Q1.) Create a JSP page to accept a number from an user and display it in words: Example: 123 One Two Three. The output should be in red color.

```
<%@ page import="java.io.*, java.util.*" %>
<html>
<head>
   <title>Number to Words</title>
</head>
<body>
   <form method="post">
      Enter a Number: <input type="text" name="num">
      <input type="submit" value="Show in Words">
   </form>

   <%
      String num = request.getParameter("num");
      if (num != null && num.matches("\\d+")) {
         out.print("<h3 style='color:red;'>");
         for (char digit : num.toCharArray()) {
            switch (digit) {
               case '0': out.print("Zero "); break;
               case '1': out.print("One "); break;
               case '2': out.print("Two "); break;
               case '3': out.print("Three "); break;
               case '4': out.print("Four "); break;
               case '5': out.print("Five "); break;
               case '6': out.print("Six "); break;
               case '7': out.print("Seven "); break;
```

```
                case '8': out.print("Eight "); break;
                case '9': out.print("Nine "); break;
            }
        }
        out.print("</h3>");
    }
  %>
</body>
</html>
```

Q.2) Write a JDBC program to display all the details of the Person table in proper format on the screen. Create a Person table with fields as PID, name, gender, birth_year. Insert values in Person table.

```
CREATE TABLE Person (
   PID INT PRIMARY KEY,
   name VARCHAR(50),
   gender VARCHAR(10),
   birth_year INT
);

INSERT INTO Person VALUES (1, 'Om', 'Male', 2002);
INSERT INTO Person VALUES (2, 'Asha', 'Female', 1999);

import java.sql.*;

public class PersonDisplay {
```

```
public static void main(String[] args) {
    try {
        // Load JDBC driver
        Class.forName("com.mysql.cj.jdbc.Driver");

        // Connect to database
        Connection con = DriverManager.getConnection(
            "jdbc:mysql://localhost:3306/your_db", "root", "your_password");

        // Query execution
        Statement stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery("SELECT * FROM Person");

        // Display results
        System.out.println("PID\tName\tGender\tBirth Year");
        System.out.println("---------------------------------");
        while (rs.next()) {
            System.out.println(
                rs.getInt("PID") + "\t" +
                rs.getString("name") + "\t" +
                rs.getString("gender") + "\t" +
                rs.getInt("birth_year"));
        }

        con.close();
    } catch (Exception e) {
        System.out.println(e);
```

```
    }
  }
}
```

Q.1) The following JSP program calculates factorial values for an integer number, while the input is taken from an HTML form.

```
<html>
<head>
  <title>Factorial Calculator</title>
</head>
<body>
  <form method="post">
    Enter a Number: <input type="text" name="num">
    <input type="submit" value="Calculate Factorial">
  </form>

  <%
    String input = request.getParameter("num");
    if (input != null && input.matches("\\d+")) {
      int n = Integer.parseInt(input);
      int fact = 1;
      for (int i = 1; i <= n; i++) {
        fact *= i;
      }
      out.println("<h3>Factorial of " + n + " is: " + fact + "</h3>");
    }
```

```
    %>
</body>
</html>
```

Q.2) Write a JDBC program to display all the countries located in the West Region. Create a table Country with fields (Name, continent, Capital,Region). Insert values in the table.

```
CREATE TABLE Country (
    Name VARCHAR(50),
    continent VARCHAR(50),
    Capital VARCHAR(50),
    Region VARCHAR(50)
);

INSERT INTO Country VALUES ('USA', 'North America', 'Washington D.C.', 'West');
INSERT INTO Country VALUES ('Germany', 'Europe', 'Berlin', 'West');
INSERT INTO Country VALUES ('India', 'Asia', 'New Delhi', 'South');
INSERT INTO Country VALUES ('Brazil', 'South America', 'Brasília', 'East');

import java.sql.*;

public class CountryDisplay {
    public static void main(String[] args) {
        try {
            // Load JDBC Driver
            Class.forName("com.mysql.cj.jdbc.Driver");
```

```
        // Connect to Database
        Connection con = DriverManager.getConnection(
            "jdbc:mysql://localhost:3306/your_db", "root", "your_password");

        // SQL Query
        String query = "SELECT * FROM Country WHERE Region = 'West'";
        Statement stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery(query);

        // Print Results
        System.out.println("Name\tContinent\tCapital\t\tRegion");
        System.out.println("-----------------------------------------------");
        while (rs.next()) {
            System.out.println(
                rs.getString("Name") + "\t" +
                rs.getString("continent") + "\t" +
                rs.getString("Capital") + "\t" +
                rs.getString("Region"));
        }

        con.close();
    } catch (Exception e) {
        System.out.println(e);
    }
  }
}
```

The following JSP program shows the Fibonacci series up to a particular term, while the input is taken from an HTML form.

```
<html>
<head>
   <title>Fibonacci Series</title>
</head>
<body>
   <form method="post">
      Enter Number of Terms: <input type="text" name="term">
      <input type="submit" value="Generate Series">
   </form>

   <%
      String termStr = request.getParameter("term");
      if (termStr != null && termStr.matches("\\d+")) {
         int n = Integer.parseInt(termStr);
         int a = 0, b = 1, c;
         out.print("<h3>Fibonacci Series: </h3>");
         for (int i = 1; i <= n; i++) {
            out.print(a + " ");
            c = a + b;
            a = b;
            b = c;
         }
      }
   %>
```

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

Q.2) Write a JDBC program to insert the records into the table Employee(ID,name,salary) using PreparedStatement interface. Accept details of Employees from user.

```
CREATE TABLE Employee (
   ID INT PRIMARY KEY,
   name VARCHAR(50),
   salary DOUBLE
);
```

```
import java.sql.*;
import java.util.Scanner;

public class InsertEmployee {
   public static void main(String[] args) {
      try {
         // Load driver
         Class.forName("com.mysql.cj.jdbc.Driver");

         // Connection
         Connection con = DriverManager.getConnection(
            "jdbc:mysql://localhost:3306/your_db", "root", "your_password");

         // Accept input
         Scanner sc = new Scanner(System.in);
```

```
            System.out.print("Enter ID: ");
            int id = sc.nextInt();
            sc.nextLine(); // consume newline
            System.out.print("Enter Name: ");
            String name = sc.nextLine();
            System.out.print("Enter Salary: ");
            double salary = sc.nextDouble();

            // Insert query using PreparedStatement
            String query = "INSERT INTO Employee VALUES (?, ?, ?)";
            PreparedStatement pst = con.prepareStatement(query);
            pst.setInt(1, id);
            pst.setString(2, name);
            pst.setDouble(3, salary);

            int i = pst.executeUpdate();
            if (i > 0)
                System.out.println("Record Inserted Successfully!");
            else
                System.out.println("Insertion Failed.");

            con.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

Q.1) Write a Program to make use of following JSP implicit objects:

i. out: To display current Date and Time.

ii. request: To get header information.

iii. response: To Add Cookie

iv. config: get the parameters value defined in <init-param>

v. application: get the parameter value defined in <context-param>

vi. session: Display Current Session ID

vii. pageContext: To set and get the attributes.

viii. page: get the name of Generated Servlet

```
<%@ page language="java" contentType="text/html" %>
<%@ page import="java.util.Date" %>

<%-- Set context and init param in web.xml to test config & application --%>
<html>
<head><title>JSP Implicit Objects Demo</title></head>
<body>

<%
   // i. out – Display current date & time
   out.println("<b>Current Date and Time:</b> " + new Date() + "<br>");

   // ii. request – Get header info
   out.println("<b>User-Agent:</b> " + request.getHeader("User-Agent") + "<br>");

   // iii. response – Add a cookie
   Cookie ck = new Cookie("User", "Om");
```

```
    response.addCookie(ck);
    out.println("<b>Cookie Added:</b> User = Om<br>");

    // iv. config – Get init param (defined in web.xml)
    String param1 = config.getInitParameter("myParam");
    out.println("<b>Init Param from config:</b> " + param1 + "<br>");

    // v. application – Get context-param
    String contextVal = application.getInitParameter("appParam");
    out.println("<b>Context Param:</b> " + contextVal + "<br>");

    // vi. session – Show session ID
    out.println("<b>Session ID:</b> " + session.getId() + "<br>");

    // vii. pageContext – Set and get attributes
    pageContext.setAttribute("course", "Java");
    out.println("<b>PageContext Attribute:</b> " + pageContext.getAttribute("course") + "<br>");

    // viii. page – Get name of generated servlet class (for demo)
    out.println("<b>Page Class:</b> " + page.getClass().getName());
%>

</body>
</html>
```

Web.xml

```
<context-param>
```

```
    <param-name>appParam</param-name>
    <param-value>JSP Practical</param-value>
</context-param>

<servlet>
    <servlet-name>JspObjectsDemo</servlet-name>
    <jsp-file>/JspObjectsDemo.jsp</jsp-file>
    <init-param>
        <param-name>myParam</param-name>
        <param-value>Hello Config</param-value>
    </init-param>
</servlet>
```

Q.2) Write a JDBC program to update number_of_students of “BCA Science” to 1000.Create a table Course (Code,name, department,number_of_students). Insert values in the table.

```
CREATE TABLE Course (
    Code INT,
    name VARCHAR(50),
    department VARCHAR(50),
    number_of_students INT
);

INSERT INTO Course VALUES (101, 'BCA Science', 'CS', 450);
INSERT INTO Course VALUES (102, 'BBA', 'Commerce', 300);
```

```
import java.sql.*;


public class UpdateCourse {
   public static void main(String[] args) {
      try {
         // Load JDBC driver
         Class.forName("com.mysql.cj.jdbc.Driver");


         // DB Connection
         Connection con = DriverManager.getConnection(
            "jdbc:mysql://localhost:3306/your_db", "root", "your_password");


         // Update query
         String query = "UPDATE Course SET number_of_students = 1000 WHERE name = 'BCA
Science'";
         Statement stmt = con.createStatement();
         int i = stmt.executeUpdate(query);


         if (i > 0)
            System.out.println("Record Updated Successfully!");
         else
            System.out.println("Update Failed.");


         con.close();
      } catch (Exception e) {
         System.out.println(e);
      }
   }
```

```
}
```

Q.1)Write a JSP script to display all the prime number's between 1 to n in Red Color.

```
<html>
<head>
  <title>Prime Numbers</title>
</head>
<body>
  <form method="post">
    Enter N: <input type="text" name="num">
    <input type="submit" value="Show Prime Numbers">
  </form>

  <%
    String numStr = request.getParameter("num");
    if (numStr != null && numStr.matches("\\d+")) {
      int n = Integer.parseInt(numStr);
      out.println("<h3 style='color:red;'>Prime Numbers from 1 to " + n + ":</h3>");
      for (int i = 2; i <= n; i++) {
        boolean prime = true;
        for (int j = 2; j <= Math.sqrt(i); j++) {
          if (i % j == 0) {
            prime = false;
            break;
          }
        }
        if (prime) {
```

```
                out.print("<span style='color:red;'>" + i + " </span>");
            }
        }
    }
%>
</body>
</html>
```

Q2: Client-Server Program to Display Server's Date & Time on Client

ServerTime.java

```
import java.io.*;
import java.net.*;
import java.util.Date;

public class ServerTime {
    public static void main(String[] args) throws IOException {
        ServerSocket ss = new ServerSocket(5000);
        System.out.println("Server is running...");

        Socket s = ss.accept();
        PrintWriter out = new PrintWriter(s.getOutputStream(), true);

        Date date = new Date();
        out.println("Server Date & Time: " + date);

        out.close();
```

```
        s.close();
        ss.close();
    }
}
```

ClientTime.java

```
import java.io.*;
import java.net.*;

public class ClientTime {
    public static void main(String[] args) throws IOException {
        Socket s = new Socket("localhost", 5000);
        BufferedReader in = new BufferedReader(new InputStreamReader(s.getInputStream()));

        String serverDate = in.readLine();
        System.out.println(serverDate);

        in.close();
        s.close();
    }
}
```

Q1: JSP Application to Accept & Display Employee Details in Table

```
<html>
<head><title>Employee Details</title></head>
<body>
```

```
<form method="post">
    <label>Employee No:</label>
    <input type="text" name="eno" required><br><br>

    <label>Employee Name:</label>
    <input type="text" name="ename" required><br><br>

    <label>Salary:</label>
    <input type="text" name="salary" required><br><br>

    <input type="submit" value="Submit">
</form>

<%
    String eno = request.getParameter("eno");
    String ename = request.getParameter("ename");
    String salary = request.getParameter("salary");

    if (eno != null && ename != null && salary != null) {
%>
    <h3>Employee Details:</h3>
    <table border="1" style="border-collapse: collapse;">
        <tr>
            <th>Employee No</th>
            <th>Employee Name</th>
            <th>Salary</th>
        </tr>
```

```
        <tr>
            <td><%= eno %></td>
            <td><%= ename %></td>
            <td><%= salary %></td>
        </tr>
    </table>
<%
    }
%>
</body>
</html>
```

Q2: Java Thread Class – Print Text N Times with Parameters

```
class PrintText_Thread extends Thread {
    String text;
    int count;

    PrintText_Thread(String text, int count) {
        this.text = text;
        this.count = count;
    }

    public void run() {
        for (int i = 1; i <= count; i++) {
            System.out.println(text);
        }
    }
```

```
    public static void main(String[] args) {
        PrintText_Thread t1 = new PrintText_Thread("I am in FY", 10);
        PrintText_Thread t2 = new PrintText_Thread("I am in SY", 20);
        PrintText_Thread t3 = new PrintText_Thread("I am in TY", 30);

        t1.start();
        t2.start();
        t3.start();
    }
}
```

Q.1)Write a JSP application to accept a user name and greets to the user according to the system Time.

```
<%@ page import="java.util.Calendar" %>
<html>
<head><title>Greeting</title></head>
<body>
    <form method="post">
        Enter Your Name: <input type="text" name="username" required>
        <input type="submit" value="Greet Me">
    </form>

<%
    String name = request.getParameter("username");
    if (name != null) {
        Calendar cal = Calendar.getInstance();
```

```
        int hour = cal.get(Calendar.HOUR_OF_DAY);
        String greeting = "";

        if (hour < 12)
            greeting = "Good Morning";
        else if (hour < 17)
            greeting = "Good Afternoon";
        else
            greeting = "Good Evening";

        out.println("<h2 style='color:green;'>" + greeting + ", " + name + "!</h2>");
    }
%>
</body>
</html>
```

Q.2) Design a servlet to display “Welcome IP address of client” to first time visitor. Display Welcome-back IP address of client” if the user is revisiting the page. (Use Cookies) (Hint: Use req.getRemoteAddr() to get IP address of client)

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class WelcomeServlet extends HttpServlet {
    protected void doGet(HttpServletRequest req, HttpServletResponse res) throws IOException {
        res.setContentType("text/html");
```

```
        PrintWriter out = res.getWriter();

        String ip = req.getRemoteAddr();
        boolean isReturning = false;

        Cookie[] cookies = req.getCookies();
        if (cookies != null) {
            for (Cookie c : cookies) {
                if (c.getName().equals("visited") && c.getValue().equals("yes")) {
                    isReturning = true;
                    break;
                }
            }
        }

        if (isReturning) {
            out.println("<h2>Welcome-back " + ip + "</h2>");
        } else {
            Cookie c = new Cookie("visited", "yes");
            c.setMaxAge(60 * 60 * 24); // 1 day
            res.addCookie(c);
            out.println("<h2>Welcome " + ip + "</h2>");
        }
    }
}
```

Q.1)Write a JSP application to accept the details of Teacher (TID, TName, Salary) and display it in tabular format.

```
<html>
<head><title>Teacher Details</title></head>
<body>
  <form method="post">
    TID: <input type="text" name="tid"><br><br>
    TName: <input type="text" name="tname"><br><br>
    Salary: <input type="text" name="salary"><br><br>
    <input type="submit" value="Submit">
  </form>

<%
  String tid = request.getParameter("tid");
  String tname = request.getParameter("tname");
  String salary = request.getParameter("salary");

  if (tid != null && tname != null && salary != null) {
%>
    <h3>Teacher Details:</h3>
    <table border="1" style="border-collapse: collapse;">
      <tr>
        <th>TID</th><th>TName</th><th>Salary</th>
      </tr>
      <tr>
        <td><%= tid %></td>
        <td><%= tname %></td>
```

```
            <td><%= salary %></td>
          </tr>
       </table>
<%
   }
%>
</body>
</html>
```

Q.2) Design the table User (username, password) using Database. Design HTML login screen. Accept the user name and password from the user. Write a servlet program to accept the login name and password and validates it from the database you have created. If it is correct then display Welcome.html otherwise display Error.html.

```
CREATE TABLE User (
   username VARCHAR(50),
   password VARCHAR(50)
);

INSERT INTO User VALUES ('admin', '12345');
```

(HTML Login Form (login.html)

```
<html>
<head><title>Login</title></head>
<body>
   <form action="LoginServlet" method="post">
      Username: <input type="text" name="username"><br><br>
```

```
    Password: <input type="password" name="password"><br><br>
    <input type="submit" value="Login">
  </form>
</body>
</html>
```

Step 3: **Servlet Code: LoginServlet.java**

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class LoginServlet extends HttpServlet {
  protected void doPost(HttpServletRequest req, HttpServletResponse res) throws IOException, ServletException {
    res.setContentType("text/html");
    String uname = req.getParameter("username");
    String pass = req.getParameter("password");

    try {
      Class.forName("com.mysql.jdbc.Driver"); // or com.mysql.cj.jdbc.Driver
      Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/yourdbname", "root", "password");
      PreparedStatement ps = con.prepareStatement("SELECT * FROM User WHERE username=? AND password=?");
      ps.setString(1, uname);
      ps.setString(2, pass);

      ResultSet rs = ps.executeQuery();
      if (rs.next()) {
```

```
            RequestDispatcher rd = req.getRequestDispatcher("Welcome.html");
            rd.forward(req, res);
        } else {
            RequestDispatcher rd = req.getRequestDispatcher("Error.html");
            rd.forward(req, res);
        }

    } catch (Exception e) {
        res.getWriter().println("Error: " + e.getMessage());
    }
  }
}
```

Step 4: Welcome.html

html

CopyEdit

```
<h1>Welcome! You have logged in successfully!</h1>
```

Step 5: Error.html

html

CopyEdit

```
<h1 style="color:red;">Invalid username or password!</h1>
```

Q.1) Write a JSP program to display the details of Hospital (HNo, HName, Address) in tabular form on browser

```
<html>
```

```
<head><title>Hospital Details</title></head>
<body>
  <form method="post">
    HNo: <input type="text" name="hno"><br><br>
    HName: <input type="text" name="hname"><br><br>
    Address: <input type="text" name="address"><br><br>
    <input type="submit" value="Show Details">
  </form>

<%
  String hno = request.getParameter("hno");
  String hname = request.getParameter("hname");
  String address = request.getParameter("address");

  if (hno != null && hname != null && address != null) {
%>
  <h3>Hospital Details:</h3>
  <table border="1" style="border-collapse: collapse;">
    <tr>
      <th>HNo</th><th>HName</th><th>Address</th>
    </tr>
    <tr>
      <td><%= hno %></td>
      <td><%= hname %></td>
      <td><%= address %></td>
    </tr>
  </table>
```

```
<%
   }
%>
</body>
</html>
```

Write a servlet which counts how many times a user has visited a web page. If the user is visiting the page for the first time, display a welcome message. If the user is re-visiting the page, display the number of times visited. (Use cookies).

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class VisitCounterServlet extends HttpServlet {
   public void doGet(HttpServletRequest req, HttpServletResponse res) throws IOException {
      res.setContentType("text/html");
      PrintWriter out = res.getWriter();

      int visitCount = 1;
      boolean isNewVisitor = true;

      Cookie[] cookies = req.getCookies();
      if (cookies != null) {
         for (Cookie c : cookies) {
            if (c.getName().equals("visitCount")) {
               visitCount = Integer.parseInt(c.getValue()) + 1;
```

```
                isNewVisitor = false;
                break;
            }
        }
    }

    Cookie visitCookie = new Cookie("visitCount", String.valueOf(visitCount));
    visitCookie.setMaxAge(60 * 60 * 24 * 365); // 1 year
    res.addCookie(visitCookie);

    if (isNewVisitor) {
        out.println("<h2>Welcome! This is your first visit.</h2>");
    } else {
        out.println("<h2>Welcome back! You have visited this page " + visitCount + " times.</h2>");
    }
  }
}
```

Q.1) Write a JSP program to calculate sum of first and last digit of a given number. Display sum in Red Color with font size 18.

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1" pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
  <meta charset="ISO-8859-1">
  <title>Sum of First and Last Digit</title>
</head>
```

```
<body>
    <h2>Enter a number:</h2>
    <form method="post">
        <input type="text" name="number" required />
        <input type="submit" value="Calculate" />
    </form>

    <%
        String numberStr = request.getParameter("number");
        if (numberStr != null && !numberStr.isEmpty()) {
            int number = Integer.parseInt(numberStr);
            int lastDigit = number % 10;
            int firstDigit = Integer.parseInt(String.valueOf(number).charAt(0));

            int sum = firstDigit + lastDigit;
    %>

    <p style="color:red; font-size:18px;">Sum of first and last digit: <%= sum %></p>

    <%
        }
    %>
</body>
</html>
```

Q.2) Write a java program using multithreading to execute the threads sequentially.

(Use Synchronized Method)

```
class PrintTask implements Runnable {
    private String message;

    public PrintTask(String message) {
        this.message = message;
    }

    @Override
    public void run() {
        synchronized (this) {
            System.out.println(message);
        }
    }
}

public class SequentialThreads {
    public static void main(String[] args) {
        PrintTask task1 = new PrintTask("Task 1");
        PrintTask task2 = new PrintTask("Task 2");
        PrintTask task3 = new PrintTask("Task 3");

        Thread thread1 = new Thread(task1);
        Thread thread2 = new Thread(task2);
        Thread thread3 = new Thread(task3);
```

```
        // Start threads sequentially
        try {
            thread1.start();
            thread1.join(); // Wait for thread1 to finish

            thread2.start();
            thread2.join(); // Wait for thread2 to finish

            thread3.start();
            thread3.join(); // Wait for thread3 to finish
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```

Q.1) Write a JSP program to check whether given number is perfect or not.

**(Use include directive) **

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1" pageEncoding="ISO-8859-1"%>
<%@ include file="header.jsp" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>Perfect Number Check</title>
</head>
<body>
```

```
<h2>Enter a number to check if it's perfect:</h2>
<form method="post">
    <input type="text" name="number" required />
    <input type="submit" value="Check" />
</form>

<%
    String numberStr = request.getParameter("number");
    if (numberStr != null && !numberStr.isEmpty()) {
        int number = Integer.parseInt(numberStr);
        int sum = 0;
        for (int i = 1; i <= number / 2; i++) {
            if (number % i == 0) {
                sum += i;
            }
        }

        if (sum == number) {
%>
            <p style="color:green; font-size:18px;">The number <%= number %> is a perfect number.</p>
<%
        } else {
%>
            <p style="color:red; font-size:18px;">The number <%= number %> is not a perfect number.</p>
<%
        }
```

```
    }
  %>
</body>
</html>
```

Q.2) Write a java program using Inter Thread Communication.

```
class SharedData {
  private int number;
  private boolean isProduced = false;

  // Producer thread
  public synchronized void produce(int number) throws InterruptedException {
    while (isProduced) {
      wait(); // Wait until the consumer consumes the number
    }
    this.number = number;
    System.out.println("Produced: " + number);
    isProduced = true;
    notify(); // Notify consumer thread
  }

  // Consumer thread
  public synchronized void consume() throws InterruptedException {
    while (!isProduced) {
      wait(); // Wait until the producer produces the number
    }
    System.out.println("Consumed: " + number);
```

```
        isProduced = false;
        notify(); // Notify producer thread
    }
}

class Producer extends Thread {
    private SharedData data;

    public Producer(SharedData data) {
        this.data = data;
    }

    @Override
    public void run() {
        try {
            for (int i = 1; i <= 5; i++) {
                data.produce(i);
                Thread.sleep(1000); // Simulate time taken to produce
            }
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}

class Consumer extends Thread {
    private SharedData data;
```

```
    public Consumer(SharedData data) {
        this.data = data;
    }

    @Override
    public void run() {
        try {
            for (int i = 1; i <= 5; i++) {
                data.consume();
                Thread.sleep(1500); // Simulate time taken to consume
            }
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}

public class InterThreadCommunicationExample {
    public static void main(String[] args) {
        SharedData data = new SharedData();

        Producer producer = new Producer(data);
        Consumer consumer = new Consumer(data);

        producer.start();
        consumer.start();
```

```
    }
}
```

Q.1) Write a JSP program to accept name and age of a Person and check whether he is eligible for Voting or not.

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1" pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>Voting Eligibility Check</title>
</head>
<body>
    <h2>Enter Your Details:</h2>
    <form method="post">
        Name: <input type="text" name="name" required /><br><br>
        Age: <input type="number" name="age" required /><br><br>
        <input type="submit" value="Check Eligibility" />
    </form>

    <%
        String name = request.getParameter("name");
        String ageStr = request.getParameter("age");

        if (name != null && !name.isEmpty() && ageStr != null && !ageStr.isEmpty()) {
```

```
        int age = Integer.parseInt(ageStr);

        if (age >= 18) {
    %>
        <p style="color:green; font-size:18px;">Hello <%= name %>, you are eligible for voting.</p>
    <%
        } else {
    %>
        <p style="color:red; font-size:18px;">Hello <%= name %>, you are not eligible for voting.</p>
    <%
        }
      }
    %>
</body>
</html>
```

Q.2) Write a multithreading program using Runnable interface to blink Text on the frame.

```
import javax.swing.*;
import java.awt.*;

public class BlinkingText implements Runnable {
    private JLabel label;
    private boolean isVisible = true;

    public BlinkingText(JLabel label) {
        this.label = label;
    }
```

```
@Override
public void run() {
    while (true) {
        try {
            Thread.sleep(500); // Wait for 500 milliseconds (half a second)
            if (isVisible) {
                label.setText(""); // Hide text
            } else {
                label.setText("Blinking Text!"); // Show text
            }
            isVisible = !isVisible; // Toggle visibility
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}

public static void main(String[] args) {
    // Create JFrame (window)
    JFrame frame = new JFrame("Blinking Text Example");
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setSize(400, 200);
    frame.setLayout(new FlowLayout());

    // Create JLabel for blinking text
    JLabel label = new JLabel("Blinking Text!");
```

```
        label.setFont(new Font("Arial", Font.BOLD, 30));

        // Add the label to the frame
        frame.add(label);

        // Create a BlinkingText object (Runnable) and pass the JLabel to it
        BlinkingText blinkingText = new BlinkingText(label);

        // Create a new thread and start it
        Thread blinkingThread = new Thread(blinkingText);
        blinkingThread.start();

        // Make the frame visible
        frame.setVisible(true);
    }
}
```

Q.1) Write a JSP program to check whether a given number is Armstrong or not.

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1" pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
  <meta charset="ISO-8859-1">
  <title>Armstrong Number Check</title>
</head>
<body>
  <h2>Enter a number to check if it is Armstrong:</h2>
```

```
<form method="post">
    Number: <input type="text" name="number" required /><br><br>
    <input type="submit" value="Check" />
</form>

<%
    String numberStr = request.getParameter("number");
    if (numberStr != null && !numberStr.isEmpty()) {
        int number = Integer.parseInt(numberStr);
        int originalNumber = number;
        int sum = 0;
        int digits = String.valueOf(number).length();

        while (number > 0) {
            int digit = number % 10;
            sum += Math.pow(digit, digits);
            number /= 10;
        }

        if (sum == originalNumber) {
%>
            <p style="color:green; font-size:18px;">The number <%= originalNumber %> is an Armstrong number.</p>
<%
        } else {
%>
            <p style="color:red; font-size:18px;">The number <%= originalNumber %> is not an Armstrong number.</p>
```

```
<%
        }
    }
%>
</body>
</html>
```

Q.2) Write a program to display information about the ResultSet like number of columns available in the ResultSet and SQL type of the column. Use Person table.
(Use ResultSetMetaData).

```
import java.sql.*;

public class ResultSetMetadataExample {
    public static void main(String[] args) {
        // Database connection variables
        String url = "jdbc:mysql://localhost:3306/your_database_name"; // Change to your database URL
        String username = "your_username";  // Change to your database username
        String password = "your_password";  // Change to your database password

        // SQL query to fetch all data from the Person table
        String query = "SELECT * FROM Person";

        try (Connection connection = DriverManager.getConnection(url, username, password);
            Statement statement = connection.createStatement();
            ResultSet resultSet = statement.executeQuery(query)) {
```

```
            // Getting ResultSetMetaData
            ResultSetMetaData rsMetaData = resultSet.getMetaData();

            // Get the number of columns in the ResultSet
            int columnCount = rsMetaData.getColumnCount();
            System.out.println("Number of columns: " + columnCount);

            // Display column information
            for (int i = 1; i <= columnCount; i++) {
                String columnName = rsMetaData.getColumnName(i);
                int columnType = rsMetaData.getColumnType(i);
                String columnTypeName = rsMetaData.getColumnTypeName(i);

                System.out.println("Column " + i + ": ");
                System.out.println("Name: " + columnName);
                System.out.println("SQL Type: " + columnType + " (" + columnTypeName + ")");
                System.out.println();
            }

        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

Q.1) Write a JSP program to check whether given number is Armstrong or not.

(Use include directive)

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

<!DOCTYPE html>

<html>

<head>

  <meta charset="ISO-8859-1">

  <title>Armstrong Number Check</title>

</head>

<body>

  <h2>Enter a number to check if it is Armstrong:</h2>

  <form method="post">

    Number: <input type="text" name="number" required /><br><br>

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

  </form>


  <%

    String numberStr = request.getParameter("number");

    if (numberStr != null && !numberStr.isEmpty()) {

      int number = Integer.parseInt(numberStr);

      int originalNumber = number;

      int sum = 0;

      int digits = String.valueOf(number).length();


      while (number > 0) {

        int digit = number % 10;

        sum += Math.pow(digit, digits);
```

```
            number /= 10;
        }

        if (sum == originalNumber) {
    %>
            <p style="color:green; font-size:18px;">The number <%= originalNumber %> is an Armstrong number.</p>
    <%
        } else {
    %>
            <p style="color:red; font-size:18px;">The number <%= originalNumber %> is not an Armstrong number.</p>
    <%
        }
      }
    %>
</body>
</html>
```

Q.2) Write a JDBC program to display all the countries located in the West Region. Create a table Country with fields (Name, continent, Capital,Region). Insert values in the table. [20 Marks

```
-- Create the Country Table
CREATE TABLE Country (
    Name VARCHAR(100),
    Continent VARCHAR(50),
    Capital VARCHAR(100),
```

```
    Region VARCHAR(50)
);

-- Insert some sample data into the Country table
INSERT INTO Country (Name, Continent, Capital, Region)
VALUES ('United States', 'North America', 'Washington, D.C.', 'West'),
       ('Canada', 'North America', 'Ottawa', 'West'),
       ('Mexico', 'North America', 'Mexico City', 'West'),
       ('Brazil', 'South America', 'Brasilia', 'West'),
       ('Argentina', 'South America', 'Buenos Aires', 'West'),
       ('Australia', 'Oceania', 'Canberra', 'West');

import java.sql.*;

public class WestRegionCountries {
    public static void main(String[] args) {
        // Database connection variables
        String url = "jdbc:mysql://localhost:3306/your_database_name"; // Change to your database URL
        String username = "your_username";  // Change to your database username
        String password = "your_password";  // Change to your database password

        // SQL query to select countries in the West region
        String query = "SELECT Name, Capital FROM Country WHERE Region = 'West'";

        try (Connection connection = DriverManager.getConnection(url, username, password);
             Statement statement = connection.createStatement();
             ResultSet resultSet = statement.executeQuery(query)) {
```

```
            // Display the header
            System.out.println("Countries in the West Region:");
            System.out.println("----------------------------------------------");

            // Iterate through the ResultSet and print each country's name and capital
            while (resultSet.next()) {
                String countryName = resultSet.getString("Name");
                String capital = resultSet.getString("Capital");
                System.out.println("Country: " + countryName + ", Capital: " + capital);
            }

        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

Q.1) Write a JSP program accept a number and display reverse number.

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1" pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>Reverse Number</title>
</head>
```

```
<body>
    <h2>Enter a number to get its reverse:</h2>
    <form method="post">
        Number: <input type="text" name="number" required /><br><br>
        <input type="submit" value="Reverse" />
    </form>

    <%
        String numberStr = request.getParameter("number");
        if (numberStr != null && !numberStr.isEmpty()) {
            int number = Integer.parseInt(numberStr);
            int reversedNumber = 0;

            // Reversing the number
            while (number != 0) {
                int digit = number % 10;
                reversedNumber = reversedNumber * 10 + digit;
                number /= 10;
            }
    %>
            <p style="color:green; font-size:18px;">The reverse of the number is: <%= reversedNumber
%></p>
    <%
        }
    %>
</body>
</html>
```

Q.2) Write a JDBC program to perform search operation on Person table.

1. Search all the person born in the year 1986.

2. Search all the females born between 2000- 2005.

```
-- Create the Person Table
CREATE TABLE Person (
   id INT PRIMARY KEY AUTO_INCREMENT,
   name VARCHAR(100),
   gender VARCHAR(10),
   birth_year INT
);


-- Insert sample data into Person table
INSERT INTO Person (name, gender, birth_year)
VALUES ('Alice', 'Female', 1986),
      ('Bob', 'Male', 1986),
      ('Charlie', 'Male', 2000),
      ('Diana', 'Female', 2003),
      ('Eva', 'Female', 2001),
      ('Frank', 'Male', 1990);


import java.sql.*;


public class PersonSearch {


   public static void main(String[] args) {
       // Database connection variables
      String url = "jdbc:mysql://localhost:3306/your_database_name";  // Change to your database
URL
```

```
String username = "your_username";  // Change to your database username
String password = "your_password";  // Change to your database password

// Query 1: Search all persons born in 1986
String query1 = "SELECT * FROM Person WHERE birth_year = 1986";

// Query 2: Search all females born between 2000 and 2005
String query2 = "SELECT * FROM Person WHERE gender = 'Female' AND birth_year BETWEEN 2000 AND 2005";

try (Connection connection = DriverManager.getConnection(url, username, password);
     Statement statement = connection.createStatement()) {

    // Execute first query: Persons born in 1986
    ResultSet resultSet1 = statement.executeQuery(query1);
    System.out.println("Persons born in 1986:");
    while (resultSet1.next()) {
        String name = resultSet1.getString("name");
        String gender = resultSet1.getString("gender");
        int birthYear = resultSet1.getInt("birth_year");
        System.out.println(name + " | " + gender + " | " + birthYear);
    }

    System.out.println("\n---------------------------------\n");

    // Execute second query: Females born between 2000 and 2005
    ResultSet resultSet2 = statement.executeQuery(query2);
    System.out.println("Females born between 2000 and 2005:");
```

```
            while (resultSet2.next()) {
                String name = resultSet2.getString("name");
                String gender = resultSet2.getString("gender");
                int birthYear = resultSet2.getInt("birth_year");
                System.out.println(name + " | " + gender + " | " + birthYear);
            }

        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```