```
//1
import java.util.*;

class primeNumber

{

   void primeMethod() {

      int i, m = 0, flag = 0;
      int n; // it is no to be checked

      System.out.println("Enter number");
      Scanner rk = new Scanner(System.in);
      n = rk.nextInt();

      m = n / 2;

      if (n == 0 || n == 1) {
         System.out.println(n + "is not prime number");
      } else {
         for (i = 2; i <= m; i++) {
            if (n % i == 0) {
               System.out.println(n + "is not prime number");
               flag = 1;
               break;
            }
         }
         if (flag == 0) {

            System.out.println(n + "is prime number");
         }
      }
   }}

class armstrongNumber

{

   void armsMethod()

   {
      int number, originalNumber, remainder, result = 0;

      System.out.println("Enter number");

      Scanner rks = new Scanner(System.in);

      number = rks.nextInt();

      originalNumber = number;
```

```
        while (originalNumber != 0)

        {

            remainder = originalNumber % 10;
            result += Math.pow(remainder, 3);
            originalNumber /= 10;

        }

        if (result == number) {
            System.out.println(number + " is an Armstrong number.");
        } else {

            System.out.println(number + " is not an Armstrong number.");
        }
    }}

public class rks

{

    public static void main(String[] args)

    {

        primeNumber prime = new primeNumber();
        prime.primeMethod();
        armstrongNumber arms = new armstrongNumber();
        arms.armsMethod();

    }}


//2
class Shape {
    public void print_shape() {
        System.out.println("this is shape");
    }}

class Rectangle extends Shape {
    public void print_rectangle() {
        System.out.println("this is rectangle shape");
    }
}

class Circle extends Shape {
    public void print_circle() {
        System.out.println("this is circle shape");
    }}
```

```
class Square extends Rectangle {
   public void print_square() {
      System.out.println("this is square shape");
   }}

public class inhri_java {
   public static void main(String[] args) {
      Square sq = new Square();
      sq.print_shape();
      sq.print_rectangle();

      Circle ca = new Circle();
      ca.print_circle();
 }}


//3
abstract class Shape
{
abstract void RectangleArea(float length, float breadth);
abstract void SquareArea(float radius);
abstract void CircleArea(float side);
}
class Area extends Shape
{
double Area=0;
void RectangleArea(float length,float breadth)
{
Area=length*breadth;
System.out.println("Area of rectangle is:"+Area);
}
void SquareArea(float Side)
{
Area=Side*Side;
System.out.println("Area of Square is:"+Area);
}
void CircleArea(float radius)
{
Area=(radius*radius)*3.14;
System.out.println("Area of Circle is:"+Area);
}}
public class objArea
{
 public static void main(String[] args)
{
Area a=new Area();
a.RectangleArea(5.5f,7f);
a.SquareArea(5f);
a.CircleArea(4);
}}
```

```
//4
//sql query if nedded: CREATE TABLE IF NOT EXISTS users (id INT PRIMARY KEY AUTO_INCREMENT, name VARCHAR(255), age INT)
import java.sql.*;


public class myDB {


   public static void main(String[] args) {
      Connection connection = null;
      Statement statement = null;
      PreparedStatement preparedStatement = null;


      try {


         String url = "jdbc:mysql://localhost:3306/rk";
         String username = "root";
         String password = "RK123@";


         // Establish connection
         connection = DriverManager.getConnection(url, username, password);


         // Create a table (if it doesn't exist)
         statement = connection.createStatement();


         // Corrected SQL statement for table creation
         String createTableSql = "CREATE TABLE IF NOT EXISTS users (id INT PRIMARY KEY AUTO_INCREMENT, name VARCHAR(255), age INT)";
         statement.execute(createTableSql);
         statement.close();


         // Insert values into the table (using PreparedStatement for security)
```

```
            preparedStatement = connection.prepareStatement("INSERT INTO users (name, age) VALUES
(?, ?)");
            preparedStatement.setString(1, "Rushikesh Kore");
            preparedStatement.setInt(2, 20);
            preparedStatement.executeUpdate();
            preparedStatement.close();

            System.out.println("Values inserted successfully!");

        } catch (SQLException e) {
            e.printStackTrace();
        } finally {
            try {
                if (connection != null) {
                    connection.close();
                }
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }
}

//5.1
//sql query if nedded: CREATE TABLE IF NOT EXISTS users (id INT PRIMARY KEY AUTO_INCREMENT,
name VARCHAR(255), age INT)

import java.sql.*;

public class dbmd {
    public static void main(String[] args) {
        try {
```

```
        Class.forName("com.mysql.cj.jdbc.Driver"); // MySQL database connection

        String url = "jdbc:mysql://localhost:3306/rk";
        String username = "root", password = "RK123@";
        Connection conn = DriverManager.getConnection(url, username, password);

        DatabaseMetaData dbmd = conn.getMetaData();
        System.out.println("Driver Name :" + dbmd.getDriverName());
        System.out.println("Driver Vsersion :" + dbmd.getDriverVersion());
        System.out.println("UserName :" + dbmd.getUserName());
        System.out.println("Database Product Name:" + dbmd.getDatabaseProductName());
        System.out.println("Database Product Version :" + dbmd.getDatabaseProductVersion());

        conn.close();

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

//5.2
//sql query if nedded: CREATE TABLE IF NOT EXISTS users (id INT PRIMARY KEY AUTO_INCREMENT,
name VARCHAR(255), age INT)

import java.sql.*;

public class dbmd2 {

   public static void main(String[] args) {
      String databaseName = "rk"; // Replace with your actual database name
```

```
        String username = "root"; // Replace with your actual username
        String password = "RK123@"; // Replace with your actual password
        String query = "select * from users"; // Replace with your desired query

        try (Connection connection = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/" + databaseName, username, password)) {

            PreparedStatement preparedStatement = connection.prepareStatement(query);
            ResultSet resultSet = preparedStatement.executeQuery();
            ResultSetMetaData rsmd = resultSet.getMetaData();

            // Print total number of columns
            System.out.println("Total Number of columns: " + rsmd.getColumnCount());

            System.out.println("1 st Column name:" + rsmd.getColumnLabel(1));
            System.out.println("2 nd Column name:" + rsmd.getColumnLabel(2));
            System.out.println("3 rd Column name:" + rsmd.getColumnLabel(3));
        } catch (SQLException e) {
            System.err.println("Error connecting to database or executing query:");
            e.printStackTrace();
        }
    }
}

//6
package rushi;

import java.io.IOException;
import java.io.PrintWriter;
import jakarta.servlet.ServletException;
import jakarta.servlet.http.HttpServlet;
```

```
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

public class firstservlet extends HttpServlet {

   protected void processRequest(HttpServletRequest request, HttpServletResponse response)
         throws ServletException, IOException {
      response.setContentType("text/html;charset=UTF-8");
      try (PrintWriter out = response.getWriter()) {

         out.println("<!DOCTYPE html>");
         out.println("<html>");
         out.println("<head>");
         out.println("<title>Servlet firstservlet</title>");
         out.println("</head>");
         out.println("<body>");
         out.println("<h1>Hello World...!"+ "</h1>");
         out.println("</body>");
         out.println("</html>");
      }
   }

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

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

```
        processRequest(request, response);
    }

    @Override
    public String getServletInfo() {
        return "Short description";
    }
}

//7
import java.sql.*;

public class App {
    public static final String DBURL = "jdbc:mysql://localhost:3306/rk";
    public static final String DBUSER = "root";
    public static final String DBPASS = "RK123@";

    public static void main(String[] args) {

        try {
            // Driver loading
            // Class.forName("com.mysql.cj.jdbc.Driver");
            // Connection object
            Connection con = DriverManager.getConnection(DBURL, DBUSER, DBPASS);
            // create table
            String createTableSql = "CREATE TABLE IF NOT EXISTS emp(emp_id INT  PRIMARY KEY
AUTO_INCREMENT, empname VARCHAR(255), email VARCHAR(100),city VARCHAR(70))";
            PreparedStatement mps = con.prepareStatement(createTableSql);
            mps.execute(createTableSql);
            mps.close();
```

```
// Inert the record
String sql = "INSERT INTO emp(emp_id,empname,email,city) VALUES(?,?,?,?)";
PreparedStatement ps = con.prepareStatement(sql);
ps.setInt(1, 101);
ps.setString(2, "RK");
ps.setString(3, "rk0947@gmail.com");
ps.setString(4, "Pune");
int i = ps.executeUpdate();
if (i > 0) {
   System.out.println("A New Record inserted successfully");
} else {
   System.out.println("Sorry...!Record not inserted");
}

// Display the record
String sql1 = "SELECT * FROM emp";
Statement st = con.createStatement();
ResultSet rs = st.executeQuery(sql1);

while (rs.next()) {
   System.out.println(
         rs.getInt(1) + "" + rs.getString(2) + "" + rs.getString(3) + "" + rs.getString(4) + "");
}

// update the record
String sql2 = "UPDATE emp SET email=? WHERE empname=?";
PreparedStatement ps1 = con.prepareStatement(sql2);
ps1.setString(1, "rk@gmail.com");
ps1.setString(2, "RK");
int j = ps1.executeUpdate();
if (j > 0) {
```

```
            System.out.println("Record Updated successfully");
        } else {
            System.out.println("Sorry...!Record not Updated");
        }

        // Delete the record
        String sql3 = "DELETE FROM emp WHERE empname=?";
        PreparedStatement ps2 = con.prepareStatement(sql3);
        ps2.setString(1, "RK");

        int k = ps2.executeUpdate();
        if (k > 0) {
            System.out.println("Record deleted successfully");
        } else {
            System.out.println("Sorry...!Record not deleted");
        }

      } catch (SQLException e) {
         e.printStackTrace();
      }
   }
}
//8.1
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Home Page</title>
   </head>
   <body>
```

```
      <h1>This is First Page</h1>
      <a href="2.jsp"><h2>Click here to open Second page &#128516</h2></a>
   </body>
</html>

//8.2
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Second Page</title>
   </head>
   <body>
       <h1>This is Second Page</h1>
      <a href="3.jsp"><h2>Click here to open Last page &#128578</h2></a>
   </body>
</html>

//8.3
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Last Page</title>
   </head>
   <body>
      <h1>This is Last Page &#128557</h1>
   </body>
</html>
```