NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 17 : Write a java program to print the names of all files and sub directory of specified directory.

Objective : The objective of this program is to list and print the names of all files and subdirectories within a specified directory.

Code:

```
package java_practice_sets;

import java.io.File;

public class Practice_Question_17 {

   public static void main(String[] args) {

      File directory = new File("D:\\Coding World\\Java Programming\\Java Files");

      if (directory.isDirectory())

      {

         File[] files = directory.listFiles();

         if (files != null)

         {

            for (File file : files)

            {

               System.out.println(file.getName());

            }

         }

         else

         {

            System.out.println("The directory is empty.");

         }

      }

      else
```

```
        {
            System.out.println("Specified path is not a directory.");
        }
    }
}
```

Output :

```
Exercise 1
Exercise 2
Java_Practice_Set
out

Process finished with exit code 0
```

NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE :- BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 18: Write a java program to print only files name of a given directory.

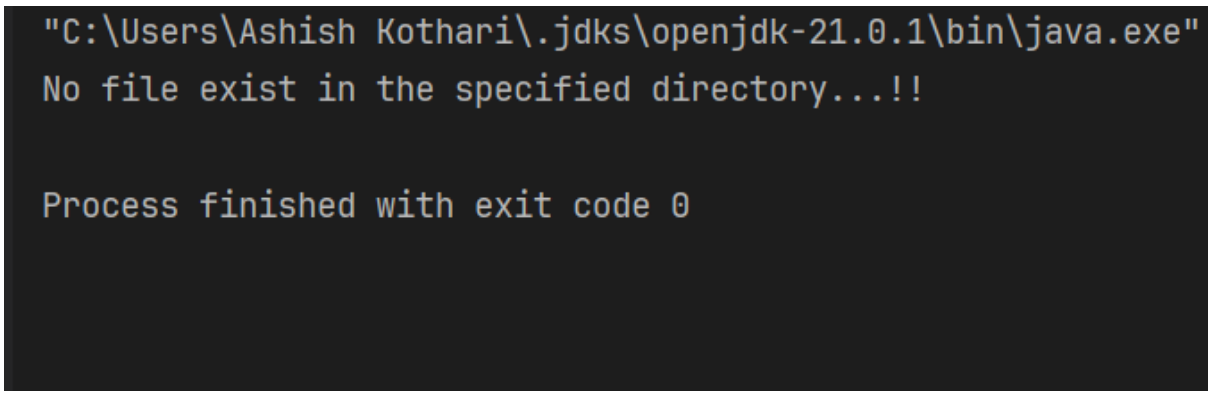
Objective: The objective of this program is to list and print only the names of files (not directories) within a specified directory.

Code :

```
package java_practice_sets;

import java.io.File;

public class Practice_Question_18 {

   public static void main(String[] args)

   {

      int count = 0;

      File directory = new File("D:\\Coding World\\Java Programming\\Java Files");

      if (directory.isDirectory())

      {

         File[] files = directory.listFiles();

         if(files != null)

         {

            for(File file: files)

            {

               if (file.isFile())

               {

                  System.out.println(file.getName());

                  count++;

               }

            }

            if(count == 0)

            {
```

```
                System.out.println("No file exist in the specified directory...!!");
            }
        }
        else
        {
            System.out.println("The directory is empty.");
        }
    }
    else
    {
        System.out.println("Specified path is not a directory.");
    }
  }
}
```

Output :-



NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE : PBC302**

Problem Statement 19 : Write a java program to print only sub-directories name of a given directory.

Objective : The objective of this program is to list and print only the names of subdirectories (not files) within a specified directory.

Code :

```
package java_practice_sets;

import java.io.File;


public class Practice_Question_19 {

   public static void main(String[] args)

   {

      int count = 0;

      File directory = new File("D:\\Coding World\\Java Programming\\Java Files");

      if (directory.isDirectory())

      {

         File[] files = directory.listFiles();

         if (files != null)

         {

            for (File file: files)

            {

               if (file.isDirectory())

               {

                  System.out.println(file.getName());

                  count++;

               }
```

```
            }
            if (count == 0)
            {
                System.out.println("No subdirectory exists in the specified directory");
            }
        }
        else
        {
            System.out.println("The directory is empty.");
        }
    }
    else
    {
        System.out.println("Specified path is not a directory.");
    }
  }
}
```

Output :

```
Exercise 1
Exercise 2
Java_Practice_Set
out

Process finished with exit code 0
```

NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 20 : Write a java program to create file named as file1.txt. Ask a user to enter three digit number. Write this 3 digit number in file1.txt using Scanner class. Now read the content from ile1.txt and check whether the number is palindrome or not. If the number is palindrome then display in console.

Objective : The objective of this program is to create a file named file1.txt then enter three numbers and check whether it is palindrome or not.

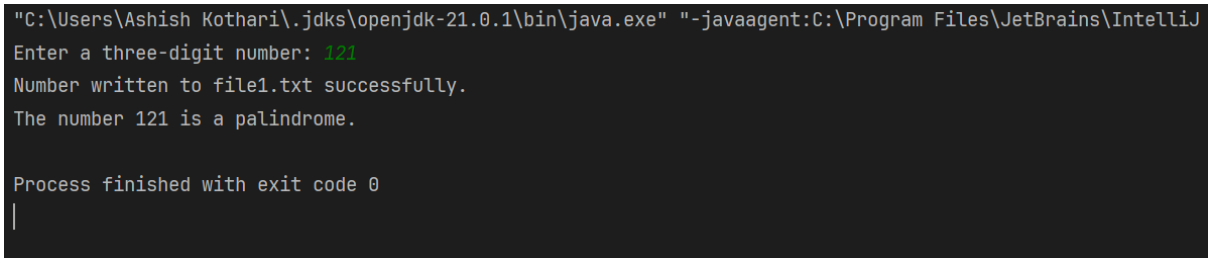
Code :

```
package java_practice_sets;

import java.io.File;

import java.io.PrintWriter;

import java.io.FileReader;

import java.io.BufferedReader;

import java.io.IOException;

import java.util.Scanner;

public class Practice_Question_20 {

   public static void main(String[] args)

   {

      WriteNumber();

      checkPalindrome();

   }

   private static void WriteNumber() {

      try {

         File file = new File("D:\\Coding World\\Java Programming\\Java
Files\\Java_Practice_Set\\src\\file1.txt");

         Scanner scanner = new Scanner(System.in);

         System.out.print("Enter a three-digit number: ");

         int number = scanner.nextInt();
```

```
            PrintWriter writer = new PrintWriter(file);
            writer.println(number);
            writer.close();
            System.out.println("Number written to file1.txt successfully.");
        }
        catch (IOException e)
        {
            e.printStackTrace();
        }
    }
    private static void checkPalindrome() {
        try {
            File file = new File("D:\\Coding World\\Java Programming\\Java
Files\\Java_Practice_Set\\src\\file1.txt");
            BufferedReader reader = new BufferedReader(new FileReader(file));
            String content = reader.readLine();
            reader.close();
            if (isPalindrome(content)) {
                System.out.println("The number " + content + " is a palindrome.");
            } else {
                System.out.println("The number " + content + " is not a palindrome.");
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    private static boolean isPalindrome(String str) {
        String reversed = new StringBuilder(str).reverse().toString();
        return str.equals(reversed);
```

```
    }

}
```

Output :


"C:\Users\Ashish Kothari\.jdks\openjdk-21.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ
Enter a three-digit number: 121
Number written to file1.txt successfully.
The number 121 is a palindrome.
Process finished with exit code 0

NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE :- BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 21 : Write a java program to read content from file1.txt using BufferedReader class. Use Integer.parseInt() method to convert character data into integer number and check that number is palindrome or not.

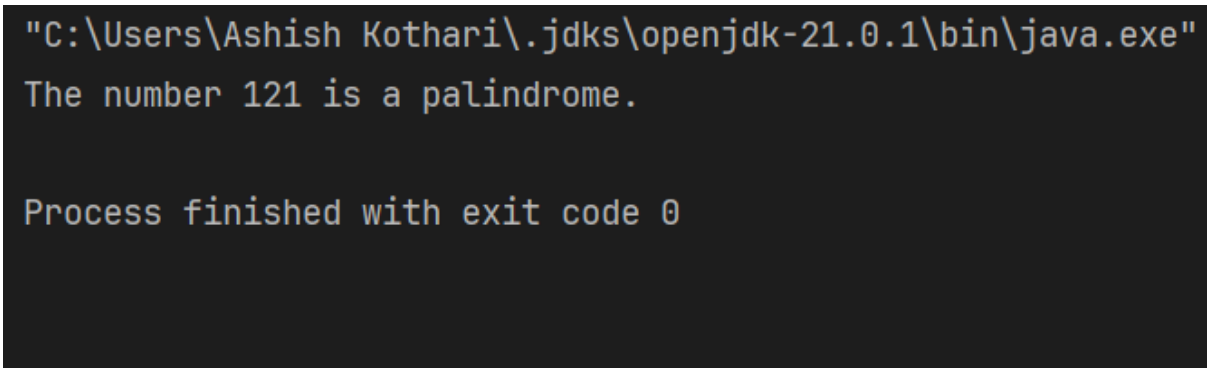
Objective : The objective of this program is to to read content from file1.txt using BufferedReader class.Then use Integer.parseInt() method to convert character data into ineger number and check that number is palindrome or not.

Code :

```
package java_practice_sets;
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

public class Practice_Question_21 {
   public static void main(String[] args)
   {
      String content = readFileContent();
      if (content != null) {
         try {
            int number = Integer.parseInt(content);
            if (isPalindrome(number)) {
               System.out.println("The number " + number + " is a palindrome.");
            }
            else {
               System.out.println("The number " + number + " is not a palindrome.");
            }
         }
         catch (NumberFormatException e) {
            System.out.println("Error: The content of file1.txt is not a valid integer.");
         }
      }
   }

   private static String readFileContent() {
      try {
         BufferedReader reader = new BufferedReader(new FileReader("D:\\Coding
World\\Java Programming\\Java Files\\Java_Practice_Set\\src\\file1.txt"));
```

```
            String content = reader.readLine();
            reader.close();
            return content;
        }
        catch (IOException e) {
            System.out.println("Error reading file1.txt: " + e.getMessage());
            return null;
        }
    }

    private static boolean isPalindrome(int number) {
        int originalNumber = number;
        int reversedNumber = 0;

        while (number > 0) {
            int digit = number % 10;
            reversedNumber = reversedNumber * 10 + digit;
            number /= 10;
        }

        return originalNumber == reversedNumber;
    }
}
```

Output :



NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE :- BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 22 : Create two files named as file2.txt and file3.txt. Write character data in both files using PrintWriter class. Now create one more file file4.txt, read the content of file2.txt and file3.txt and merge the data of file2.txt and file3.txt into file4.txt.

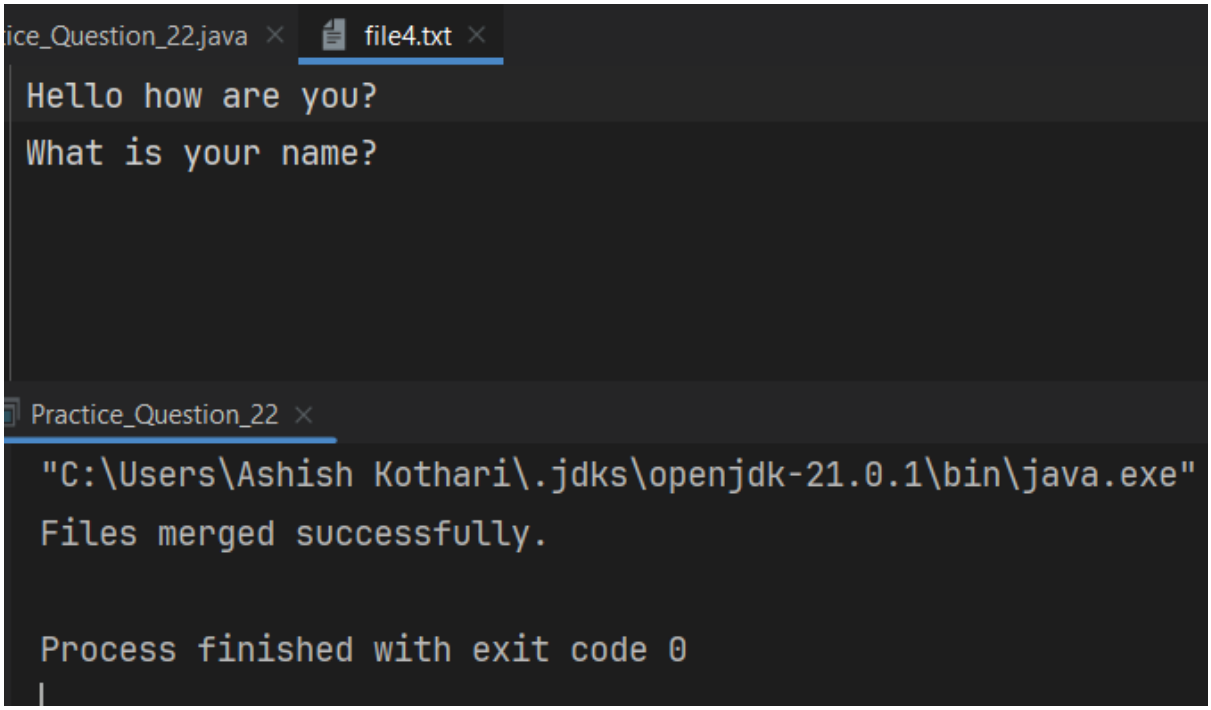
Objecive : The objective of this program is to create two files named as file2.txt and file3.txt. Write character data in both files using PrintWriter class. Now create one more file file4.txt, read the content of file2.txt and file3.txt and merge the data of file2.txt and file3.txt into file4.txt.

Code :

```
package java_practice_sets;

import java.io.*;

public class Practice_Question_22 {

   public static void main(String[] args) {

      writeToFile("D:\\Coding World\\Java Programming\\Java
Files\\Java_Practice_Set\\src\\file2.txt", "Hello how are you?");

      writeToFile("D:\\Coding World\\Java Programming\\Java
Files\\Java_Practice_Set\\src\\file3.txt", "What is your name?");

      mergeFiles("D:\\Coding World\\Java Programming\\Java
Files\\Java_Practice_Set\\src\\file2.txt", "D:\\Coding World\\Java Programming\\Java
Files\\Java_Practice_Set\\src\\file3.txt", "D:\\Coding World\\Java Programming\\Java
Files\\Java_Practice_Set\\src\\file4.txt");

   }

   private static void writeToFile(String fileName, String content) {

      try (PrintWriter writer = new PrintWriter(new FileWriter(fileName))) {

         writer.println(content);

      }

      catch (IOException e) {

         System.out.println(e);;

      }

   }
```

```
    private static void mergeFiles(String inputFile1, String inputFile2, String outputFile) {
        try (BufferedReader reader1 = new BufferedReader(new FileReader(inputFile1));
             BufferedReader reader2 = new BufferedReader(new FileReader(inputFile2));
             PrintWriter writer = new PrintWriter(new FileWriter(outputFile))) {
            String line;
            while ((line = reader1.readLine()) != null) {
                writer.println(line);
            }
            while ((line = reader2.readLine()) != null) {
                writer.println(line);
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
        System.out.println("Files merged successfully.");
    }
}
```

Output :



NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 23 : Create a swing application using Applet and awt to add and subtract two numbers using TextField, Label and Button.

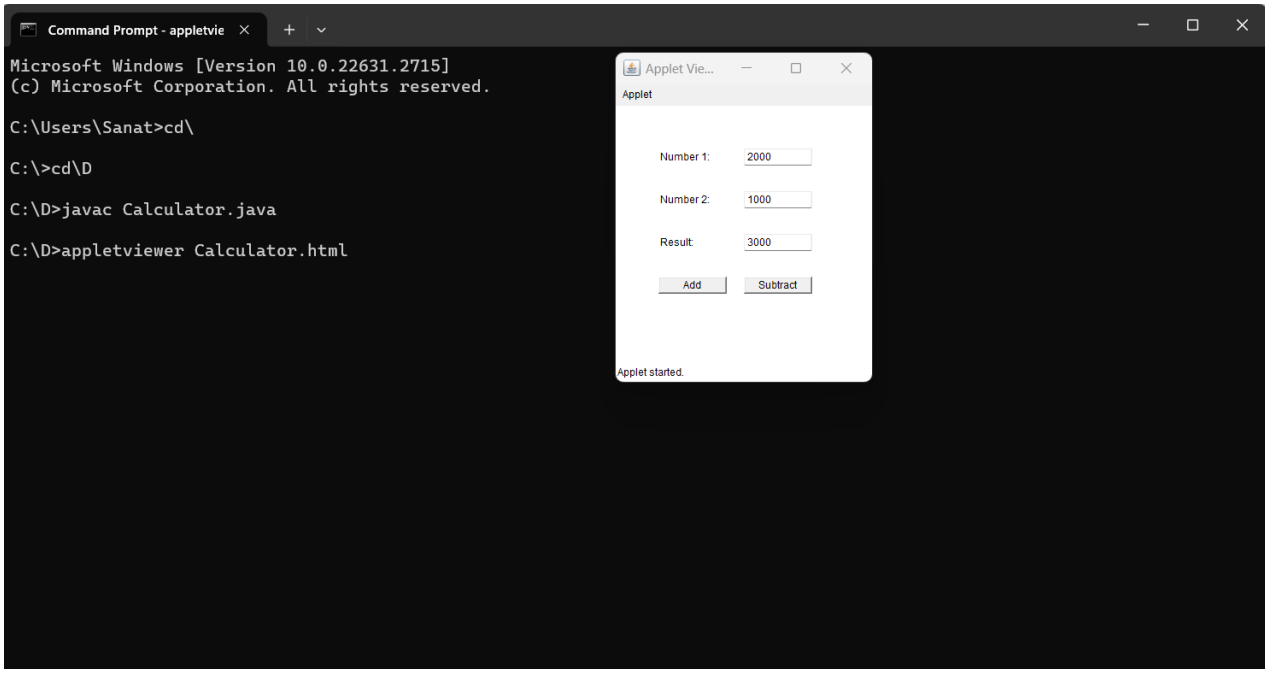
Objective :- The objective of this program to create a swing application using Applet and awt to add and subtract two numbers using TextField, Label and Button.

Code :

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class Calculator extends Applet implements ActionListener {
   Label l1, l2, l3;
   TextField t1, t2, t3;
   Button b1, b2;
   public void init() {
      l1 = new Label("Number 1:");
      t1 = new TextField(10);
      l2 = new Label("Number 2:");
      t2 = new TextField(10);
      l3 = new Label("Result:");
      t3 = new TextField(10);
      b1 = new Button("Add");
      b2 = new Button("Subtract");
      l1.setBounds(50, 50, 80, 20);
      t1.setBounds(150, 50, 80, 20);
      l2.setBounds(50, 100, 80, 20);
      t2.setBounds(150, 100, 80, 20);
      l3.setBounds(50, 150, 80, 20);
      t3.setBounds(150, 150, 80, 20);
      b1.setBounds(50, 200, 80, 20);
      b2.setBounds(150, 200, 80, 20);
      add(l1);
      add(t1);
      add(l2);
      add(t2);
      add(l3);
      add(t3);
      add(b1);
      add(b2);
```

```
        b1.addActionListener(this);
        b2.addActionListener(this);
        setLayout(null);
    }
    public void actionPerformed(ActionEvent e)
                    {
        try {
            int num1 = Integer.parseInt(t1.getText());
            int num2 = Integer.parseInt(t2.getText());
            int result = 0;
            if (e.getSource() == b1) {
                result = num1 + num2;
            } else if (e.getSource() == b2) {
                result = num1 - num2;
            }
            t3.setText(String.valueOf(result));
        } catch (NumberFormatException ex) {
            t3.setText("Invalid input");
        }
    }
}
```

Output :-



NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 24 : Write a java program to demonstrate parameter passing in Applet.

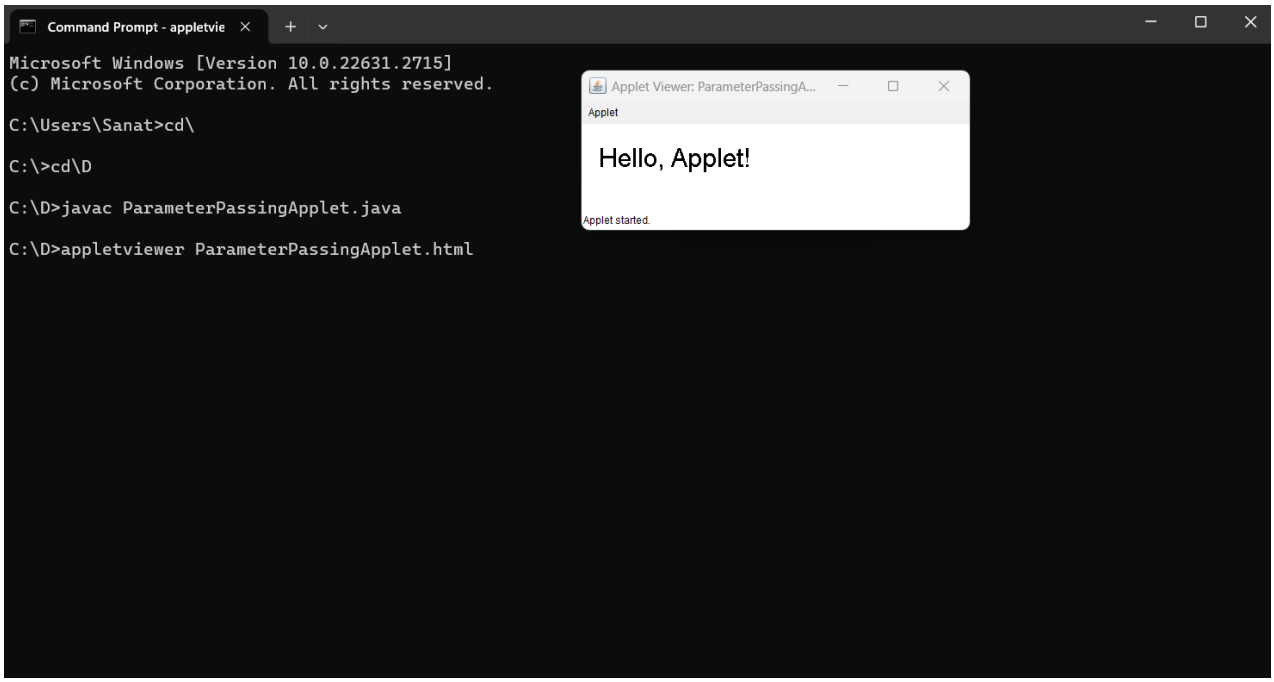
Objective : The objective of this program is to demonstrate parameter passing in Applet.

Code :

```
import java.applet.Applet;
import java.awt.Graphics;
import java.util.StringTokenizer;
public class ParameterPassingApplet extends Applet {
   private String message;
   private int fontSize;
   public void init() {
      message = getParameter("message");
      if (message == null) {
         message = "Hello, World!";
      }
      String fontSizeParam = getParameter("fontSize");
      if (fontSizeParam == null) {
         fontSize = 20;
      } else {
         try {
            fontSize = Integer.parseInt(fontSizeParam);
         } catch (NumberFormatException e) {
            fontSize = 20; // Use default if parsing fails
         }
      }
   }
```

```
    public void paint(Graphics g) {
        g.setFont(g.getFont().deriveFont((float) fontSize));
        g.drawString(message, 20, 50);
    }
}
```

Output :



NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

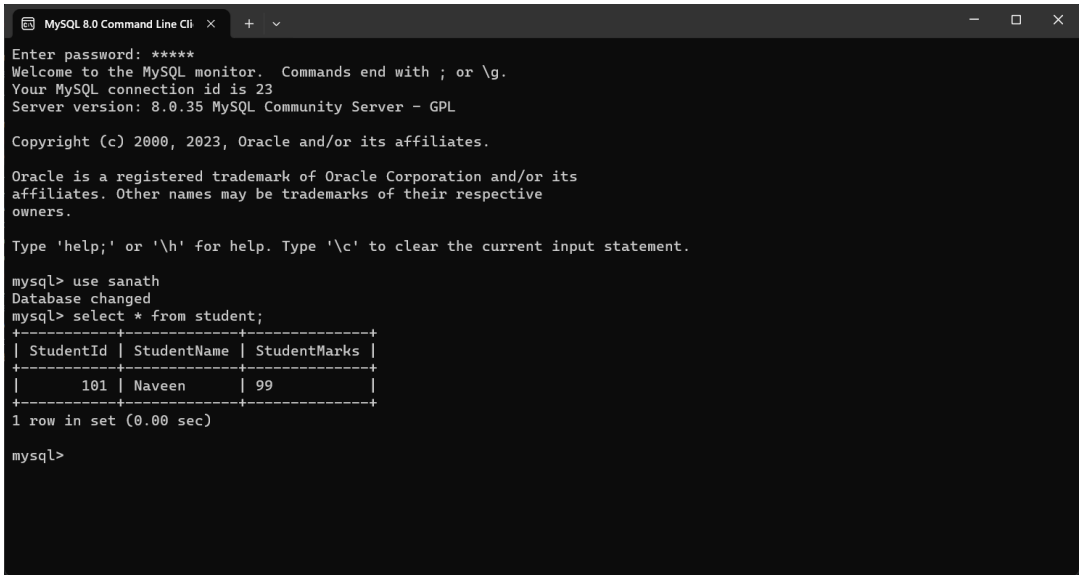
Problem Statement 25 : Assume a table in mysql/oracle database, Write a JDBC program to insert data in database table using Statement interface.

Objective : The objective of this program is to create a table in mysql dbms then write a JDBC program to insert data in database table using Statement interface.

Code :

```
package java_practice_sets;

import java.sql.*;

public class Practice_Question_25 {

   public static void main(String[] args) throws Exception

   {

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

      String url = "jdbc:mysql://localhost:3306/sanath";

      String usr = "root";

      String psw = "happy";

      Connection con = DriverManager.getConnection(url,usr,psw);

      String query = "insert into student values(101,'Naveen',99)";

      Statement st = con.createStatement();

      st.executeUpdate(query);

      System.out.println("Data insert operation done");

   }

}
```

Output :-


MySQL 8.0 Command Line Cli
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 23
Server version: 8.0.35 MySQL Community Server - GPL
Copyright (c) 2000, 2023, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> use sanath
Database changed
mysql> select * from student;
+-----------+-------------+--------------+
| StudentId | StudentName | StudentMarks |
+-----------+-------------+--------------+
| 101 | Naveen | 99 |
+-----------+-------------+--------------+
1 row in set (0.00 sec)
mysql>

NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 26: Assume a table in mysql/oracle database, Write a JDBC program to insert multiple records in database table using PreparedStatement interface.

Objectve : The objective of this program is to create a table in oracle dbms then. Write a JDBC program to insert multiple records in database table using PreparedStatement interface.

Code :

```
package com.bca;

import java.sql.*;

import java.util.Scanner;

public class InsertMultipledata {


    public static void main(String[] args) throws Exception

    {

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

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

            String usr="C##Sanath";

            String psw="hello";

            Connection con=DriverManager.getConnection(url,usr,psw);

            System.out.println("Driver Loaded");

            String q="insert into product values(?,?,?)";

            PreparedStatement pst=con.prepareStatement(q);

            String ch=null;

            int f=0;

            Scanner sc=new Scanner(System.in);

            do

            {
```

```
                System.out.println("Enter product id ");
                int id=sc.nextInt();
                System.out.println("Enter product Name ");
                String name=sc.next();
                System.out.println("Enter product price ");
            double price=sc.nextDouble();
            pst.setInt(1, id);
            pst.setString(2, name);
            pst.setDouble(3, price);
            f=pst.executeUpdate();
            System.out.println("Do you want to continue");
            ch=sc.next();

        }
while(ch.equalsIgnoreCase("y"));
        if(f>0)
        {
                System.out.println("Insert operation done");
        }
        else
        {
                System.out.println("Not done");
        }
        sc.close();
    }
}
```

Output :

```
Driver Loaded
Enter product id
101
Enter product Name
laptop
Enter product price
80000
Do you want to continue
y
Enter product id
102
Enter product Name
mobile
Enter product price
45000
Do you want to continue
y
Enter product id
103
Enter product Name
speaker
Enter product price
10000
Do you want to continue
n
Insert operation done
```

SQL Plus

```
SQL> conn
Enter user-name: C##Sanath
Enter password:
Connected.
SQL> select * from product;

 PRODUCTID PRODUCTNAM PRODCTPRICE
---------- ---------- ------------
       101 laptop           80000
       102 mobile           45000
       103 speaker          10000

SQL>
```

NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 27 : Assume a table in mysql/oracle database, Write a JDBC program to fetch a specific data from database table using ResultSet absolute() method.

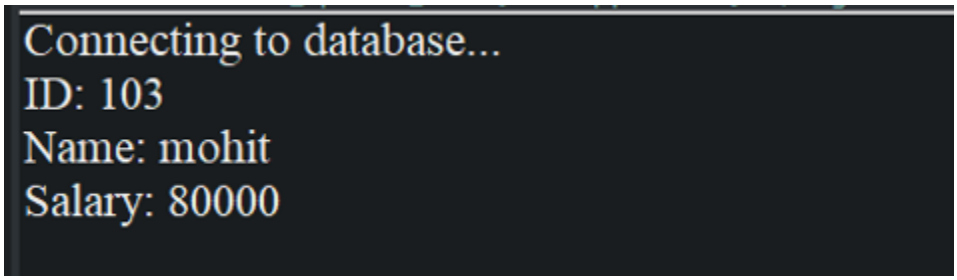
Objective : The objective of this program to write a java program and fetch specfic data from msql.

Code :-

```
package com.bca;

import java.sql.*;

public class Fetch_Specific_Data {

static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";

static final String DB_URL = "jdbc:mysql://localhost:3306/sanath";

static final String USER = "root";

static final String PASS = "happy";

public static void main(String[] args) {

    Connection conn = null;

    PreparedStatement stmt = null;

    ResultSet rs = null;

    try {

       Class.forName(JDBC_DRIVER);

       System.out.println("Connecting to database...");

       conn = DriverManager.getConnection(DB_URL, USER, PASS);

       String sql = "SELECT * FROM employee WHERE id = ?";

       stmt = conn.prepareStatement(sql);

       stmt.setInt(1, 103);

       rs = stmt.executeQuery();

       boolean found = rs.next();

       if (found) {
```

```
            int id = rs.getInt("id");
            String name = rs.getString("name");
            int salary = rs.getInt("salary");
            System.out.println("ID: " + id);
            System.out.println("Name: " + name);
            System.out.println("Salary: " + salary);
        } else {
            System.out.println("Row not found.");
        }
    } catch (SQLException se) {
        se.printStackTrace();
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        try {
            if (rs != null) rs.close();
            if (stmt != null) stmt.close();
            if (conn != null) conn.close();
        } catch (SQLException se) {
            se.printStackTrace();
        }
    }
  }
}
```

Output :



```
mysql> select * from employee;
+------+-------+--------+
| id   | name  | salary |
+------+-------+--------+
|  101 | amit  |  40000 |
|  102 | rohit |  50000 |
|  103 | mohit |  80000 |
+------+-------+--------+
3 rows in set (0.00 sec)
```

NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE : BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 28 : Assume a table in mysql/oracle database, Write a JDBC program to fetch all the data from database table using Statement interface.

Objective :- The objective of this program to write a java program and fetch all data from mysql database.

Code :

```
package com.bca;

import java.sql.*;

public class Fetch_All_Data  {

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";

   static final String DB_URL = "jdbc:mysql://localhost:3306/sanath";

   static final String USER = "root";

   static final String PASS = "happy";

   public static void main(String[] args) {

      Connection conn = null;

      Statement stmt = null;

      ResultSet rs = null;

      try {

         Class.forName(JDBC_DRIVER);

         System.out.println("Connecting to database...");

         conn = DriverManager.getConnection(DB_URL, USER, PASS);

         stmt = conn.createStatement();

         String sql = "SELECT * FROM employee";

         rs = stmt.executeQuery(sql);

         while (rs.next()) {

            int id = rs.getInt("id");

            String name = rs.getString("name");
```

```
                int salary = rs.getInt("salary");
                System.out.println("ID: " + id);
                System.out.println("Name: " + name);
                System.out.println("Salary: " + salary);
                System.out.println("----------------------");
            }
        } catch (SQLException se) {
            se.printStackTrace();
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            try {
                if (rs != null) rs.close();
                if (stmt != null) stmt.close();
                if (conn != null) conn.close();
            } catch (SQLException se) {
                se.printStackTrace();
            }
        }
    }
}
```

Output :

```
Connecting to database...
ID: 101
Name: amit
Salary: 40000
----------------------
ID: 102
Name: rohit
Salary: 50000
----------------------
ID: 103
Name: mohit
Salary: 80000
----------------------
```

```
mysql> select * from employee;
+------+-------+--------+
| id   | name  | salary |
+------+-------+--------+
|  101 | amit  |  40000 |
|  102 | rohit |  50000 |
|  103 | mohit |  80000 |
+------+-------+--------+
3 rows in set (0.00 sec)
```

NAME : ASHISH KOTHARI
ROLL NO : 17 **COURSE :- BCA**
SECTION : D1 **SEMESTER : 3**
SUBJECT : JAVA PROGRAMMING LAB **SUBJECT CODE: PBC302**

Problem Statement 29 :- Assume a table in mysql/oracle database, Write a JDBC program to perform following operation in database table.

a. Insert
b. Update
c. Delete

Using batchUpdate

Objective :- The objective of this program to write a java program too perform the Insert, Update, Delete operations from msql database.

Code :

```
package java_practice_sets;

import java.sql.*;

public class Batch_Update {

   static final String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";

   static final String DB_URL = "jdbc:mysql://localhost:3306/sanath";

   static final String USER = "root";

   static final String PASS = "happy";

   public static void main(String[] args) {

      Connection conn = null;

      Statement stmt = null;

      try {

         Class.forName(JDBC_DRIVER);

         System.out.println("Connecting to database...");

         conn = DriverManager.getConnection(DB_URL, USER, PASS);

         conn.setAutoCommit(false);

         stmt = conn.createStatement();

         // Insert
```

```
        String insertSql = "INSERT INTO employee (id, name, salary) VALUES (104, 'John',
50000)";
        stmt.addBatch(insertSql);
        // Update
        String updateSql = "UPDATE employee SET salary = 55000 WHERE id = 101";
        stmt.addBatch(updateSql);
        // Delete
        String deleteSql = "DELETE FROM employee WHERE id = 102";
        stmt.addBatch(deleteSql);
        int[] result = stmt.executeBatch();
        conn.commit();
        System.out.println("Rows affected for insert: " + result[0]);
        System.out.println("Rows affected for update: " + result[1]);
        System.out.println("Rows affected for delete: " + result[2]);
    } catch (SQLException se) {
        se.printStackTrace();
        try {
            if (conn != null) {
                conn.rollback();
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        try {
            if (stmt != null) stmt.close();
            if (conn != null) {
                conn.setAutoCommit(true);
```

```
            conn.close();
          }
        } catch (SQLException se) {
          se.printStackTrace();
        }
      }
    }
}
```

Output :


```
mysql> select * from employee;
+------+-------+--------+
| id   | name  | salary |
+------+-------+--------+
|  101 | amit  |  55000 |
|  103 | mohit |  80000 |
|  104 | John  |  50000 |
+------+-------+--------+
3 rows in set (0.00 sec)

mysql>
```