

**1 Understanding J2EE Architecture using JSP, SERVLETS, Beans and JDBC**

**2 Understanding JSP**

**Develop a JSP application to display access count of the webpage using sessions.**

**02.  Assume the existence of database named “Inventory” which contains the table**

**a.        “Product” with necessary information like Product ID (varchar), Product name (varchar), supplier ID (varchar), stock available(number), opening stock (number), last supply date (Date), unit price (number).**

**b.      “Supplier” containing Supplier ID (varchar), Suppler Name (varchar), Email (varchar), Mobile Phone number (varchar)**

**c.       “User” containing UserId (varchar),  Username (varchar), password (varchar), role (varchar- Allowed values are {DEO,POS,MGR})**

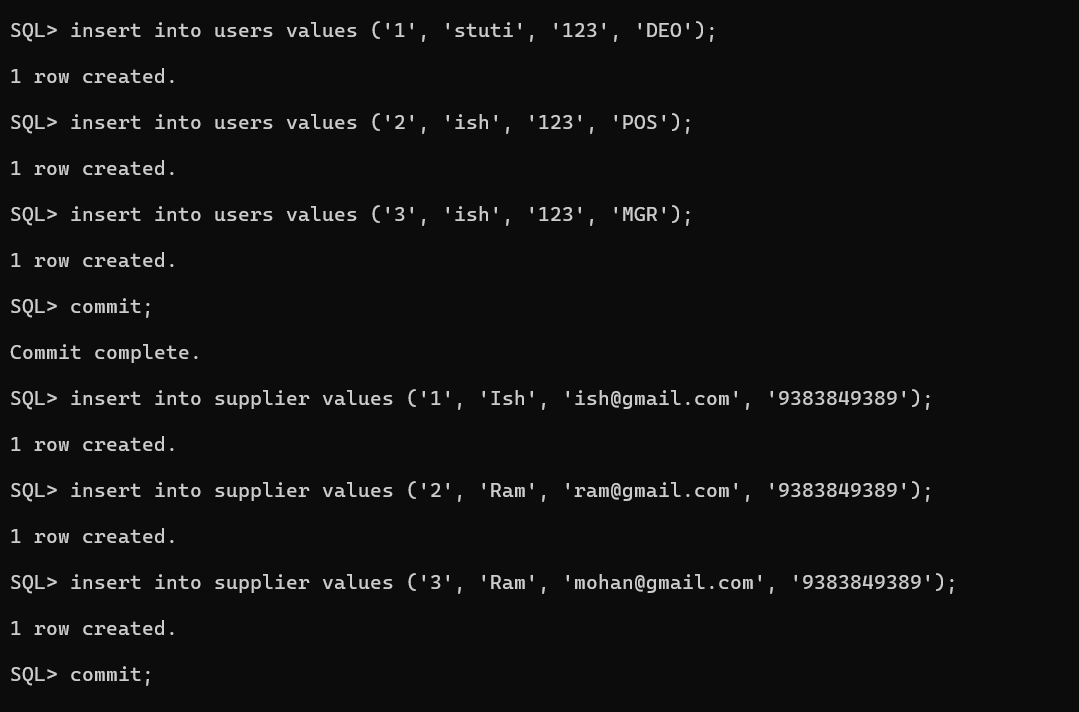
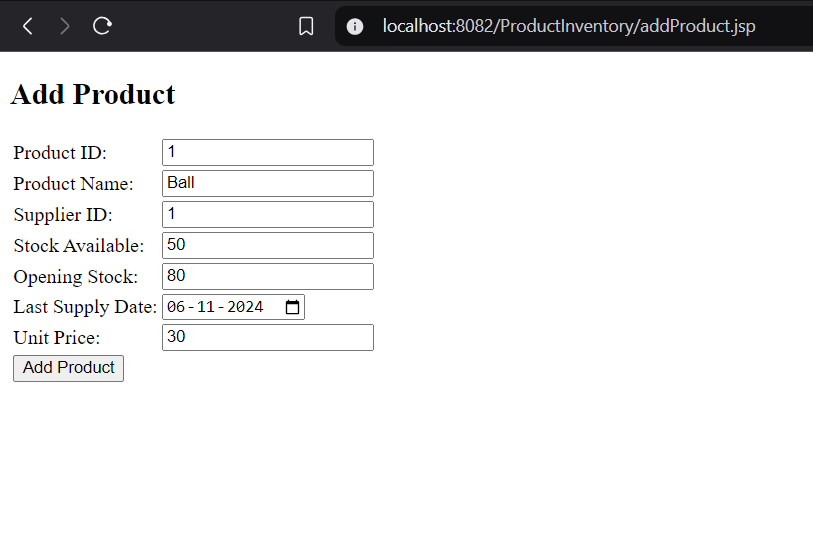
**Develop a JSP page to**

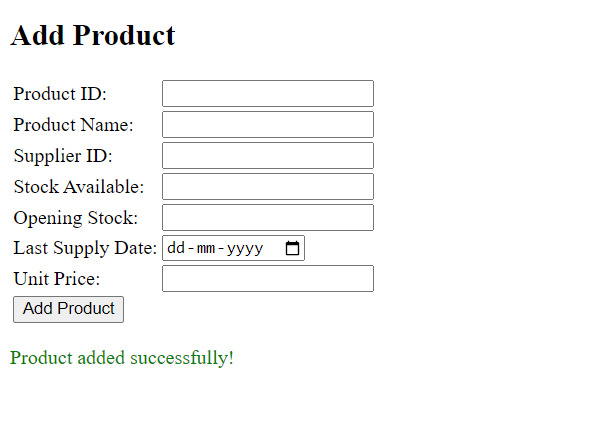
**d.      Add new product. Prevent duplication of Product ID and entering non existing supplier ID.**

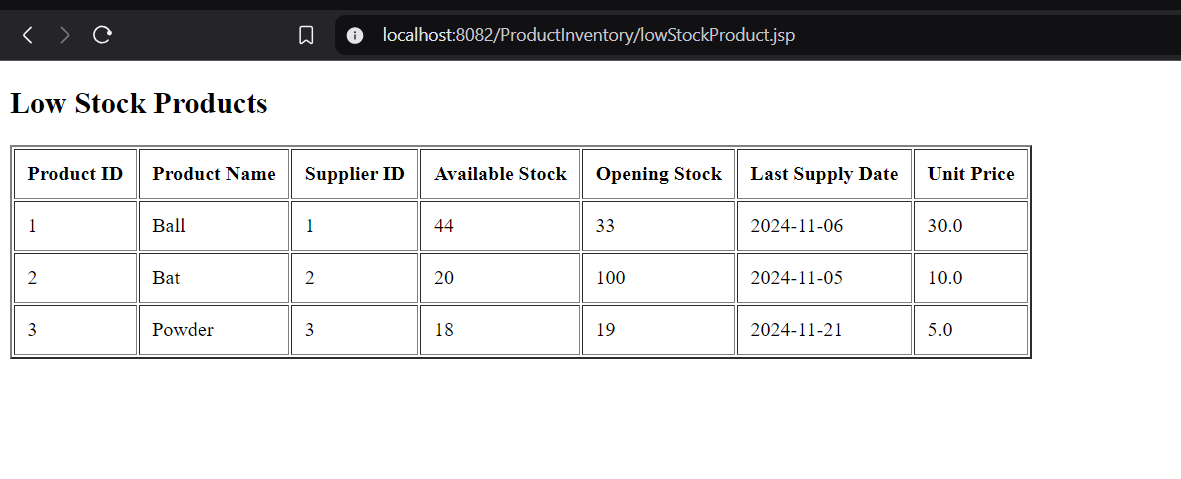
**e.       Fetch product based on ID**

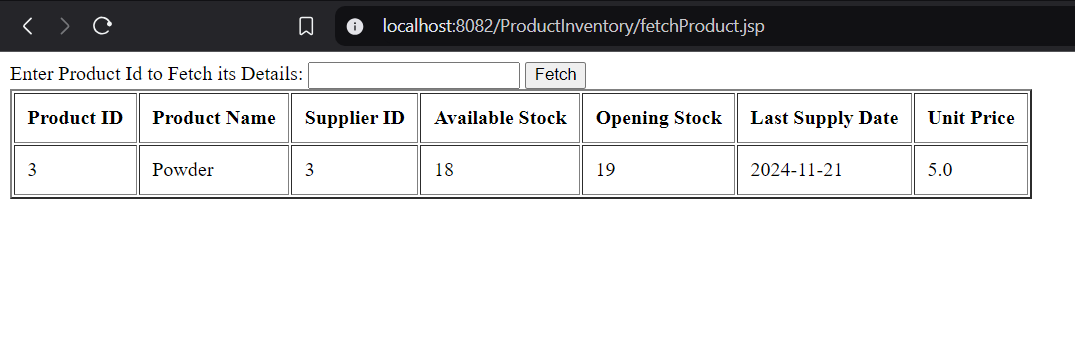
**f. View the list of products for which stock available is less than 50**

**DATABASE**

****



****



**Index.jsp**

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

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

<!**DOCTYPE** html>

<**html**>

<**head**>

<**meta** charset=*"UTF-8"*>

<**title**>Login</**title**>

</**head**>

<**body**>

<%

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

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

String errorMessage = null;

if (username != null && password != null) {

Connection conn = null;

PreparedStatement stmt = null;

ResultSet rs = null;

try {

// Load the Oracle JDBC driver

Class.forName("oracle.jdbc.driver.OracleDriver");

// Database connection details

String dbURL = "jdbc:oracle:thin:@localhost:1521:xe";

String dbUser = "system";

String dbPassword = "admin";

// Establish the database connection

conn = DriverManager.getConnection(dbURL, dbUser, dbPassword);

// SQL query to check if the user exists

String sql = "SELECT role FROM users WHERE username LIKE ? AND password LIKE ?";

stmt = conn.prepareStatement(sql);

stmt.setString(1, username);

stmt.setString(2, password);

rs = stmt.executeQuery();

if (rs.next()) {

// User authenticated successfully, create session

session = request.getSession(true);

session.setAttribute("username", username);

response.sendRedirect("welcome.jsp?role="+rs.getString("role").toString());

} else {

// Authentication failed

errorMessage = "Invalid Username or Password!";

}

} catch (Exception e) {

e.printStackTrace();

errorMessage = "Database connection error!";

} finally {

// Close the resources

if (rs != null) rs.close();

if (stmt != null) stmt.close();

if (conn != null) conn.close();

}

}

%>

<**h2**>Login Page</**h2**>

<**form** method=*"post"*>

<**table**>

<**tr**>

<**td**>Username:</**td**>

<**td**><**input** type=*"text"* name=*"username"* required></**td**>

</**tr**>

<**tr**>

<**td**>Password:</**td**>

<**td**><**input** type=*"password"* name=*"password"* required></**td**>

</**tr**>

<**tr**>

<**td** colspan=*"2"*>

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

</**td**>

</**tr**>

</**table**>

</**form**>

<%

if (errorMessage != null) {

%>

<**p** style="color: *red*;"><%= errorMessage %></**p**>

<%

}

%>

</**body**>

</**html**>

**Welcome.jsp**

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

pageEncoding=*"UTF-8"*%>

<!**DOCTYPE** html>

<**html**>

<**head**>

<**meta** charset=*"UTF-8"*>

<**title**>Welcome</**title**>

</**head**>

<**body**>

Hi <%= session.getAttribute("username") %><**br**><**br**>

<%

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

if(role.equals("POS")){%>

<**strong**><**a** href = *"fetchProduct.jsp"*>Fetch Product</**a**></**strong**><**br**><**br**>

<%}

if(role.equals("DEO")){%>

<**strong**><**a** href = *"addProduct.jsp"* >Add Product</**a**></**strong**><**br**><**br**>

<%}

if(role.equals("MGR")){%>

<**strong**><**a** href = *"lowStockProduct.jsp"*>Low Stock Product</**a**></**strong**><**br**><**br**>

<%}

%>

</**body**>

</**html**>

**fetchProduct.jsp**

<%@ **page** language=*"java"* import=*"java.sql.\*"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<!**DOCTYPE** html>

<**html**>

<**head**>

<**meta** charset=*"UTF-8"*>

<**title**>Fetch Product</**title**>

</**head**>

<**body**>

<**form** method=*"post"*>

Enter Product Id to Fetch its Details: <**input** name=*"ProductID"* type=*"text"* required/>

<**input** type=*"submit"* value=*"Fetch"*/>

</**form**>

<%

String pid = request.getParameter("ProductID");

if (pid != null) {

Connection conn = null;

PreparedStatement pstmt = null;

ResultSet rs = null;

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "system", "admin");

String sql = "SELECT \* FROM PRODUCT WHERE ProductID = ?";

pstmt = conn.prepareStatement(sql);

pstmt.setString(1, pid);

rs = pstmt.executeQuery(); // Execute the prepared statement

if (rs.next()) {

// Display table headers

%>

<**table** border=*"2"* cellpadding=*"10"*>

<**tr**>

<**th**>Product ID</**th**>

<**th**>Product Name</**th**>

<**th**>Supplier ID</**th**>

<**th**>Available Stock</**th**>

<**th**>Opening Stock</**th**>

<**th**>Last Supply Date</**th**>

<**th**>Unit Price</**th**>

</**tr**>

<%

// Display the product details

%>

<**tr**>

<**td**><%= rs.getString("ProductID") %></**td**>

<**td**><%= rs.getString("ProductName") %></**td**>

<**td**><%= rs.getString("SupplierID") %></**td**>

<**td**><%= rs.getInt("StockAvailable") %></**td**>

<**td**><%= rs.getInt("OpeningStock") %></**td**>

<**td**><%= rs.getDate("LastSupplyDate") %></**td**>

<**td**><%= rs.getDouble("UnitPrice") %></**td**>

</**tr**>

</**table**>

<%

} else {

%>

<**p**>No products found. <**a** href=*"addProduct.jsp"*><**strong**>Add Products</**strong**></**a**></**p**>

<%

}

} catch (Exception e) {

e.printStackTrace();

} finally {

if (rs != null) rs.close();

if (pstmt != null) pstmt.close(); // Corrected from stmt to pstmt

if (conn != null) conn.close();

}

}

%>

</**body**>

</**html**>

**AddProduct.jsp**

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

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

<!**DOCTYPE** html>

<**html**>

<**head**>

<**meta** charset=*"UTF-8"*>

<**title**>Add Product</**title**>

</**head**>

<**body**>

<%

String productID = request.getParameter("ProductID");

String productName = request.getParameter("ProductName");

String supplierID = request.getParameter("SupplierID");

String stockAvailable = request.getParameter("StockAvailable");

String openingStock = request.getParameter("OpeningStock");

String lastSupplyDate = request.getParameter("LastSupplyDate");

String unitPrice = request.getParameter("UnitPrice");

String errorMessage = null;

boolean productAdded = false;

if (productID != null && productName != null && supplierID != null && stockAvailable != null && openingStock != null && lastSupplyDate != null && unitPrice != null) {

Connection conn = null;

PreparedStatement stmt = null;

ResultSet rs = null;

try {

// Load Oracle JDBC Driver

Class.forName("oracle.jdbc.driver.OracleDriver");

// Connect to the Oracle Database

String dbURL = "jdbc:oracle:thin:@localhost:1521:xe";

String dbUser = "system";

String dbPassword = "admin";

conn = DriverManager.getConnection(dbURL, dbUser, dbPassword);

// Check if ProductID already exists

String checkProductSQL = "SELECT ProductID FROM Product WHERE ProductID = ?";

stmt = conn.prepareStatement(checkProductSQL);

stmt.setString(1, productID);

rs = stmt.executeQuery();

if (rs.next()) {

errorMessage = "Product ID already exists!";

} else {

// Check if SupplierID exists in Supplier table

String checkSupplierSQL = "SELECT SupplierID FROM Supplier WHERE SupplierID = ?";

stmt = conn.prepareStatement(checkSupplierSQL);

stmt.setString(1, supplierID);

rs = stmt.executeQuery();

if (!rs.next()) {

errorMessage = "Supplier ID does not exist!";

} else {

// Insert the new product

String insertSQL = "INSERT INTO Product (ProductID, ProductName, SupplierID, StockAvailable, OpeningStock, LastSupplyDate, UnitPrice) VALUES (?, ?, ?, ?, ?, TO\_DATE(?, 'YYYY-MM-DD'), ?)";

stmt = conn.prepareStatement(insertSQL);

stmt.setString(1, productID);

stmt.setString(2, productName);

stmt.setString(3, supplierID);

stmt.setInt(4, Integer.parseInt(stockAvailable));

stmt.setInt(5, Integer.parseInt(openingStock));

stmt.setString(6, lastSupplyDate); // Date in format 'YYYY-MM-DD'

stmt.setDouble(7, Double.parseDouble(unitPrice));

stmt.executeUpdate();

productAdded = true;

}

}

} catch (Exception e) {

e.printStackTrace();

errorMessage = "Database error occurred!";

} finally {

// Close resources

if (rs != null) rs.close();

if (stmt != null) stmt.close();

if (conn != null) conn.close();

}

}

%>

<**h2**>Add Product</**h2**>

<**form** method=*"post"*>

<**table**>

<**tr**>

<**td**>Product ID:</**td**>

<**td**><**input** type=*"text"* name=*"ProductID"* required></**td**>

</**tr**>

<**tr**>

<**td**>Product Name:</**td**>

<**td**><**input** type=*"text"* name=*"ProductName"* required></**td**>

</**tr**>

<**tr**>

<**td**>Supplier ID:</**td**>

<**td**><**input** type=*"text"* name=*"SupplierID"* required></**td**>

</**tr**>

<**tr**>

<**td**>Stock Available:</**td**>

<**td**><**input** type=*"number"* name=*"StockAvailable"* required></**td**>

</**tr**>

<**tr**>

<**td**>Opening Stock:</**td**>

<**td**><**input** type=*"number"* name=*"OpeningStock"* required></**td**>

</**tr**>

<**tr**>

<**td**>Last Supply Date:</**td**>

<**td**><**input** type=*"date"* name=*"LastSupplyDate"* required></**td**>

</**tr**>

<**tr**>

<**td**>Unit Price:</**td**>

<**td**><**input** type=*"number"* step=*"0.01"* name=*"UnitPrice"* required></**td**>

</**tr**>

<**tr**>

<**td** colspan=*"2"*>

<**input** type=*"submit"* value=*"Add Product"*>

</**td**>

</**tr**>

</**table**>

</**form**>

<% if (errorMessage != null) { %>

<**p** style="color: *red*;"><%= errorMessage %></**p**>

<% } else if (productAdded) { %>

<**p** style="color: *green*;">Product added successfully!</**p**>

<% } %>

</**body**>

</**html**>

**lowStockProduct.jsp**

<%@ **page** language=*"java"* import = *"java.sql.\*"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<!**DOCTYPE** html>

<**html**>

<**head**>

<**meta** charset=*"UTF-8"*>

<**title**>Low Stock Product</**title**>

</**head**>

<**body**>

<**h2**>Low Stock Products</**h2**>

<%

Connection conn = null;

Statement stmt = null;

ResultSet rs = null;

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "system", "admin");

String sql = "SELECT \* FROM PRODUCT WHERE StockAvailable < 50";

stmt = conn.createStatement();

rs = stmt.executeQuery(sql);

if (rs.next()) {

// Display table headers

%>

<**table** border=*"2"* cellpadding=*"10"*>

<**tr**>

<**th**>Product ID</**th**>

<**th**>Product Name</**th**>

<**th**>Supplier ID</**th**>

<**th**>Available Stock</**th**>

<**th**>Opening Stock</**th**>

<**th**>Last Supply Date</**th**>

<**th**>Unit Price</**th**>

</**tr**>

<%

do {

%>

<**tr**>

<**td**><%= rs.getString("ProductID") %></**td**>

<**td**><%= rs.getString("ProductName") %></**td**>

<**td**><%= rs.getString("SupplierID") %></**td**>

<**td**><%= rs.getInt("StockAvailable") %></**td**>

<**td**><%= rs.getInt("OpeningStock") %></**td**>

<**td**><%= rs.getDate("LastSupplyDate") %></**td**>

<**td**><%= rs.getDouble("UnitPrice") %></**td**>

</**tr**>

<%

} while (rs.next());

%>

</**table**>

<%

} else {

%>

<**p**>No products found. <**a** href=*"addProduct.jsp"*><**strong**>Add Products</**strong**></**a**></**p**>

<%

}

} catch (Exception e) {

e.printStackTrace();

} finally {

if (rs != null) rs.close();

if (stmt != null) stmt.close();

if (conn != null) conn.close();

}

%>

</**body**>

</**html**>

**Understanding Java Networking and RMI**

**01. Develop a TCP based client –server application using java.net.\*. Let the client transfers the operation in the format of ‘A + B’ and server in turn interprets the operator, perform the requested operation using the operands and transmits the result to get displayed in client.**

**TCPClient.java**

**package TCPOperand;**

**import java.io.BufferedReader;**

**import java.io.InputStreamReader;**

**import java.io.PrintWriter;**

**import java.net.Socket;**

**public class TCPClient {**

**public static void main(String[] args) {**

**String serverAddress = "localhost";**

**int port = 1234;**

**try (Socket socket = new Socket(serverAddress, port);**

**BufferedReader input = new BufferedReader(new InputStreamReader(socket.getInputStream()));**

**PrintWriter output = new PrintWriter(socket.getOutputStream(), true);**

**BufferedReader consoleInput = new BufferedReader(new InputStreamReader(System.in))) {**

**System.out.println("Enter an operation in the format 'A + B':");**

**String operation = consoleInput.readLine();**

**output.println(operation);**

**String response = input.readLine();**

**System.out.println("Server response: " + response);**

**} catch (Exception e) {**

**System.out.println("Client exception: " + e.getMessage());**

**}**

**}**

**}**

**TCPServer.java**

**package TCPOperand;**

**import java.io.BufferedReader;**

**import java.io.InputStreamReader;**

**import java.io.PrintWriter;**

**import java.net.ServerSocket;**

**import java.net.Socket;**

**public class TCPServer {**

**public static void main(String[] args) {**

**try (ServerSocket serverSocket = new ServerSocket(1234)) {**

**System.out.println("Server is listening on port 1234");**

**while (true) {**

**try (Socket socket = serverSocket.accept();**

**BufferedReader input = new BufferedReader(new InputStreamReader(socket.getInputStream()));**

**PrintWriter output = new PrintWriter(socket.getOutputStream(), true)) {**

**String expression = input.readLine();**

**String[] tokens = expression.split(" ");**

**int operand1 = Integer.parseInt(tokens[0]);**

**String operator = tokens[1];**

**int operand2 = Integer.parseInt(tokens[2]);**

**int result = 0;**

**switch (operator) {**

**case "+":**

**result = operand1 + operand2;**

**break;**

**case "-":**

**result = operand1 - operand2;**

**break;**

**case "\*":**

**result = operand1 \* operand2;**

**break;**

**case "/":**

**if (operand2 != 0) {**

**result = operand1 / operand2;**

**} else {**

**output.println("Error: Division by zero");**

**continue;**

**}**

**break;**

**default:**

**output.println("Error: Unknown operator");**

**continue;**

**}**

**output.println("Result: " + result);**

**} catch (Exception e) {**

**System.out.println("Error: " + e.getMessage());**

**}**

**}**

**} catch (Exception e) {**

**System.out.println("Server exception: " + e.getMessage());**

**}**

**}**

**}**

**02. Design a UDP application in which sender broadcasts the message and the receiver computes count of each character in a message.**

import java.net.DatagramPacket;

import java.net.DatagramSocket;

import java.net.InetAddress;

import java.net.SocketException;

public class UDPCountChar {

public static DatagramSocket ds;

public static byte buffer[] = new byte[1024];

public static void server() {

System.out.println("SERVER is up");

try {

int pos = 0;

while(true) {

int ch = System.in.read();

switch(ch) {

case '\n' : ds.send(new DatagramPacket(buffer, pos, InetAddress.getLocalHost(),6666));

pos = 0;

break;

default : buffer[pos++] = (byte)ch;

break;

}

}

} catch (Exception e) {

e.printStackTrace();

}

}

public static void client() {

System.out.println("CLIENT is up");

try {

int pos = 0;

while(true) {

DatagramPacket p = new DatagramPacket(buffer, buffer.length);

ds.receive(p);

String str = new String(p.getData(), 0 ,p.getLength()).trim();

System.out.println("Received : "+ str);

int cnt[] = new int[26];

for(int i=0; i<str.length(); i++) {

if(str.charAt(i) - 'a' < 26){

cnt[str.charAt(i) - 'a']++;

}

}

System.out.println("Character counts :- ");

for(int i=0; i<26; i++) {

if(cnt[i]!= 0)

System.out.println((char)(i+'a') + " : " + cnt[i]);

}

}

} catch (Exception e) {

e.printStackTrace();

}

}

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

if (args.length == 1){

ds = new DatagramSocket(5555);

server();

} else {

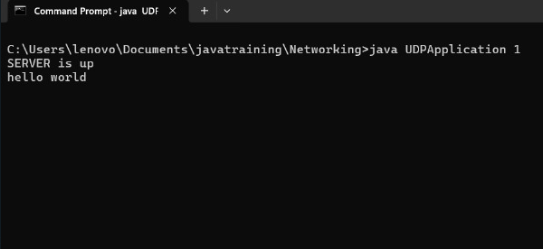
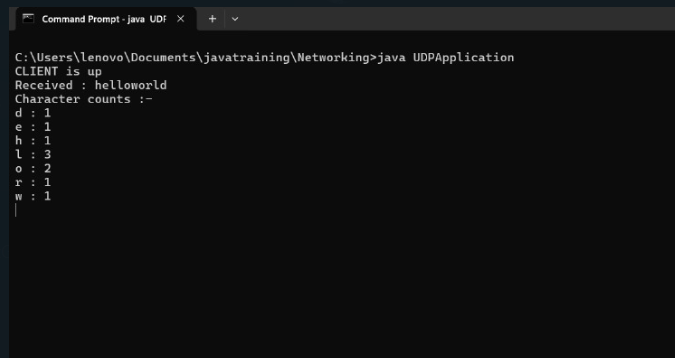
ds = new DatagramSocket(6666);

client();

}

}

}

**03. Implement an RMI application to invoke a factorial method with appropriate parameter deployed in server and display the result in client.**

import java.rmi.Remote;

import java.rmi.RemoteException;

public interface Factorial extends Remote {

int computeFactorial(int n) throws RemoteException;

}

import java.rmi.RemoteException;

import java.rmi.server.UnicastRemoteObject;

public class FactorialImpl extends UnicastRemoteObject implements Factorial {

protected FactorialImpl() throws RemoteException {

super();

}

@Override

public int computeFactorial(int n) throws RemoteException {

if (n <= 1) return 1;

return n \* computeFactorial(n - 1);

}

}

import java.rmi.Naming;

import java.rmi.registry.LocateRegistry;

public class FactorialServer {

public static void main(String[] args) {

try {

LocateRegistry.createRegistry(1099);

FactorialImpl factorial = new FactorialImpl();

Naming.rebind("rmi://localhost:1099/FactorialService", factorial);

System.out.println("Factorial Service is running...");

} catch (Exception e) {

System.err.println("Server exception: " + e.toString());

e.printStackTrace();

}

}

}

import java.rmi.Naming;

public class FactorialClient {

public static void main(String[] args) {

try {

Factorial factorial = (Factorial) Naming.lookup("rmi://localhost:1099/FactorialService");

int number = 5; // Example number to calculate factorial

int result = factorial.computeFactorial(number);

System.out.println("Factorial of " + number + " is " + result);

} catch (Exception e) {

System.err.println("Client exception: " + e.toString());

e.printStackTrace();

}

}

}