**Distributed and Cloud Computing Laboratory**

**B.Tech. 6thSemester**



**Name : Deepak R**

**Roll Number : 18ETCS002041**

**Department : Computer Science and Engineering**

**Faculty of Engineering & Technology**

**Ramaiah University of Applied Sciences**

**Ramaiah University of Applied Sciences**

Private University Established in Karnataka State by Act No. 15 of 2013

|  |  |
| --- | --- |
| Faculty | Engineering & Technology |
| Programme | B. Tech. in Computer Science and Engineering |
| Year/Semester | 2nd Year / 6th Semester |
| Name of the Laboratory | Distributed and Cloud Computing Laboratory |
| Laboratory Code | 19CSL316A |

# Laboratory 1

**Title of the Laboratory Exercise: Multithreaded Programs in Java**

1. **Introduction and Purpose of Experiment**

Multithreading is the ability of a single core or a multi-core processor to execute multiple [threads](https://en.wikipedia.org/wiki/Thread_(computer_science)) concurrently, supported by Java run time system. By solving this students will be able to manipulate multiple threads in a Java program.

**Aim and Objectives**

**Aim**

* To develop Java multithreaded programs

1. **Experimental Procedure**
   * 1. Analyse the problem statement
     2. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
     3. Implement the algorithm in Java language
     4. Compile the Java program
     5. Test the implemented program
     6. Document the Results
     7. Analyse and discuss the outcomes of your experiment
2. **Questions**

**Implement the following:**

* 1. Create two Java threads and display Hello World by them
  2. Create four Java threads and display the results of addition, subtraction, multiplication and division of two numbers by each thread.

1. **Calculations/Computations/Algorithms**

**4.1 Algorithm for Program to Create two Java threads and display Hello World by them**

1: Start

2: Main Class should be Created

3: Two objects of Thread class Should be declared

4: We Should Pass newly created objects of the subclass for each Thread class object with threadNo as parameter of the constructor to be declared in the subclass

5: We should Use the created thread objects to implement Runnable interface

6: Create a subclass that implements Runnable interface

7: Parameterized constructor to initialize threadNo Should be Declared.

8: Declare run() method for subclass to print message “Hello World”

9: End

**4.2 Algorithm for Program Create four Java threads and display the results of addition, subtraction, multiplication and division of two numbers by each thread.**

1: Start

2: Create a Main class

3: Declare and input two integers

4: Declare four objects of Thread class

5: Create four subclasses for Addition, Subtraction, Multiplication and Division where each implements Runnable interface

6: In the Main class, pass newly created objects of the subclasses for each of the Thread class objects with threadNo, x and y as parameters of the constructor to be declared in the subclasses

7: Declare a constructor to initialize num1, num2 and threadNo inside each of the subclasses with the values passed by the objects of Main class

8: Declare run() method for each subclass to compute sum, difference, product and quotient and print the results when invoked by the respective thread object.

9: Use the thread objects to implement Runnable interface for each subclass

10: End

1. **Presentation of Results**

**1.Program to Create two Java threads and display Hello World by them**

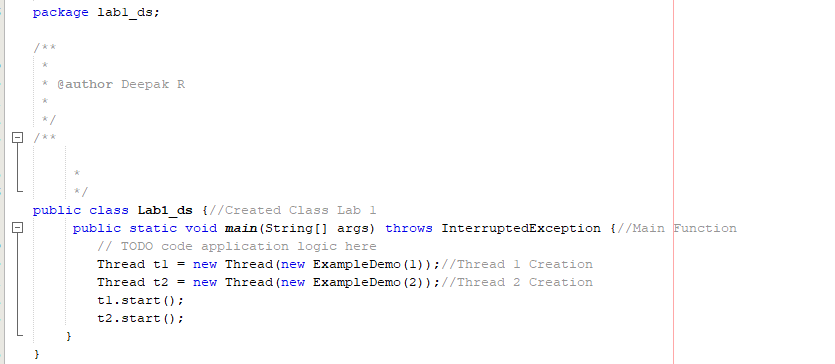


Fig 5.1 Main Class with 2 Threads

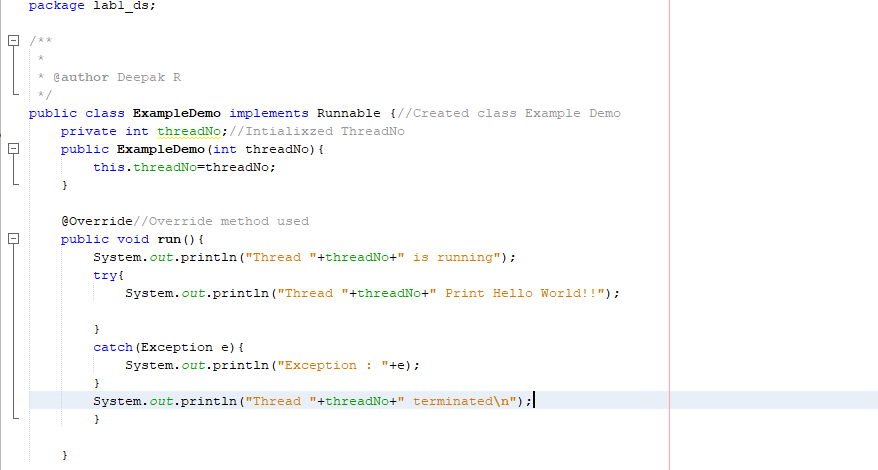


Fig 5.2 Implementation in Subclass

**2.Program to Create four Java threads and display the results of addition, subtraction, multiplication and division of two numbers by each thread.**

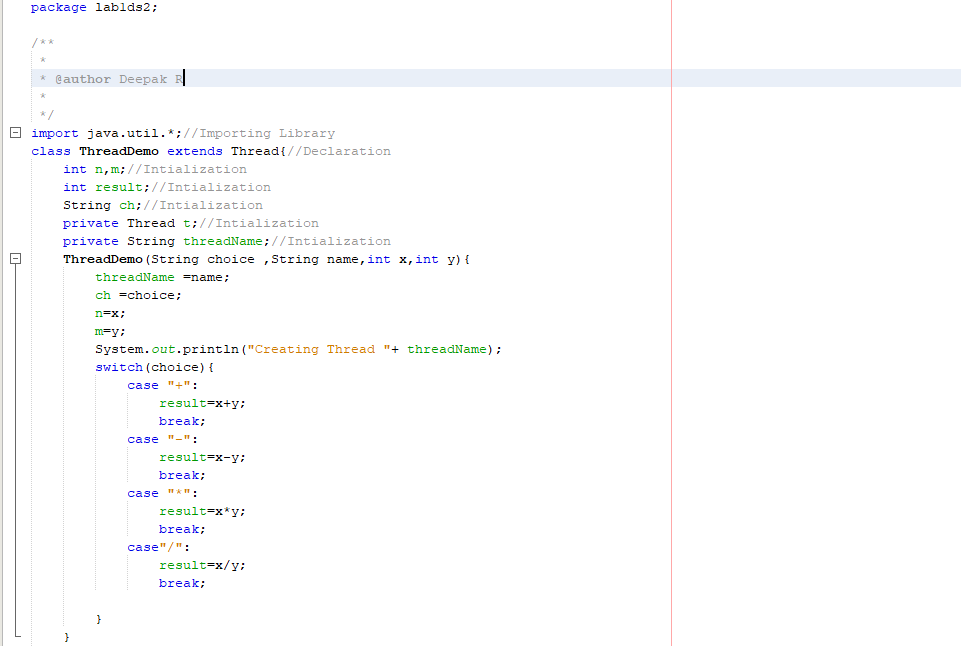
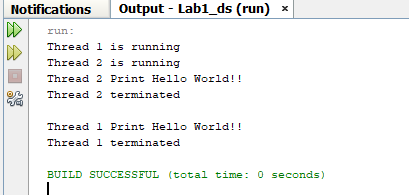
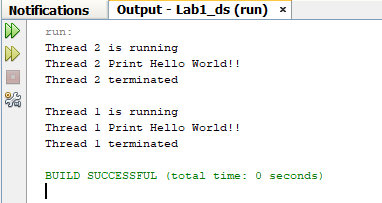
 

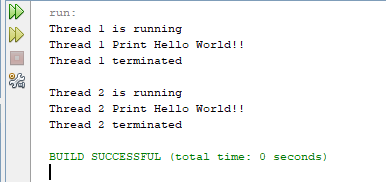
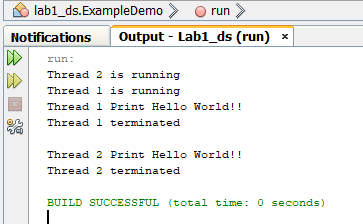
Fig 5.3 Implementation of Program 2

**Output**

**Test Cases**

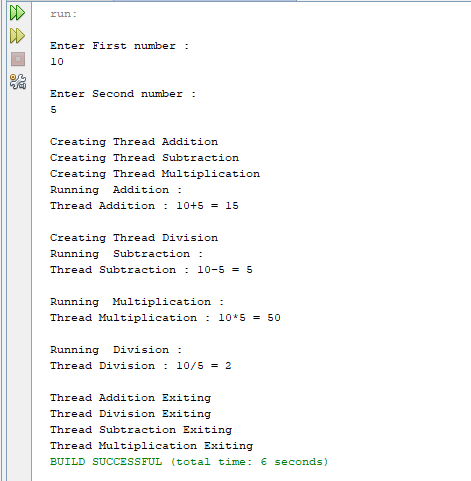
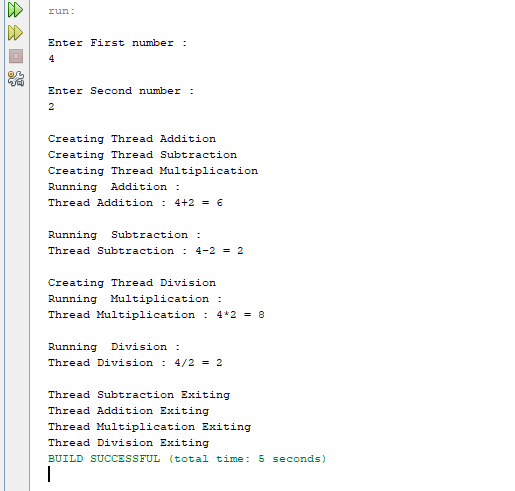
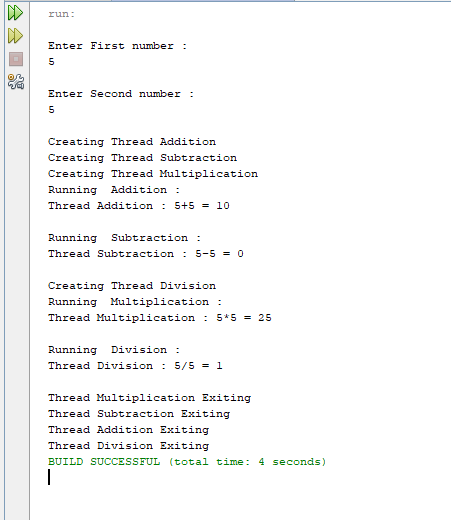
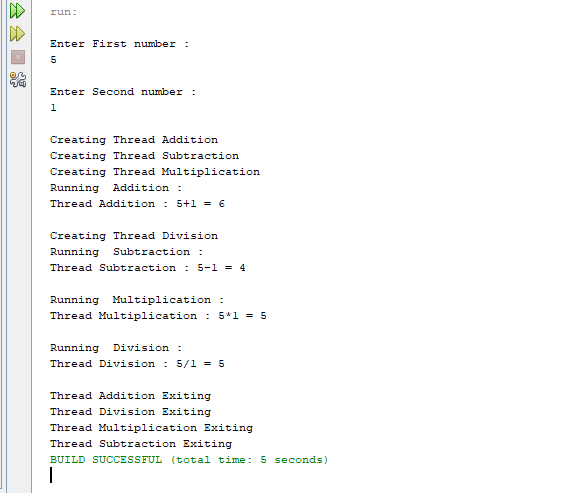
**Output for Program to Create two Java threads and display Hello World by them**

**Test cases**

**Output for** **Program to Create four Java threads and display the results of addition, subtraction, multiplication and division of two numbers by each thread.**

**Analysis and Discussions**

A multithreaded program contains two or more parts that can run concurrently. Each part of such a program is called a thread, and each thread defines a separate path of execution. Thus, multithreading is a specialized form of multitasking. Here we have done arithmetic operation by using Multithreaded method.

**1. Limitations of Experiments**

* Difficulty of writing code

Multithreaded applications are not easy to write.

* Difficulty of debugging

It is much harder to replicate an error in a multithreaded orapplication than it is to do so in a single-threaded, single-contexted application. As a result, it is more difficult, in the former case, to identify and verify root causes when errors occur.

* Difficulty of managing concurrency

The task of managing concurrency among threads is difficult and has the potential to introduce new problems into an application.

2. **Limitations of Results**

Unpredictable Results Sometime

3. **Learning happened**

We learnt how to do Multithreaded operations and its application.

4. **Recommendations**

Above code can be further optimized.