ONLINE GIFT SHOP

A Full Stack Development Web Application / Project report

Submitted in the partial fulfillment of the requirements for

the award of the Degree of

Bachelor of Technology
in
Computer Science and Engineering
(Artificial Intelligence & Machine Learning)

by

Kotapati. Sravanthi 21761A4233

Under the guidance of
Dr. S. Nagarjuna Reddy
Associate Professor, Dept. of CSE

Department of Computer Science and Engineering

Lakireddy Bali Reddy College of Engineering (Autonomous)

Accredited by NAAC with 'A' Grade & NBA (Under Tier - I), ISO 9001:2015 Certified Institution

Approved by AICTE, New Delhi and Affiliated to JNTUK, Kakinada

L.B. REDDY NAGAR, MYLAVARAM, NTR DIST., A.P.-521 230.

Academic Year: 2022 – 23

LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING

(AUTONOMOUS)
Accredited by NAAC with 'A' Grade & NBA (Under Tier - I), ISO 9001:2015 Certified Institution
Approved by AICTE, New Delhi and Affiliated to JNTUK, Kakinada
L.B. REDDY NAGAR, MYLAVARAM, NTR DIST., A.P.-521 230.
cseoffice@Lbrce.ac.in, hodcse@Lbrce.ac.in Phone: 08659-222933, Fax: 08659-222931

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the **"20CSS2-Web Application Development using Full Stack Module-II (Backend Development)"** project entitled "**Online Gift Shop**" is being submitted by **Kotapati Sravanthi** in partial fulfilment for the award of B. Tech in Computer Science & Engineering (AI&ML) to the Jawaharlal Nehru Technological University Kakinada is a record of bonafide work carried out by him/her under our guidance.

The results embodied in this Skill Oriented Course Web Application / Project report have not been submitted to any other University or Institute for the award of any degree or diploma.

Project Guide	**Head of the department**
Dr. S. Nagarjuna Reddy	Dr. D. Veeraiah,
Associate Professor.	Professor.

External Examiner

ACKNOWLEDGEMENT

I would like to thank **Dr. S. Nagarjuna Reddy,** Associate Professor**,** CSE department for the encouraging and support in carrying out this Skill Oriented Course Web Application.

I also take the privilege to record my thanks to **Dr. D. Veeraiah,** Professor, Head of the Department of CSE whose encouragement, cooperation and valuable support crown my success.

I express my thanks to the support given by management in completing my Web Application / Project. I also express my sincere gratitude & deep sense of respect to the Principal, **Dr. K. Appa Rao** for making us available all the required assistance and his support and inspiration to carry out this Skill Oriented Course Web Application / Project in the Institute.

I am thankful to the teaching and non-teaching staff of CSE department for their direct as well as indirect help in my Skill Oriented Course Web Application / Project.

I am elated to avail my selves to this opportunity to express my deep sense of gratitude to my parents.

Kotapati. Sravanthi **21761A4233**

ONLINE GIFT SHOP

Abstract:

The advent of the internet has revolutionized the way we shop, making it more convenient and accessible than ever before. Online retail has witnessed significant growth in recent years, and one of its thriving segments is the online gift shop industry. This abstract aims to provide an overview of an online gift shop, highlighting its key characteristics, benefits, challenges, and potential for success.

An online gift shop is a digital platform that allows customers to browse and purchase a wide variety of gifts for various occasions, such as birthdays, anniversaries, holidays, and special events. The shop typically offers an extensive range of products, including personalized items, novelty gifts, home decor, jewelry, and much more. Customers can conveniently explore the catalog, select products of interest, make secure online payments, and have their chosen gifts delivered to the desired recipient's doorstep.

The key advantages of an online gift shop are numerous. Firstly, it provides customers with a vast selection of gift options, eliminating the need to visit multiple physical stores in search of the perfect present. Additionally, it offers the convenience of shopping from anywhere and at any time, making it ideal for busy individuals or those who prefer to avoid crowded retail environments. The online platform also facilitates comparison shopping, enabling customers to compare prices, read reviews, and make informed purchasing decisions. Moreover, online gift shops often provide gift-wrapping services and personalized messages, enhancing the overall gifting experience.

CONTENTS

CONCEPTS **Page No**

1.Introduction:

Introduction to Online Gift Shop:

In today's fast-paced and digital world, online shopping has become increasingly popular, offering convenience, accessibility, and a wide range of products at our fingertips. Among the flourishing sectors of online retail, the online gift shop industry stands out as a vibrant marketplace that caters to the diverse gifting needs of people worldwide. This introduction provides an overview of online gift shops, highlighting their significance, benefits, and the unique experiences they offer to both customers and businesses.

An online gift shop serves as a virtual marketplace where customers can explore and purchase an extensive array of gifts for various occasions and recipients. It provides a convenient platform for individuals to find and buy gifts without the need to visit physical stores, saving time and effort. Whether it's a birthday celebration, anniversary, graduation, or any other special event, online gift shops offer a wide selection of products to suit different tastes, preferences, and budgets.

The benefits of online gift shops are multifaceted. Firstly, they offer unparalleled convenience. Customers can browse through countless gift options, compare prices, read reviews, and make purchases from the comfort of their own homes or on the go. The 24/7 availability of online shops allows people to shop at any time that suits them, regardless of their location. This flexibility is particularly valuable for individuals with busy schedules or those living in remote areas with limited access to physical gift stores.

In conclusion, online gift shops have transformed the way people shop for gifts, providing convenience, variety, and an enjoyable experience. Whether you're searching for a thoughtful present or looking to start your own gifting business, the online gift shop industry offers tremendous potential. By harnessing the power of technology, embracing innovation, and focusing on customer satisfaction, online gift shops continue to redefine the art of gifting in the digital age.

2)Modules:

MODULE 1:-Connection of the ORACLE database by using the JDBC string.

JDBC is an acronym for Java Database Connectivity. It’s an advancement for ODBC (Open Database Connectivity). JDBC is a standard API specification developed in order to move data from frontend to the backend. This API consists of classes and interfaces written in Java. It basically acts as an interface (not the one we use in Java) or channel between your Java program and databases i.e. it establishes a link between the two so that a programmer could send data from Java code and store it in the database for future use.

Steps For Connectivity Between Java Program and Database

1. Import the Packages.
2. Load the drivers using the *forName() method.*
3. Register the drivers *using DriverManager.*
4. Establish a connection *using the Connection class object.*
5. Create a statement.
6. Execute the query.
7. Close the connections.

Example:-

```
<%@ page import="java.sql.*"%>
Class.forName("oracle.jdbc.driver.OracleDriver");
Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","username","pass
word");
String qry="select * from table_name";
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery(qry);
```

Module 2:- Usage of the PREPARED STATEMENTS for retrieving data.

A **PreparedStatement** is a pre-compiled SQL statement. It is a subinterface of **Statement**. Prepared Statement objects have some useful additional features than Statement objects. Instead of hard coding queries, PreparedStatement object provides a feature to execute a parameterized query.

Advantages of PreparedStatement

- When PreparedStatement is created, the SQL query is passed as a parameter. This Prepared Statement contains a pre-compiled SQL query, so when the PreparedStatement is executed, DBMS can just run the query instead of first compiling it.
- We can use the same PreparedStatement and supply with different parameters at the time of execution.
- An important advantage of PreparedStatements is that they prevent SQL injection attacks.

Methods of PreparedStatement:

- **setInt(int, int):** This method can be used to set integer value at the given parameter index.
- **setString(int, string):** This method can be used to set string value at the given parameter index.
- **setFloat(int, float):** This method can be used to set float value at the given parameter index.
- **setDouble(int, double):** This method can be used to set a double value at the given parameter index.
- **executeUpdate():** This method can be used to create, drop, insert, update, delete etc. It returns int type.
- **executeQuery():** It returns an instance of ResultSet when a select query is executed.

EXAMPLE :

```
Class.forName("oracle.jdbc.driver.OracleDriver");
Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Username","password");
String qry="insert into Table_name values(?,?,?,?,?)";
PreparedStatement pstmt=con.prepareStatement(qry);
pstmt.setString(1,name);
pstmt.setString(2,fname);
pstmt.setString(3,DOB);
pstmt.setString(4,Gender);
pstmt.setString(5,CNIC);
int res=pstmt.executeUpdate();
```

Creating a table using the ResultSet.

```
String qry2="select * from results";
Statement stmt2=con.createStatement();
ResultSet rs2=stmt2.executeQuery(qry2);

while(rs2.next()){
if(rs2.getString(1).equals("1")) {
count=count+1;
}
if(rs2.getString(2).equals("1")) {
count1=count1+1;
}
}

%>
<b><u><i><h1>RESULTS</b></i></u></h1>
<table border="2px" align="center" bgcolor="3E567C">
<tr>
<th colspan='9'> <p style="color:yellow">FINAL RESULT </th>
</tr>
<tr>
<th> <p style="color:yellow">YSR</th>
<th> <p style="color:yellow">TDP</th>
<th> <p style="color:yellow">JSP</th>
<th> <p style="color:yellow">CONGRESS</th>
<th> <p style="color:yellow">NOTA</th>
<th> <p style="color:yellow">BJP</th>
<th> <p style="color:yellow">CPI</th>
<th> <p style="color:yellow">PSP</th>
<th> <p style="color:yellow">TRS</th>
</tr>
<tr>
```

```
<td> <%= count %> </td>
<td> <%= count1 %> </td>
<td> <%= count2 %> </td>
<td> <%= count3 %> </td>
<td> <%= count4 %> </td>
<td> <%= count5 %> </td>
<td> <%= count6 %> </td>
<td> <%= count7 %> </td>
<td> <%= count8 %> </td>
</tr>
</table>
```

Module 3: Including the JSP to the FrontEnd page.

Java Server Page (JSP) is a technology for controlling the content or appearance of Web pages through the use of servlets. Small programs that are specified in the Web page and run on the Web server to modify the Web page before it is sent to the user who requested it.

There are total three Scripting Element in JSP

1. Scriptlet tag.
2. Expression tag.
3. Declaration tag.

Scriptlet Tag:

This tag allow user to insert java code in JSP. The statement which is written will be moved to jspservice() using JSP container while generating servlet from JSP. When client make a request, JSP service method is invoked and after that the content which is written inside the scriptlet tag executes.

```
<html>

<body>

<% out.print("GeeksforGeeks"); %>  <!-- scriptlet tag -->

</body>
```

```
</html>
```

Expression tag:

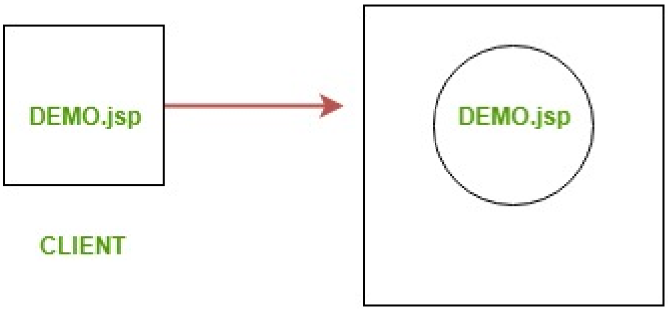
Expression tag is one of the scripting elements in JSP. Expression Tag in JSP is used for writing your content on the client-side. We can use this tag for displaying information on the client’s browser. The JSP Expression tag transforms the code into an expression statement that converts into a value in the form of a string object and inserts into the implicit output object.
Syntax: JSP tag

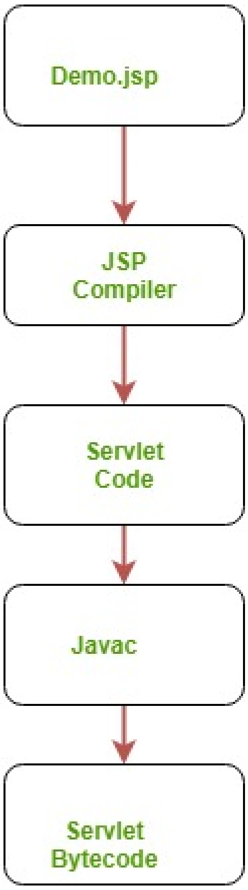
```
<%= expression %>
```

Declaration Tag:

Declaration tag is one of the scripting elements in JSP.This Tag is used for declare the variables. Along with this, Declaration Tag can also declare method and classes. Jsp initializer scans the code and find the declaration tag and initializes all the variables, methods, and classes. JSP container keeps this code outside of the service method (_JSPService()) to make them class level variables and methods.

Syntax of JSP-Declaration Tag
<%! inside this tag we can initialize our variables, methods and classes %>
The flow chart of the JSP:-





<u>MODULE 4:-</u> Creating the table in jsp by retrieving data from Database.

```
<%@ page import="java.sql.*"%>

<%
try{

int count=0,count1=0,count2=0,count3=0,count4=0,count5=0,count6=0,count7=0,count8=0;
Class.forName("oracle.jdbc.driver.OracleDriver");
Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","Uname","password");
String qry="select * from Login3";
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery(qry);



%>
```

```
<html>
<body align="center" style="background-image:url("https://cdn.staticcr/effects/footagecrate-
low-poly-background-black-wall-prev-full.pngate.com/stock-hd")">
<b><u><i><h1>DETAILS OF VOTERS</b></i></u></h1>
<table border="2px" align="center" bgcolor="3E567C">
<tr>
<th>NAME</th>
<th>FATHER NAME</th>
<th>DATE_OF_BIRTH</th>
<th>GENDER</th>
<th>CNIC_NUMBER</th>
</tr>
<%
while(rs.next())
{

%>
<tr>
<td> <p style="color:yellow"><%=rs.getString(1) %> </p></td>
<td> <p style="color:yellow"><%=rs.getString(2)%> </p></td>
<td> <p style="color:yellow"><%=rs.getString(3)%> </p></td>
<td> <p style="color:yellow"><%=rs.getString(4) %> </p></td>
<td> <p style="color:yellow"><%=rs.getString(5)%> </p></td>
</tr>
<%
}
%>
</table>

<%

String qry1="select * from results";
Statement stmt1=con.createStatement();
ResultSet rs1=stmt1.executeQuery(qry1);
```

```
%>
<h1><b><u><i>ELECTION RESULTS</u></i></b></h1>
<table border="2px" align="center" bgcolor="3E567C">
<tr>
<th> <p style="color:yellow">YSR</th>
<th> <p style="color:yellow">TDP</th>
<th> <p style="color:yellow">JSP</th>
<th> <p style="color:yellow">CONGRESS</th>
<th> <p style="color:yellow">NOTA</th>
<th> <p style="color:yellow">BJP</th>
<th> <p style="color:yellow">CPI</th>
<th> <p style="color:yellow">PSP</th>
<th> <p style="color:yellow">TRS</th>

</tr>
<%
while(rs1.next())
{

%>
<tr>
<td> <p style="color:yellow"><%=rs1.getString(1)%></td>
<td> <p style="color:yellow"><%=rs1.getString(2)%></td>
<td> <p style="color:yellow"><%=rs1.getString(3)%></td>
<td> <p style="color:yellow"><%=rs1.getString(4)%></td>
<td> <p style="color:yellow"><%=rs1.getString(5)%></td>
<td> <p style="color:yellow"><%=rs1.getString(6)%></td>
<td> <p style="color:yellow"><%=rs1.getString(7)%></td>
<td> <p style="color:yellow"><%=rs1.getString(8)%></td>
<td> <p style="color:yellow"><%=rs1.getString(9)%></td>
</tr>
<%
```

```
}
%>
</table>
<%

String qry2="select * from results";
Statement stmt2=con.createStatement();
ResultSet rs2=stmt2.executeQuery(qry2);

while(rs2.next()){
if(rs2.getString(1).equals("1")) {
count=count+1;
}
if(rs2.getString(2).equals("1")) {
count1=count1+1;
}
if(rs2.getString(3).equals("1")) {
count2=count2+1;
}
if(rs2.getString(4).equals("1")) {
count3=count3+1;
}
if(rs2.getString(5).equals("1")) {
count4=count4+1;
}
if(rs2.getString(6).equals("1")) {
count5=count5+1;
}
if(rs2.getString(7).equals("1")) {
count6=count6+1;
}
if(rs2.getString(8).equals("1")) {
count7=count7+1;
}
```

```
if(rs2.getString(9).equals("1")) {
count8=count8+1;
}
}

%>
<b><u><i><h1>RESULTS</b></i></u></h1>
<table border="2px" align="center" bgcolor="3E567C">
<tr>
<th colspan='9'> <p style="color:yellow">FINAL RESULT </th>
</tr>
<tr>
<th> <p style="color:yellow">YSR</th>
<th> <p style="color:yellow">TDP</th>
<th> <p style="color:yellow">JSP</th>
<th> <p style="color:yellow">CONGRESS</th>
<th> <p style="color:yellow">NOTA</th>
<th> <p style="color:yellow">BJP</th>
<th> <p style="color:yellow">CPI</th>
<th> <p style="color:yellow">PSP</th>
<th> <p style="color:yellow">TRS</th>
</tr>
<tr>
<td> <p style="color:yellow"><%= count %> </td>
<td> <p style="color:yellow"><%= count1 %> </td>
<td> <p style="color:yellow"><%= count2 %> </td>
<td> <p style="color:yellow"><%= count3 %> </td>
<td> <p style="color:yellow"><%= count4 %> </td>
<td> <p style="color:yellow"><%= count5 %> </td>
<td> <p style="color:yellow"><%= count6 %> </td>
<td> <p style="color:yellow"><%= count7 %> </td>
<td> <p style="color:yellow"><%= count8 %> </td>
</tr>
</table>
```

```
</body>
</html>

<%
}
catch(Exception e)
{
out.println(e);
}
%>
```

3)Technologies Used:

Hyper Text Markup Language

(HTML) is the backbone of any website development process, without which a web page does not exist. Hypertext means that text has links, termed hyperlinks, embedded in it. When a user clicks on a word or a phrase that has a hyperlink, it will bring another web-page. A markup language indicates text can be turned into images, tables, links, and other representations. It is the HTML code that provides an overall framework of how the site will look.

HTML stands for Hypertext Markup Language and it is a widely used programming language used to develop web pages. Through this HTML Tutorial, we'll understand what HTML means, the features of HTML, the basic tags and elements used, and more.

- HyperText stands for Link between web pages.
- Markup Language means Text between tags that define the structure.

HTML is a markup language that is used to create web pages. It defines how the web page looks and how to display content with the help of elements. It forms or defines the structure of our Web Page, thus it forms or defines the structure of our Web Page. We must remember to save your file with .html extension.

Features Of HTML

- The learning curve is very easy (easy to modify)
- Creating effective presentations
- Adding Links wherein we can add references
- Can display documents on platforms like Mac, Windows, Linux, etc

- Adding videos, graphics, and audios making it more attractive

HTML Editor

- Simple editor: Notepad
- Notepad++
- Best editor: Sublime Text.

ii) Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of content and presentation, including layout, colors, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate css file, which reduces complexity and repetition in the structural content; and enable the css file to be cached to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

CSS is created and maintained through a group of people within the W3C called the CSS Working Group.

The CSS Working Group creates documents called specifications. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

Advantages of CSS

- **CSS saves time** − You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- **Pages load faster** − If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
- **Easy maintenance** − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

- **Superior styles to HTML** − CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- **Multiple Device Compatibility** − Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.
- **Global web standards** − Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

iii) Java Database Connectivity (JDBC)

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of Java SE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database. There are four types of JDBC drivers:

- JDBC-ODBC Bridge Driver,
- Native Driver,
- Network Protocol Driver, and
- Thin Driver

We can use JDBC API to access tabular data stored in any relational database. By the help of JDBC API, we can save, update, delete and fetch data from the database. It is like Open Database Connectivity (ODBC) provided by Microsoft.

Components of JDBC:

There are generally four main components of JDBC through which it can interact with a database. They are as mentioned below:

1. JDBC API: It provides various methods and interfaces for easy communication with the database. It provides two packages as follows, which contain the java SE and Java EE platforms to exhibit WORA(write once run anywhere) capabilities.

It also provides a standard to connect a database to a client application.

2. JDBC Driver manager: It loads a database-specific driver in an application to establish a connection with a database. It is used to make a database-specific call to the database to process the user request.

3. JDBC Test suite: It is used to test the operation(such as insertion, deletion, updation) being performed by JDBC Drivers.

4. JDBC-ODBC Bridge Drivers: It connects database drivers to the database. This bridge translates the JDBC method call to the ODBC function call. It makes use of the sun.jdbc.odbc package which includes a native library to access ODBC characteristics.

JDBC Drivers:

JDBC drivers are client-side adapters (installed on the client machine, not on the server) that convert requests from Java programs to a protocol that the DBMS can understand. There are 4 types of JDBC drivers:

Type-1 driver or JDBC-ODBC bridge driver

Type-2 driver or Native-API driver

Type-3 driver or Network Protocol driver

Type-4 driver or Thin driver

Interfaces of JDBC API:

A list of popular interfaces of JDBC API is given below:

Driver interface, Connection interface, Statement interface ,PreparedStatement interface, CallableStatement interface, ResultSet interface, ResultSetMetaData interface, DatabaseMetaData interface, RowSet interface.

Classes of JDBC API:

1.DriverManager class

2.Blob class

3.Clob class

iv) JSP

JSP technology is used to create web application just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc.

A JSP page consists of HTML tags and JSP tags. The JSP pages are easier to maintain than Servlet because we can separate designing and development. It provides some additional features such as Expression Language, Custom Tags, etc.

Advantages of JSP over Servlet:

There are many advantages of JSP over the Servlet. They are as follows:

1) Extension to Servlet

JSP technology is the extension to Servlet technology. We can use all the features of the Servlet in JSP. In addition to, we can use implicit objects, predefined tags, expression language and Custom tags in JSP, that makes JSP development easy.

2) Easy to maintain

JSP can be easily managed because we can easily separate our business logic with presentation logic. In Servlet technology, we mix our business logic with the presentation logic.

3) Fast Development

No need to recompile and redeploy.

If JSP page is modified, we don't need to recompile and redeploy the project..

4) Less code than Servlet

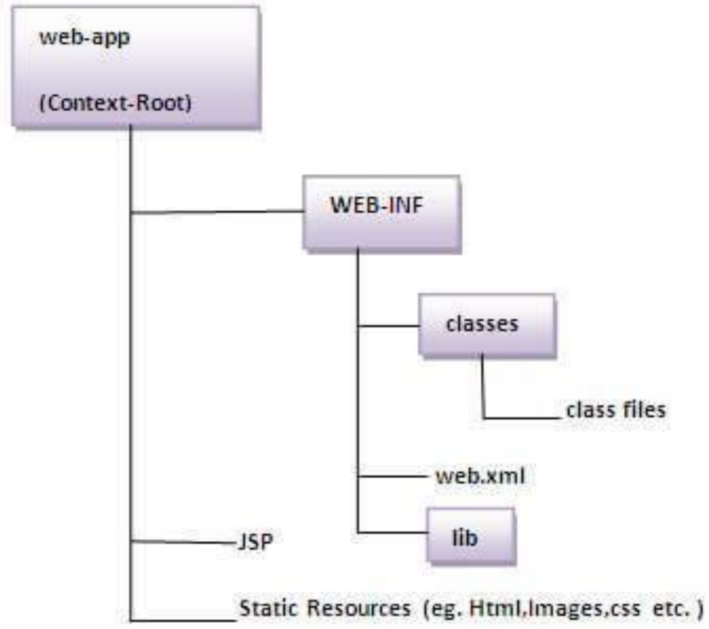
In JSP, we can use many tags such as action tags, JSTL, custom tags, etc. that reduces the code. Moreover, we can use EL, implicit objects, etc.

Do I need to follow the directory structure to run a simple JSP?

No, there is no need of directory structure if you don't have class files or TLD files. For example, put JSP files in a folder directly and deploy that folder. It will be running fine. However, if you are using Bean class, Servlet or TLD file, the directory structure is required.

The Directory structure of JSP

The directory structure of JSP page is same as Servlet. We contain the JSP page outside the WEB-INF folder or in any directory.



The Lifecycle of a JSP Page

The JSP pages follow these phases:

- Translation of JSP Page
- Compilation of JSP Page
- Classloading (the classloader loads class file)
- Instantiation (Object of the Generated Servlet is created).
- Initialization (the container invokes jspInit() method).
- Request processing (the container invokes_jspService() method).
- Destroy
- the container invokes jspDestroy() method.

4)Implementation:

a) <u>Html Code and Css code</u>

1) <u>Signin Page:</u>

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {font-family: Arial, Helvetica, sans-serif;}
form {border: 3px solid #f1f1f1;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  margin: 8px 0;
  display: inline-block;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #04AA6D;
  color: white;
  padding: 14px 20px;
  margin: 8px 0;
  border: none;
  cursor: pointer;
  width: 100%;
}

button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 15%;
  border-radius: 20%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
```

```
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}
</style>
</head>
<body>

<h2>Login Form</h2>

<form action="signin.jsp" method="post">
  <div class="imgcontainer">
    <img src="r2.jpg" alt="Avatar" class="avatar">
  </div>

  <div class="container">
    <label for="uname"><b>username</b></label>
    <input type="text" placeholder="Enter Username" name="username" required>

    <label for="psw"><b>password</b></label>
    <input type="password" placeholder="Enter Password" name="password" required>

    <button type="submit">Login</button>
    <label>
      <input type="checkbox" checked="checked" name="remember"> Remember me
    </label>
  </div>
<center><h3>Not have an account ?<a href="signup.html">signup</a></center>
 </div>
</form>
</body>
</html>
```

2)Signup Page:

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {font-family: Arial, Helvetica, sans-serif;}
form {border: 3px solid #f1f1f1;}

input[type=text], input[type=password] {
  width: 100%;
  padding: 12px 20px;
  margin: 8px 0;
  display: inline-block;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

button {
  background-color: #04AA6D;
  color: white;
  padding: 14px 20px;
  margin: 8px 0;
  border: none;
  cursor: pointer;
  width: 100%;
}
```

```
button:hover {
  opacity: 0.8;
}

.cancelbtn {
  width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}

.imgcontainer {
  text-align: center;
  margin: 24px 0 12px 0;
}

img.avatar {
  width: 11%;
  border-radius: 20%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
     width: 100%;
  }
}
</style>
</head>
<body>

<h2>Register</h2>

<form action="signup.jsp" method="post">
  <div class="imgcontainer">
    <img src="r2.jpg" alt="Avatar" class="avatar">
  </div>

  <div class="container">
    <label for="uname"><b>username</b></label>
    <input type="text" placeholder="Enter Username" name="username" required>

    <label for="psw"><b>password</b></label>
    <input type="password" placeholder="Enter Password" name="password" required>

    <button type="submit">Register</button>

  </div>
<center><h3>Already have an account ?<a href="signin.html">signin</a></center>

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

3)Admin Login Page:

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>login</title>
  <link rel="icon" type="image/x-icon" href="https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcS8W9ok3roBDzpYCGuq8msXgLd0V5Cjg1RPfdH
66AbAxrmIBO9ARqo4ZosA4zarrjtenWA&usqp=CAU">
  <style>
  body, html
  {
    font-family: cursive,bold;
  }
  *
  {
    box-sizing: border-box;
  }
  .bg-img
  {
    background-image:
url("https://i.pinimg.com/736x/c7/7e/f8/c77ef815d90d42f4ad86d9bc6324567b.jpg");
    background-repeat: no-repeat;
    background-size: cover;
    position: fixed;
    width: 100%;
    height: 100%;
    top: 0;
    left: 0;
    overflow: hidden;
  }
  .container
  {
```

```
  border-radius: 5%;
  position: fixed;
  left: 35%;
  max-width: 300px;
  padding: 16px;
  top: 13%;
  background-color: #005A9C.;
  margin-left: auto;
  margin-right: auto;
  width: 50%;
}
.container:hover
{
  box-shadow: 0 8px 16px 0 rgba(0,0,0,0.2);
}
input[type=text], input[type=password]
{
  width: 100%;
  padding: 15px;
  margin: 5px 0 22px 0;
  border: none;
  background: #f1f1f1;
}
input[type=text]:focus, input[type=password]:focus
{
  background-color: #ddd;
  outline: none;
}
.center
{
  margin: 0;
  position: absolute;
  top:100%;
  left: 50%;
```

```
      -ms-transform: translate(-50%, -50%);
      transform: translate(-50%, -50%);
    }
    a
    {
      border-radius: 100%;
      font-size: 100%;
      opacity: 0.9;
      padding: 16px 20px;
      background-color: #03548f;
      color: white;
      text-decoration: none;
    }
  </style>
</head>
<body>
  <div class="bg-img">
    <form action="adminpage.jsp" method="post" class="container">
      <h1>Admin Login</h1>
      <label for="username"><b>Username</b></label>
      <input type="text" placeholder="Enter Email" name="username" required>
      <label for="password"><b>Password</b></label>
      <input type="password" placeholder="Enter Password" name="password" required>
      <div class="center">
       <input type="submit" name="Login" value="Login">
      </div>
    </form>
  </div>
</body>
</html>
```

4) Index Page

```
<!DOCTYPE.html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>homepage</title>
    <link rel="stylesheet" type="text/css" href="homepage">
</head>
<style>
*{
        padding: 0;
        margin: 0;
}
.body {
        width: 1170px;
        margin:auto;
}
header {
        background: linear-gradient(rgba(0,0,0,0),rgba(0,0,0,0)),url("r1.jpg");
        height: 100vh;
        -webkit-background-size:cover;
        background-size:cover;
        background-position: center center;
        position: relative;
}
.nav-area {
        float: right;
        list-style: none;
        margin-top; 20px;

}
.nav-area li {
        display: inline-block;
```

```
                background-color: inherit;
}
.nav-area li a {

            text-decoration: none;
            padding: 5px 20px;
            font-family: poppins;
            font-size: 23px;
            color:Red;
}
.nav-area li a:hover{
            background: #fff;
            color: #333
}
a:link {
  text-decoration: none;
}
.dropbtn {


  padding: 16px;
  font-size: 16px;
  border: none;
}

.dropdown {
  position: relative;
  display: inline-block;
}

.dropdown-content {
  display: none;
  position: absolute;
  min-width: 160px;
  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;

}

.dropdown-content a {
  color: black;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
  font-size: 15px;
}

.dropdown:hover .dropdown-content {display: block;}
</style>
<body>
        <header>
<center>
        <h1 style="color:white; font-size:50px" >GIFTS SHOP</h></center><br><br><br>

        <div class="mainproject">
      </div>
        <ul class="nav-area">
              <li><a href="signin.html">SHOP NOW</a></li>
              <li><a href="signup.html">Sign Up</a></li>
             <li>
               <div class="dropdown">
                  <a class="dropbtn" style="font-size: 24px;">Login</a>
                  <div class="dropdown-content">
                    <a href="adminlogin.html" style="font-size: 19px;">AdminLogin</a>
                    <a href="signin.html" style="font-size: 19px;">Login</a>
             </div>
          </div></li>
          <li><a href="thank.html">MY CART</a></li>
        </ul>
      </div>
</header>
 </body>
</html>
```

5)Home page:-mainproject.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>giftsbuystore</title>
  <link rel="icon" type="image/x-icon" href="https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcS8W9ok3roBDzpYCGuq8msXgLd0V5Cjg1RPfdH
66AbAxrmIBO9ARqo4ZosA4zarrjtenWA&usqp=CAU">
<style>
  .navbar
  {
    overflow: hidden;
    border: 5px outset mediumvioletred;
    background-image: linear-gradient(140deg, #EADEDB 0%, #BC70A4 50%, #BFD641
120%);
    position: fixed;
    top: 55px;
    width:100%;
    height: 130px;
    z-index: 9999;
  }
  .divide1
  {
    overflow: hidden;
  width: 100%;
  top: 0;
  text-decoration: overline;
  z-index: 9999;
  position: fixed;
 }
 #grad1
 {
  background-image: linear-gradient(140deg, #EADEDB 0%, #BC70A4 50%, #BFD641
100%);
  opacity: 0.95;
 }
 *
 {
  box-sizing: border-box;
 }
 body
 {
  background-color: #f6f0fc;
  margin: 0;
 }
 .column
 {
  float: left ;
  width: 25%;
  padding: 0px 10px;
 }
 .columnew
 {
  float: left ;
  width: 50%;
  padding: 0px 10px;
 }
 .column1
```

```
{
 float: left ;
 width: 33%;
 padding: 0px 10px;
}
.row
{
 margin: 30px -5px;
}
.row:after
{
 content: "";
 display: table;
 clear: both;
}
.card
{
 box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
 padding: 1%;
 text-align: center;
 border-radius: 5%;
}
p.c
{
 font-family: cursive,bold;
 font-size: 20px;
}
.cardd
{
 position: relative;
 display: inline-block;
}
.cardd .img-top
{
  display: none;
  position: absolute;
  top: 0;
  left: 0;
  z-index: 99;
}
.cardd:hover .img-top
{
  display: inline;
}
.main
{
  width: 100%;
  margin-top: 170px;
  height: 1170px;
  padding: 25px;
}
.card:hover
{
  box-shadow: 0 8px 16px 0 rgba(0,0,0,0.2);
}
.image img
{
  width:100%;
  overflow: hidden;
}
.des
{
  text-align: center;
  padding: 0;
}
.top-left
{
  position: absolute;
  top: 90px;
  left: 560px;
}
.button
{
  padding: 10px 30px;
  font-size: 20px;
  text-align: center;
  cursor: pointer;
  outline: none;
  color: black;
  background-color: #f6f0fc;
  border: none;
  border-radius: 100%;
  box-shadow: #a6a2a1;
}
.button:hover
{
  background-color: #BC70A4
}
.button:active
{
  background-color: #EADEDB;
  box-shadow: #3051ab;
  transform: translateY(4px);
}
.responsive
{
  width: 95%;
}
a
{
  color: black;
  text-decoration: none;
```

```
 }
 .subnav
 {
   float: left;
   overflow: hidden;
 }
 .subnav .subnavbtn
 {
   font-size: 20px;
   border: none;
   outline: none;
   color: black;
   padding: 14px 16px;
   background-color: inherit;
   font-family: cursive,bold;
   margin: 0;
 }
 .navbar a:hover, .subnav:hover .subnavbtn
 {
   background-color: none;
 }
 .subnav-content
 {
   display: none;
   position: absolute;
   left: 0;
   background-image: linear-gradient(140deg, #EADEDB 0%, #BC70A4 50%, #BFD641
120%);
   width: 100%;
   z-index: 1;
 }
 .subnav-content a
 {
   float: left;
```

```
  color: black;
  text-decoration: none;
  font-family: cursive,bold;
  font-size: 18px;
  background-color: none;
  padding: 0px 50px;
 }
 .subnav-content a:hover
 {
  background-color: #eee;
  color: black;
 }
 .subnav:hover .subnav-content
 {
  display: block;
 }

 </style>
</head>
<body>
 <div class="divide1">
  <div id="grad1" style="text-align:center;margin:auto;color:black;padding-top: 10px;font-family: cursive,bold;font-size: 30px;">ONLINE GIFTS SHOP
 </div>
 </div>
  <div class="navbar">
   <div class="row">
    <div class="columnew">
     <div class="card">
      <a href="index.html">
       <p class="c">Home</p></a>
     </div>
    </div>
    <div class="column">
```

```
     <div class="subnav">
      <button class="subnavbtn">Buy Products<i class="fa fa-caret-down"></i></button>
       <div class="subnav-content">
        <a href=cakes.html>Cakes</a>
        <a href=chocolates.html>Chocolates</a>
        <a href=bouquet.html>Bouquet</a>
        <a href=plants.html>Plants</a>
        <a href=decor.html>Decor</a>
        <a href=personalized.html>Personalized</a>
        <a href=handicrafts.html>Handicrafts</a>
       </div>
      </div>
     </div>
    </div>
   </div>
  <div class="main">
   <div class="cardd">
    <img src="https://media.istockphoto.com/id/1383842766/photo/gift-image-for-fathers-day-and-men.jpg?b=1&s=170667a&w=0&k=20&c=Ndk5R-GxsHlrSMV9DNjACm4RzW4JFP9YIDHx8VUhTTQ=" alt="Card Front" width="1300px" height="400px">
    <img src="https://c4.wallpaperflare.com/wallpaper/1001/776/991/colorful-gifts-wood-box-wallpaper-preview.jpg" class="img-top" alt="Card Back" width="1300px" height="400px">
    <div class="row">
     <div class="column">
      <a href=Cakes.html>
      <div class="card">
       <div class="image">
        <img src="https://funcakes.com/content/uploads/2020/03/Rainbow_Surprise_Cake-960x960-c-default.jpg" height="200px">
       </div>
       <p class="c">
       Cakes
```

```
              </p>
              <div class="des">
               <button class="button">Click</button>
              </div>
            </div>
          </a>
        </div>
      <div class="column">
        <a href=chocolates.html>
        <div class="card">
          <div class="image">
            <img src="https://images2.alphacoders.com/768/76822.jpg" height="200">
          </div>
          <p class="c">Chocolates</p>
          <div class="des">
            <button class="button">Click</button>
          </div>
        </div>
      </a>
    </div>
    <div class="column">
      <a href=bouquet.html>
      <div class="card">
        <div class="image">
          <img src="https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRnaftjYcqSvQgwKydK6qWc892LcyEzcf-
fog&usqp=CAU" height="200">
          </div>
          <p class="c">Bouquet</p>
          <div class="des">
             <button class="button">Click</a></button>
          </div>
        </div>
      </a>
    </div>
    <div class="column">
      <a href=plants.html>
      <div class="card">
        <div class="image">
          <img src="https://wallpapercave.com/wp/wp9481707.jpg" height="200">
        </div>
        <p class="c">Plants</p>
        <div class="des">
          <button class="button">Click</button>
        </div>
      </div>
    </div>
  </a>
</div>
<div class="row">
  <div class="column1">
    <a href=decor.html>
    <div class="card">
      <div class="image">
        <img src="https://images.pexels.com/photos/1099816/pexels-photo-
1099816.jpeg?auto=compress&cs=tinysrgb&dpr=1&w=500" height="200">
      </div>
      <p class="c">Decor</p>
      <div class="des">
        <button class="button">Click</button>
      </div>
    </div>
  </a>
</div>
<div class="column1">
  <a href=personalized.html>
  <div class="card">
    <div class="image">
    <img
src="https://cdn.shopify.com/s/files/1/0645/5594/2108/products/createyouown_700x.webp?v
=1660114866" height="200">
  </div>
  <p class="c">Personalized</p>
  <div class="des">
   <button class="button">Click</button>
  </div>
</div>
</a>
</div>
<div class="column1">
<a href=handicrafts.html>
<div class="card">
  <div class="image">
    <img src="https://i.pinimg.com/236x/55/00/eb/5500eb7e3c88a0f194e79b7939df5bce--
butterfly-frame-picture-frames.jpg" height="200">
    </div>
    <p class="c">Handicrafts</p>
    <div class="des">
     <button class="button">Click</button>
    </div>
  </div>
</a>
</div>
</div>
</div>
</div><br>
<h4><center><a href="index.html"><button name="button"
type="button">Home</button></a></center>
</body>
</html>
font-weight: 700;
              cursor: pointer;
              outline: none;
          }
          input[type="submit"]:hover{
              border-color: green;
              transition: .5s;
          }
      </style
  </head>
  <body>
```

5.1:Cakes.html

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>cakes</title>
  <link rel="icon" type="image/x-icon" href="https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcS8W9ok3roBDzpYCGuq8msXgLd0V5Cjg1RPfdH66
AbAxrmIBO9ARqo4ZosA4zarrjtenWA&usqp=CAU">
<style>
* {
  box-sizing: border-box;
}
.column {
  float: left;
}
.column1 {
  float: left ;
  width: 25%;
  padding: 0px 10px;
  height: 20%;
}
.left {
  width: 35%;
  overflow: hidden;
}
.right {
  width: 65%;
  overflow: hidden;
}
body {
  font-family: cursive,bold;
}

.row {margin: 0 -5px;}

.row:after {
  content: "";
  display: table;
  clear: both;
}

@media screen and (max-width: 600px) {
  .column {
    width: 100%;
    display: block;
    margin-bottom: 20px;
  }
}

.card {
  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
  padding: 1%;
  text-align: center;
  border-radius: 5%;
  background-color: #f1f1f1;
  overflow:hidden;
}
```

```
.card:hover {
  box-shadow: 0 8px 16px 0 rgba(0,0,0,0.2);
  }

  .imgg {
    transform: scaleY(-1);
  }

</style>
</head>
<body>
<div class="row">

  <div class="column left" style="background-color:white;">
      <img src="https://funcakes.com/content/uploads/2020/03/Rainbow_Surprise_Cake-
960x960-c-default.jpg" width="170%" height="100%">
      <div class="imgg">
      <img src="https://funcakes.com/content/uploads/2020/03/Rainbow_Surprise_Cake-
960x960-c-default.jpg" width="170%" height="100%">
    </div>
  </div>

  <div class="column right" style="background-color:white;">
    <div class="row">
  <div class="column1">
    <div class="card">
    <img src="https://livforcake.com/wp-content/uploads/2017/07/black-forest-cake-thumb.jpg"
width="100%">
      <p>&#8377;500</p>
      <p>Angel cake</p>
      <button onclick="myFunction('demo1')">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
    <div class="card">
      <img
src="https://i.pinimg.com/originals/26/9a/e0/269ae08020d0dddd050cda68a0e1a35b.png"
width="100%">
      <p>&#8377;500</p>
      <p>Apple cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
    <div class="card">
      <img
src="https://i.pinimg.com/originals/ea/14/4a/ea144a6c14d810faa9705cb2a28ebee6.jpg"
width="100%">
      <p>&#8377;500</p>
      <p>Applesauce cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
    <div class="card">
      <img src="https://ourtableforseven.com/wp-content/uploads/chocolate-rainbow-bundt-cake-
main-ourtableforseven.com_-480x480.jpg" width="100%">
      <p>&#8377;500</p>
      <p>Babka</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
</div>

<br>

<div class="row">
  <div class="column1">
    <div class="card">
    <img
src="https://imagesvc.meredithcorp.io/v3/mm/image?url=https%3A%2F%2Fimg1.cookinglight
.timeinc.net%2Fsites%2Fdefault%2Ffiles%2Fstyles%2F4_3_horizontal_-
_1200x900%2Fpublic%2F1542062283%2Fchocolate-and-cream-layer-cake-1812-
cover.jpg%3Fitok%3DR_xDiShk" width="133%">
      <p>&#8377;500</p>
      <p>Banana Cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
    <div class="card">
      <img src="https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcRQeAkYFwdu97_3LZl9KTHhKAlc7SsEMLjMbA&u
sqp=CAU" width="150%">
      <p>&#8377;500</p>
      <p>Batik cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
```

```
    <div class="card">
      <img src="https://cdn.shopify.com/s/files/1/1147/6896/products/IMG_8455_2048x2048.jpg?v=1653581105" width="100%">
      <p>&#8377;500</p>
      <p>Baumkuchen</p>
      <button onclick="myFunction('demo')">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
    <div class="card">
      <img src=" https://hpbd.name/uploads/w450/2023/03/03/almond-birthday-cake-with-name-editing-1_95f87.jpg" width="150%">
      <p>&#8377;500</p>
      <p>Brownie</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
</div>

<br>

<div class="row">
  <div class="column1">
    <div class="card">
    <img src="https://iambaker.net/wp-content/uploads/2014/03/green-ombre-cake.jpg" width="70%">
      <p>&#8377;500</p>
      <p>Blondie</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
    <div class="card">
      <img src="https://aclassictwist.com/wp-content/uploads/2016/05/Makeup-Cake-2-
720x540.jpg" width="140%">
      <p>&#8377;500</p>
      <p>Cheesecake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
    <div class="card">
      <img
src="https://i.pinimg.com/originals/24/b7/93/24b7931733dc45de216bbfdcb2c706e6.jpg"
width="105%">
      <p>&#8377;500</p>
      <p>Bundt cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>

  <div class="column1">
    <div class="card">
      <img src="
https://i.pinimg.com/originals/5b/56/b8/5b56b836e36f5909d38b5837c219cf8d.jpg"
width="100%">
      <p>&#8377;500</p>
      <p>Butter Cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
</div>

<br>
<div class="row">
  <div class="column1">
    <div class="card">
    <img src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSb8t1QbI44duplCVIm7_4h2i4hnPocJT5Dkg&usqp=CAU" width="150%">
      <p>&#8377;500</p>
      <p>Chiffon cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
<div class="column1">
    <div class="card">
      <img src="https://faircake.co.uk/wp-content/uploads/2020/06/buttercreamflowerscakeclass.jpg" width="67%">
      <p>&#8377;500</p>
      <p>Dundee cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
  <div class="column1">
    <div class="card">
      <img src="https://i.pinimg.com/originals/25/0a/da/250ada05b2f17cea276f4c47bbebe26c.jpg" width="76%">
      <p>&#8377;500</p>
      <p>Coconut cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
  <div class="column1">
    <div class="card">
      <img src="https://i.pinimg.com/736x/f2/df/5d/f2df5db9d9ef1b59bfceb274df244dd6--
homemade-baby-cake-decorations.jpg" width="75%">
      <p>&#8377;500</p>
      <p>Batik cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
</div>
<br>
<div class="row">
  <div class="column1">
    <div class="card">
    <img src="https://images.pexels.com/photos/1702373/pexels-photo-
1702373.jpeg?cs=srgb&dl=pexels-brent-keane-1702373.jpg&fm=jpg" width="141%">
      <p>&#8377;500</p>
      <p>Chocolate cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
<div class="column1">
    <div class="card">
      <img src="https://www.freshboo.com/wp-content/uploads/2014/10/pinkblue-baby-shower-
cake-400x330.jpg" width="122%">
      <p>&#8377;500</p>
      <p>Birthday cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
  <div class="column1">
  <div class="card">
      <img src=" https://1.bp.blogspot.com/-
EX4QjTBykKI/YTzrEBLCjhI/AAAAAAALbw0/bMqEjeIp18kLKakGLhCPb81yb1n1dkwaQ
CLcBGAsYHQ/s4032/IMG_0910%2B%25281%2529.JPG" width="75%">
      <p>&#8377;500</p>
      <p>Blackout cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
  <div class="column1">
    <div class="card">
      <img src="https://edibletimes.com/wp-content/uploads/2020/08/cake-decorating-tips-
1024x683.jpg"width="149%">
      <p>&#8377;500</p>
      <p>Chestnut cake</p>
      <button onclick="myFunction()">Add to Cart</button>
      <p id="demo"></p>
    </div>
  </div>
</div>
  </div>
<script>
function myFunction() {
  let text = "Add to Cart";
  if (confirm(text) == true) {
    text = "Added to cart";
  } else {
    text = "Cancelled";
  }
  document.getElementById("demo").innerHTML = text;
}
</script>
<script>
function myFunction() {
  let text = "Add to Cart";
  if (confirm(text) == true) {
    text = "Added to cart";
  } else {
    text = "Cancelled";
  }
  document.getElementById("demo").innerHTML = text;
}
</script>
</div><br>
<h4><center><a href="index.html"><button name="button"
type="button">Home</button></a></center></h4>
<h4><center><a href="mainproject.html"><button name="button"
type="button">Back</button></a></center></h4>
</body>
</html>
```

6)Adminloginpage: admin1.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Table Example</title>
  <style>
    table {
      border-collapse: collapse;
      width: 100%;
    }
    th, td {
      border: 1px solid black;
      padding: 8px;
      text-align: left;
    }
  </style>
</head>
<body>
  <table>
    <tr>
      <th>Name</th>
      <th>Email</th>
      <th>Address</th>
      <th>City</th>
      <th>Zip</th>
      <th>State</th>
    </tr>
    <tr>
      <td>Sravanthi</td>
      <td>kotapati3@gmail.com</td>
      <td>Ongole</td>
      <td>Pamur</td>
      <td>523108</td>
      <td>Ap</td>
    </tr>
    <tr>
      <td>revanth</td>
      <td>revanth0@gmail.com</td>
      <td>guntur</td>
      <td>pamur</td>
      <td>528709</td>
      <td>Ts</td>
    </tr>
  </table><br><br>
<center><a href="hiii.html"><button>Delete</button></a><center><br><br>
<button onclick="printPage()">Print</button><br><br>
```

```
<a href="index.html"><button>Home</button></a>
<script>
    function printPage() {
      window.print();
    }
  </script>
</body>
</html>
```

7)AddToCart.css

```
/* Body Styles */
body {
  background-image: url('https://m.media-amazon.com/images/I/71+17bVYHxL.jpg');
}

/* Container Styles */
.container {
  max-width: 800px;
  margin: 0 auto;
  padding: 20px;
  background-color: #fff;
  border-radius: 4px;
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
}

/* Heading Styles */
h1 {
  text-align: center;
}

/* Table Styles */
table {
  width: 100%;
  border-collapse: collapse;
  margin-top: 20px;
```

```
}

table th,
table td {
  padding: 10px;
  text-align: left;
}

table th {
  background-color: #f2f2f2;
}

/* Quantity Buttons Styles */
button {
  background-color: #4CAF50;
  color: #fff;
  border: none;
  padding: 5px 10px;
  border-radius: 4px;
  cursor: pointer;
}

button:hover {
  background-color: #45a049;
}

input[type="number"] {
  padding: 5px;
  text-align: center;
}

/* Remove Button Styles */
a img {
  width: 50px;
  height: 50px;
}

/* Total Styles */
.total {
  margin-top: 20px;
  text-align: right;
}

.total span {
  font-weight: bold;
}

/* Order Now Button Styles */
.btn-container {
  text-align: center;
  margin-top: 20px;
}

.btn {
  display: inline-block;
  padding: 10px 20px;
  font-size: 16px;
  text-align: center;
  text-decoration: none;
  background-color: #4CAF50;
  color: #fff;
  border-radius: 4px;
  border: none;
  cursor: pointer;
  transition: background-color 0.3s;
}

.btn:hover {
 background-color: #45a049;
}
```

8) MyCart.html:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Online Gift Shop</title>
    <style>
        .product {
            margin-bottom: 20px;
        }

        .product h2 {
            margin-bottom: 5px;
        }

        .product p {
            margin-top: 0;
        }

        .add-to-cart {
            display: inline-block;
            padding: 5px 10px;
            background-color: #4CAF50;
            color: white;
            border: none;
            cursor: pointer;
            font-size: 14px;
            border-radius: 4px;
        }

        .add-to-cart:hover {
            background-color: #45a049;
        }

        .cart {
          margin-top: 20px;
        }

        .cart table {
          width: 100%;
          border-collapse: collapse;
        }

        .cart th,
        .cart td {
          border: 1px solid #ddd;
          padding: 8px;
          text-align: left;
        }

        .cart th {
          background-color: #f2f2f2;
        }
    </style>
</head>
<body>
    <h1>Welcome to the Online Gift Shop</h1>
    <div class="container-fluid" style="overflow-x: hidden;">
    <div class="row">
    <div class="product">
        <h2>Product 1</h2>
        <p>Price: $500</p>
        <button class="add-to-cart" onclick="addToCart('Product 1', 500)">Buy Now</button>
    </div>
    </div>
```

```
 <div class="container-fluid" style="overflow-x: hidden;">
 <div class="product">
    <h2>Product 2</h2>
    <p>Price: $450</p>
    <button class="add-to-cart" onclick="addToCart('Product 2', 450)">Buy Now</button>
 </div>
 </div>
</div>
 <div class="cart">
    <h2>Your Cart</h2>
    <table>
        <thead>
            <tr>
                <th>Product</th>
                <th>Price</th>
            </tr>
        </thead>
        <tbody id="cartItems">
        </tbody>
        <tfoot>
            <tr>
                <td colspan="2" align="right">
                    <strong>Total:</strong> $<span id="cartTotal">0</span>
                </td>
            </tr>
        </tfoot>
    </table>
 </div>
<center><a href="payment.html"><button>BUY</button></a></center>

 <script>
    var cartItems = [];
    var cartTotal = 0;
```

```
function addToCart(productName, productPrice) {
   // Create a new cart item object
   var cartItem = {
      name: productName,
      price: productPrice
   };

   // Add the item to the cart
   cartItems.push(cartItem);

   // Update the cart display
   updateCartDisplay();
}

function updateCartDisplay() {
   // Get the cart items element
   var cartItemsElement = document.getElementById("cartItems");

   // Clear the current items
   cartItemsElement.innerHTML = "";

   // Update the cart items
   cartItems.forEach(function(item) {
      // Create a new row for the cart item
      var row = document.createElement("tr");

      // Create cells for product name and price
      var nameCell = document.createElement("td");
      nameCell.textContent = item.name;
      var priceCell = document.createElement("td");
      priceCell.textContent = "$" + item.price;

      // Append cells to the row
      row.appendChild(nameCell);
```

```
            row.appendChild(priceCell);

            // Append the row to the cart items table
            cartItemsElement.appendChild(row);
         });

         // Update the cart total
         updateCartTotal();
      }

      function updateCartTotal() {
         // Get the cart total element
         var cartTotalElement = document.getElementById("cartTotal");

         // Calculate the total price
         cartTotal = cartItems.reduce(function(total, item) {
            return total + item.price;
         }, 0);

         // Update the cart total display
         cartTotalElement.textContent = cartTotal.toFixed(2);
      }
   </script>
</body>
</html>
```

9)Payment.html:

```
<html>
   <head>
      <title>Check out</title>
      <meta name="viewport" content="width=device-width, initial-scale=1">
      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
   </head>
```

```
<style>
  h1{
    text-align: center;
    color: Black;
  }

  .payment{
    align-content: center;
    width:30%;
    background-color: pink;
    padding:25px 15px 25px 15px;
    border-radius: 8px;
    font-family: 'Confortaa', cursive;
  }
  .billing{
    align-content: center;
    width:30%;
    background-color: pink;
    padding:25px 15px 25px 15px;
    border-radius: 8px;
    backdrop-filter: 50px;
    font-family: 'Confortaa', cursive;
  }

  input[type=number] {
    background-color: #558599;
    width: 100%;
    padding: 5px 20px;
    margin: 8px 0;
    box-sizing: border-box;
    border: none;
    border-bottom: 2px solid #f1f9fc;
   }
   input[type=text] {
```

```
  background-color: #558599;
  width: 100%;
  padding: 5px 20px;
  margin: 8px 0;
  box-sizing: border-box;
  border: none;
  border-bottom: 2px solid #f1f9fc;
}
.date input[type=number]{
    width:40%;
}
.icon-container {
  margin-bottom: 10px;
  padding: 0px 0;
  font-size: 24px;
}
.payment input[type=checkbox]{
    margin-left: 30px;
}
button{
    padding: 8px 8px 8px 8px;
    color: #f1f9fc(243, 230, 230);
    background-color: #e0eeef;
    font-family: 'Confortaa', cursive;
    font-weight: bold;
    width: 40%;
    border-radius: 5px;
}
body{
    background-image: url("
https://png.pngtree.com/thumb_back/fh260/background/20190221/ourmid/pngtree-click-buy-gesture-pay-image_19038.jpg");
    background-repeat: no-repeat;
    background-size: cover;
```

```
            }
        </style>
        <body>
            <h1 style="color:Black;">Payment</h1>
            <div class="checkout">
                <table width=700px align="center">
                    <tr>
                        <td class="billing">
            <!--   <div class="billing">      -->
                <form action="payments.jsp" method ="post">

                    <h3>SHIFTING ADDRESS</h3><br><br><br>

                    <label for="name"><i class="fa fa-user"></i> Full Name</label>
                    <input class="card" type="text" name="name" id="name"><br><br>
                    <label for="email"><i class="fa fa-envelope"></i> Email</label>
                    <input type="text" id="email" name="email" >
                    <label for="adr"><i class="fa fa-address-card-o"></i> Address</label>
                    <input type="text" id="adr" name="address">
                    <label for="city"><i class="fa fa-institution"></i> City</label>
                    <input type="text" id="city" name="city">
                    <label for="zip">Pincode</label>
                    <input type="text" id="zip" name="zip" placeholder="10001">
                    <label for="state">State</label>
                    <input type="text" id="state" name="state" placeholder="NY">

                    <input type="checkbox" nname="sameadr" >
                    <label for="sameadr" >Shipping address is same as Billing address.</label><br><br>

               <br><center> <img src="h1.jpg" alt="Paytm" width="200" height="200"></center><br>
    <br>
                <input type="checkbox" nname="sameadr" >
                <label for="cod">Cash On Delivery</label><br><br>
                <center><a href="success.html"><button>PAY</button></a></center>
            </form>
       <!-- </div>   -->
        </td>
        </tr>
        </table>
    </div>
    <br>
            <h4><center><a href="index.html"><button name="button"
    type="button">Home</button></a></center></h4>
    <h4><center><a href="mainproject.html"><button name="button" type="button">Back
    </button></a></center></h4>
        </body>
    </html>
```

b) List of Backend Tables:

1. Signup Table
2. Admin Login Table
3. Cart Table
4. Payment Table
5. Contact Table

c)**Back-end code:**

1) Signin.jsp:

```
<%@ page import="java.sql.*" %>
<%@ page import="java.io.*" %>
<%
   String Name = request.getParameter("username");
   String Password = request.getParameter("password");
   try{
      Class.forName("oracle.jdbc.driver.OracleDriver");
      Connection conn =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","21761A");
      String sql = "select * from signup where username=? and password=?";
      PreparedStatement pstmt = conn.prepareStatement(sql);
      pstmt.setString(1,Name);
      pstmt.setString(2,Password);
      int result = pstmt.executeUpdate();
       if(result==1){
              session.setAttribute("username",Name);
              session.setMaxInactiveInterval(5*60*60);
              response.sendRedirect("mainproject.html");
           }
      else{ %>
         response.sendRedirect("signin.jsp? error=1");
           <script>alert("Invalid Email or password");</script>
         <jsp:include page="signin.html"/>
      <%
      }
   }
   catch(Exception e){
      out.println(e);%>
     <jsp:include page="signin.html"/>
   <%
  }
```

```
   %>
```

2)Signup.jsp:

```
<%@ page import="java.sql.*" %>
<%@ page import="java.io.*" %>
<%
  String Name = request.getParameter("username");
  String Password = request.getParameter("password");
   try{
     Class.forName("oracle.jdbc.driver.OracleDriver");
     Connection conn =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","21761A");
     String sql = "insert into signup values(?,?)";

     PreparedStatement pstmt = conn.prepareStatement(sql);

     pstmt.setString(1,Name);

     pstmt.setString(2,Password);



     int result = pstmt.executeUpdate();

     if(result==1){ %>

                     <jsp:include page="signin.html"/>
     <%
     }
     else{ %>
        <script>alert("Some error occured. Please try again")</script>
        <jsp:include page="signup.html"/>
     <%
     }
  }
  catch(Exception e){ %>
```

```
       <%
     out.println(e);
    }
%>
```

3)adminpage.jsp:

```
<%@ page import="java.sql.*" %>
<html>
<head>
   <style>
       table {
           border-collapse: collapse;
           width: 100%;
       }

       th, td {
           border: 1px solid #ddd;
           padding: 10px;
       }

       th {
           background-color: #f2f2f2;
           font-weight: bold;
       }
                   body {
           font-family: Arial, sans-serif;
           color: #333;
                   }
   </style>
</head>
<body>
   <center><table>
     <h3>Online Gift Shop</h3>
<%
     String username=request.getParameter("username");
```

```
        String password=request.getParameter("password");
      try
      {
            Class.forName("oracle.jdbc.driver.OracleDriver");
            Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","system","21761A");
            if(username.equals("sravanthi") && password.equals("2004"))
            {
           response.sendRedirect("admin1.html");
            }
          else
           {%>
           <script>alert("Invalid admin !!!! Enter correct credentials")</script>
                   <jsp:include page="adminlogin.html"/>
          <%
          }
          }
        catch(Exception e)
         {%>
    <%
               System.out.println(e);
        }
 %>
               </table><center>

 </body>
</html>
```

4)addtocart.jsp:

```
<%@ page import="java.sql.*" %>
<%@ page import="java.io.*" %>
<html>
<head>
  <title>My Cart</title>
  <link rel="stylesheet" href="addtocart.css"/>
```

```
<script>
  function decrement(n) {
    var q = document.getElementById("quantity" + n);
    var val = parseInt(q.value);
     if (val > 1) {
      val = val - 1;
      q.value = val;
    }
   var totalval = document.getElementById("total" + n);
    var qcost = parseInt(document.getElementById("p" + n).innerHTML);
    totalval.innerHTML = val * qcost;
    calculateTotal();
  }
    function increment(n) {
    var q = document.getElementById("quantity" + n);
    var val = parseInt(q.value);

    if (val < 10) {
      val = val + 1;
      q.value = val;
    }
     var totalval = document.getElementById("total" + n);
    var qcost = parseInt(document.getElementById("p" + n).innerHTML);
    totalval.innerHTML = val * qcost;
    calculateTotal();
  }

  function calculateTotal() {
    var gt = 0;
    var totalElements = document.getElementsByClassName("subtotal");

    for (var i = 0; i < totalElements.length; i++) {
      gt += parseInt(totalElements[i].innerHTML);
    }
```

```
        var grandTotal = document.getElementById("gt");
        grandTotal.innerHTML = gt;
      }
  </script>
</head>
<body style="background-image: url('https://m.media-
amazon.com/images/I/71+17bVYHxL.jpg');">
<%
try {
    Class.forName("oracle.jdbc.driver.OracleDriver");
    Connection conn =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","21761A");

    String sql = "select * from cart where Email=?";
    PreparedStatement pstmt = conn.prepareStatement(sql);
    String Email = (String) session.getAttribute("email");
    pstmt.setString(1, Email);
    ResultSet rs = pstmt.executeQuery();
    int i = 1;
%>
<div class="container">
  <h1>My Cart</h1>
  <table border="1px">
    <thead>
      <tr>
        <th>Product Name</th>
        <th>Price</th>
        <th>Quantity</th>
        <th>Subtotal</th>
        <th>Remove</th>
      </tr>
    </thead>
<%
    while (rs.next()) {
```

```
%>
    <tr>
        <td><%= rs.getString(2) %></td>
        <td id="p<%= i %>"><%= rs.getString(3) %></td>
        <td>
          <button onclick="decrement('<%= i %>')">-</button>
          &nbsp;&nbsp;
          <input type="number" id="quantity<%= i %>" readonly size="10" min="1" max="10"
value="1">
          &nbsp;&nbsp;
          <button onclick="increment('<%= i %>')">+</button>
        </td>
        <td class="subtotal" id="total<%= i %>"><%= rs.getString(3) %></td>
        <td><a href="deleteItem.jsp?productname=<%= rs.getString(2) %>"><img
src="https://cdn.pixabay.com/photo/2023/03/30/17/34/ai-generated-7888395_960_720.png"
width="50px"></a></td>
       </tr>
<%
    i++;
}
conn.close();
}
catch(Exception e){
    out.println(e);
}

%>
  </table>
<div class="total">
    <span>Total:</span>
    <span id="gt" name="total">0</span>
  </div>
  <div class="btn-container">
    <button class="btn" onclick="t()">Order Now</button>
  </div>
</div>
<script>
  calculateTotal();
</script>
<script>
  function t()
  {
var tat = document.getElementById("gt").innerHTML;
var total =parseInt(tat);
window.location.href="http://localhost:8081/FSD-2/order.jsp?gt="+total;
}
</script>
</body>
</html>
```

5)Cart.jsp:

```
<%@ page import="java.sql.*" %>
<%@ page import="java.io.*" %>
<%
    try {
        String productname=request.getParameter("name");
        String Price=request.getParameter("price");
        String Email = (String)session.getAttribute("email");
                Class.forName("oracle.jdbc.driver.OracleDriver");
                Connection conn =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","21761A");
                String sql = "INSERT INTO cart VALUES (?, ?, ?)";

                PreparedStatement pstmt = conn.prepareStatement(sql);
                pstmt.setString(1, Email);
                pstmt.setString(2, productname);
                pstmt.setString(3, Price);
                int result = pstmt.executeUpdate();
                if(Email==null){ %>
                <script> window.alert("you should login") </script>
                <jsp:include page="mycart.html"/>
            <%
            }
            else{

              if(result==1){ %>
                <script> window.alert("Added to cart") ;
                 window.location.href = "http://localhost:8081/FSD-2/index.jsp#cat";
            </script>
            <%
            }
           }
      }
          catch (Exception e) {
              out.println(e);
      }%>
     %>
```

6)payments.jsp:

```
<%@ page import="java.sql.*" %>
<%@ page import="java.io.*" %>
<%
 String Name = request.getParameter("name");
 String Email= request.getParameter("email");
 String Address= request.getParameter("address");
 String City= request.getParameter("city");
 String Zip= request.getParameter("zip");
 String State= request.getParameter("state");
 try{
        Class.forName("oracle.jdbc.driver.OracleDriver");
        Connection conn =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","21761A");
        String sql = "insert into payment values(?,?,?,?,?,?)";
        PreparedStatement pstmt = conn.prepareStatement(sql);
        pstmt.setString(1,Name);
        pstmt.setString(2,Email);
        pstmt.setString(3,Address);
        pstmt.setString(4,City);
        pstmt.setString(5,Zip);
        pstmt.setString(6,State);
        int result = pstmt.executeUpdate();
       if(result==1){ %>
                <jsp:include page="success.html"/>
        <%
        }
        else{ %>
           <script>alert("Some error occured. Please try again")</script>
           <jsp:include page="payment.html"/>
        <%
        }
    }
    catch(Exception e){ %>

    <%
     out.println(e);
    }
%>
```

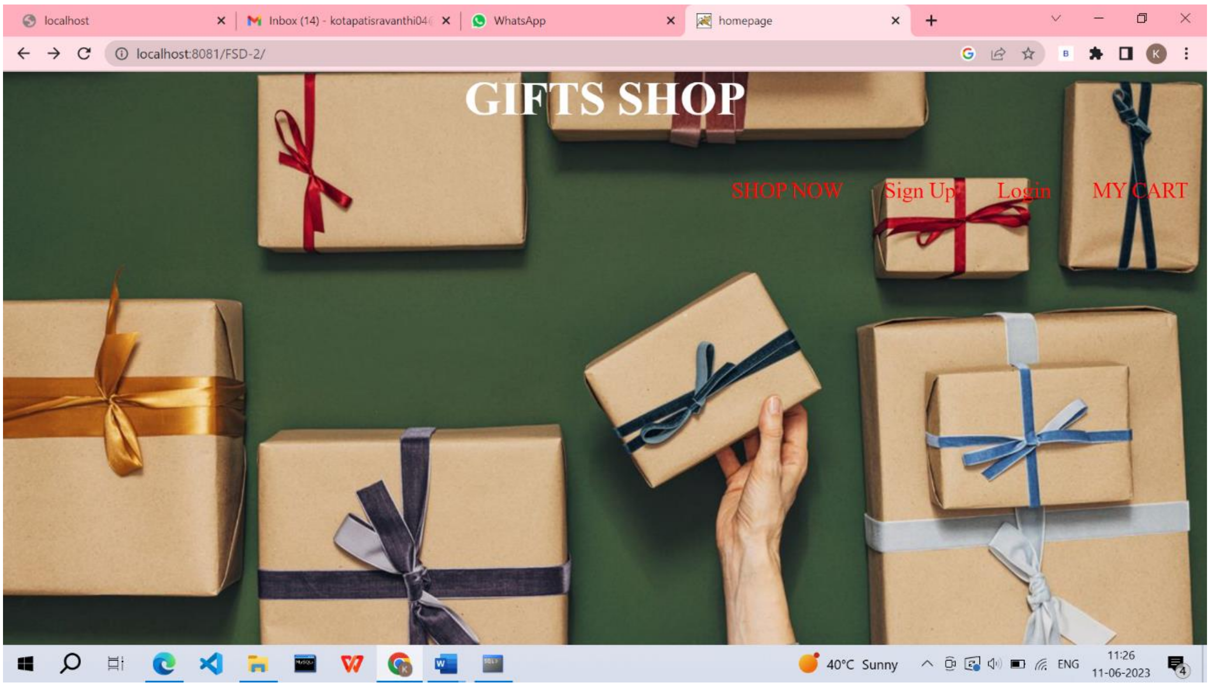
7)delete.jsp:

```
      <%@ page import="java.sql.*" %>
      <%@ page import="java.io.*" %>
      <%  String email = request.getParameter("email");
          try{
              Class.forName("oracle.jdbc.driver.OracleDriver");
              Connection conn =
      DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","21761A");
              String sql = "delete from users where email=?";
              PreparedStatement pstmt = conn.prepareStatement(sql);
```

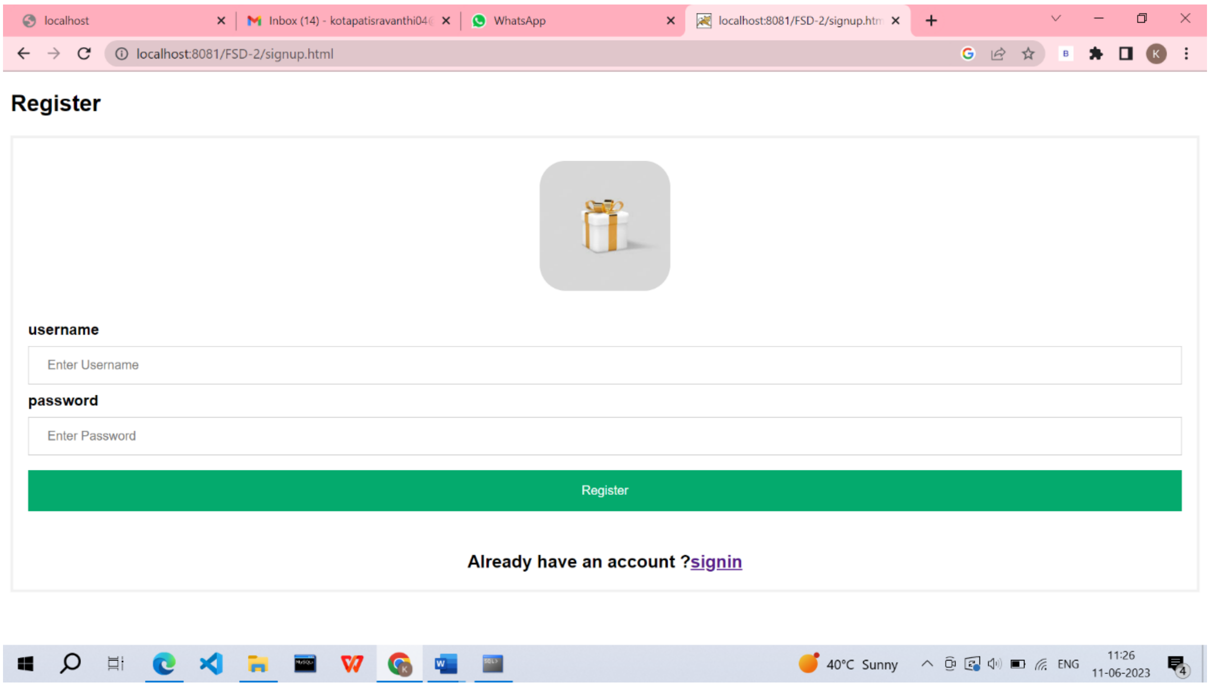
```
        pstmt.setString(1,email);
        int res = pstmt.executeUpdate();
        if(res == 1){ %>
          <script>
          window.location.href = "http://localhost:8081/FSD/admin1.jsp"
        </script>
        <%
        }
        else{  %>
          <script>
            alert("Something went wrong");
            window.location.href = "http://localhost:8081/FSD/admin1.jsp"
          </script>

        <%}}
      catch(Exception e){
        out.print(e);  }%>
```

5) Screenshorts:

1) Index page:



2)Signup Page:


localhost
Inbox (14) - kotapatisravanthi04
WhatsApp
localhost:8081/FSD-2/signup.htm
localhost:8081/FSD-2/signup.html
40°C Sunny
ENG
11:26
11-06-2023

Register

username

Enter Username

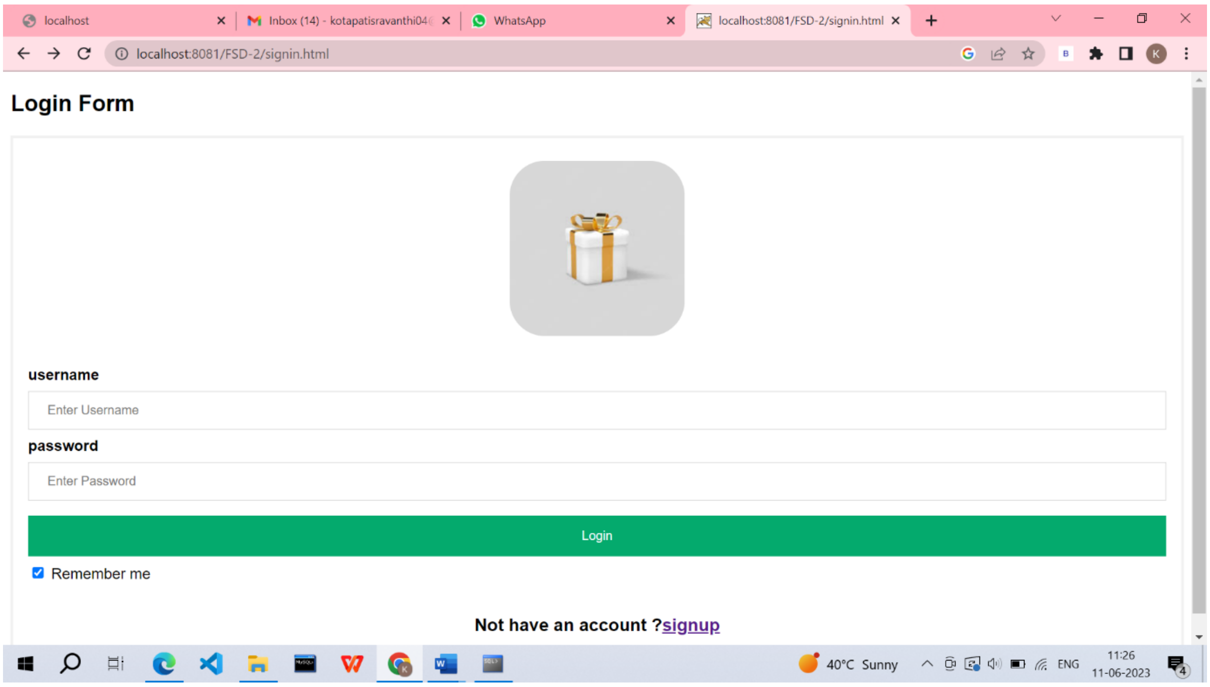
password

Enter Password

Register

Already have an account ?signin

3)Signin Page:


localhost
Inbox (14) - kotapatisravanthi04
WhatsApp
localhost:8081/FSD-2/signin.html
localhost:8081/FSD-2/signin.html
40°C Sunny
ENG
11:26
11-06-2023

Login Form

username

Enter Username

password

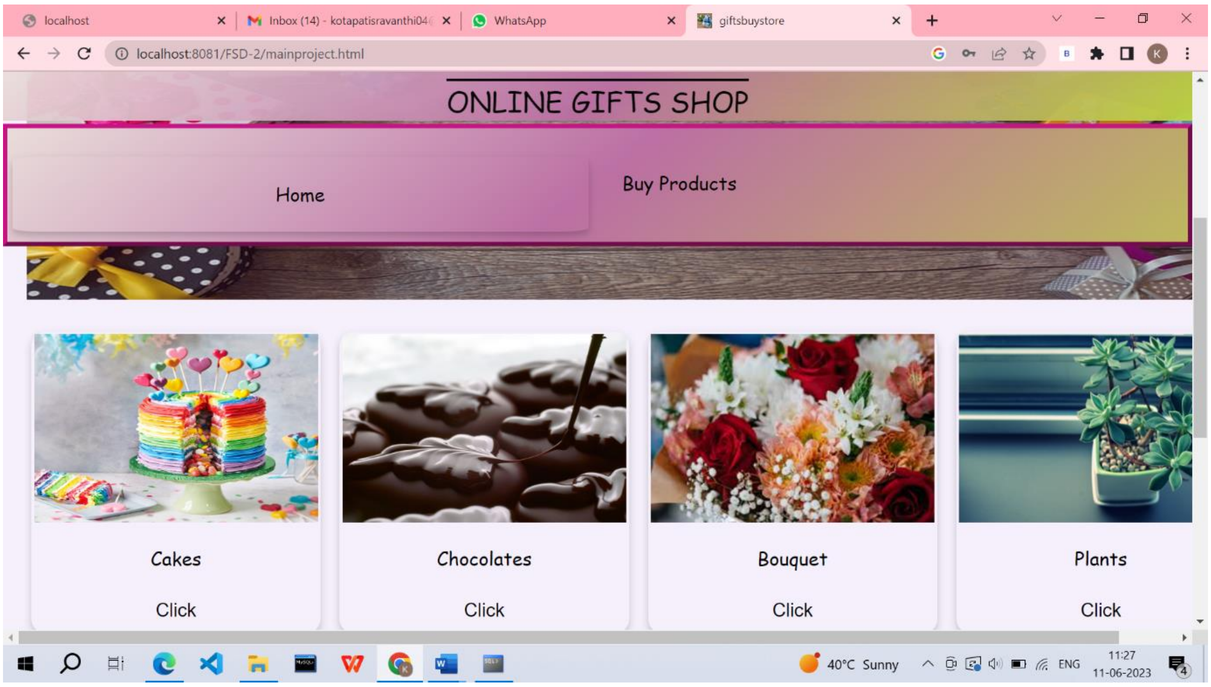
Enter Password

Login

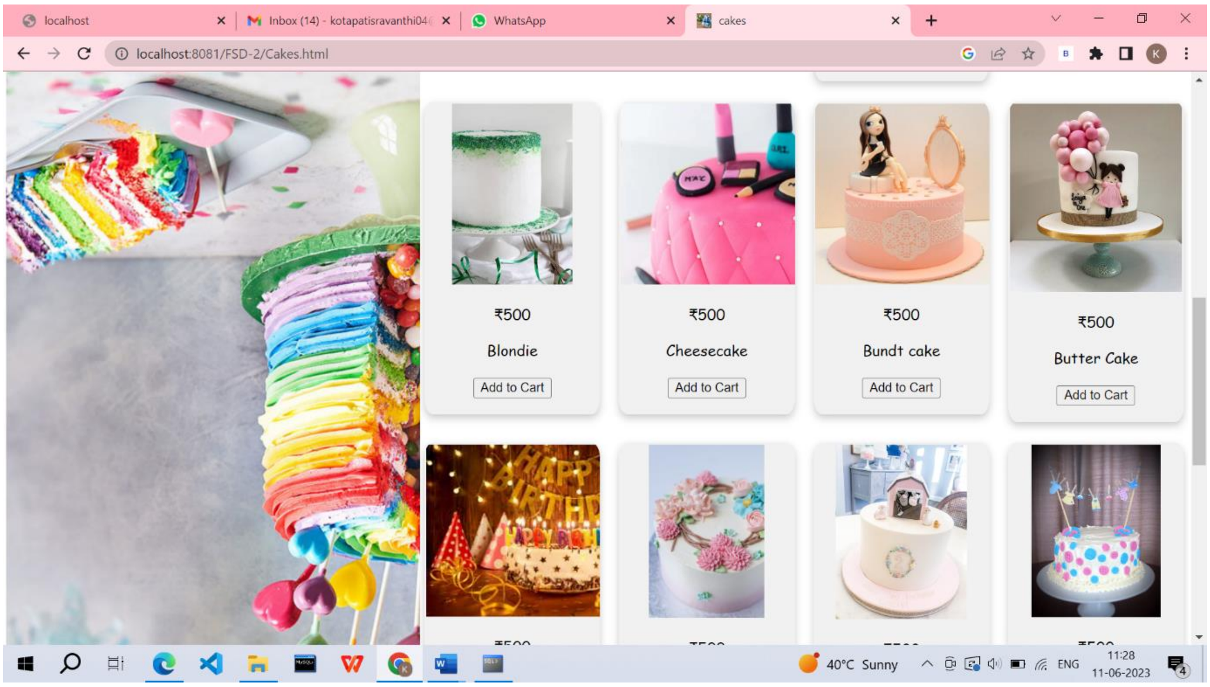
☑ Remember me

Not have an account ?signup

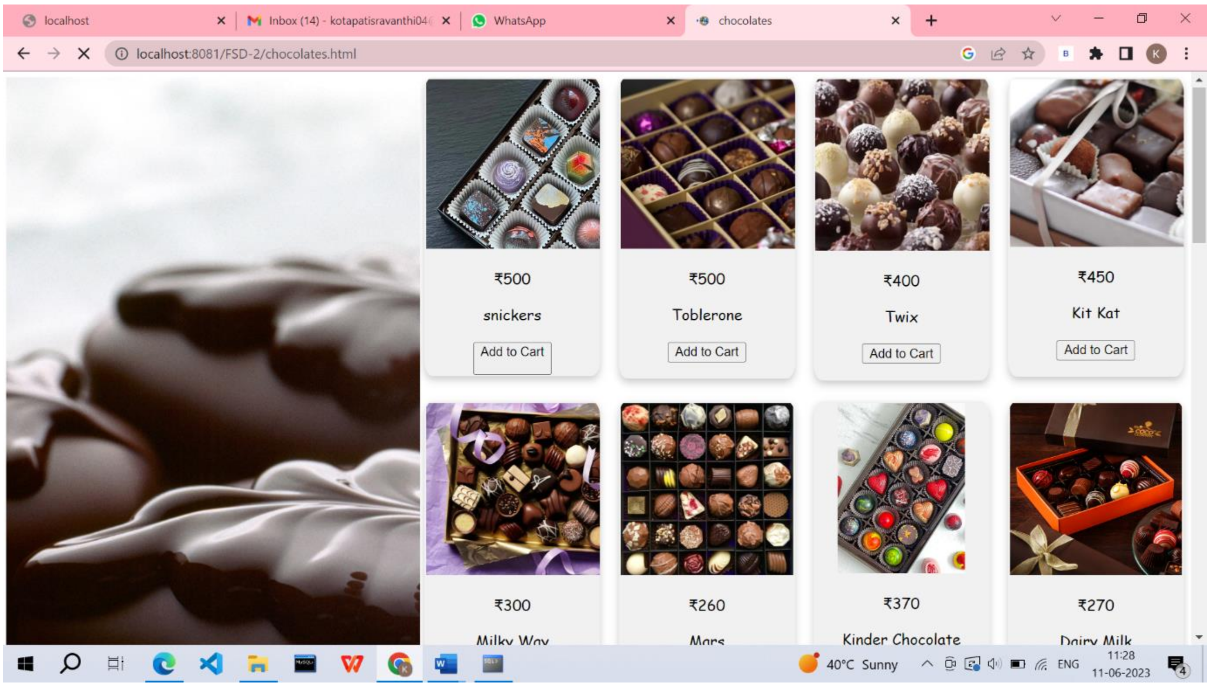
4)BuyProducts Page:



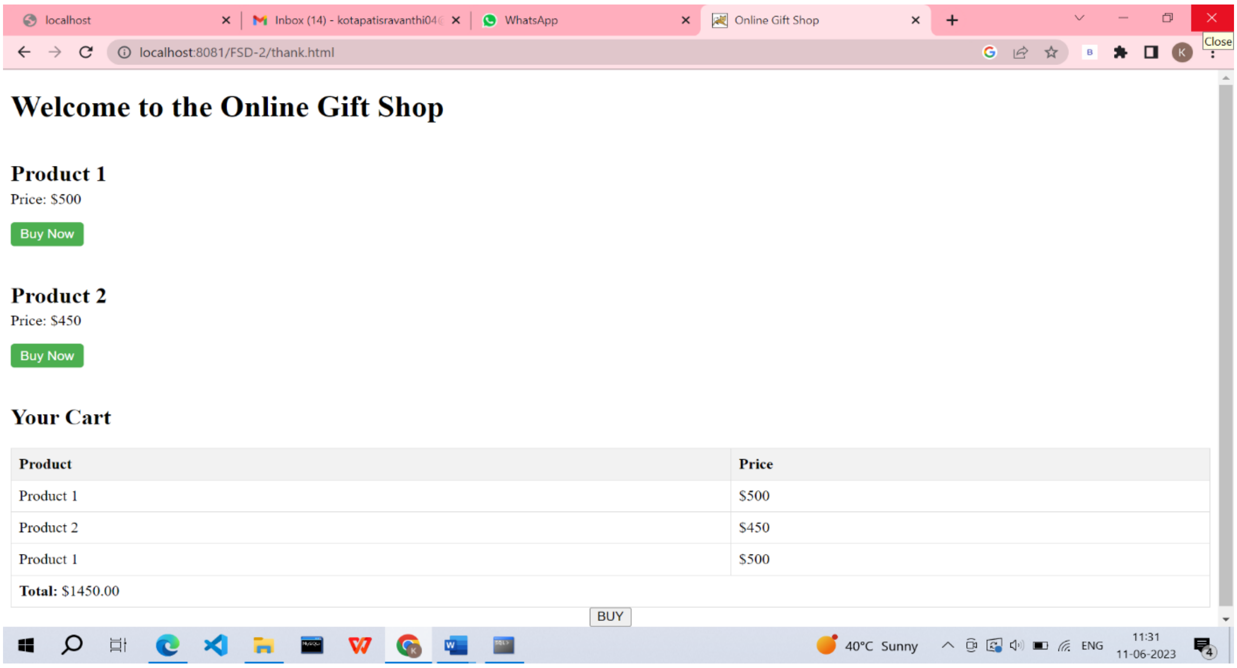
1)cakes page:



2)chocolates page:



5)Cart page:



Welcome to the Online Gift Shop

Product 1

Price: $500

Buy Now

Product 2

Price: $450

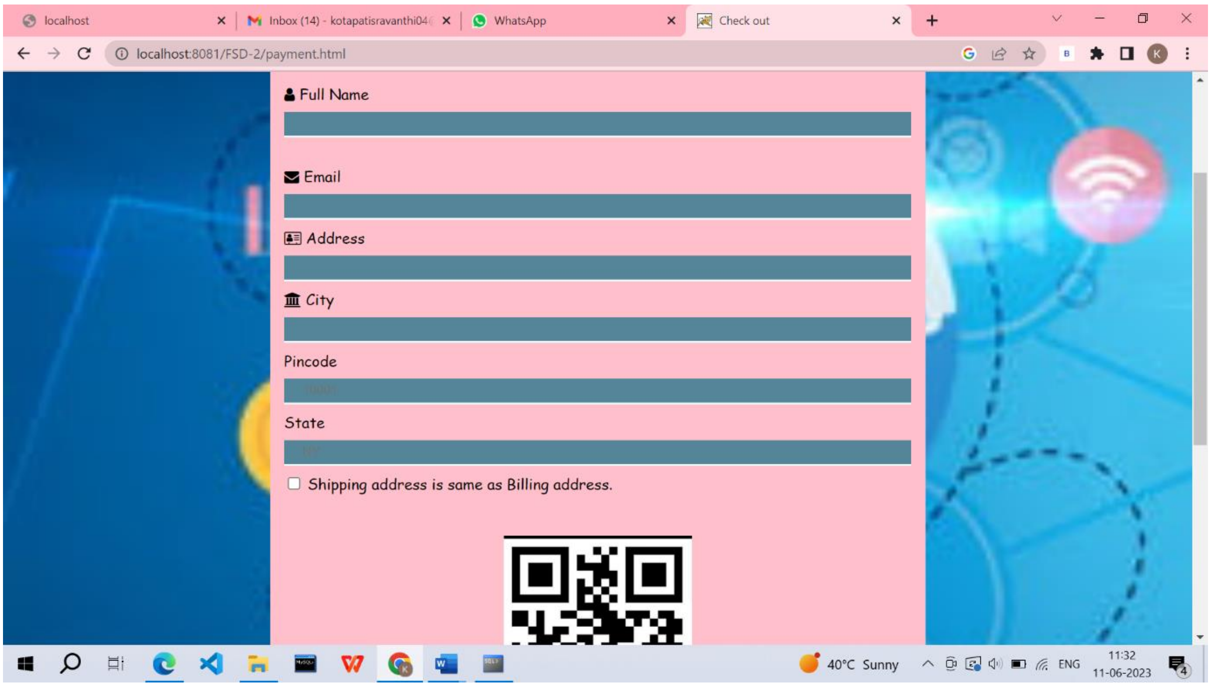
Buy Now

Your Cart

Product	Price
Product 1	$500
Product 2	$450
Product 1	$500
Total: $1450.00	

BUY

6)Payments page:



localhost
Inbox (14) - kotapatisravanthi04
WhatsApp
Check out
localhost:8081/FSD-2/payment.html
40°C Sunny
ENG
11:32
11-06-2023

Full Name

Email

Address

City

Pincode

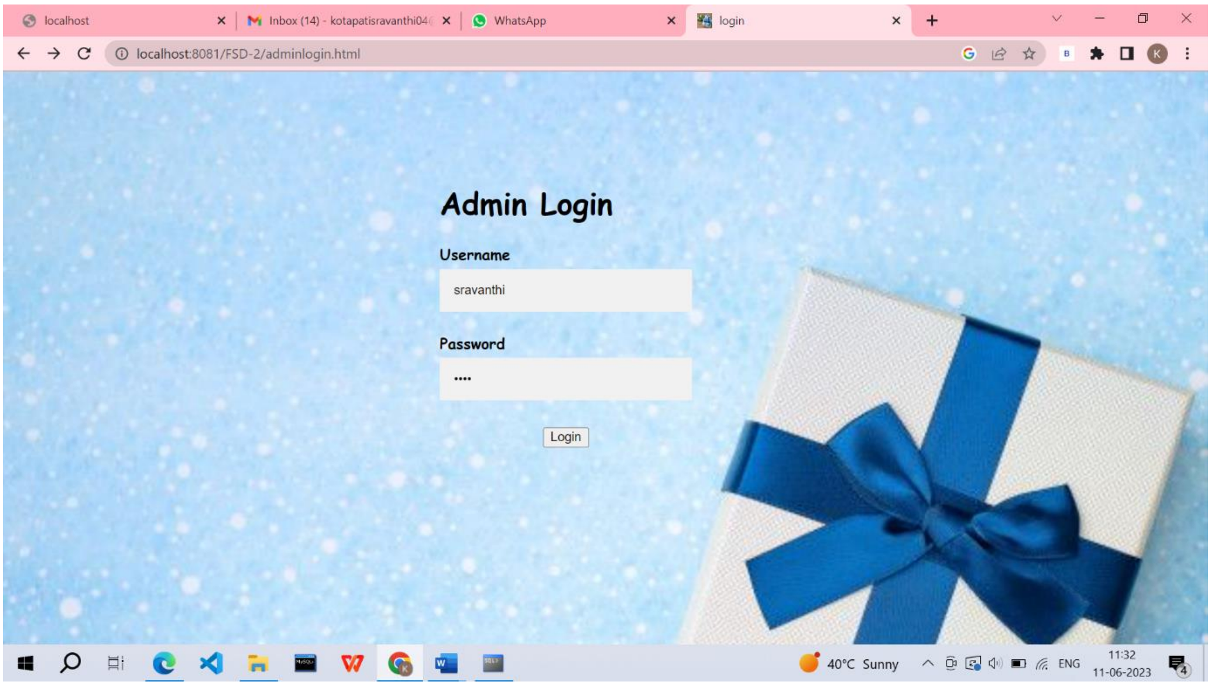
10001

State

NY

□ Shipping address is same as Billing address.

7)Admin Page:



localhost
Inbox (14) - kotapatisravanthi04
WhatsApp
login
localhost:8081/FSD-2/adminlogin.html
40°C Sunny
ENG
11:32
11-06-2023

Admin Login

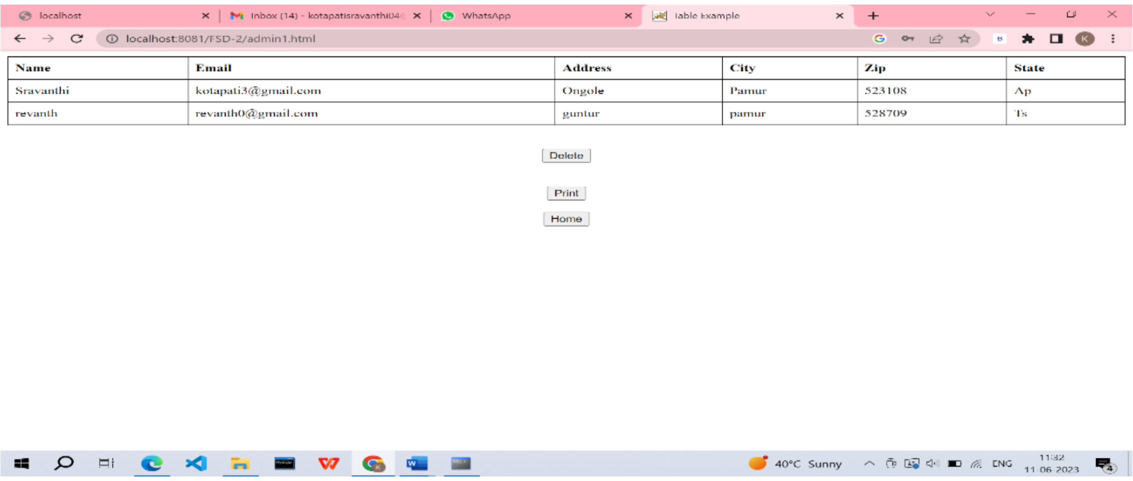
Username

sravanthi

Password

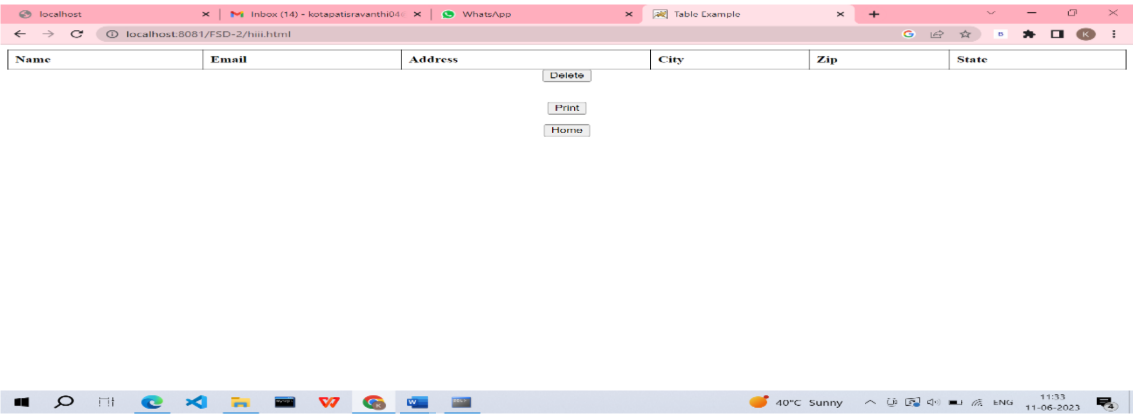
••••

8)Payment Details:


localhost
Inbox (14) - kotapatisravanthi04
WhatsApp
Table Example
localhost:8081/FSD-2/admin1.html
40°C Sunny
ENG
11:32
11-06-2023

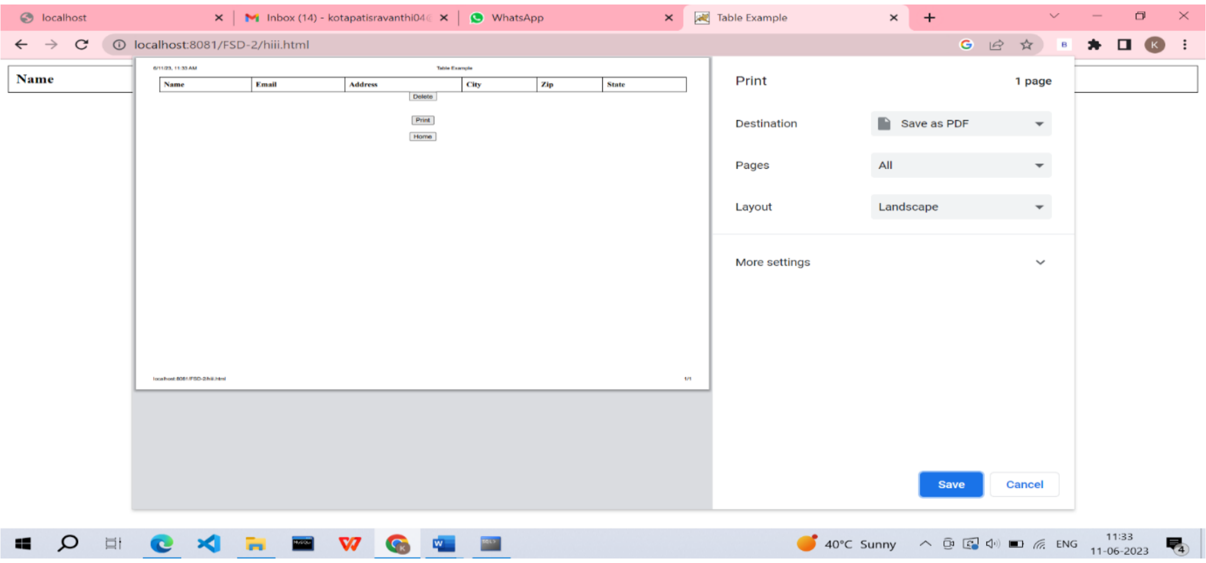
Name	Email	Address	City	Zip	State
Sravanthi	kotapati3@gmail.com	Ongole	Pamur	523108	Ap
revanth	revanth0@gmail.com	guntur	pamur	528709	Ts

9)Delete Page:


localhost
Inbox (14) - kotapatisravanthi04
WhatsApp
Table Example
localhost:8081/FSD-2/hiii.html
40°C Sunny
ENG
11:33
11-06-2023

Name	Email	Address	City	Zip	State

10)Print Page:



Name	Email	Address	City	Zip	State

Print 1 page

Destination: Save as PDF

Pages: All

Layout: Landscape

More settings

Conclusion

In conclusion, online gift shops offer a convenient and accessible way to purchase gifts for various occasions. They provide a wide range of options, allowing customers to find unique and personalized gifts for their loved ones. Here are some key points to consider:

1. Convenience: Online gift shops eliminate the need to visit physical stores, saving time and effort. Customers can browse through a vast selection of gifts from the comfort of their homes or on-the-go using their smartphones or computers.

2. Variety: Online gift shops often offer a diverse range of products, catering to different tastes, preferences, and occasions. Whether it's birthdays, anniversaries, holidays, or special events, customers can find specific gifts tailored to the recipient's interests.

3. Personalization: Many online gift shops provide options for personalization, allowing customers to add custom messages, names, or photos to their chosen gifts. This adds a special touch and makes the gift more meaningful and memorable.

4. Easy Comparison and Research: With online gift shops, customers can easily compare prices, read reviews, and research products before making a purchase. This helps in making informed decisions and ensures that the gift meets their expectations.

5.Worldwide Access: Online gift shops enable people to send gifts to loved ones living in different cities or even countries. They offer international shipping options, making it possible to send gifts globally, which is particularly beneficial for long-distance relationships or distant family members.

It's important to note that when choosing an online gift shop, it's advisable to consider factors such as the shop's reputation, customer reviews, payment security, and return policies. By doing so, customers can have a positive shopping experience and delight their loved ones with thoughtful gifts.

7) References:

http://nsr-materials.blogspot.com/2017/02/web-technologies-material_30.html

https://www.w3schools.com/html/

https://www.wedmegood.com/

https://www.w3schools.com/w3css/defaulT.asp

https://www.w3schools.com/html/

https://www.w3schools.com/js/