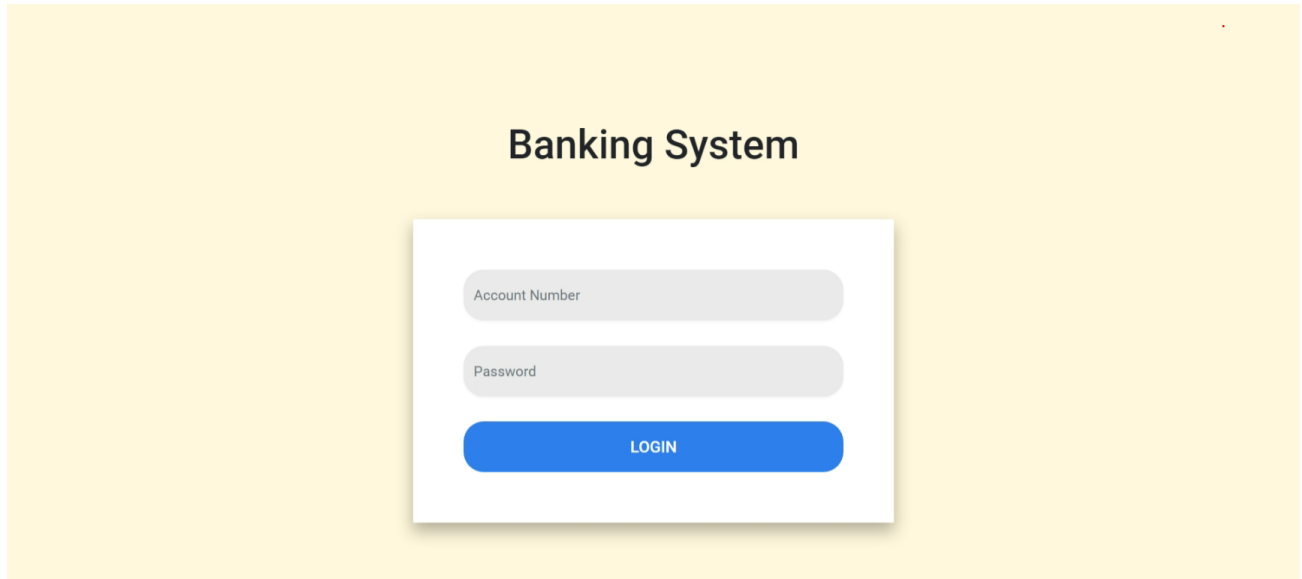
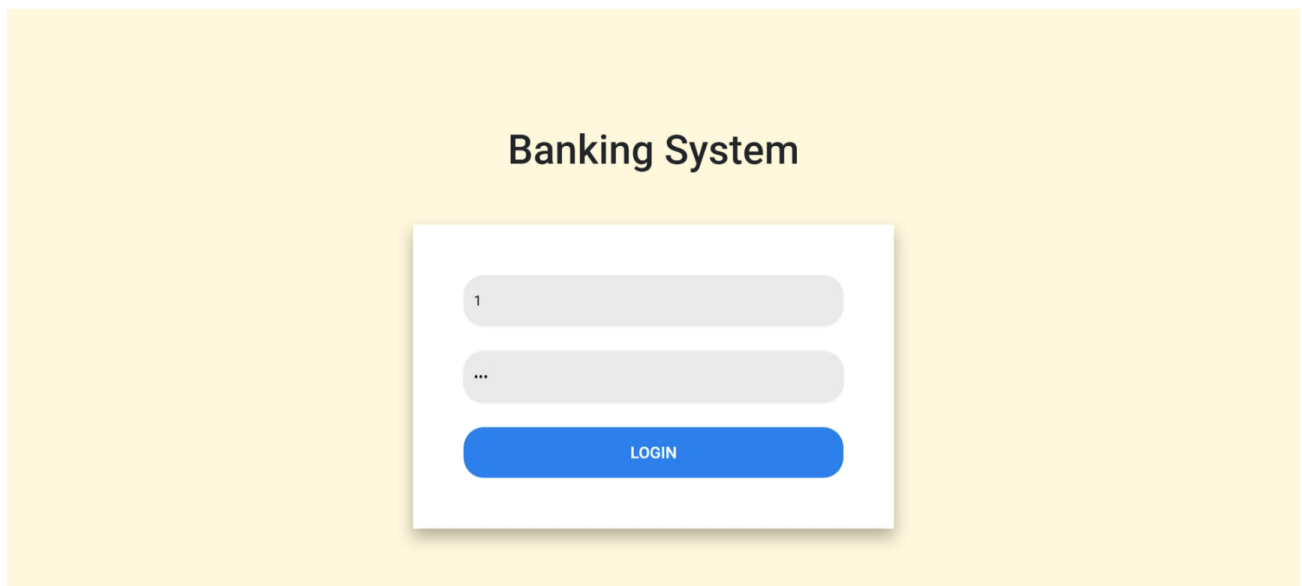


Project 1

User Login Screen - <http://localhost:8080/BankingSystem/index.jsp>



The image shows a user login screen for a banking system. It features a light yellow background. At the top center, the text "Banking System" is displayed in a bold, black font. Below this, there is a white rectangular box with rounded corners and a subtle drop shadow. Inside this box, there are two input fields: the first is labeled "Account Number" and the second is labeled "Password". Both labels are in a small, gray font. Below the input fields is a blue button with rounded corners and the word "LOGIN" in white, uppercase letters.



This image shows the same banking system login screen as above, but with some input in the fields. The "Account Number" field now contains the digit "1". The "Password" field contains three dots "...". The "LOGIN" button remains unchanged.

Main Menu Display - (<http://localhost:8080/BankingSystem/LoginCheck>)

Account Number Logged In: 1

Create Account

Debit or Credit

Transactions

Authorize Credit Card

Logout

Creating Account – (<http://localhost:8080/BankingSystem/createAccount.jsp?>)

Enter Name

Name

Select Date of Birth

dd—yyyy

Enter Address

Address

Enter Email

Email

Choose a type of bank Account:

Salary Account

Create Account

Enter Name

Shubham Kumar

Select Date of Birth

01-Oct-1998

Enter Address

C5A/322B, Janakpuri
New Delhi

Enter Email

kumar.shubham@ivycomptech.com

Choose a type of bank Account:

Salary Account

Create Account

Account Created –
(<http://localhost:8080/BankingSystem/LoginCheck>)

New Account Created with Account Number : 772332616170

Card Number allocated to New Account : 6004090456837989

CVV allocated to New Card : 456

Account Number Logged In: 1

Create Account

Debit or Credit

Transactions

Authorize Credit Card

Logout

Debit or Credit Transaction – (<http://localhost:8080/BankingSystem/debitOrCredit.jsp>)

Make a Transaction

Enter Beneficiary Account Number

Enter Amount

Enter Description

Enter Cheque Number

Debit From

Credit To

Homepage

Make a Transaction

Transaction Successfull!

Enter Beneficiary Account Number

Enter Amount

Enter Description

Enter Cheque Number

Debit From

Credit To

Homepage

Error Display – (<http://localhost:8080/BankingSystem/debitOrCredit.jsp>)

Make a Transaction

Account Number and Amount can't be empty !

Enter Beneficiary Account Number

Enter Amount

Enter Description

Enter Cheque Number

Debit From

Credit To

Homepage

Make a Transaction

Enter Valid Account Number!

Enter Beneficiary Account Number

Enter Amount

Enter Description

Enter Cheque Number

Debit From

Credit To

Homepage

Transaction between Dates – (<http://localhost:8080/BankingSystem/transactions.jsp>)

Date Range

From Date

01-Oct-2021

To Date

31-Oct-2021

Submit

Homepage

(<http://localhost:8080/BankingSystem/transactionsTable.jsp>)

Transactions								
SL. NO.	ACCOUNT NUMBER	BENEFICIARY ACCOUNT NO.	DATE	DESCRIPTION	CHEQUE NO.	WITHDRAW	DEPOSIT	AVAILABLE BALANCE
1	1	56483765809	2021-10-30			0.0	3.1241321E7	-2.5459799E7
2	1	56483765809	2021-10-30			1321.0	0.0	-2.5458478E7
3	1	56483765809	2021-10-30			0.0	7.531312313E9	-7.556770791E9
4	1	56483765809	2021-10-30			32432.0	0.0	6607271.0
5	1	56483765809	2021-10-30			42432.0	0.0	6617271.0
6	1	56483765809	2021-10-30			31321.0	0.0	6606160.0
7	1	56483765809	2021-10-30			0.0	31321.0	6543518.0
8	1	96626266159	2021-10-30			0.0	313213.0	6261626.0
9	1	96626266159	2021-10-30			213213.0	0.0	6788052.0
10	1	557238349934	2021-10-30			0.0	132.0	6574707.0
11	1	557238349933	2021-10-30			0.0	123.0	6574716.0
12	1	557238349933	2021-10-30			0.0	21213.0	6553626.0
13	1	557238349934	2021-10-30			0.0	3213.0	6571626.0
14	1	557238349933	2021-10-30			0.0	121.0	6574718.0
15	1	557238349933	2021-10-30			123.0	0.0	6574962.0
16	1	557238349933	2021-10-30			123.0	0.0	6574962.0
17	1	557238349933	2021-10-30			0.0	123.0	6574716.0
18	1	557238349934	2021-10-30			7987.0	0.0	6582703.0
19	1	557238349934	2021-10-30			3213.0	0.0	6585916.0
20	1	557238349933	2021-10-30			0.0	132.0	6585784.0
21	1	56483765809	2021-10-30			0.0	3131.0	6582653.0
22	1	96626266159	2021-10-30			123323.0	0.0	6705976.0
23	1	96626266159	2021-10-30			123.0	0.0	6706099.0
24	1	96626266159	2021-10-30			123.0	0.0	6706222.0
25	1	96626266159	2021-10-30	132		232.0	0.0	6706454.0
26	1	557238349934	2021-10-30			0.0	57.0	6706397.0
27	1	557238349934	2021-10-30			3245.0	0.0	6709642.0
28	1	557238349934	2021-10-30			12345.0	0.0	6721987.0
29	1	557238349934	2021-10-30	Debit From		0.0	12345.0	6709642.0

Go Back

Homepage

Authorize Credit Card –

(<http://localhost:8080/BankingSystem/authorizeCreditCard.jsp>)

The screenshot shows a web form titled "Authorize Credit Card" on a light yellow background. The form contains several input fields and two buttons. The fields are labeled "Enter Name", "Enter Card Number", "Enter CVV/CVV2", and "Amount". The "Name" field contains the text "Name", the "Credit Card Number" field contains "Credit Card Number", the "CVV" field contains "CVV", and the "Amount" field contains "0". Below the fields are two buttons: a green "Authorize" button and a blue "Homepage" button.

Authorize Credit Card

Enter Name

Name

Enter Card Number

Credit Card Number

Enter CVV/CVV2

CVV

Amount

0

Authorize

Homepage

Card Authorization Successful –

(<http://localhost:8080/BankingSystem/Authorize>)

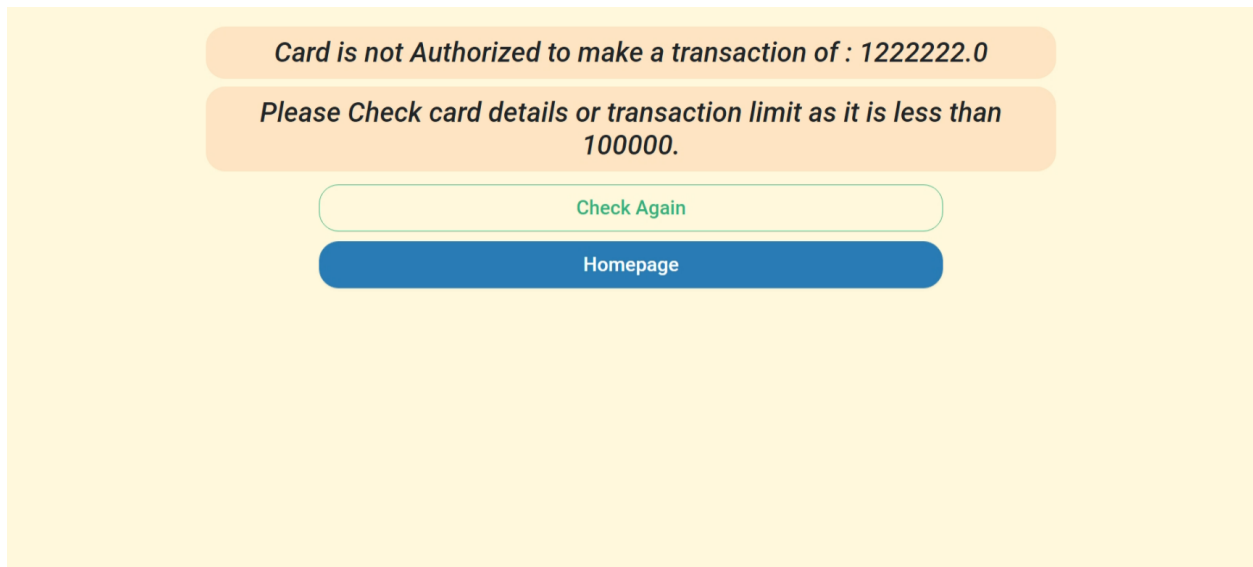
The screenshot shows a web page with a light yellow background. At the top, there is a message in an orange box: "Card is Authorized to make a transaction of : 12090.0". Below this message are two buttons: a green "Check other cards" button and a blue "Homepage" button.

Card is Authorized to make a transaction of : 12090.0

Check other cards

Homepage

Card Authorization Failed – (<http://localhost:8080/BankingSystem/Authorize>)



Card is not Authorized to make a transaction of : 1222222.0

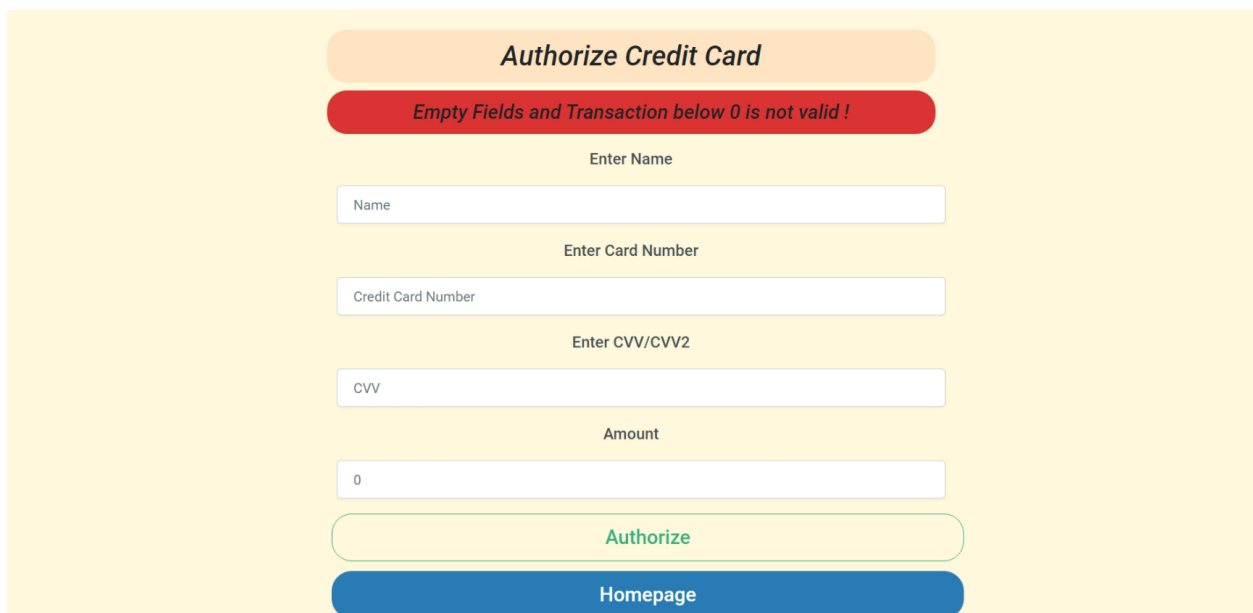
Please Check card details or transaction limit as it is less than 100000.

Check Again

Homepage

This screenshot shows a web interface for card authorization. It features a light yellow background. At the top, there are two orange rounded rectangular boxes. The first box contains the text "Card is not Authorized to make a transaction of : 1222222.0". The second box contains the text "Please Check card details or transaction limit as it is less than 100000.". Below these boxes, there are two buttons: a light green button with the text "Check Again" and a blue button with the text "Homepage".

Error Handling Display – (<http://localhost:8080/BankingSystem/authorizeCreditCard.jsp>)



Authorize Credit Card

Empty Fields and Transaction below 0 is not valid !

Enter Name

Name

Enter Card Number

Credit Card Number

Enter CVV/CVV2

CVV

Amount

0

Authorize

Homepage

This screenshot shows a web interface for authorizing a credit card. It features a light yellow background. At the top, there is an orange rounded rectangular box with the text "Authorize Credit Card". Below this, there is a red rounded rectangular box with the text "Empty Fields and Transaction below 0 is not valid !". Below the red box, there are four input fields, each with a label above it: "Enter Name" (input field contains "Name"), "Enter Card Number" (input field contains "Credit Card Number"), "Enter CVV/CVV2" (input field contains "CVV"), and "Amount" (input field contains "0"). At the bottom, there are two buttons: a light green button with the text "Authorize" and a blue button with the text "Homepage".

Authorize Credit Card

Card Not Found !

Enter Name

Name

Enter Card Number

Credit Card Number

Enter CVV/CVV2

CVV

Amount

0

Authorize

Homepage

Code:

Styles.css

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
body {
  min-height: 100vh;
  background-color: cornsilk;
  display: flex;
  font-family: "Roboto";
}
.container {
  margin: auto;
  width: 500px;
  max-width: 90%;
}
.container h1 {
  text-align: center;
  margin: 50px;
  font-family: "Roboto";
}
.container form {
  width: 100%;
  height: 100%;
  padding: 20px;
  background: white;
  border-radius: 1px;
```

```

    box-shadow: 0 8px 16px rgba(0, 0, 0, 0.3);
    padding-top: 50px;
    padding-right: 50px;
    padding-left: 50px;
    padding-bottom: 50px;
}
.container form .form-control {
    width: 100%;
    height: 50px;
    color: black;
    background: #eaeaea;
    border-radius: 1px;
    border: 0px solid rgb(2, 93, 153);
    margin: 0px 0 25px 0;
    padding: 0 10px;
    font-family: "Roboto";
    border-radius: 20px;
}
.container form .btn {
    margin-left: 50%;
    transform: translateX(-50%);
    width: 100%;
    height: 50px;
    border: none;
    outline: none;
    background: #2d80ec;
    cursor: pointer;
    font-size: 16px;
    text-transform: uppercase;
    color: white;
    border-radius: 1px;
    transition: 0.3s;
    font-family: "Roboto";
    border-radius: 20px;
}
.container form .btn:hover {
    opacity: 0.7;
}
.h1 {
    text-align: center;
    margin-bottom: 24px;
    margin: auto;
    color: #222;
    font-family: "Roboto";
}
::placeholder {
    color: #b8b8b8;
    opacity: 1;
}
.center {
    display: block;
    margin-left: auto;
    margin-right: auto;
}

```

```
.homepage {  
  text-align: center;  
  width: 90vw;  
}  
  
.homepage .btn {  
  width: 50vw;  
  margin: 20px;  
  font-size: 20px;  
  border-radius: 20px;  
}  
  
.homepage h4 {  
  background-color: bisque;  
  border-radius: 20px;  
  padding: 20px;  
  margin: 20px;  
}  
  
.homepage h5 {  
  background-color: rgb(57, 211, 26);  
  border-radius: 20px;  
  padding: 20px;  
  margin: 20px;  
}  
  
.create {  
  text-align: center;  
  width: 50vw;  
  align-items: center;  
  justify-content: center;  
  margin-top: 10px;  
}  
  
.create label {  
  margin: 10px;  
}  
  
.create input {  
  margin: 10px;  
}  
  
.create .btn {  
  width: 50vw;  
  margin: 10px;  
  margin-top: 0%;  
  font-size: 15px;  
  border-radius: 20px;  
}  
  
.dbcd {  
  text-align: center;  
  width: 50vw;  
  align-items: center;  
  justify-content: center;
```

```
    margin-top: 10px;
}

.dbcd h2 {
    font-style: italic;
    background-color: bisque;
    padding: 5px;
    border-radius: 20px;
}

.dbcd label {
    margin: 5px;
}

.dbcd input {
    margin: 5px;
}

.dbcd textarea {
    margin: 5px;
}

.dbcd .btn {
    width: 45vw;
    margin: 10px;
    margin-top: 5px;
    font-size: 12px;
    border-radius: 20px;
}

.dbcd h5 {
    font-style: italic;
    background-color: rgb(219, 51, 51);
    padding: 10px;
    border-radius: 20px;
}

.transactions {
    text-align: center;
    width: 50vw;
    align-items: center;
    justify-content: center;
    margin-top: 20px;
}

.transactions h2 {
    font-style: italic;
    background-color: bisque;
    padding: 10px;
    border-radius: 20px;
}

.transactions input {
    margin: 10px;
}
```

```
.transactions label {  
  margin: 10px;  
}  
  
.transactions .btn {  
  width: 50vw;  
  margin: 10px;  
  margin-top: 5px;  
  font-size: 20px;  
  border-radius: 20px;  
}  
  
.tr-table {  
  text-align: center;  
  width: 100vw;  
  align-items: center;  
  justify-content: center;  
  margin-top: 10px;  
}  
  
.tr-table h3 {  
  font-style: italic;  
  background-color: bisque;  
  padding: 10px;  
  border-radius: 20px;  
}  
  
.tr-table .btn {  
  width: 90vw;  
  margin: 10px;  
  margin-top: 5px;  
  font-size: 20px;  
  border-radius: 20px;  
}  
  
.authorize {  
  text-align: center;  
  width: 50vw;  
  align-items: center;  
  justify-content: center;  
  margin-top: 20px;  
}  
  
.authorize input {  
  margin: 10px;  
}  
  
.authorize label {  
  margin: 5px;  
}  
  
.authorize .btn {  
  width: 50vw;  
  margin: 5px;
```

```
margin-top: 5px;
font-size: 20px;
border-radius: 20px;
}

.authorize h3 {
  font-style: italic;
  background-color: bisque;
  padding: 10px;
  border-radius: 20px;
}

.authorize h5 {
  font-style: italic;
  background-color: rgb(219, 51, 51);
  padding: 10px;
  border-radius: 20px;
}

.authorized {
  text-align: center;
  width: 70vw;
  align-items: center;
  justify-content: center;
  margin-top: 20px;
}

.authorized .btn {
  width: 50vw;
  margin: 5px;
  margin-top: 5px;
  font-size: 20px;
  border-radius: 20px;
}

.authorized h3 {
  font-style: italic;
  background-color: bisque;
  padding: 10px;
  border-radius: 20px;
}

.unauthorized {
  text-align: center;
  width: 70vw;
  align-items: center;
  justify-content: center;
  margin-top: 20px;
}

.unauthorized .btn {
  width: 50vw;
  margin: 5px;
  margin-top: 5px;
  font-size: 20px;
}
```

```

    border-radius: 20px;
}

.unauthorized h3 {
    font-style: italic;
    background-color: bisque;
    padding: 10px;
    border-radius: 20px;
}

```

index.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<title>Banking System</title>
<meta charset="windows-1252" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
<link rel="stylesheet" href="styles.css" type="text/css" />
</head>
<body>
<div class="container-fluid container">
<h1>Banking System</h1>
<%session.invalidate(); %>
<form action="LoginCheck" method="post">
<div class="form-group">
<input
    type="number"
    class="form-control"
    name="accountNumber"
    placeholder="Account Number"
/>
</div>
<div class="form-group">
<input
    type="password"
    class="form-control"
    name="password"
    placeholder="Password"
/>
</div>
<input type="submit" class="btn" value="Login" />
</form>
</div>
</body>
</html>

```

debitOrCredit.jsp


```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="ISO-8859-1" />
    <title>Debit or Credit Amount</title>
    <link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
    <link rel="stylesheet" href="styles.css">
  </head>
  <body>
    <div class="container-fluid dbcd">
      <h2>Make a Transaction</h2>
      <% String message = ""; if(!(request.getSession().getAttribute("message")
      == null)){ message =
      request.getSession().getAttribute("message").toString(); }
      if(!message.isEmpty()){ %>
      <h5><%= message %></h5>
      <%}%>
      <form action="DebitCredit" method="post">
        <div class="form-group">
          <label class="form-label" for="beneficiaryAccountNumber">
            >Enter Beneficiary Account Number</label>
          <input
            type="number"
            class="form-control"
            id="beneficiaryAccountNumber"
            name="beneficiaryAccountNumber"
            placeholder="Beneficiary Account Number"
          />
        </div>

        <div class="form-group">
          <label class="form-label" for="amount">Enter Amount</label>
          <input
            id="amount"
            type="number"
            class="form-control"
            name="amount"
            placeholder="Amount"
            value="0"
          />
        </div>

        <div class="form-group">
          <label class="form-label" for="description">Enter Description</label>
          <textarea
            id="description"
            class="form-control"
            name="description"
            placeholder="Description"
          ></textarea>
        </div>
      </form>
    </div>
  </body>
</html>

```

```

<div class="form-group">
  <label class="form-label" for="chequeNo">Enter Cheque Number</label>
  <input
    id="chequeNo"
    type="text"
    class="form-control"
    name="chequeNo"
    placeholder="Cheque No."
    value=""
  />
</div>

<input
  type="submit"
  class="btn btn-outline-success"
  name="submitButton"
  value="Debit From"
/>
<input
  type="submit"
  class="btn btn-outline-danger"
  name="submitButton"
  value="Credit To"
/>
</form>
<form action="homepage.jsp">
  <input type="submit" class="btn btn-info" name="submitButton"
value="Homepage" />
</form>
</div>
</body>
</html>

```

Transactions.jsp

```

<%@page import="java.sql.ResultSet"%> <%@ page language="java"
contentType="text/html; charset=ISO-8859-1" pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>
<html>
  <head>
    <meta charset="ISO-8859-1" />
    <title>All Transactions</title>
    <link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
    <link rel="stylesheet" href="styles.css">
  </head>
  <body>
    <div class="container-fluid transactions">
      <form method="post" action="transactionsTable.jsp">
        <h2>Date Range</h2>
        <div class="form-group">
          <label class="form-label" for="fromDate">From Date</label>
          <input

```

```

        type="date"
        class="form-control"
        id="fromDate"
        name="fromDate"
        placeholder="From Date"
    />
</div>
<div class="form-group">
    <label class="form-label" for="toDate">To Date</label>
    <input
        type="date"
        class="form-control"
        id="toDate"
        name="toDate"
        placeholder="To Date"
    />
</div>

    <input type="submit" class="btn btn-outline-success" value="Submit" />
</form>
<form action="homepage.jsp">
    <input type="submit" class="btn btn-info" name="submitButton"
value="Homepage" />
</form>
</div>
</body>
</html>

```

transactionsTable.jsp

```

<%@page import="java.sql.DriverManager"%> <%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Statement"%> <%@page import="java.sql.Connection"%> <%
String from_date=request.getParameter("fromDate"); String
to_date=request.getParameter("toDate"); String accountNumber =
request.getSession().getAttribute("session_accountNumber").toString(); String
driver = "com.mysql.cj.jdbc.Driver"; String connectionUrl =
"jdbc:mysql://localhost:3306/"; String database = "bank"; String userid =
"root"; String password = "kumar.shubham"; try { Class.forName(driver); } catch
(ClassNotFoundException e) { e.printStackTrace(); } Connection connection =
null; Statement statement = null; ResultSet resultSet = null; %>
<!DOCTYPE html>

<html>
<head>
    <link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <div class="container-fluid tr-table">
        <h3>Transactions</h3>
        <table class="table table-hover" border="1">
            <thead>
                <tr>

```

```

        <th>SL. No.</th>
        <th>Account Number</th>
        <th>Beneficiary Account No.</th>
        <th>Date</th>
        <th>Description</th>
        <th>Cheque No.</th>
        <th>Withdraw</th>
        <th>Deposit</th>
        <th>Available Balance</th>
    </tr>
</thead>
<tbody>
    <% try{ connection =
        DriverManager.getConnection(connectionUrl+database, userid, password);
        statement=connection.createStatement(); String sql ="select * from
transactions where date between '" + from_date + "' and '" + to_date + "' and
accountNumber = '" + accountNumber + "'";
        resultSet = statement.executeQuery(sql); int i=1; while(resultSet.next()) {
%>

        <tr>
            <td><%= i %></td>
            <%i++; %>
            <td><%=resultSet.getString("accountNumber") %></td>
            <td><%=resultSet.getString("beneficiaryAccountNumber") %></td>
            <td><%=resultSet.getString("date") %></td>
            <td><%=resultSet.getString("description") %></td>
            <td><%=resultSet.getString("chequeNo") %></td>
            <td><%=resultSet.getDouble("withdraw") %></td>
            <td><%=resultSet.getDouble("deposit") %></td>
            <td><%=resultSet.getDouble("balance") %></td>
        </tr>
        <% } connection.close(); } catch (Exception e) { e.printStackTrace();
        } %>
    </tbody>
</table>

<form method="post" action="transactions.jsp">
    <input type="submit" class="btn btn-outline-success" value="Go Back" />
</form>

<form method="post" action="homepage.jsp">
    <input type="submit" class="btn btn-info" value="Homepage" />
</form>
</div>
</body>
</html>

```

createAccount.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>

```

```

<html>
<head>
  <meta charset="ISO-8859-1" />
  <title>Create Account</title>
  <link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
  <link rel="stylesheet" href="styles.css">
</head>

<body>
  <div class="container-fluid create">
    <form action="CreateAccount" method="post">
      <div class="form-group">
        <label class="form-label" for="name">Enter Name</label>
        <input
          type="text"
          class="form-control"
          name="name"
          placeholder="Name"
        />
      </div>

      <div class="form-group">
        <label class="form-label" for="dob">Select Date of Birth</label>
        <input
          type="date"
          class="form-control"
          id="dob"
          name="dob"
          placeholder="Date of Birth"
        />
      </div>

      <div class="form-group">
        <label class="form-label" for="address">Enter Address</label>
        <textarea
          class="form-control"
          id="address"
          name="address"
          placeholder="Address"
        ></textarea>
      </div>

      <div class="form-group">
        <label class="form-label" for="email">Enter Email</label>
        <input
          type="email"
          id="email"
          class="form-control"
          name="email"
          placeholder="Email"
        />
      </div>

      <div class="form-group">

```

```

        <label class="form-Label" for="accountType">Choose a type of bank
Account:</label>
        <select name="accountType" class="form-select mb-3" id="accountType">
            <option value="salary">Salary Account</option>
            <option value="saving">Savings Account</option>
            <option value="current">Current Account</option>
        </select>
    </div>
    <input type="submit" class="btn btn-primary" value="Create Account" />
</form>
</div>
</body>
</html>

```

Homepage.jsp

```

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

<!DOCTYPE html>

<%@page import="java.sql.*" %> <% String BankAccount = ""; if
(request.getParameter("BankAccountNumber") == null) { BankAccount = ""; } else {
BankAccount = request.getParameter("BankAccountNumber"); } String accountNumber
= request.getSession().getAttribute("session_accountNumber").toString(); String
password = request.getSession().getAttribute("session_password").toString();
String userAccountNumberCreated="";
if(!(request.getSession().getAttribute("userAccountCreated") == null)){
userAccountNumberCreated =
request.getSession().getAttribute("userAccountCreated").toString(); } String
userCardAllocated="";
if(!(request.getSession().getAttribute("userCardAllocated") == null)){
userCardAllocated =
request.getSession().getAttribute("userCardAllocated").toString(); } String
userCvv=""; if(!(request.getSession().getAttribute("userCvv") == null)){ userCvv
= request.getSession().getAttribute("userCvv").toString(); } %>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Home Page</title>
    <link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <div class="container-fluid homepage" >
        <%if(!userAccountNumberCreated.isEmpty()){%>
        <h5>
            New Account Created with Account Number : <%= userAccountNumberCreated
            %>
        </h5>
        <%}%> <%if(!userCardAllocated.isEmpty()){%>
        <h5>Card Number allocated to New Account : <%= userCardAllocated %></h5>
        <%}%> <%if(!userCvv.isEmpty()){%>
        <h5>CVV allocated to New Card : <%= userCvv %></h5>

```

```

<%=}%> <%if(accountNumber.isEmpty()) {%>
<h5>Go Back and Login Again !</h5>
<%=}%> else { %>
<h4>Account Number Logged In: <%= accountNumber %></h4>
<%=}%>
<form action="createAccount.jsp" method="Get">
  <% if (accountNumber.isEmpty()) {%>
    <input
      type="submit"
      class="btn btn-lg btn-primary"
      value="Create Account"
      disabled
    />
  <%=}%> else {%>
    <input type="submit" class="btn btn-primary" value="Create Account" />
  <%=}%>
</form>
<form action="debitOrCredit.jsp" method="Post">
  <% if (accountNumber.isEmpty()) {%>
    <input
      type="submit"
      class="btn btn-secondary btn-lg"
      value="Debit or Credit"
      disabled
    />
  <%=}%> else {%>
    <input type="submit" class="btn btn-outline-primary" value="Debit or Credit"
  />
  <%=}%>
</form>
<form action="transactions.jsp" method="Post">
  <% if (accountNumber.isEmpty()) {%>
    <input
      type="submit"
      class="btn btn-secondary btn-lg"
      value="Transactions"
      disabled
    />
  <%=}%> else {%>
    <input type="submit" class="btn btn-outline-primary" value="Transactions" />
  <%=}%>
</form>
<form action="authorizeCreditCard.jsp" method="Post">
  <% if (accountNumber.isEmpty()) {%>
    <input
      type="submit"
      class="btn btn-secondary btn-lg"
      value="Authorize Credit Card"
      disabled
    />
  <%=}%> else {%>
    <input
      type="submit"
      class="btn btn-outline-primary"
      value="Authorize Credit Card"

```

```

        />
        <%}%>
    </form>
    <form action="index.jsp" method="Get">
        <input type="submit" class="btn btn-danger" value="Logout" />
    </form>
</div>
</body>
</html>

```

authorizeCreditCard.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="ISO-8859-1" />
        <title>Authorize Credit Card</title>
        <link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
        <link rel="stylesheet" href="styles.css">
    </head>
    <body>
        <div class="container-fluid authorize">
            <h3>Authorize Credit Card</h3>
            <% String message = ""; if(!(request.getSession().getAttribute("message")
            == null)){ message =
            request.getSession().getAttribute("message").toString(); }
            if(!message.isEmpty()){ %>
            <h5><%= message %></h5>
            <%}%>
            <form action="Authorize" method="post">
                <div class="form-group">
                    <label class="form-label" class="form-label class="form-label""
for="name">Enter Name</label>
                    <input
                        type="text"
                        class="form-control"
                        name="name"
                        placeholder="Name"
                    />
                </div>

                <div class="form-group">
                    <label class="form-label" class="form-label" for="cardNumber">Enter Card
Number</label>
                    <input
                        type="number"
                        class="form-control"
                        id="cardNumber"
                        name="cardNumber"
                        placeholder="Credit Card Number"
                    />
                </div>
            </form>
        </div>
    </body>
</html>

```



```

</div>

<div class="form-group">
  <label class="form-label" for="cvv">Enter CVV/CVV2</label>
  <input
    type="number"
    class="form-control"
    id="cvv"
    name="cvv"
    placeholder="CVV"
  />
</div>

<div class="form-group">
  <label class="form-label" for="amount">Amount</label>
  <input
    type="number"
    class="form-control"
    name="amount"
    id="amount"
    placeholder="0"
  />
</div>
<input type="submit" class="btn btn-outline-success" value="Authorize" />
</form>
<form action="homepage.jsp">
  <input type="submit" class="btn btn-info" name="submitButton"
value="Homepage" />
</form>
</div>
</body>
</html>

```

Authorized.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="ISO-8859-1" />
    <title>Card Authorized</title>
    <link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <div class="container-fluid authorized">
      <h3>
        Card is Authorized to make a transaction of :
        <%=request.getSession().getAttribute("session_amount").toString() %>
      </h3>

      <form method="post" action="authorizeCreditCard.jsp">

```

```

        <input type="submit" class="btn btn-outline-success" value="Check other
cards" />
    </form>
    <form action="homepage.jsp">
        <input type="submit" class="btn btn-info" name="submitButton"
value="Homepage" />
    </form>
</div>
</body>
</html>

```

unAuthorized.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1" />
<title>Card Unauthorized</title>
<link rel="stylesheet" href="bootstrap.min.css" type="text/css" />
<link rel="stylesheet" href="styles.css" />
</head>
<body>
<div class="container-fluid unauthorized">
<h3>
    Card is not Authorized to make a transaction of :
    <%=request.getSession().getAttribute("session_amount").toString() %>
</h3>
<h3>
    Please Check card details or transaction limit as it is less than
    100000.
</h3>
<form method="post" action="authorizeCreditCard.jsp">
    <input type="submit" class="btn btn-outline-success" value="Check Again" />
</form>
<form action="homepage.jsp">
    <input type="submit" class="btn btn-info" name="submitButton"
value="Homepage" />
</form>
</div>
</body>
</html>

```

LoginCheck.java

```

package com.user;

import java.io.IOException;

import java.io.PrintWriter;

```

```
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.sql.*;

import javax.servlet.RequestDispatcher;
import javax.servlet.http.HttpSession;
import javax.servlet.http.Cookie;

@WebServlet(urlPatterns = { "/LoginCheck" })
public class LoginCheck extends HttpServlet {
    private static final long serialVersionUID = 1L;

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        // super.doGet(request, response);
        processRequest(request, response);
    }

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        // super.doPost(request, response);
        processRequest(request, response);
    }
}
```

```
void processRequest(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

    HttpSession session = request.getSession(true);

    response.setContentType("text/html;charset=UTF-8");

    try (PrintWriter out = response.getWriter()) {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");

            String url = "jdbc:mysql://localhost:3306/bank?autoReconnect=true&useSSL=false";

            String user = "root";

            String password = "kumar.shubham";

            Connection connection = null;

            Statement statement = null;

            connection = DriverManager.getConnection(url, user, password);

            String query = "SELECT * FROM user_pass";

            statement = connection.createStatement();

            String userAccountNumber = request.getParameter("accountNumber");

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

            if (userAccountNumber.isEmpty() || userPassword.isEmpty()) {

                RequestDispatcher dispatcher = request.getRequestDispatcher("index.jsp");

                dispatcher.include(request, response);

                out.println("<script type=\"text/javascript\">");

                out.println("alert('Empty Account Number or Password or Not Authorized');");

                out.println("</script>");

            } else {

                int flag = 0;
```

```

String correctAccountNumber = "", correctPassword = "";

ResultSet resultSet = statement.executeQuery(query);

System.out.println(userAccountNumber);

while (resultSet.next()) {

    if (resultSet.getString("accountNumber").equals(userAccountNumber)
        && resultSet.getString("password").equals(userPassword)) {

        flag++;

        correctAccountNumber = resultSet.getString("accountNumber");

    }

}

statement.close();

if (flag != 0) {

    /*
     * if (session.isNew() == false) { session.invalidate(); session =
     * request.getSession(true); }
     */

    session.setAttribute("session_accountNumber", correctAccountNumber);

    session.setAttribute("session_password", correctPassword);

    Cookie cookie_ID = new Cookie("cookie_accountNumber", correctAccountNumber);

    cookie_ID.setMaxAge(365 * 24 * 60 * 60);

    cookie_ID.setPath("/");

    response.addCookie(cookie_ID);

    request.setAttribute("BankAccountNumber", correctAccountNumber);

    RequestDispatcher dispatcher = request.getRequestDispatcher("homepage.jsp");

    dispatcher.forward(request, response);

} else {

```

```
package com.user;
```

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.Statement;
import java.time.LocalDate;
```

```
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

@WebServlet(urlPatterns = { "/DebitCredit" })
public class DebitCredit extends HttpServlet {

    private static final long serialVersionUID = 1L;

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

        HttpSession session = request.getSession(true);
        response.setContentType("text/html;charset=UTF-8");

        try (PrintWriter out = response.getWriter()) {
            try {

                Class.forName("com.mysql.cj.jdbc.Driver");

                String url =
"jdbc:mysql://localhost:3306/bank?autoReconnect=true&useSSL=false";

                String user = "root";

                String password = "kumar.shubham";

                Connection connection = null;

                Statement statement = null;
```

```
connection = DriverManager.getConnection(url, user, password);

statement = connection.createStatement();

String beneficiaryAccountNumber =
request.getParameter("beneficiaryAccountNumber").toString();

Double amount = 0.0;

if (!request.getParameter("amount").toString().isEmpty()) {
    amount =
Double.parseDouble(request.getParameter("amount"));
}

String description = request.getParameter("description");
String chequeNo = request.getParameter("chequeNo");

LocalDate todayLocalDate = LocalDate.now();
java.sql.Date currentDate = java.sql.Date.valueOf(todayLocalDate);

Double deposit = 0.0;
Double withdraw = 0.0;

if (beneficiaryAccountNumber.isEmpty() || amount == 0.0) {
    session.setAttribute("message", "Account Number and
Amount can't be empty !");

    RequestDispatcher dispatcher =
request.getRequestDispatcher("debitOrCredit.jsp");

    dispatcher.include(request, response);
    dispatcher.forward(request, response);
    out.println("<script type=\"text/javascript\">");
```



```

        out.println("alert('Account Number and Amount can't be
empty !');");

        out.println("</script>");

    } else {
        String sql = "CREATE TABLE IF NOT EXISTS " + "transactions"
            + " (accountNumber      varchar(200),"
            + " beneficiaryAccountNumber
varchar(200)," + " date      date,"
            + " description      varchar(200)," + "
chequeNo  VARCHAR(100),"
            + " withdraw  double," + " deposit
double," + " balance double)";

        statement = connection.createStatement();
        Boolean status = statement.execute(sql);

        sql = "select * from account_details where accountNumber="
+ beneficiaryAccountNumber + """;

        ResultSet rs = statement.executeQuery(sql);

        if (rs.next()) {
            sql = "Select accountNumber, balance from
account_details where accountNumber="
            +
            session.getAttribute("session_accountNumber") + """;
            rs = statement.executeQuery(sql);

            String accountNumber = "";
            Double balance = 0.0;

```

```

while (rs.next()) {
    accountNumber =
rs.getString("accountNumber");

    balance = rs.getDouble("balance");
}

sql = "Select accountNumber, balance from
account_details where accountNumber="

        + beneficiaryAccountNumber + """;
rs = statement.executeQuery(sql);

Double beneficiaryBalance = 0.0;
while (rs.next()) {
    beneficiaryBalance = rs.getDouble("balance");
}

String submitButton =
request.getParameter("submitButton");

if (submitButton.equals("Debit From")) {
    balance += amount;
    beneficiaryBalance -= amount;
    withdraw = amount;
} else if (submitButton.equals("Credit To")) {
    balance -= amount;
    beneficiaryBalance += amount;
    deposit = amount;
} else {
    RequestDispatcher dispatcher =
request.getRequestDispatcher("debitOrCredit.jsp");

```

```
dispatcher.include(request, response);  
}
```

```
PreparedStatement ps = connection  
    .prepareStatement("UPDATE  
account_details SET balance = ? where accountNumber = ?;");
```

```
ps.setDouble(1, balance);  
ps.setString(2, accountNumber);  
status = ps.execute();
```

```
ps.setDouble(1, beneficiaryBalance);  
ps.setString(2, beneficiaryAccountNumber);  
status = ps.execute();
```

```
ps = connection.prepareStatement(  
    "insert into  
transactions(accountNumber, beneficiaryAccountNumber, date, description, chequeNo, withdraw,  
deposit, balance) values(?,?,?,?,?,?,?,?)");
```

```
ps.setString(1, accountNumber);  
ps.setString(2, beneficiaryAccountNumber);  
ps.setDate(3, currentDate);  
ps.setString(4, description);  
ps.setString(5, chequeNo);  
ps.setDouble(6, withdraw);  
ps.setDouble(7, deposit);  
ps.setDouble(8, balance);
```

```
status = ps.execute();
```

```

String query = "SELECT balance from account_details
where accountNumber = " + accountNumber;

Boolean checkBalance1 = false;

ResultSet resultSet = statement.executeQuery(query);

while (resultSet.next()) {
    if (resultSet.getDouble("balance") == balance)
    {
        checkBalance1 = true;
    }
}

query = "SELECT balance from account_details where
accountNumber = " + beneficiaryAccountNumber;

Boolean checkBalance2 = false;

resultSet = statement.executeQuery(query);

while (resultSet.next()) {
    if (resultSet.getDouble("balance") ==
beneficiaryBalance) {
        checkBalance2 = true;
    }
}

statement.close();

ps.close();

if (checkBalance1 && checkBalance2) {
    session.setAttribute("message", "Transaction
Successfull!");
}

```

```

RequestDispatcher dispatcher =
request.getRequestDispatcher("debitOrCredit.jsp");

dispatcher.include(request, response);
dispatcher.forward(request, response);
out.println("<script type='text/javascript'>");
out.println("alert('Transaction Successfull!');");
out.println("</script>");

} else {

session.setAttribute("message", "Transaction
Failed, Try Again!");

RequestDispatcher dispatcher =
request.getRequestDispatcher("debitOrCredit.jsp");

dispatcher.include(request, response);
dispatcher.forward(request, response);
out.println("<script type='text/javascript'>");
out.println("alert('Transaction Failed, Try
Again!');");

out.println("</script>");

}

} else {

session.setAttribute("message", "Enter Valid Account
Number!");

RequestDispatcher dispatcher =
request.getRequestDispatcher("debitOrCredit.jsp");

dispatcher.include(request, response);
dispatcher.forward(request, response);
out.println("<script type='text/javascript'>");
out.println("alert('Enter Valid Account Number!');");
out.println("</script>");

}

```

```

    }

    } catch (Exception e) {
        System.err.println("Got an exception! ");
        e.printStackTrace();
    }
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

}

```

CreateAccount.java

```
package com.user;
```

```
import java.io.IOException;
```

```
import java.io.PrintWriter;
```

```
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.sql.*;
import java.util.*;
import javax.servlet.RequestDispatcher;
import javax.servlet.http.HttpSession;

@WebServlet(urlPatterns = { "/CreateAccount" })
public class CreateAccount extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public String generateRandomNumbers(int n) {
        Random rnd = new Random();
        char[] digits = new char[n];
        digits[0] = (char) (rnd.nextInt(9) + '1');
        for (int i = 1; i < digits.length; i++) {
            digits[i] = (char) (rnd.nextInt(10) + '0');
        }
        return (new String(digits));
    }

    void processRequest(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
        HttpSession session = request.getSession(true);
        response.setContentType("text/html;charset=UTF-8");
    }
}
```

```
try (PrintWriter out = response.getWriter()) {  
    try {  
        Class.forName("com.mysql.cj.jdbc.Driver");  
        String url =  
"jdbc:mysql://localhost:3306/bank?autoReconnect=true&useSSL=false";  
        String user = "root";  
        String password = "kumar.shubham";  
  
        Connection connection = null;  
        Statement statement = null;  
  
        connection = DriverManager.getConnection(url, user, password);  
  
        statement = connection.createStatement();  
  
        String userName = request.getParameter("name");  
        String userDOB = request.getParameter("dob").toString();  
        String userAdress = request.getParameter("address");  
        String userEmail = request.getParameter("email").toString();  
        String userAccountNumber = generateRandomNumbers(12);  
        String userCvv = generateRandomNumbers(3);  
        String userCardNumber = generateRandomNumbers(16);  
        String userAccountType =  
request.getParameter("accountType").toString();  
  
        if (userName.isEmpty() || userDOB.isEmpty() || userAdress.isEmpty()  
|| userEmail.isEmpty()  
|| userAccountType.isEmpty()) {
```



```

        RequestDispatcher dispatcher =
request.getRequestDispatcher("homepage.jsp");

        dispatcher.include(request, response);

        out.println("<script type=\"text/javascript\">");
        out.println("alert('All the fields are mandatory, Fill All of Them
!");");

        out.println("</script>");

    } else {

        String sqlCreate = "CREATE TABLE IF NOT EXISTS " +
"user_pass"

                                + " (accountNumber      Varchar(100)," + "
password VARCHAR(100))";

        Boolean status = statement.execute(sqlCreate);

        sqlCreate = "CREATE TABLE IF NOT EXISTS " +
"account_details" + " (name      varchar(200),"

                                + " dob      varchar(100)," + " address
varchar(200),"

                                + " email      varchar(200)," + "

        typeOfAccount  VARCHAR(100),"

                                + " accountNumber  VARCHAR(100)," + "

        cardNumber  VARCHAR(100),"

                                + " cvv  VARCHAR(10)," + " balance
double)";

        statement = connection.createStatement();
        status = statement.execute(sqlCreate);

```

```

        PreparedStatement ps = connection.prepareStatement(
            "insert into account_details(name, dob,
address, email, typeofAccount, accountNumber, cardNumber, cvv, balance) values(?,?,?,?,?,?,?,?,?)");

        ps.setString(1, userName);
        ps.setString(2, userDOB);
        ps.setString(3, userAdress);
        ps.setString(4, userEmail);
        ps.setString(5, userAccountType);
        ps.setString(6, userAccountNumber);
        ps.setString(7, userCardNumber);
        ps.setString(8, userCvv);
        ps.setDouble(9, 0.0);

        status = ps.execute();

        ps = connection.prepareStatement("insert into
user_pass(accountNumber, password) values(?,?)");

        ps.setString(1, userAccountNumber);
        ps.setString(2, "");

        status = ps.execute();

        int flag = 0;

        String query = "SELECT accountNumber from account_details
where accountNumber = "

            + userAccountNumber;

        String correctAccountNumber = "";

        ResultSet resultSet = statement.executeQuery(query);

```

```
        while (resultSet.next()) {  
            System.out.println("inside loop");  
  
            if  
(resultSet.getString("accountNumber").equals(userAccountNumber)) {  
                flag++;  
                correctAccountNumber =  
resultSet.getString("accountNumber");  
            }  
        }  
        statement.close();  
        ps.close();  
  
        if (flag != 0) {  
  
            session.setAttribute("userAccountCreated",  
correctAccountNumber);  
  
            session.setAttribute("userCardAllocated",  
userCardNumber);  
  
            session.setAttribute("userCvv", userCvv);  
  
            RequestDispatcher dispatcher =  
request.getRequestDispatcher("homepage.jsp");  
            dispatcher.include(request, response);  
            out.println("<script type=\"text/javascript\">");  
            out.println("alert('Account Created!');");  
            out.println("</script>");  
            dispatcher.forward(request, response);  
        } else {  
            RequestDispatcher dispatcher =  
request.getRequestDispatcher("homepage.jsp");
```

```

        dispatcher.include(request, response);
        out.println("<script type=\"text/javascript\">");
        out.println("alert(`Account not created!`);");
        out.println("</script>");
    }

}

} catch (Exception e) {
    System.err.println("Got an exception! ");
    e.printStackTrace();
}
}

}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}
}

```

Authorize.java

```
package com.user;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

@WebServlet(urlPatterns = { "/Authorize" })
public class Authorize extends HttpServlet {
    private static final long serialVersionUID = 1L;

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        // super.doGet(request, response);
        processRequest(request, response);
    }
}
```

@Override

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)  
    throws ServletException, IOException {  
    // super.doPost(request, response);  
    processRequest(request, response);  
}
```

```
void processRequest(HttpServletRequest request, HttpServletResponse response) throws  
ServletException, IOException {
```

```
    HttpSession session = request.getSession(true);
```

```
    response.setContentType("text/html;charset=UTF-8");
```

```
try (PrintWriter out = response.getWriter()) {
```

```
    try {
```

```
        Class.forName("com.mysql.cj.jdbc.Driver");
```

```
        String url = "jdbc:mysql://localhost:3306/bank?autoReconnect=true&useSSL=false";
```

```
        String user = "root";
```

```
        String password = "kumar.shubham";
```

```
        Connection connection = null;
```

```
        Statement statement = null;
```

```
        connection = DriverManager.getConnection(url, user, password);
```

```
        String query = "";
```

```
        statement = connection.createStatement();
```

```
String name = request.getParameter("name");
String cardNumber = request.getParameter("cardNumber");
String cvv = request.getParameter("cvv");
Double amount = 0.0;
if (!request.getParameter("amount").toString().isEmpty()) {
    amount = Double.parseDouble(request.getParameter("amount"));
}

if (name.isEmpty() || cardNumber.isEmpty() || cvv.isEmpty() || amount <= 0) {

    RequestDispatcher dispatcher = request.getRequestDispatcher("authorizeCreditCard.jsp");
    dispatcher.include(request, response);

    out.println("<script type=\"text/javascript\">");
    out.println("alert('Empty Fields and Transaction below 0 is not valid !');");
    out.println("</script>");
    session.setAttribute("message", "Empty Fields and Transaction below 0 is not valid !");
    dispatcher.forward(request, response);

} else {
    int flag = 0;

    query = "select * from account_details where cardNumber=" + cardNumber;
    ResultSet resultSet = statement.executeQuery(query);

    if (!resultSet.next()) {
        RequestDispatcher dispatcher =
request.getRequestDispatcher("authorizeCreditCard.jsp");
        dispatcher.include(request, response);
```

```
        out.println("<script type=\"text/javascript\">");
        out.println("alert('Card Not Found !');");
        out.println("</script>");
        session.setAttribute("message", "Card Not Found !");
        dispatcher.forward(request, response);
    }
    resultSet = statement.executeQuery(query);
    while (resultSet.next()) {

        if (resultSet.getString("cardNumber").equals(cardNumber)
            && resultSet.getString("cvv").equals(cvv) && amount > 0 && amount <= 100000) {
            flag++;
        }
    }
    session.setAttribute("session_amount", amount);
    statement.close();

    System.out.println(flag);

    if (flag != 0) {
        RequestDispatcher dispatcher = request.getRequestDispatcher("authorized.jsp");
        dispatcher.include(request, response);
    } else {
        RequestDispatcher dispatcher = request.getRequestDispatcher("unAuthorized.jsp");
        dispatcher.include(request, response);
    }
}
```



```
    } catch (Exception e) {  
        System.err.println("Got an exception! ");  
        e.printStackTrace();  
    }  
}  
}  
  
}
```