Placement Management Portal using Full stack Web Development with Java

ABSTRACT

The **Placement Management Portal** is a comprehensive full-stack web application developed to streamline and enhance the campus placement process for students, recruiters, and college administrators. Built using **Java(JSP)** for the backend, **SQL for database management** and integrated with modern frontend technologies **HTML 5 and CSS 3**.

The project was developed as a group effort by –

Harsh Prasad (001CSE22GT017) , **Sunil Kumar (001CSE22GT024)** and **Dulari Kumari (001CSE22GT023)**, under the valuable mentorship of **Mr**. **Shashi Kumar Tanti Sir**.

The primary objective of the portal is to automate and manage various aspects of the placement process, including student registration, job postings, application tracking, eligibility verification, interview scheduling, and final placement status updates. The system ensures secure login for different user roles—students, recruiters, and administrators—each with tailored functionalities. The backend is powered by robust Java-based frameworks ensuring data consistency and performance, while the frontend delivers a responsive and user-friendly interface.

This project not only addresses the inefficiencies of traditional placement procedures but also provides an easily maintainable and scalable solution for educational institutions. It reflects the collaborative effort and practical implementation of full-stack web development skills, project management, and real-world problem-solving.

Acknowledgement

We would like to express our sincere gratitude to everyone who contributed to the successful completion of our project titled **"Placement Management Portal"** developed using full-stack web technologies including **Java (JSP),** and **SQL database**.

First and foremost, we are deeply thankful to our mentor, **Mr Shashi Kumar Tanti Sir**, for his invaluable guidance, continuous support, and encouragement throughout the development of this project. His technical expertise and constructive feedback played a crucial role in shaping our understanding and refining our implementation.

We also extend our appreciation to our institution and faculty members for providing the necessary resources and a conducive environment for learning and development.

This project was a collaborative effort, and we, **Arwin Jonathan Lakra**, **Ankit Kumar Soni**, and **Nitish Kumar Yadav**, are grateful to each other for the strong teamwork, dedication, and effort put in at every stage—from initial planning to final deployment.

Lastly, we thank our families and friends for their constant support and motivation, which inspired us to give our best.

- **Sunil Kumar**
  B.Tech(C.S.E)
  Semester – VI
  (001CSE22GT024)

Table of Contents

1. Introduction ………………………………………………………….. 4

2. Chapter 1 – Introduction to HTML …………………………………….. 5

2.1About HTML

2.2Some basic codes in HTML ………………………………………….

2.2Marquee Tag in Html

2.4Tables and Forms in Html

3 Chapter 2 – Introduction to JSP

3.1About JSP ………………………………………………………..

3.2First Program in JSP ………………………………….

3.3Linking html forms with jsp page ……………………….

4 Chapter 3 – SQL

4.1Installing SQL 10g…………………………………..

4.2Connecting to Database …………………………………..

4.3Basics queries of SQL …………………………………..

5 Chapter 4 – About Apache Tomcat Server

5.1Installation and Running of Apache Tomcat…………………………..

5.2Hosting our first webpage on local server 8080………………………

6 Chapter 5 – Developing our Group Project

6.1 Creating Frontend ……………………………………..

6.2 Creating Tables in Database ……………………………………..

6.3 Creating JSP pages for connectivity…….

6.4 Final output

7 References

Introduction

In today’s fast-paced academic and corporate environment, campus placements play a crucial role in shaping the future of students by providing them with career opportunities directly from their institutions. However, the traditional process of managing campus placements is often time-consuming, error-prone, and dependent on manual record-keeping. This not only results in inefficiencies but also adds a significant burden to placement officers and administrative staff.

To address these challenges, we have developed a **Placement Management Portal** — a web-based application designed to streamline and automate the entire placement process within an educational institution. This portal is a comprehensive solution that allows placement officers to manage job openings, recruiters to post opportunities, and students to apply for jobs—all through a centralized digital platform.

The portal is built using **Java JSP and Servlets** for the server-side logic and dynamic content generation, ensuring a robust and scalable backend. For data storage and retrieval, we utilized **SQL** to create and manage a structured database that handles user data, job listings, applications, and result records efficiently. The user interface is designed to be intuitive and accessible, ensuring a smooth user experience across different roles—student, recruiter, and administrator.

Key functionalities of the Placement Management Portal include:

- **Student Module**: Registration, login, profile management, resume upload, job browsing, and application tracking.
- **Recruiter Module**: Job posting, candidate shortlisting, and interview scheduling.
- **Admin Module**: Verification of student and recruiter accounts, job post approvals, analytics and report generation.

The Placement Management Portal not only improves the transparency and efficiency of the placement process but also provides a scalable foundation that can be enhanced with features like automated notifications, resume parsing, and integration with external recruitment platforms in future iterations.

Through this project, we aimed to bridge the gap between academic institutions and recruiters by leveraging technology to create a reliable, user-friendly, and efficient placement management system.

Chapter 1 – Introduction to HTML

1.1 About Html

HTML stands for **HyperText Markup Language**. It is the standard language used to create and structure content on the **World Wide Web**. HTML provides the basic building blocks for web pages, allowing developers to define elements like headings, paragraphs, links, images, tables, forms, and more.

What is HTML?

HTML is not a programming language; it is a **markup language**. This means it is used to "mark up" text and other content to tell a web browser how to display it. It consists of **elements** or **tags**, which are enclosed in angle brackets like `<tagname>...</tagname>`.

Basic Structure of a HTML Document

```
<!DOCTYPE html>
<html>
  <head>
    <title>My First Web Page</title>
  </head>
  <body>
    <h1>Welcome to HTML!</h1>
    <p>This is a simple paragraph.</p>
  </body>
</html>
```

Key Components:

- <!DOCTYPE html>: Declares the document type and version of HTML.
- <html>: The root element of an HTML page.
- <head>: Contains metadata, such as the page title, links to stylesheets, etc.
- <title>: Sets the title shown in the browser tab.
- <body>: Contains the content that is displayed on the webpage, such as text, images, and links.

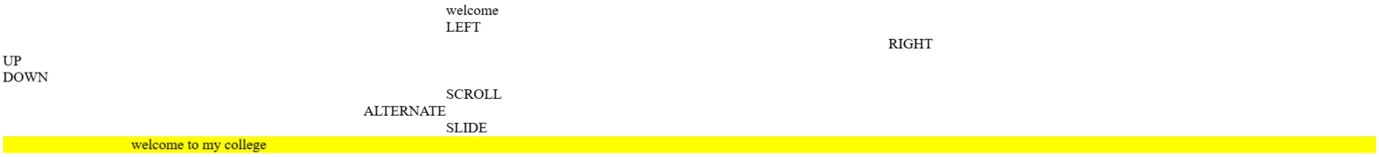
- <h1> to <h6>: Headings of different levels.
- <p>: Paragraph tag used for blocks of text.

2.2 Some basic codes in HTML

```
  <html>

<body>

Hello <br>

World <hr>

<p align="left">Raja</p>

<p align = "right">Harsh</p>

<p align = "center">Riya</p>

<h1>Ram</h1>

<h2>Shyam</h2>

<h3>Raghav</h3>

<h4>Ram</h4>

<h5>Shyam</h5>

<h6>Raghav</h6>

<font face="chiller" size="+7" color="red">Welcome</font>

<img src="a.jpg" border = 1 height=300 width=300 alt="flower">

<a href="a.html">click</a>

<a href="a.html" target ="_self">click</a>

<a href="a.html" target ="_blank">click</a>

<a href="http:www.google.com"><img src="ab.jpg" height=100 width = 100 ></a>

<table border=1>

<tr><th>Roll</th><th>Name</th></tr>

<tr><td>1</td><td>Mangru</td></tr>

</table>

</body>

</html>
```

2.3 Marquee Tag in HTML

```
<html>

<body>

<marquee>welcome</marquee>

<marquee direction="left">LEFT</marquee>

<marquee direction="right">RIGHT</marquee>

<marquee direction="up">UP</marquee>

<marquee direction="down">DOWN</marquee>

<marquee behavior="scroll">SCROLL</marquee>

<marquee behavior="alternate">ALTERNATE</marquee>

<marquee behavior="slide">SLIDE</marquee>

<marquee bgcolor="yellow" direction=right scrolldelay="300">welcome to my college</marquee>

</body>

</html>
```



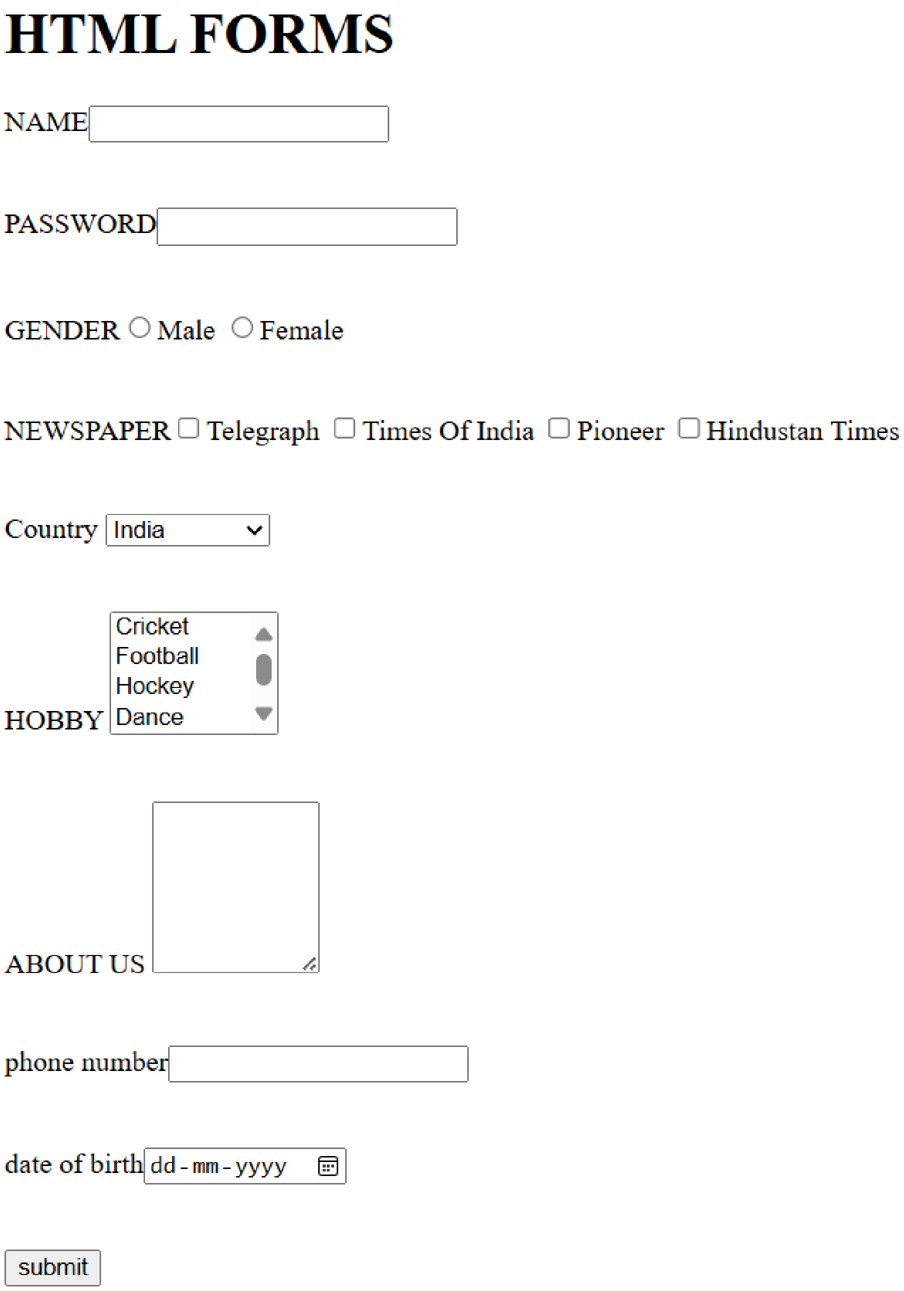
```
                                  welcome
                                  LEFT
                                                                RIGHT
UP
DOWN
                                  SCROLL
                          ALTERNATE
                                  SLIDE
          welcome to my college
```

2.4 Table and Forms in Html

Forms :

```
<html>

<body>

<h1>HTML FORMS</h1>

<form name="frm">

NAME<input type="text" name="nm" size="20"> <br><br><br>

PASSWORD<input type="password" name="pwd" size="20"> <br><br><br>

GENDER<input type="radio" name="rd" value="male">Male

<input type="radio" name="rd" value="female">Female

<br><br><br>

NEWSPAPER<input type="checkbox" name="c1" value="Telegraph">Telegraph

<input type="checkbox" name="c2" value="Times Of India">Times Of India

<input type="checkbox" name="c3" value="Pioneer">Pioneer

<input type="checkbox" name="c4" value="Hindustan Times">Hindustan Times

<br><br><br>

Country

<select name="con">

<option>India</option>

<option>Japan</option>

<option>Nepal</option>

<option>Bhutan</option>
```

```html
<option>Bangladesh</option>

</select>

<br>

<br><br>

HOBBY

<select multiple=true>

<option>Cricket</option>

<option>Football</option>

<option>Hockey</option>

<option>Dance</option>

<option>Table Tennis</option>

</select>

<br>

<br><br>

ABOUT US

<textarea rows=6 cols=10>

</textarea>

<br><br><br>

phone number<input type="number" name="ph" size=20><br><br><br>

date of birth<input type="date" name="dt" ><br><br><br>

<input type="submit" value="submit">

</form>

</body>

</html>
```



HTML FORMS

NAME

PASSWORD

GENDER ○ Male ○ Female

NEWSPAPER □ Telegraph □ Times Of India □ Pioneer □ Hindustan Times

Country India

HOBBY Cricket
Football
Hockey
Dance

ABOUT US

phone number

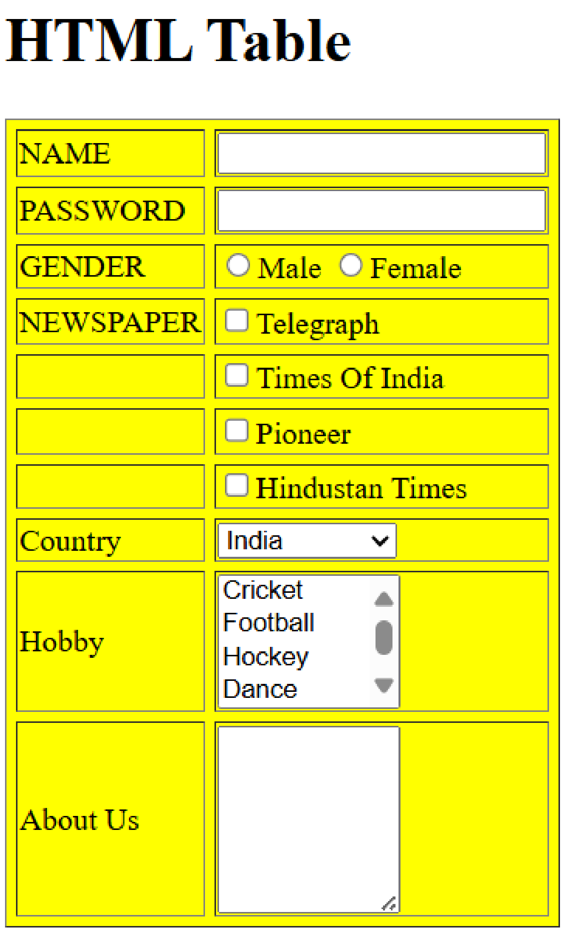
date of birth dd-mm-yyyy

submit

Tables

```
<html>

<body>

<h1>HTML Table</h1>

<form name="frm">

<table border=1 bgcolor=yellow cellspacing=5>

<tr><td>NAME</td><td>

<input type="text" name="nm" size="20"> </td></tr>

<tr><td>PASSWORD</td><td><input type="password" name="pwd" size="20"></td></tr>

<tr><td>GENDER</td><td><input type="radio" name="rd" value="male">Male

<input type="radio" name="rd" value="female">Female</td></tr>

<tr><td>NEWSPAPER</td><td><input type="checkbox" name="c1" value="Telegraph">Telegraph</td></tr>

<tr><td></td><td>

<input type="checkbox" name="c2" value="Times Of India">Times Of India</td></tr>

<tr><td></td><td><input type="checkbox" name="c3" value="Pioneer">Pioneer</td></tr>

<tr><td></td><td>

<input type="checkbox" name="c4" value="Hindustan Times">Hindustan Times</td></tr>

<tr><td>Country</td><td><select name="con">

<option>India</option>

<option>Japan</option>

<option>Nepal</option>

<option>Bhutan</option>

<option>Bangladesh</option>
```

```
</select></td></tr>

<tr><td>Hobby</td><td><select multiple=true>

<option>Cricket</option>

<option>Football</option>

<option>Hockey</option>

<option>Dance</option>

<option>Table Tennis</option>

</select></td></tr>

<tr><td>About Us</td><td><textarea rows=6 cols=10>

</textarea></td></tr>

</table>

</form>
```



HTML Table

NAME	
PASSWORD	
GENDER	○ Male ○ Female
NEWSPAPER	□ Telegraph
	□ Times Of India
	□ Pioneer
	□ Hindustan Times
Country	India
Hobby	Cricket Football Hockey Dance
About Us	

&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp

```
&nbsp&nbsp&nbsp&nbsp&nbsp

</body>

</html>
```

3. Chapter 2 – Introduction to JSP

3.1 About JSP

JSP (JavaServer Pages) is a **server-side technology** used for creating **dynamic web content** using Java. Developed by Sun Microsystems (now part of Oracle), JSP is built on top of the Java Servlet technology and allows embedding Java code directly into HTML pages using special JSP tags.

What is JSP?

JSP files are text-based documents saved with a `.jsp` extension. They contain a mix of **HTML**, **Java code**, and **JSP-specific tags**, which are executed on the **server side** before the page is sent to the user's browser.

When a client requests a JSP page:

1. The web server passes the request to a **JSP engine**.
2. The engine converts the JSP into a **Servlet** (a Java class).
3. The Servlet is compiled and executed on the server.
4. The output (usually HTML) is sent to the browser.

Why Use JSP?

- **Separation of concerns**: JSP separates presentation (HTML) from business logic (Java).
- **Reusability**: JavaBeans and custom tags can be reused across pages.
- **Integration**: Easily integrates with Java-based technologies and databases.
- **Familiar Syntax**: Developers familiar with HTML and Java can learn JSP quickly.

Basic Syntax Example:

```
<%@ page language="java" contentType="text/html" %>

<html>

  <head><title>JSP Example</title></head>

  <body>

    <h1>Welcome to JSP!</h1>

    <%

      String name = "Arwin";

      out.println("Hello, " + name + "!");

    %>

  </body>

</html>
```

Common JSP Tags:

- `<% %>`: Scriptlet — contains Java code.
- `<%= %>`: Expression — outputs a value.
- `<%@ %>`: Directive — provides instructions (like importing packages).
- `<jsp:include>`: Includes content from another file.
- `<jsp:useBean>`: Uses a JavaBean in the page.

3.2 First Program in JSP

```
<%@ page language="java" contentType="text/html; charset=UTF-8" %>

<!DOCTYPE html>

<html>

<head>

    <title>Hello World in JSP</title>
```

```
</head>
<body>
    <h1>
        <%-- JSP scriptlet to print Hello World --%>
        <%
            out.println("Hello, World!");
        %>
    </h1>
</body>
</html>
```

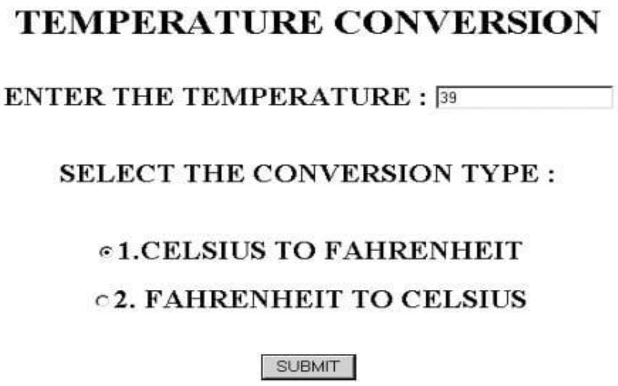
3.2 Linking Html forms with JSP

Temperature .html

```
HTML>
 <BODY>
     <CENTER> <h1> TEMPERATURE CONVERSION   </h1> <BR>
    <FORM METHOD="POST" ACTION="http://localhost:80/temp.jsp">
    <h2>ENTER THE TEMPERATURE :
          <INPUT TYPE="TEXT" NAME="temp">
          <br>
           SELECT THE CONVERSION TYPE :  <br>
           <INPUT TYPE="RADIO" NAME ="conv" VALUE="1">
                  1.CELSIUS TO FAHRENHEIT
           <INPUT TYPE="RADIO" NAME="conv"  VALUE="2">
                  2. FAHRENHEIT TO CELSIUS
     </h2> <br> <INPUT TYPE="SUBMIT" VALUE="SUBMIT">
   </FORM>
 </BODY>
</HTML>
```

Temperature.jsp

```
HTML>
<BODY>
     <CENTER><h1> TEMPERATURE CONVERSION   </h1> <br> <br>
        <%   double Tc,Tf;
                 double t=Double.parseDouble(request.getParameter("temp"));
                  int type=Integer.parseInt(request.getParameter("conv"));
                        if(type==2)
                              {
                                    Tc=0.55*(t-32);
         %>
                       <P> <h2> <%=t%> Fahrenheit = <%=Tc%>' Celcius </h2>
        <%   }
                        else
                             {
                                   Tf=(1.8*t)+32;
          %>
                        <P> <h2> <%=t%>' Celcius = <%=Tf%> Fahrenheit </h2>
         <%   }
          %>
 </BODY>
</HTML>
```



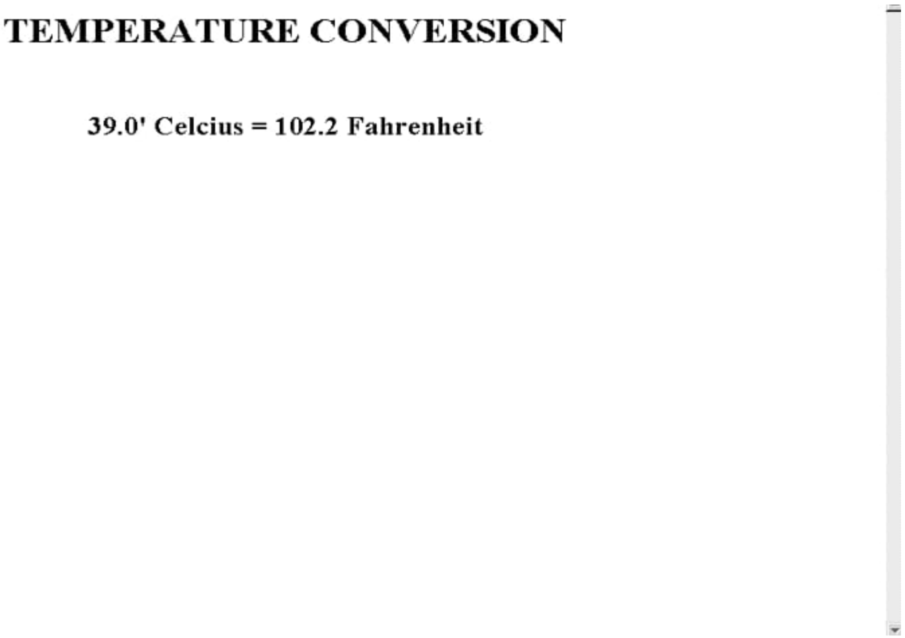
TEMPERATURE CONVERSION

ENTER THE TEMPERATURE : 39

SELECT THE CONVERSION TYPE :

● 1.CELSIUS TO FAHRENHEIT

○ 2. FAHRENHEIT TO CELSIUS



TEMPERATURE CONVERSION

39.0' Celcius = 102.2 Fahrenheit

4. Chapter 3 – SQL

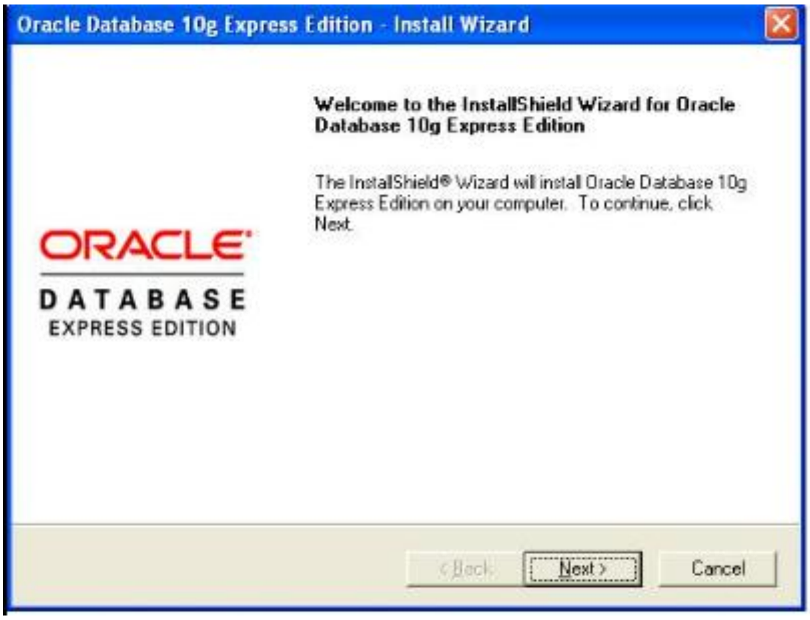
4.1 Installing SQL 10g XE

Installation Steps

1. **Extract the .zip/.rar** file if downloaded.
2. Run `setup.exe` as **Administrator**.
3. Choose **Basic Installation** or **Advanced Installation**.
4. Set:

- **Oracle Home**: Installation directory
    - **Global Database Name**: e.g., `orcl`
    - **Password**: Remember this for SQL login

5. Follow prompts and let the installation complete.



Oracle Database 10g Express Edition - Install Wizard

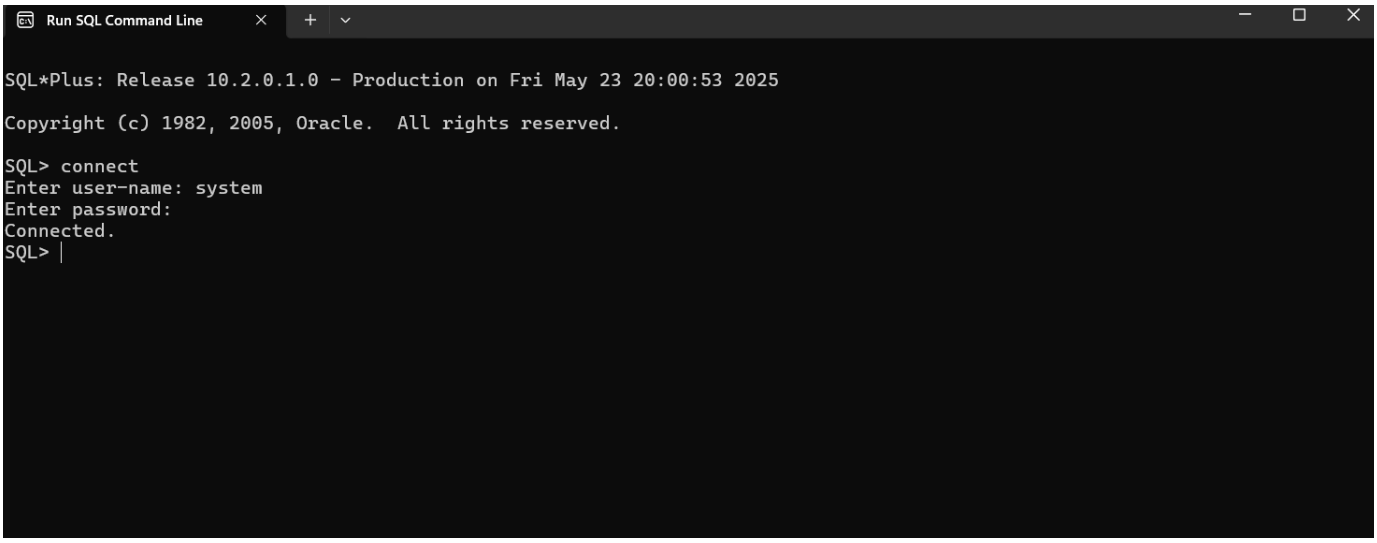
Welcome to the InstallShield Wizard for Oracle Database 10g Express Edition

The InstallShield® Wizard will install Oracle Database 10g Express Edition on your computer. To continue, click Next.

4.2 Connecting to Database :

After Installation

- Launch **SQL*Plus** from Start Menu or Desktop
- Login using: Username: system
- Password: [your password]



```
SQL*Plus: Release 10.2.0.1.0 - Production on Fri May 23 20:00:53 2025

Copyright (c) 1982, 2005, Oracle.  All rights reserved.

SQL> connect
Enter user-name: system
Enter password:
Connected.
SQL> |
```

4.3 Basic Queries of SQL

Creating a table

```
CREATE TABLE Students (
   ID INT PRIMARY KEY,
   Name VARCHAR(50),
   Age INT,
   Department VARCHAR(50)
);
```

Inserting into table :

```
INSERT INTO Students (ID, Name, Age, Department)
VALUES (1, 'ABC', 21, 'CSE');
```

Selecting all values:

```
SELECT * FROM Students;
```

Use where Clause:

```
SELECT * FROM Students
WHERE Department = 'CSE';
```

Update Data :

```
UPDATE Students
SET Age = 22
WHERE ID = 1;
```

Delete a Record:

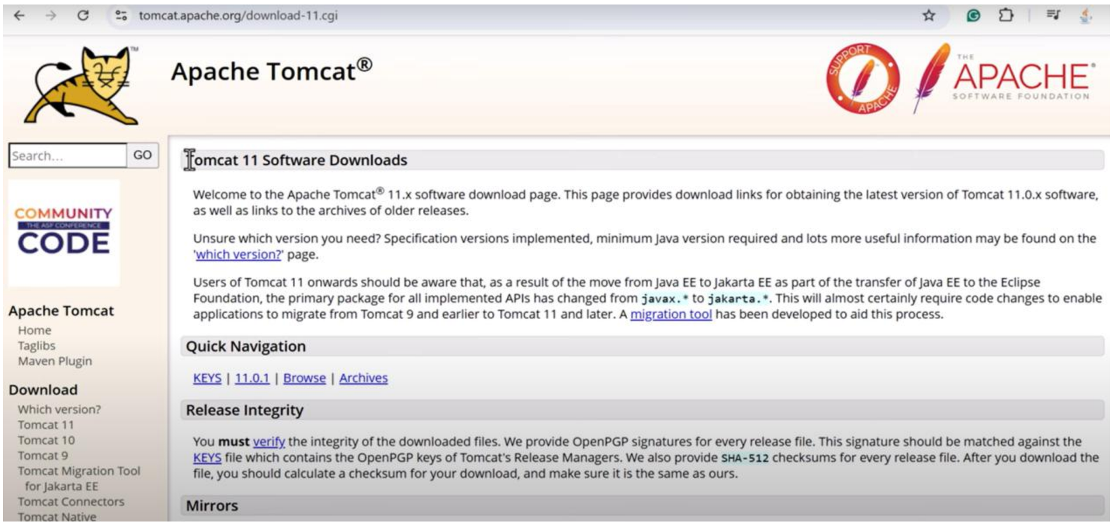
```
DELETE FROM Students
WHERE ID = 1;
```

5. Chapter 4 - About Apache Tomcat Server

Apache Tomcat is an open-source web server and servlet container developed by the Apache Software Foundation. It primarily supports Java-based web applications by executing Java Servlets and rendering JSP (JavaServer Pages). Tomcat is lightweight, efficient, and widely used for hosting dynamic websites and applications, making it a popular choice for Java developers.

5.1 Installation and Running of Apache Tomcat

Step 1 : Go to the official Apache Tomcat website and download the latest stable version of Tomcat for your operating system.



tomcat.apache.org/download-11.cgi

Apache Tomcat®

Tomcat 11 Software Downloads

Welcome to the Apache Tomcat® 11.x software download page. This page provides download links for obtaining the latest version of Tomcat 11.0.x software, as well as links to the archives of older releases.

Unsure which version you need? Specification versions implemented, minimum Java version required and lots more useful information may be found on the 'which version?' page.

Users of Tomcat 11 onwards should be aware that, as a result of the move from Java EE to Jakarta EE as part of the transfer of Java EE to the Eclipse Foundation, the primary package for all implemented APIs has changed from `javax.*` to `jakarta.*`. This will almost certainly require code changes to enable applications to migrate from Tomcat 9 and earlier to Tomcat 11 and later. A migration tool has been developed to aid this process.

Quick Navigation

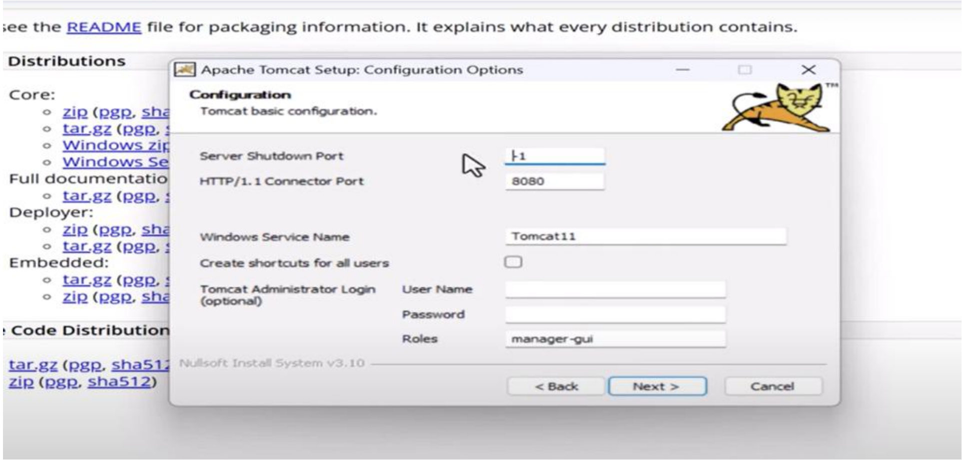
KEYS | 11.0.1 | Browse | Archives

Release Integrity

You **must** verify the integrity of the downloaded files. We provide OpenPGP signatures for every release file. This signature should be matched against the KEYS file which contains the OpenPGP keys of Tomcat's Release Managers. We also provide `SHA-512` checksums for every release file. After you download the file, you should calculate a checksum for your download, and make sure it is the same as ours.

Mirrors

Step 2 : Unzip the file and click on install and set username and password



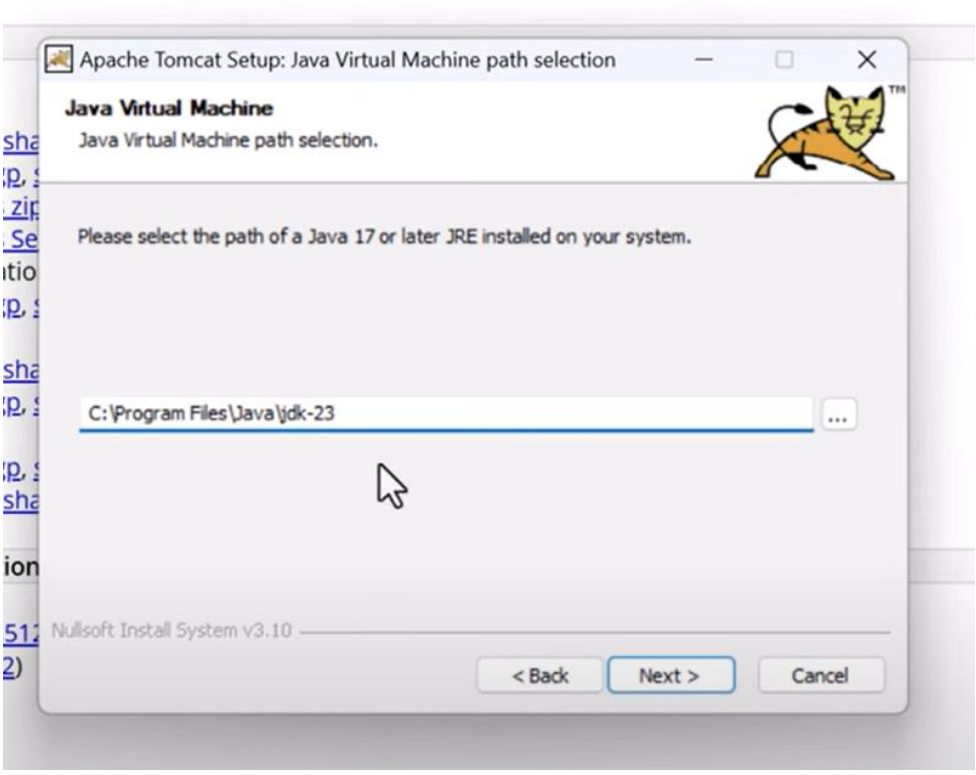
see the README file for packaging information. It explains what every distribution contains.

Apache Tomcat Setup: Configuration Options

Configuration
Tomcat basic configuration.

Server Shutdown Port: -1
HTTP/1.1 Connector Port: 8080
Windows Service Name: Tomcat11
Create shortcuts for all users: □
Tomcat Administrator Login (optional): User Name: Password: Roles: manager-gui

Step 3: Select jdk



Apache Tomcat Setup: Java Virtual Machine path selection

Java Virtual Machine

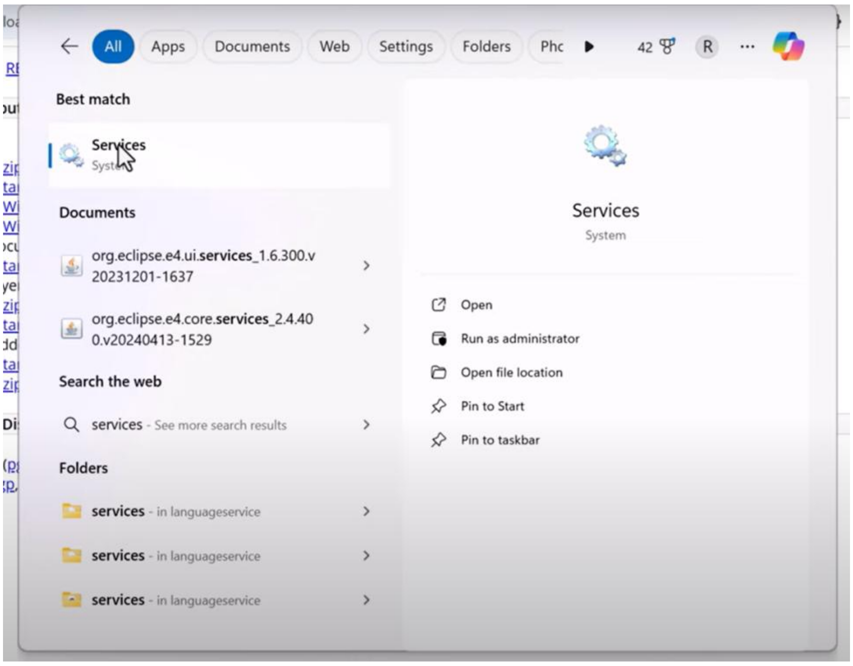
Java Virtual Machine path selection.

Please select the path of a Java 17 or later JRE installed on your system.

C:\Program Files\Java\jdk-23

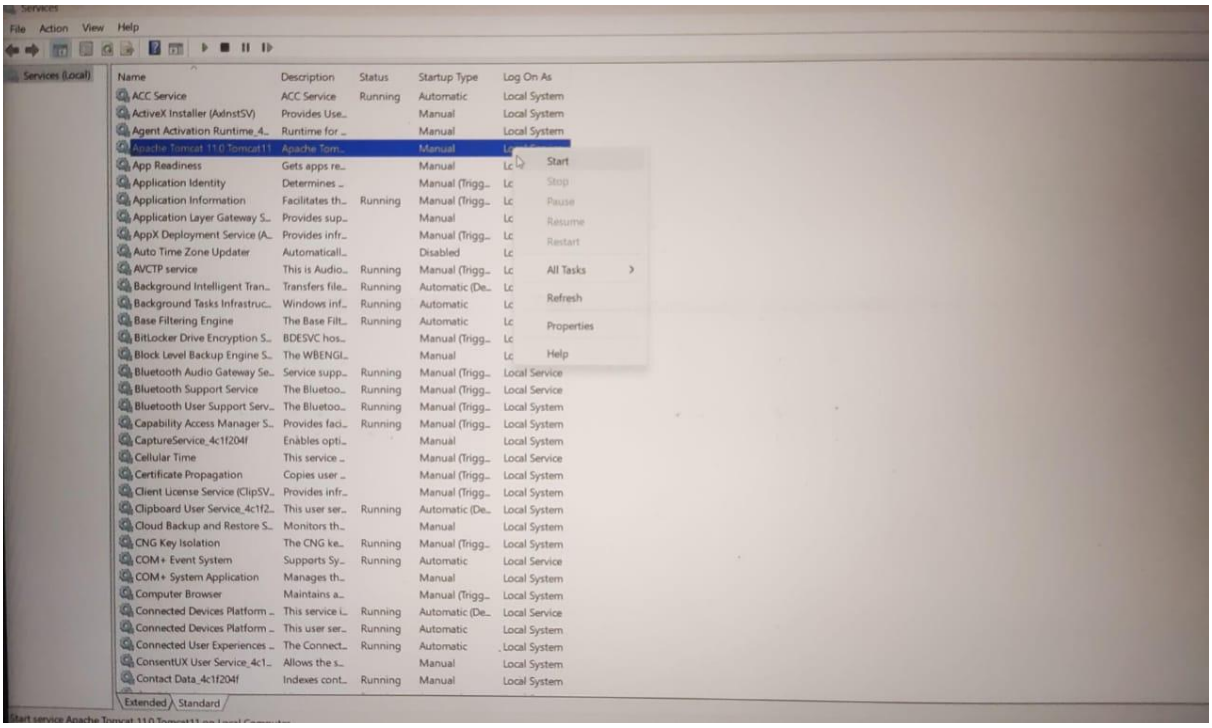
Running Apache Tomcat Server

Step 1 : Click on services



All
Apps
Documents
Web
Settings
Folders
Phc
42
R
...
Best match
Services
System
Documents
org.eclipse.e4.ui.services_1.6.300.v20231201-1637
org.eclipse.e4.core.services_2.4.40 0.v20240413-1529
Search the web
services - See more search results
Folders
services - in languageservice
services - in languageservice
services - in languageservice
Services
System
Open
Run as administrator
Open file location
Pin to Start
Pin to taskbar

Step 2 : Right click and click on "start" server

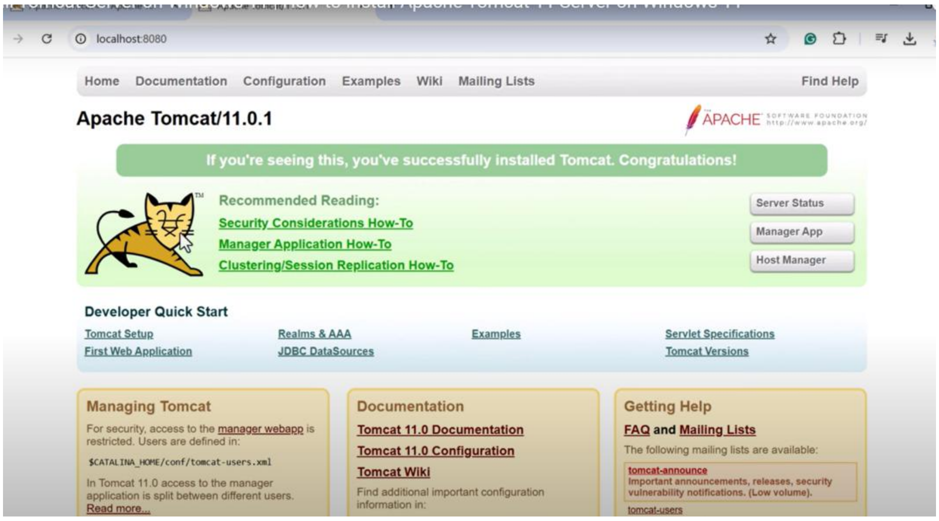


Name	Description	Status	Startup Type	Log On As
ACC Service	ACC Service	Running	Automatic	Local System
ActiveX Installer (AxInstSV)	Provides Use...		Manual	Local System
Agent Activation Runtime_4...	Runtime for ...		Manual	Local System
Apache Tomcat 11.0 Tomcat11	Apache Tom...		Manual	Lo
App Readiness	Gets apps re...		Manual	Lc
Application Identity	Determines ...		Manual (Trigg...	Lc
Application Information	Facilitates th...	Running	Manual (Trigg...	Lc
Application Layer Gateway S...	Provides sup...		Manual	Lc
AppX Deployment Service (A...	Provides infr...		Manual (Trigg...	Lc
Auto Time Zone Updater	Automaticall...		Disabled	Lc
AVCTP service	This is Audio...	Running	Manual (Trigg...	Lc
Background Intelligent Tran...	Transfers file...	Running	Automatic (De...	Lc
Background Tasks Infrastruc...	Windows inf...	Running	Automatic	Lc
Base Filtering Engine	The Base Filt...	Running	Automatic	Lc
BitLocker Drive Encryption S...	BDESVC hos...		Manual (Trigg...	Lc
Block Level Backup Engine S...	The WBENGI...		Manual	Lc
Bluetooth Audio Gateway Se...	Service supp...	Running	Manual (Trigg...	Local Service
Bluetooth Support Service	The Bluetoo...	Running	Manual (Trigg...	Local Service
Bluetooth User Support Serv...	The Bluetoo...	Running	Manual (Trigg...	Local System
Capability Access Manager S...	Provides faci...	Running	Manual (Trigg...	Local System
CaptureService_4c1f204f	Enables opti...		Manual	Local System
Cellular Time	This service ...		Manual (Trigg...	Local Service
Certificate Propagation	Copies user ...		Manual (Trigg...	Local System
Client License Service (ClipSV...	Provides infr...		Manual (Trigg...	Local System
Clipboard User Service_4c1f2...	This user ser...	Running	Automatic (De...	Local System
Cloud Backup and Restore S...	Monitors th...		Manual	Local System
CNG Key Isolation	The CNG ke...	Running	Manual (Trigg...	Local System
COM+ Event System	Supports Sy...	Running	Automatic	Local Service
COM+ System Application	Manages th...		Manual	Local System
Computer Browser	Maintains a...		Manual (Trigg...	Local System
Connected Devices Platform ...	This service i...	Running	Automatic (De...	Local Service
Connected Devices Platform ...	This user ser...	Running	Automatic	Local System
Connected User Experiences ...	The Connect...	Running	Automatic	Local System
ConsentUX User Service_4c1...	Allows the s...		Manual	Local System
Contact Data_4c1f204f	Indexes cont...	Running	Manual	Local System

Start service Apache Tomcat 11.0 Tomcat11 on Local Computer

Verifying if Apache Tomcat Server is running or not

- Go to google and type "localhost:8080"
- If you see welcome page then you have successfully run the Tomcat Server



Apache Tomcat/11.0.1

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:

Security Considerations How-To

Manager Application How-To

Clustering/Session Replication How-To

Developer Quick Start

Tomcat Setup
First Web Application
Realms & AAA
JDBC DataSources
Examples
Servlet Specifications
Tomcat Versions

Managing Tomcat

For security, access to the manager webapp is restricted. Users are defined in:

```
$CATALINA_HOME/conf/tomcat-users.xml
```

In Tomcat 11.0 access to the manager application is split between different users. Read more...

Documentation

Tomcat 11.0 Documentation

Tomcat 11.0 Configuration

Tomcat Wiki

Find additional important configuration information in:

Getting Help

FAQ and Mailing Lists

The following mailing lists are available:

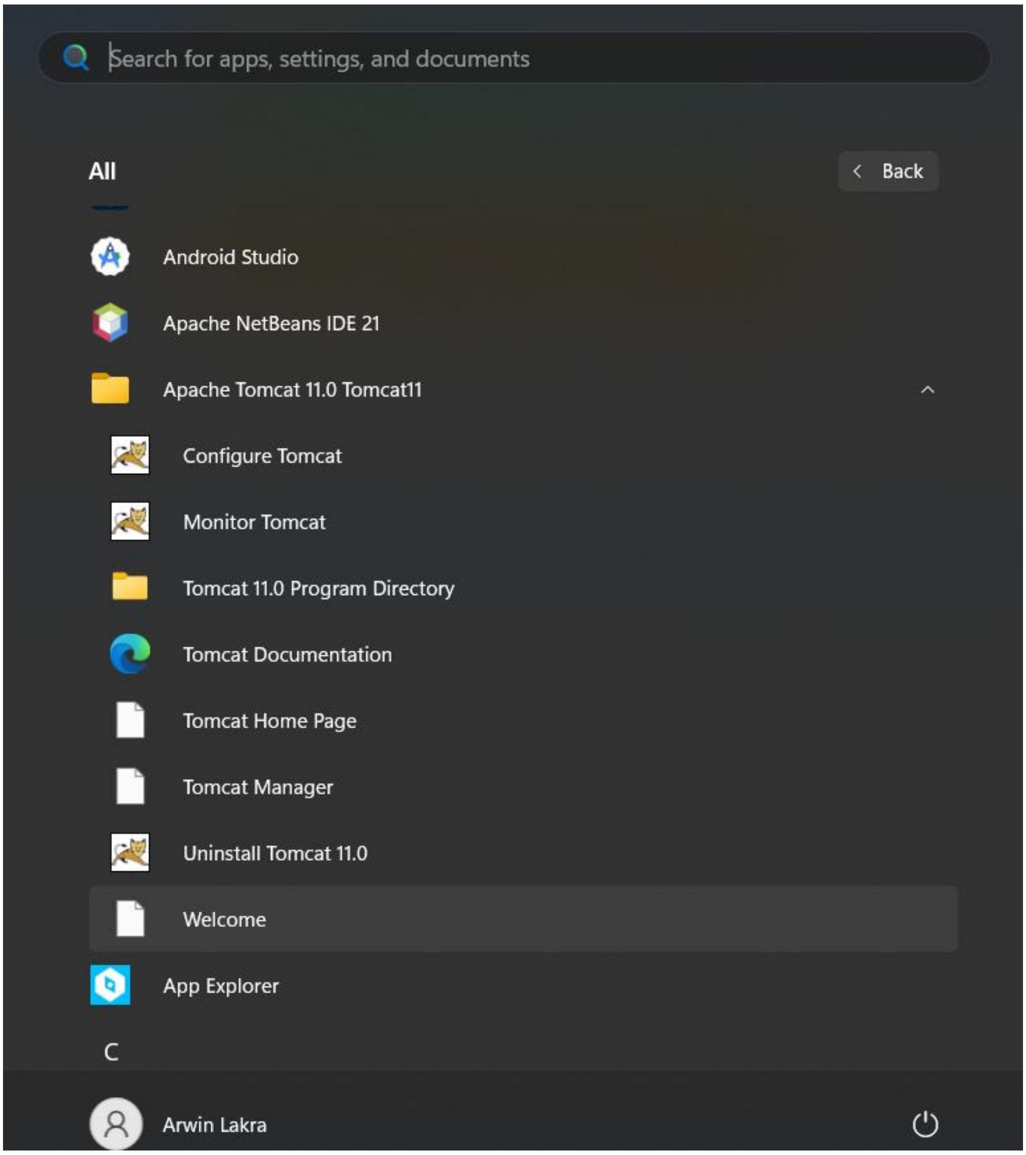
tomcat-announce
Important announcements, releases, security vulnerability notifications. (Low volume).

tomcat-users

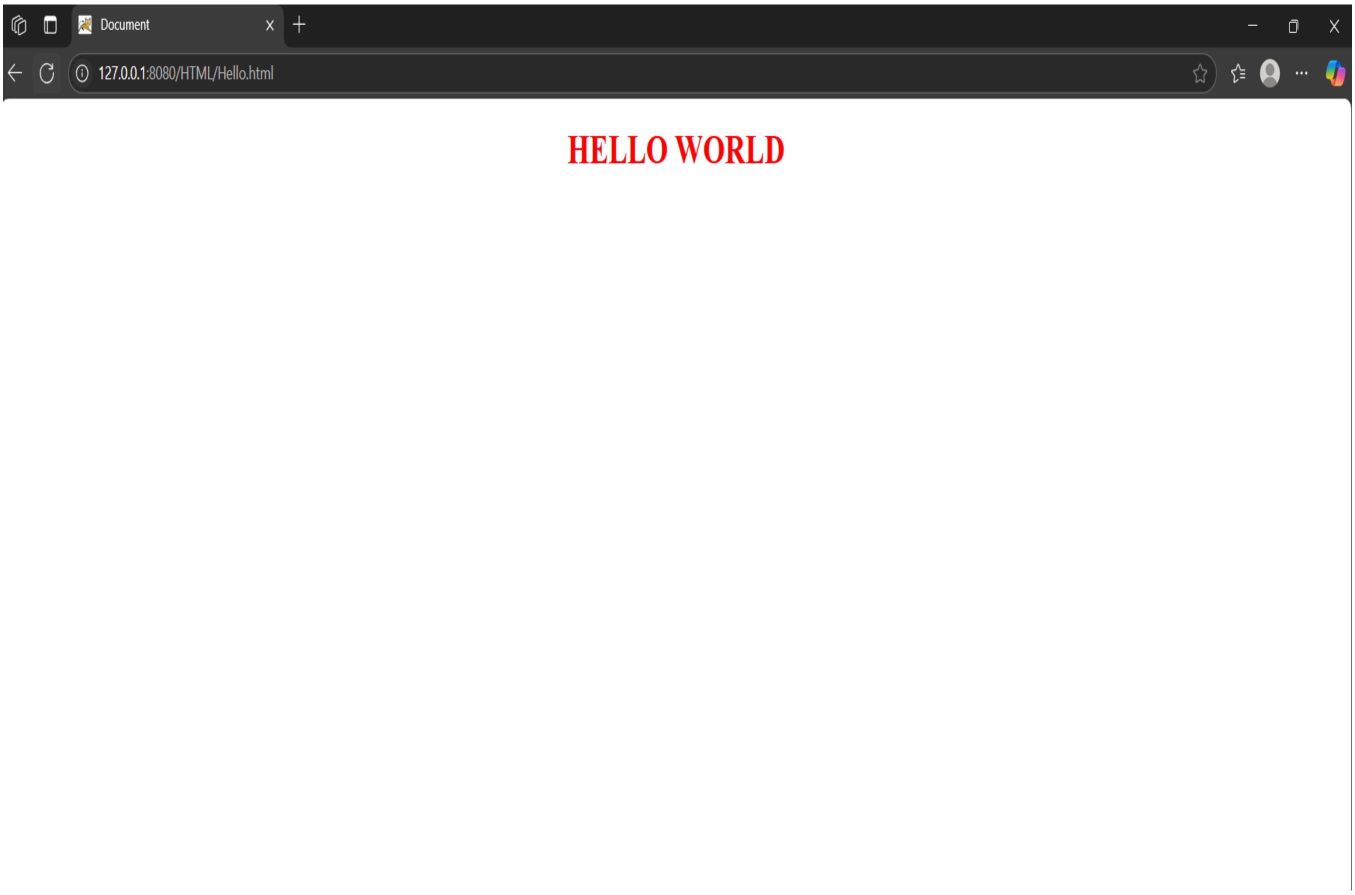
5.2 Hosting our first webpage on Tomcat server

Step 1 : Start the Tomcat server

Step 2 : Go to welcome page of Tomcat server through the Apache tomcat folder



Step 3: Write the webpage folder path in the server’s search bar :



HELLO WORLD

6 . Chapter 5 – Developing our Group Project

6.1 Creating Frontend

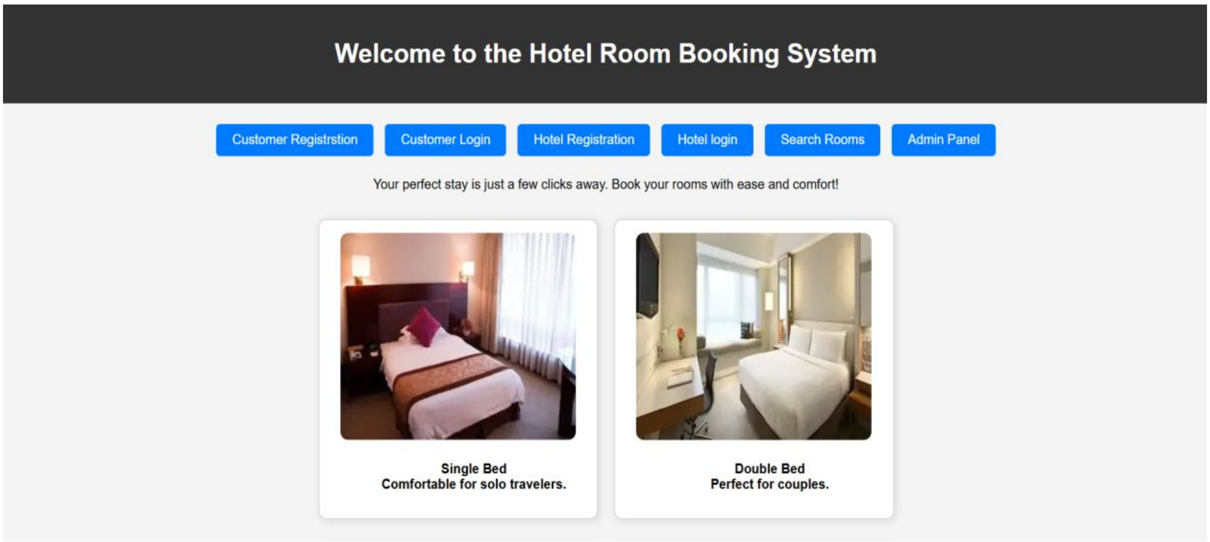
Index.html

```
<html>
<head>
  <title>Hotel Room Booking System</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      text-align: center;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
    }
    header {
      background-color: #333;
      color: #fff;
      padding: 20px;
    }
    nav {
      margin: 20px 0;
    }
    nav a {
      text-decoration: none;
      color: #fff;
      background-color: #007BFF;
      padding: 10px 20px;
      margin: 5px;
      border-radius: 5px;
      display: inline-block;
    }
    nav a:hover {
      background-color: #0056b3;
    }
    .program {
      display: flex;
      flex-wrap: wrap;
      justify-content: center;
      gap: 20px;
      margin: 2em auto;
      max-width: 1200px;
    }
    .img-para {
      display: flex;
      flex-direction: column;
      align-items: center;
      background: white;
      border: 2px solid #ddd;
      border-radius: 10px;
      padding: 15px;
      width: 320px;
      text-align: center;
      box-shadow: 2px 2px 10px rgba(0, 0, 0, 0.1);
```

```
        }
        .img img {
            width: 100%;
            height: auto;
            border-radius: 10px;
        }
        .para {
            margin-top: 10px;
        }
        .para li {
            list-style: none;
            font-weight: bold;
        }
    </style>
</head>
<body>
    <header>
        <h1>Welcome to the Hotel Room Booking System</h1>
    </header>
    <nav>
        <a href="custregistration.html">Customer Registrstion</a>
        <a href="custlogin.html">Customer Login</a>
        <a href="hotelregist.html">Hotel Registration</a>
        <a href="hotellogin.html">Hotel login</a>
        <a href="searchhotel.html">Search Rooms</a>
        <a href="adminlogin.html">Admin Panel</a>
    </nav>
    <p>Your perfect stay is just a few clicks away. Book your rooms with ease
and comfort!</p>

    <div class="program">
        <div class="img-para">
            <div class="img">
                <img src="Single.jpg" alt="Single Bed">
            </div>
            <div class="para">
                <ul>
                    <li>Single Bed</li>
                    <li>Comfortable for solo travelers.</li>
                </ul>
            </div>
        </div>
        <div class="img-para">
            <div class="img">
                <img src="Double.jpg" alt="Double Bed">
            </div>
            <div class="para">
                <ul>
                    <li>Double Bed</li>
                    <li>Perfect for couples.</li>
                </ul>
            </div>
        </div>
    </div>
```

```
<div class="program">
    <div class="img-para">
        <div class="img">
            <img src="Triple.jpg" alt="Triple Bed">
        </div>
        <div class="para">
            <ul>
                <li>Triple Bed</li>
                <li>Ideal for small families.</li>
            </ul>
        </div>
    </div>
    <div class="img-para">
        <div class="img">
            <img src="Queen.jpg" alt="Queen Bed">
        </div>
        <div class="para">
            <ul>
                <li>Queen Bed</li>
                <li>Spacious for comfort.</li>
            </ul>
        </div>
    </div>
</div>

<div class="program">
    <div class="img-para">
```

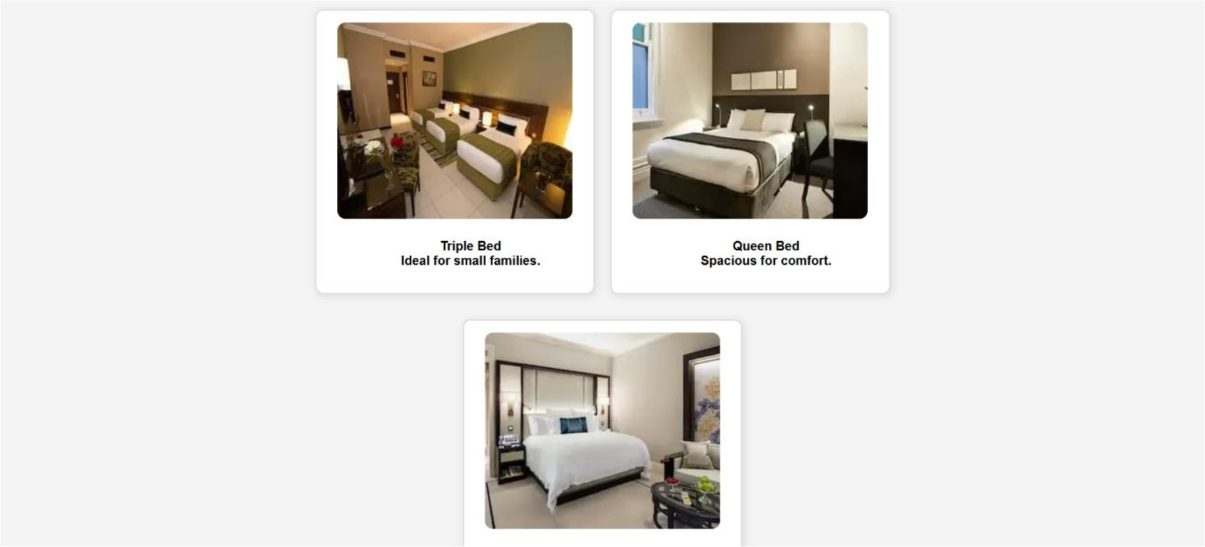
```
            <div class="img">
                <img src="King.jpg" alt="King Bed">
            </div>
            <div class="para">
                <ul>
                    <li>King Bed</li>
                    <li>Ultimate luxury and space.</li>
                </ul>
            </div>
        </div>
    </div>
</body>
</html>
```


Welcome to the Hotel Room Booking System

Your perfect stay is just a few clicks away. Book your rooms with ease and comfort!

Single Bed
Comfortable for solo travelers.

Double Bed
Perfect for couples.



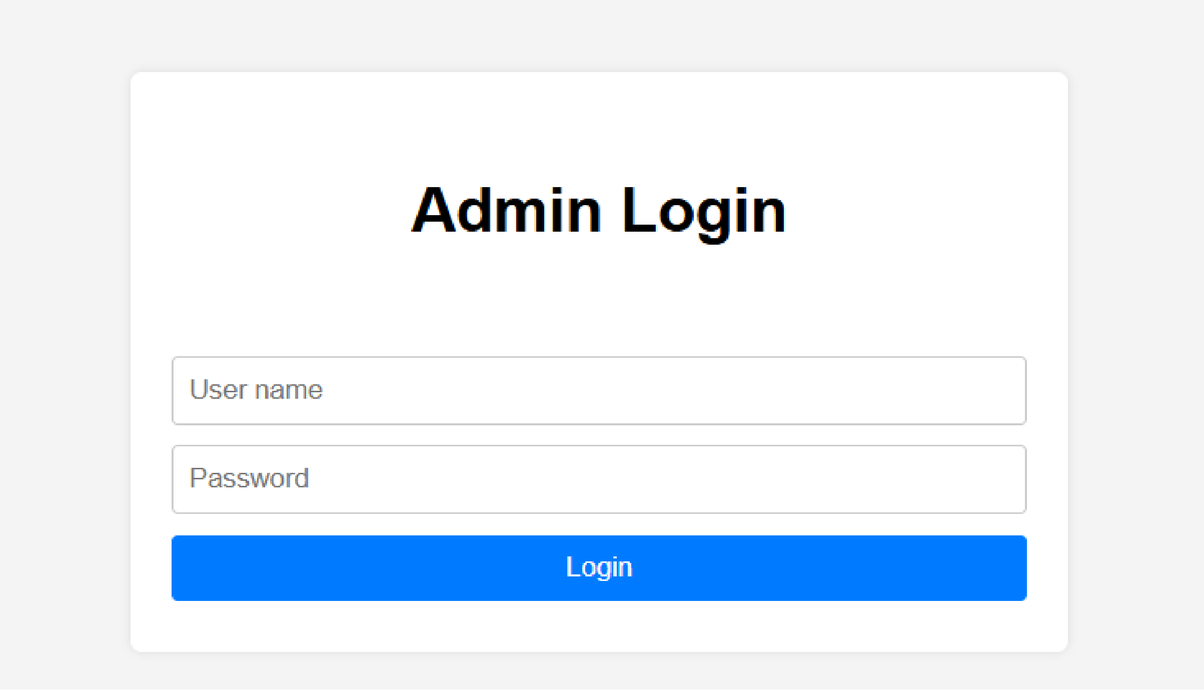
Triple Bed
Ideal for small families.

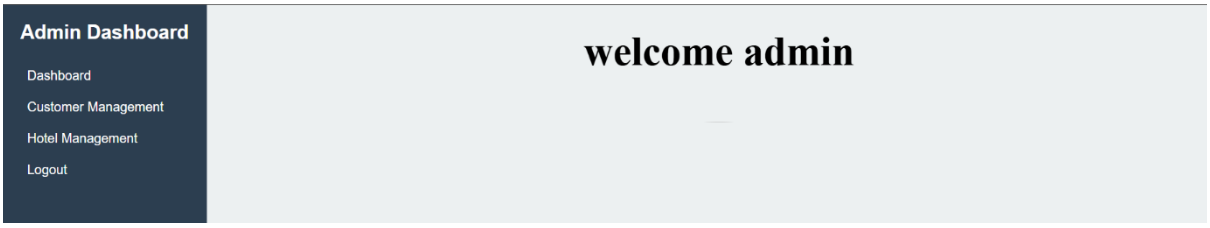
Queen Bed
Spacious for comfort.

Admin.html

```
<html><body> <form   action="adminlogin.jsp"  method="post" >
    <div class="form-container">
        <center><h3>Admin Login</h3></center><br>
        <form>
            <input type="text" placeholder="User name" required name="nm1">
            <input type="password" placeholder="Password" required name="nm2">
            <button type="submit">Login</button>
        </form>
    </div>


</form>
<style>
    body {
        font-family: Arial, sans-serif;
        display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    background: #f4f4f4;
}
.form-container h3{
    font-size: 30px;
}
.form-container {
    padding: 20px;
    background: white;
    border-radius: 5px;
    box-shadow: 0 0 5px rgba(0, 0, 0, 0.1);
}
input, button {
    width: 100%;
    padding: 8px;
    margin: 5px 0;
    border: 1px solid #ccc;
    border-radius: 3px;
}
button {
    background: #007bff;
    color: white;
    border: none;
    cursor: pointer;
}
```

```
</style></body>
</html>
```





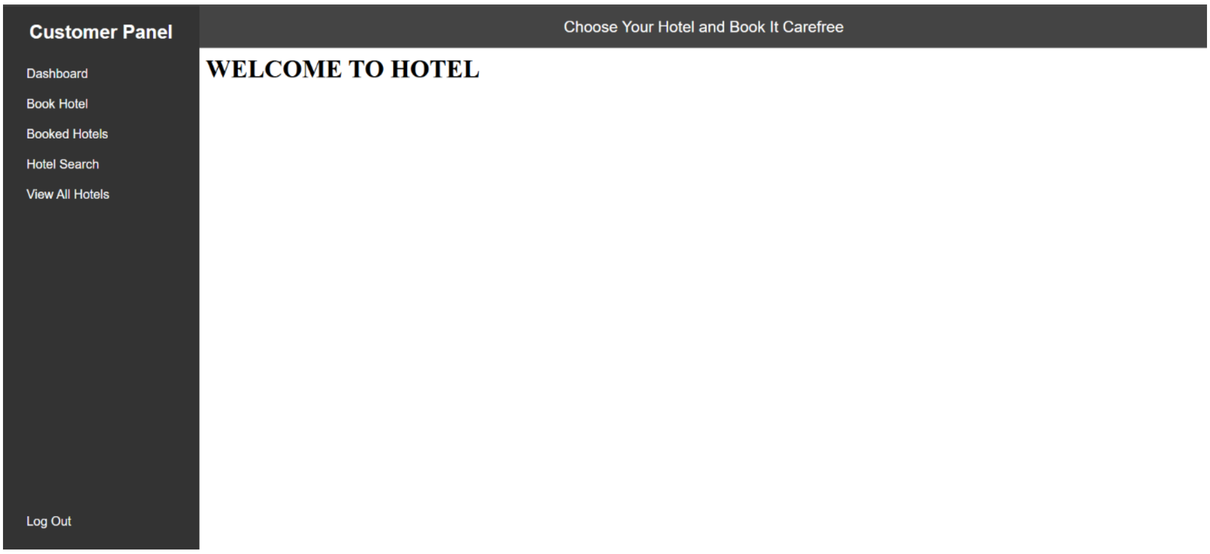
Customer.html

```
<!DOCTYPE html>
<html lang="en">
<head>
```

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Hotel Dashboard</title>
<style>
  * {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: Arial, sans-serif;
  }
  body {
    display: flex;
  }
  .sidebar {
    width: 250px;
    height: 100vh;
    background: #333;
    color: white;
    padding: 20px;
  }
  .sidebar h2 {
    text-align: center;
    margin-bottom: 20px;
  }
  .sidebar ul {
    list-style: none;
  }
```

```
.sidebar ul li {
   padding: 10px;
   cursor: pointer;
}
.sidebar ul li a {
   color: white;
   text-decoration: none;
   display: block;
}
.sidebar ul li:hover {
   background: #444;
}
.main-content {
   flex: 1;
   display: flex;
   flex-direction: column;
}
.header {
   background: #444;
   color: white;
   padding: 15px;
   text-align: center;
}
.iframe-container {
   flex: 1;
   border: none;
   width: 100%;
```

```
            height: 100%;
        }
    </style>
</head>
<body>
    <div class="sidebar">
        <h2>Admin Panel</h2>
        <ul>
            <li><a href="hoteldash.html" target="content-
frame">Dashboard</a></li>
            <li><a href="addhotel.html" target="content-frame">Hotel
Add</a></li>
            <li><a href="hoteldelete.html" target="content-frame">Hotel
delet</a></li>
            <li><a href="updatehotel.html" target="content-frame">SettingsHotel
update</a></li>
            <li><a href="searchhotel.html" target="content-frame">SettingsHotel
search</a></li>
            <li><a href="viewallhotel.jsp" target="content-frame">viewall
hotels</a></li>
        </ul>
    </div>
    <div class="main-content">
        <div class="header">Hotel Dashboard</div>
        <iframe name="content-frame" class="iframe-container"
src="hoteldash.html"></iframe>
    </div>
</body>
</html>
```



Choose Your Hotel and Book It Carefree

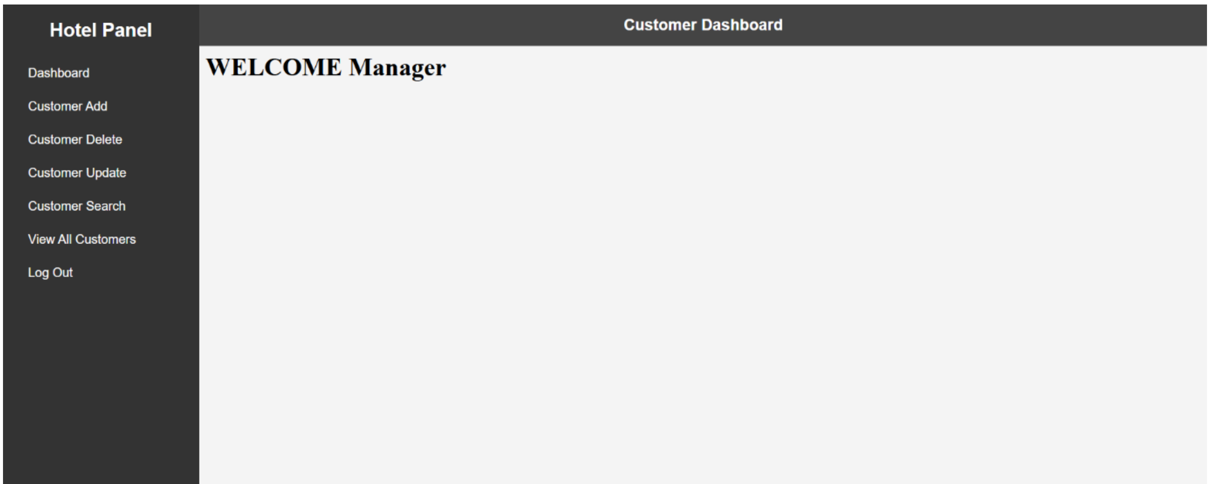
WELCOME TO HOTEL

Hotel.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Customer Dashboard</title>
    <style>
        * {
            margin: 0;
            padding: 0;
            box-sizing: border-box;
            font-family: Arial, sans-serif;
        }
        body {
            display: flex;
            min-height: 100vh;
            background: #f4f4f4;
        }
        .sidebar {
            width: 250px;
            height: 100vh;
            background: #333;
            color: white;
            padding: 20px;
            transition: width 0.3s ease-in-out;
        }
        .sidebar h2 {
            text-align: center;
            margin-bottom: 20px;
        }
        .sidebar ul {
            list-style: none;
        }
        .sidebar ul li {
            padding: 12px;
            cursor: pointer;
            transition: 0.3s ease-in-out;
        }
        .sidebar ul li a {
            color: white;
            text-decoration: none;
            display: block;
            font-size: 16px;
        }
        .sidebar ul li:hover {
            background: #444;
            padding-left: 15px;
        }
        .main-content {
            flex: 1;
            display: flex;
            flex-direction: column;
        }
        .header {
            background: #444;
            color: white;
            padding: 15px;
            text-align: center;
            font-size: 20px;
            font-weight: bold;
        }
        .iframe-container {
            flex: 1;
            border: none;
            width: 100%;
            height: 100%;
        }

        /* Responsive Design */
```

```
        @media (max-width: 768px) {
            .sidebar {
                width: 60px;
                overflow: hidden;
            }
            .sidebar h2 {
                display: none;
            }
            .sidebar ul li {
                text-align: center;
            }
            .sidebar ul li a {
                font-size: 0;
            }
            .sidebar ul li:hover a {
                font-size: 16px;
            }
        }
    </style>
</head>
<body>

    <div class="sidebar">
        <div>
        <h2>Hotel Panel</h2>
        <ul>
            <li><a href="custdash.html" target="content-
frame">Dashboard</a></li>
```

```
            <li><a href="custadd.html" target="content-frame">Customer Add</a></li>

            <li><a href="deletecust.html" target="content-frame">Customer Delete</a></li>

            <li><a href="updatecust.html" target="content-frame">Customer Update</a></li>

            <li><a href="custsearch.html" target="content-frame">Customer Search</a></li>

            <li><a href="viewallcust.jsp" target="content-frame">View All Customers</a></li>

        </ul>

    </div>

    <ul>

        <li><a href="index.html">Log Out</a></li>

    </ul>

    </div>


    <div class="main-content">

        <div class="header">Customer Dashboard</div>

        <iframe name="content-frame" class="iframe-container" src="custdash.html"></iframe>

    </div>


</body>

</html>
```



WELCOME Manager

6.2 Creating Tables in Databases

Customer table

Create table customer (Name varchar(30), Age varchar(30), Gender varchar(30), Phone varchar(30), Email varchar(30), Address varchar(30), Password varchar(30);

Hotel table

Create table hotelreg(Name varchar(30), Website varchar(30), Price varchar(30), Rooms Varchar(30), Extra varchar(30), Manager varchar(30), Email varchar(30), Address varchar(30), Password varchar(30);

6.3 Creating Jsp pages

Customer

```
<%@ page language="java" import="java.net.*, java.io.*, java.sql.*" %>
<html>
<head>
   <title>Customer Registration</title>
   <script>
```

```
        function showAlert(message, redirect) {
            alert(message);
            if (redirect) {
                window.location.href = "custlogin.html";
            }
        }
    </script>
</head>
<body>
    <%
        String s1 = request.getParameter("nm1");
        String s2 = request.getParameter("nm2");
        String s3 = request.getParameter("nm3");
        String s4 = request.getParameter("nm4");
        String s5 = request.getParameter("nm5");
        String s6 = request.getParameter("nm6");
        String s7 = request.getParameter("nm7");

        Connection con = null;
        PreparedStatement ps = null;

        try {
            // Load Oracle JDBC Driver
            Class.forName("oracle.jdbc.driver.OracleDriver");

            // Establish connection
            con =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"system", "system");

            // Prepare SQL statement
            ps = con.prepareStatement("INSERT INTO customer VALUES (?, ?, ?,
?, ?, ?, ?)");
            ps.setString(1, s1);
            ps.setString(2, s2);
            ps.setString(3, s3);
            ps.setString(4, s4);
            ps.setString(5, s5);
            ps.setString(6, s6);
            ps.setString(7, s7);

            // Execute update
            int rowsInserted = ps.executeUpdate();

            if (rowsInserted > 0) {
    %>
                <script>
                    showAlert("Login successful!", true);
                </script>
    <%
            } else {
    %>
                <script>
                    showAlert("Invalid username or password. Please try again.", true);
                </script>
    <%
            }
        } catch (Exception e) {
    %>
            <script>
                showAlert("Invalid username or password. Please try again.", true);
            </script>
    <%
            e.printStackTrace();
        } finally {
            // Close resources
            try {
                if (ps != null) ps.close();
                if (con != null) con.close();
            } catch (SQLException ex) {
                ex.printStackTrace();
            }
        }
    %>
</body>
</html>
```

Hotel

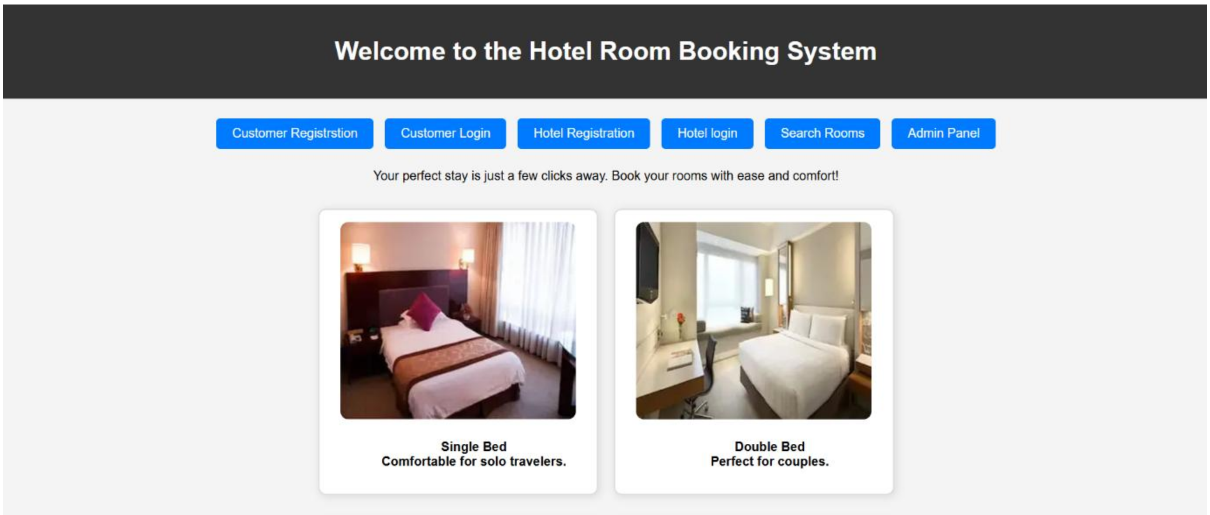
```
<%@ page language="java" import="java.net.*, java.io.*, java.sql.*" %>
<html>
<head>
```

```
    <title>Hotel Registration</title>
    <script>
        function showAlert(message, redirect) {
            alert(message);
            if (redirect) {
                window.location.href = redirect;
            }
        }
    </script>
</head>
<body>
<%
String s1 = request.getParameter("nm1");
String s2 = request.getParameter("nm2");
String s3 = request.getParameter("nm3");
String s4 = request.getParameter("nm4");
String s5 = request.getParameter("nm5");
String s6 = request.getParameter("nm6");
String s7 = request.getParameter("nm7");
String s8 = request.getParameter("nm8");
String s9 = request.getParameter("nm9");

Connection con = null;
PreparedStatement ps = null;

try {
    // Load Oracle JDBC Driver
    Class.forName("oracle.jdbc.driver.OracleDriver");
    con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"system", "system");

    // Prepare SQL Statement
    String query = "INSERT INTO hotel VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)";
    ps = con.prepareStatement(query);
    ps.setString(1, s1);
    ps.setString(2, s2);
    ps.setString(3, s3);
    ps.setString(4, s4);
    ps.setString(5, s5);
    ps.setString(6, s6);
    ps.setString(7, s7);
    ps.setString(8, s8);
    ps.setString(9, s9);

    // Execute the Query
    int rowsInserted = ps.executeUpdate();

    if (rowsInserted > 0) {
%>
        <script>
            showAlert("Registration successful!", "hotellogin.html");
        </script>
<%
    } else {
%>
```

```
        <script>
            showAlert("Registration failed. Please try again.", "hotelregist.html");
        </script>
<%
    }
} catch (Exception e) {
    e.printStackTrace(); // Log the error in server console
%>
    <script>
        showAlert("An error occurred. Please try again.", "hotelregist.html");
    </script>
<%
} finally {
    // Close Resources
    try {
        if (ps != null) ps.close();
        if (con != null) con.close();
    } catch (SQLException ex) {
        ex.printStackTrace();
    }
}
%>
</body>
</html>
```

6.4 Final Output

Home page

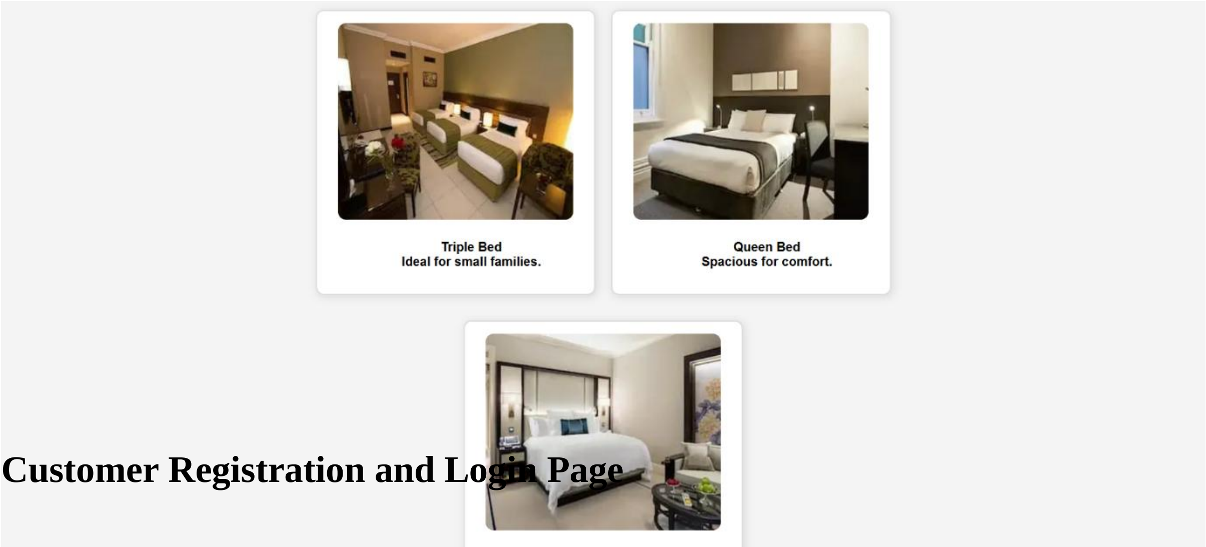


Welcome to the Hotel Room Booking System

Your perfect stay is just a few clicks away. Book your rooms with ease and comfort!

Single Bed
Comfortable for solo travelers.

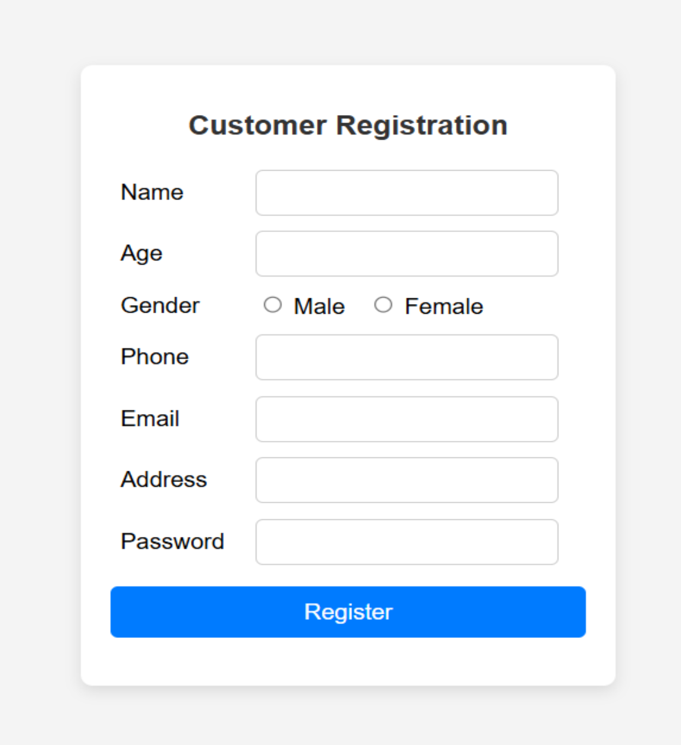
Double Bed
Perfect for couples.



Triple Bed
Ideal for small families.

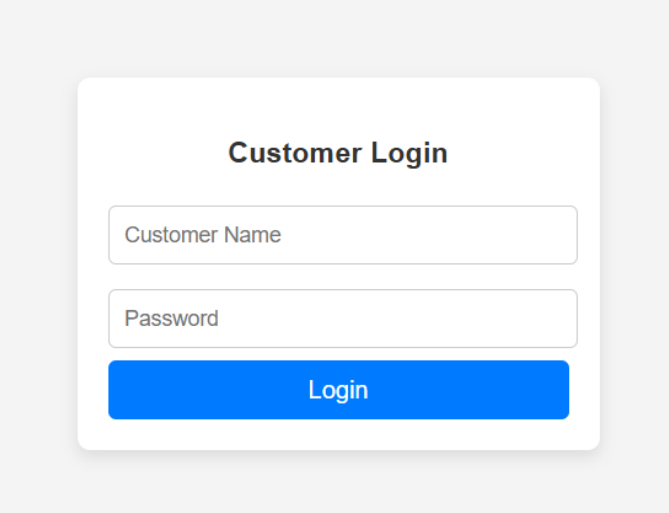
Queen Bed
Spacious for comfort.

Customer Registration and Login Page



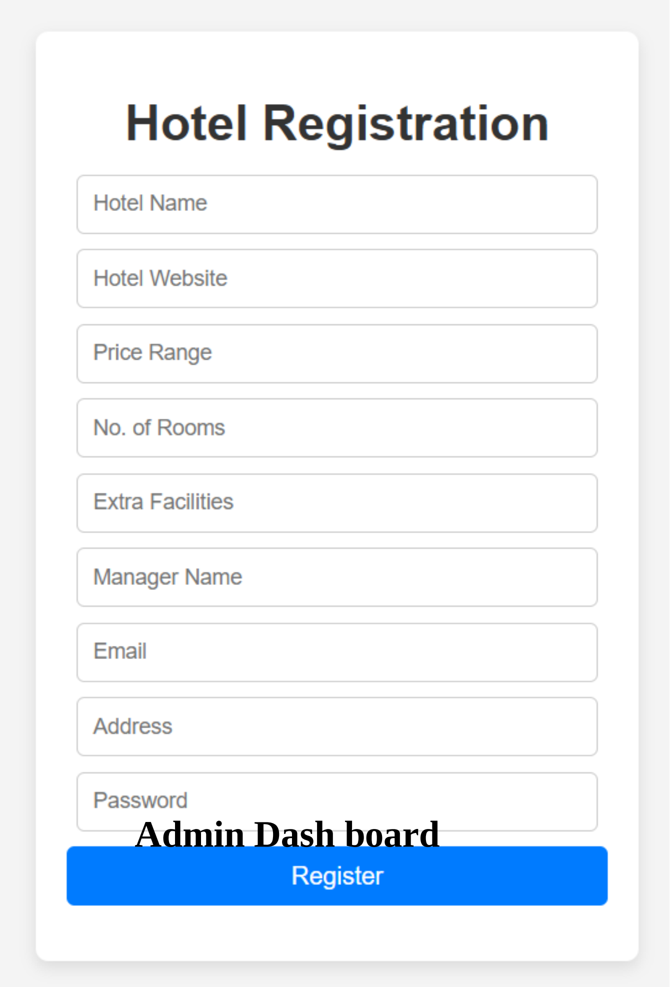
Customer Registration

Name	
Age	
Gender	○ Male ○ Female
Phone	
Email	
Address	
Password	



Customer Login

Hotel Registration and Login Page



Hotel Registration

Hotel Name

Hotel Website

Price Range

No. of Rooms

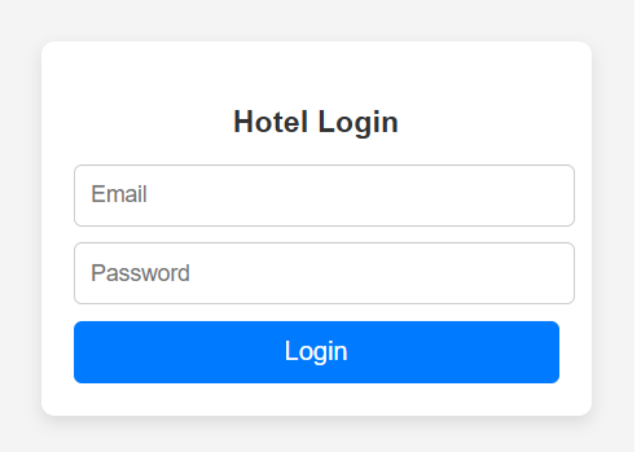
Extra Facilities

Manager Name

Email

Address

Password

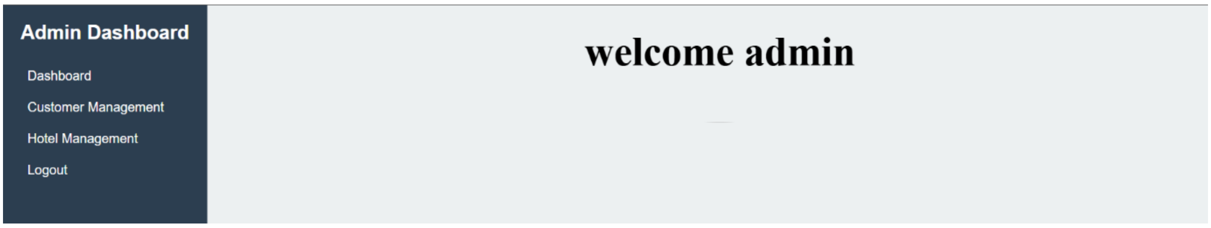


Hotel Login

Email

Password

Admin Dash board



Admin Dashboard

welcome admin

7 References

1. **Oracle.** *Java Platform, Standard Edition Documentation.* Available at: https://docs.oracle.com/javase/
Accessed on: May 2025.
2. **Spring Framework.** *Spring Boot Reference Documentation.* Available at: https://spring.io/projects/spring-boot
Accessed on: May 2025.
3. **Thymeleaf.** *Thymeleaf: Java XML/XHTML/HTML5 Template Engine.* Available at: https://www.thymeleaf.org/
Accessed on: May 2025.
4. **Mozilla Developer Network (MDN).** *HTML, CSS, and JavaScript documentation.* Available at: https://developer.mozilla.org/
Accessed on: May 2025.
5. **MySQL.** *MySQL 8.0 Reference Manual.* Available at: https://dev.mysql.com/doc/
Accessed on: May 2025.
6. GeeksforGeeks. *Full Stack Web Development Tutorial.* Available at: https://www.geeksforgeeks.org/full-stack-web-development/
Accessed on: May 2025.
7. **Baeldung.** *Guide to Spring Boot and Full Stack Web Development.* Available at: https://www.baeldung.com/spring-boot
Accessed on: May 2025.
8. **Stack Overflow.** *Community-driven Q&A for programming and software development.* Available at: https://stackoverflow.com/
Accessed on: May 2025.