TASK 7

Web Database Application and its Execution

7.1 Web database application source program and screenshots showing successful compilation

DataHandler.java Code

```
// Catherine Donner
// DBMS Individual Project
// Task 7 DataHandler Java file


package jsp_azure_test;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
public class Donner_Catherine_IP_Task7_DataHandler {
        private Connection conn;
        // Azure SQL connection credentials
        private String server = "donn0031-sql-server.database.windows.net";
        private String database = "cs-dsa-4513-sql-db";
        private String username = "donn0031";
        private String password = "RoSaLiNa5000$";
        // Resulting connection string
        final private String url =
```

```
        String.format("jdbc:sqlserver://%s:1433;database=%s;user=%s;password=%s;encrypt=true;trustServerCertificate=false;hostNameInCertificate=*.database.windows.net;loginTimeout=30;",
                                        server, database, username, password);
        // Initialize and save the database connection
        private void getDBConnection() throws SQLException {
                if (conn != null) {
                        return;
                }
                this.conn = DriverManager.getConnection(url);
        }
        // Return the result of selecting customer data within category range
        public ResultSet getAllCustomersinRange(int lower_category, int higher_category) throws SQLException {
                getDBConnection();
                final String sqlQuery = "SELECT * FROM Customer WHERE Category BETWEEN ? AND ? ORDER BY Name ASC;";
                final PreparedStatement stmt = conn.prepareStatement(sqlQuery);
                // Replace the '?' in the above statement with the given attribute values
                stmt.setInt(1, lower_category);
                stmt.setInt(2, higher_category);
                return stmt.executeQuery();
        }
        // Insert a record into the Customer table
        public boolean addCustomer(
                        String name, String address, int category) throws SQLException {
```

```
                getDBConnection(); // Prepare the database connection
                // Prepare the SQL statement
                final String sqlQuery =
                                "INSERT INTO Customer " + "(Name, Address, Category) " +
"VALUES " + "(?, ?, ?)";
                final PreparedStatement stmt = conn.prepareStatement(sqlQuery);
                // Replace the '?' in the above statement with the given attribute values
                stmt.setString(1, name);
                stmt.setString(2, address);
                stmt.setInt(3, category);
                // Execute the query, if only one record is updated, then we indicate success by
returning true
                return stmt.executeUpdate() == 1;
        }
}
```

Add_customer_form.jsp Code

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Add Customer</title>
  </head>
  <body>
    <h2>Add Customer</h2>
    <!--
      Form for collecting user input for the new Customer record.
```

```
   Upon form submission, add_customer.jsp file will be invoked.
-->
<form action="Donner_Catherine_IP_Task7_add_customer.jsp">
   <!-- The form organized in an HTML table for better clarity. -->
   <table border=1>
      <tr>
         <th colspan="2">Enter Customer Data:</th>
      </tr>
      <tr>
         <td>Customer Name </td>
         <td><div style="text-align: center;">
         <input type=text name=name>
         </div></td>
      </tr>
      <tr>
         <td>Customer Address </td>
         <td><div style="text-align: center;">
         <input type=text name=address>
         </div></td>
      </tr>
      <tr>
         <td>Customer Category (1-10) </td>
         <td><div style="text-align: center;">
         <input type=text name=category>
         </div></td>
```

```
            </tr>
            <tr>
               <td><div style="text-align: center;">
               <input type=reset value=Clear>
               </div></td>
               <td><div style="text-align: center;">
               <input type=submit value=Insert>
               </div></td>
            </tr>
         </table>
      </form>
   </body>
</html>
```

Add_customer.jsp Code (called when add_customer_form is invoked)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Query Result</title>
</head>
   <body>
   <%@page import="jsp_azure_test.Donner_Catherine_IP_Task7_DataHandler"%>
```

```
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Array"%>
<%
// The handler is the one in charge of establishing the connection.
Donner_Catherine_IP_Task7_DataHandler handler = new
Donner_Catherine_IP_Task7_DataHandler();

// Get the attribute values passed from the input form.
String name = request.getParameter("name");
String address = request.getParameter("address");
String category_str = request.getParameter("category");

/*
 * Checks if the user has not filled out name, address, and category
 */
if (name.equals("") || address.equals("") || category_str.equals("")) {
   response.sendRedirect("Donner_Catherine_IP_Task7_add_customer_form.jsp");
} else {
   int category = Integer.parseInt(category_str);

   // Now perform the query with the data from the form.
   boolean success = handler.addCustomer(name, address, category);
   if (!success) { // Something went wrong
      %>
         <h2>There was a problem inserting the customer record</h2>
```

```
            <%
        } else { // Confirm success to the user
            %>
            <h2>New Customer:</h2>

            <ul>
                <li>Name: <%=name%></li>
                <li>Address: <%=address%></li>
                <li>Category: <%=category%></li>
            </ul>

            <h2>Was successfully inserted.</h2>
            <%
        }
    }
    %>
    </body>
</html>
```

Get_customers_form.jsp Code

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>Request Customer Data in Category Range</title>
    </head>
```

```
<body>
   <h2>Request Customer Data in Category Range</h2>
   <!--
      Form for collecting user input for the customers in category range.
      Upon form submission, get_all_customers.jsp file will be invoked.
   -->
   <form action="Donner_Catherine_IP_Task7_get_all_customers.jsp">
      <!-- The form organized in an HTML table for better clarity. -->
      <table border=1>
         <tr>
            <th colspan="2">Enter Category Range:</th>
         </tr>
         <tr>
            <td>Lower Category Limit </td>
            <td><div style="text-align: center;">
            <input type=text name=lower_category>
            </div></td>
         </tr>
         <tr>
            <td>Higher Category Limit </td>
            <td><div style="text-align: center;">
            <input type=text name=higher_category>
            </div></td>
         </tr>
         <tr>
```

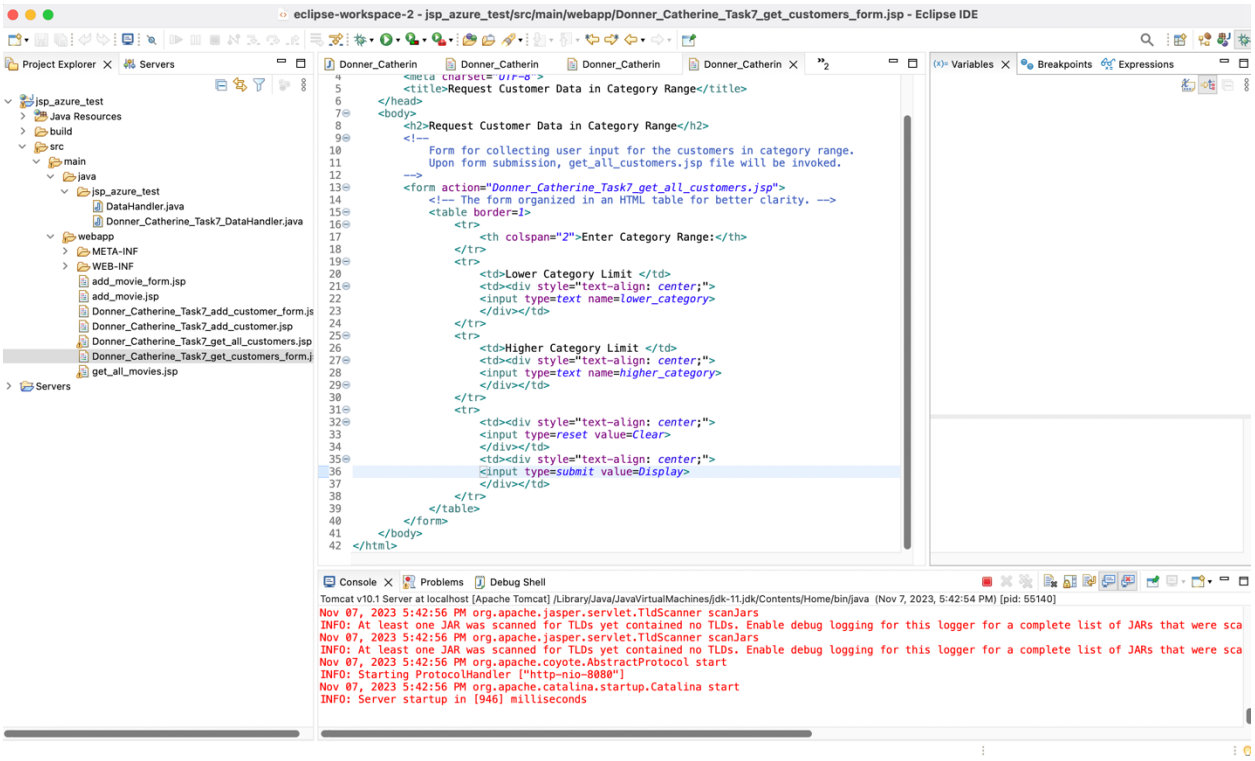
```
            <td><div style="text-align: center;">
            <input type=reset value=Clear>
            </div></td>
            <td><div style="text-align: center;">
            <input type=submit value=Display>
            </div></td>
          </tr>
        </table>
      </form>
  </body>
</html>
```

Get_all_customers.jsp Code (called when get_customers_form.jsp is invoked)

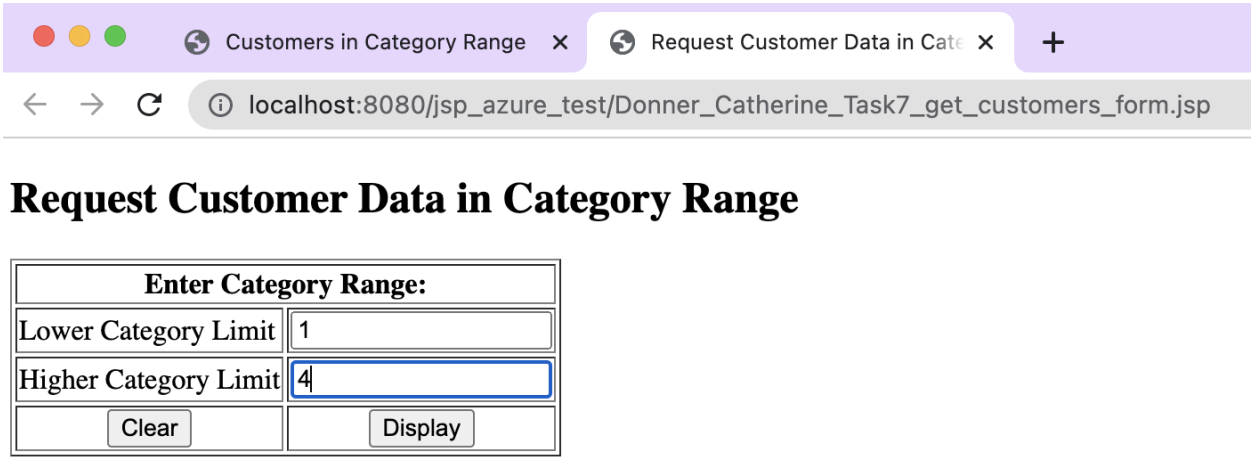
```
<%@ page language="java" contentType="text/html; charset=UTF-8"
  pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
  <meta charset="UTF-8">
    <title>Customers in Category Range</title>
  </head>
  <body>
    <%@page import="jsp_azure_test.Donner_Catherine_IP_Task7_DataHandler"%>
    <%@page import="java.sql.ResultSet"%>
    <%
            // Instantiate DataHandler
```

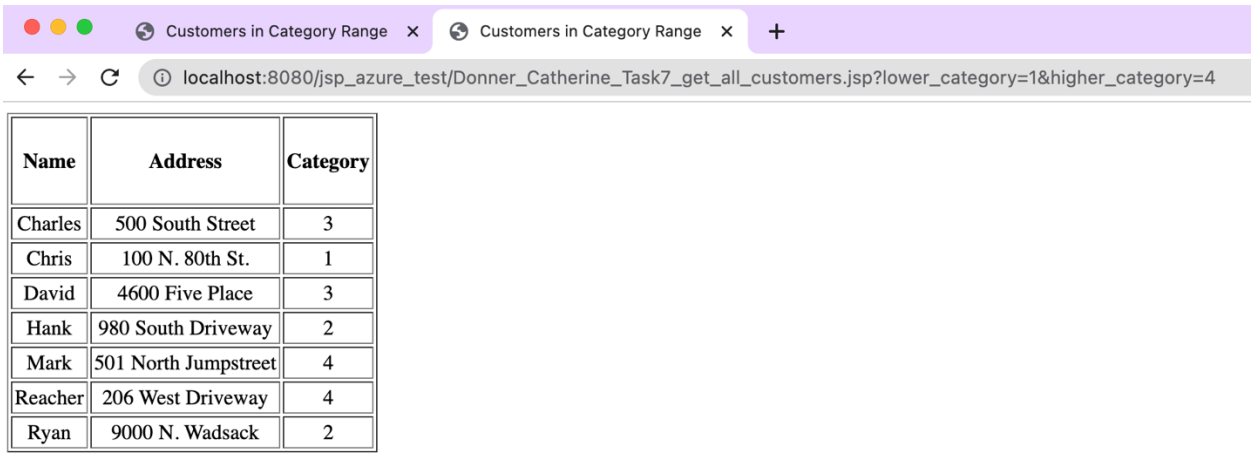
```
            final Donner_Catherine_IP_Task7_DataHandler handler = new
Donner_Catherine_IP_Task7_DataHandler();


        // Extract strings
        String lower_category_str = request.getParameter("lower_category");
        String higher_category_str = request.getParameter("higher_category");


        /*
        * Check if the user has input lower_category and higher_category
        */
        if (lower_category_str.equals("") || higher_category_str.equals("") ) {
                response.sendRedirect("Donner_Catherine_IP_Task7_get_customers_form.jsp");
        } else {
                int lower_category = Integer.parseInt(lower_category_str);
                int higher_category = Integer.parseInt(higher_category_str);
                // Retrieve customers in given range
                final ResultSet customers = handler.getAllCustomersinRange(lower_category,
higher_category);
    %>
    <!-- The table for displaying all the customer records -->
    <table cellspacing="2" cellpadding="2" border="1">
      <tr> <!-- The table headers row -->
       <td align="center">
         <h4>Name</h4>
       </td>
       <td align="center">
```

```
            <h4>Address</h4>
          </td>
          <td align="center">
            <h4>Category</h4>
          </td>
        </tr>
        <%
          while(customers.next()) { // For each customer record returned...
            // Extract the attribute values for every row returned
            final String name = customers.getString("name");
            final String address = customers.getString("address");
            final String category = customers.getString("category");

            out.println("<tr>"); // Start printing out the new table row
            out.println( // Print each attribute value
                "<td align=\"center\">" + name +
                "</td><td align=\"center\"> " + address +
                "</td><td align=\"center\"> " + category + "</td>");
            out.println("</tr>");
          }
        }
        %>
      </table>
    </body>
</html>
```

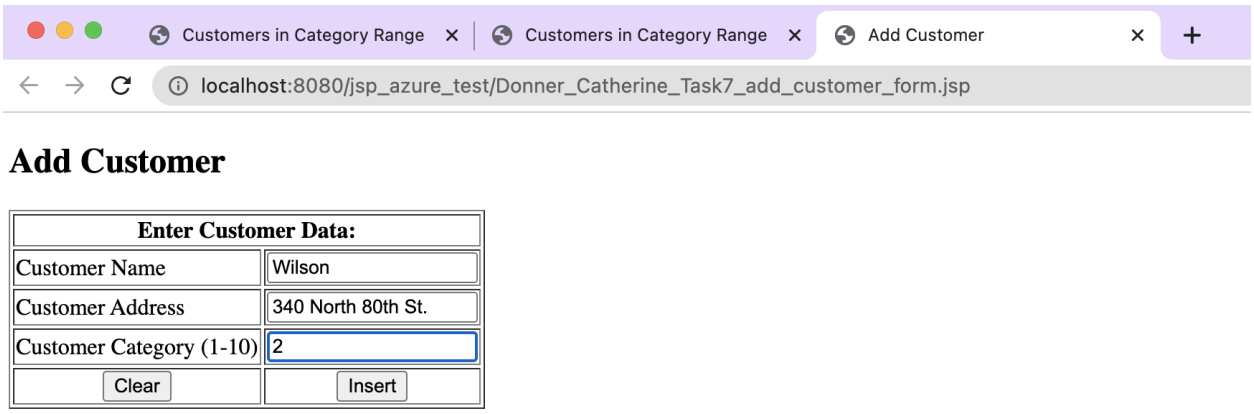


7.2 Screenshots showing the testing of the Web database application



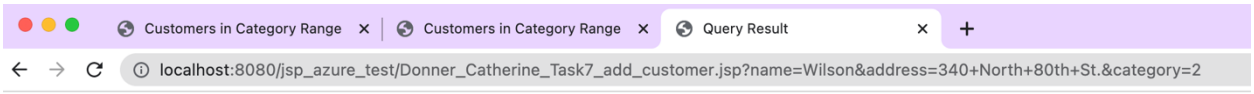


Name	Address	Category
Charles	500 South Street	3
Chris	100 N. 80th St.	1
David	4600 Five Place	3
Hank	980 South Driveway	2
Mark	501 North Jumpstreet	4
Reacher	206 West Driveway	4
Ryan	9000 N. Wadsack	2



Add Customer

Enter Customer Data:	
Customer Name	Wilson
Customer Address	340 North 80th St.
Customer Category (1-10)	2
Clear	Insert



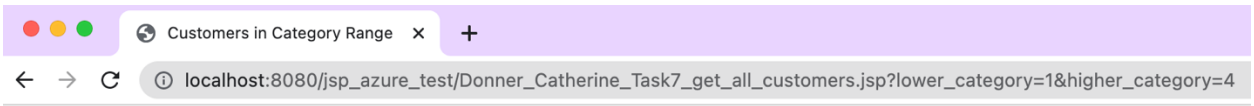
New Customer:

- Name: Wilson
- Address: 340 North 80th St.
- Category: 2

Was successfully inserted.

Request Customer Data in Category Range

Enter Category Range:	
Lower Category Limit	1
Higher Category Limit	4
Clear	Display



Name	Address	Category
Charles	500 South Street	3
Chris	100 N. 80th St.	1
David	4600 Five Place	3
Hank	980 South Driveway	2
Mark	501 North Jumpstreet	4
Reacher	206 West Driveway	4
Ryan	9000 N. Wadsack	2
Wilson	340 North 80th St.	2

***** END OF REPORT *****