestDispatcher;
import jakarta.servlet.ServletException;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
```

```
/**
 * Servlet implementation class LoginServlet
 */
public class LoginServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;


    protected void doPost(HttpServletRequest request,
HttpServletResponse response) throws ServletException, IOException {
    try {
        response.setContentType("text/html");
        PrintWriter out=response.getWriter();
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/strao","root
","root");
        String n=request.getParameter("txtName");
        String p=request.getParameter("txtPwd");
        PreparedStatement ps=con.prepareStatement("select uname
from login where uname=? and password=?");
        ps.setString(1, n);
        ps.setString(2, p);
        ResultSet rs=ps.executeQuery();
        if(rs.next())
        {
            RequestDispatcher rd=
request.getRequestDispatcher("Welcome.jsp");
            rd.forward(request, response);
        }
        else
        {
            out.println("<font color=red size=18> Login Failed!!<br>");
            out.println("<a href=Login.jsp> Try AGAIN!!</a>");
        }
    } catch (ClassNotFoundException e) {
```

```
                // TODO Auto-generated catch block
                e.printStackTrace();
        } catch (SQLException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
        }

        }

}
```

> now copy connector.jar and paste it into the server Lib folder(Apache/apache-tomcat/lib/)
> Now right click on the project and refresh to see the jar file.
> Now select Login.jsp and click on Run As/Run on Server
>login page will open and ask for userid and password; if userid and password matches with data in table then “Login Successfull!!” message will appear and if not match then it will Display “ Login Failed!!
Try AGAIN!!”.