LAB CYCLE:4 DATE:10/07/23

EXPERIMENT NO:25

PACKAGES

**Aim:** Create an arithmetic package that has classes and interfaces for 4 basic arithmetic operations. Test the package by implementing all operations on 2 given numbers.

**Algorithm:**

Step 1: Create a package Arithmetic in class Calculator.

Step 2: Create an interface operations inside class with method prototype.

Step 3: Define each the method in class which implements operations.

Step 4: Import the package in another class operations by,

import Arithmetic.Calculator;

Step 5: Create the objects for the class and call each method to display the result.

**Result**: Program executed successfully and output is verified.

LAB CYCLE:4 DATE:10/07/23

EXPERIMENT NO:26

EXCEPTION CLASS

**Aim:** Write a user defined exception class to authenticate the username and password.

**Algorithm:**

Step 1: Define two classes Userexception and Passwdexception that extends Exception class with constructor.

Step 2: Define main class read the values for username and password.

Step 3: In try statement throw the exception for username length, check the condition for password. Include the catch statement for the exception.

**Result**: Program executed successfully and output is verified.

LAB CYCLE:4 DATE:10/07/23

EXPERIMENT NO:27

IMPLEMENTING THREAD

**Aim:** Define two classes one for generating multiplication table of 5 other for displaying 1st N prime numbers. Implement using threads.

**Algorithm:**

Step 1: Define a class to print the multiplication table of 5 that extends Thread class.

Step 2: Create a class to display 1st N prime numbers that extends Thread class.

Step 3: Read the count of prime numbers, set i=1 and j=2.

if(i%j==0) then

count++

Step 4: Inside the main class create objects for the two classes and call the method.

Step 5: Print the result.

**Result**: Program executed successfully and output is verified.

LAB CYCLE:5 DATE:24/07/23

EXPERIMENT NO:28

EVENT HANDLING

**Aim:** Develop a program to handle all mouse events and window events.

**Algorithm:**

Step 1: Import the awt packages for mouse and window event handling.

Step 2: Create the class extend Frame and implement MouseListener and WindowListener class.

Step 3: Create object for Frame and Label.

Step 4: Create constructor and set setLayout, setSize, setVisible, addMouseListener, addWindowListener and setBounds for Frame.

Step 5: Create method for mouse events(mouseClicked, mouseEntered, mouseExited, mouseExited, mousePressed, mouseReleased).

Step 6: Inside the main method create the object for class.

Step 7: Create the methods for window events using @Override and define the methods.

**Result**: Program executed successfully and output is verified.

LAB CYCLE:6 DATE:24/07/23

EXPERIMENT NO:29

READ FILE CONTENTS

**Aim:** Write a program to write to a file then read from the file and display the contents on the console.

**Algorithm:**

Step 1: Declare variable to store the file name.

Step 2: In try statement, create object for Buffer and Scanner class.

Step 3: Write the file contents.

Step 4: Write catch statement for exception.

Step 5: In another try statement create FileReader object and read the file.

Step 6: Display the file contents.

**Result**: Program executed successfully and output is verified.

LAB CYCLE:6 DATE:24/07/23

EXPERIMENT NO:30

COPYING FILE CONTENTS

**Aim:** Write a program to copy one file to another.

**Algorithm:**

Step 1: Import the file read and write packages.

Step 2: Read the name of source file to be copied.

Step 3: Read the name of destination file to be copied.

Step 4: Create a FileReader(source file) and FileWriter(destination file) objects.

Step 5: Create BufferedReade(filerReader) and BufferedWriter(fileWriter) for reading and writing objects.

Step 6: Read and write the contents line by line.

while ((line = bufferedReader.readLine()) != null)

{bufferedWriter.write(line); }

Step 7: Display the contents of the copied file.

**Result**: Program executed successfully and output is verified.