Name:	Debjit Ghosal
UID:	2023300065
Experiment No.	10A

AIM:	Multithreading
Program 1	
PROBLEM STATEMENT :	Write a Java program to perform a runnable interface, take two threads t1 and t2 and fetch the names of the thread using getName() method. Sample Output: Thread names are following: Thread A Thread B

PROGRAM:	/*
	Write a Java program to perform a runnable interface, take two threads t1 and t2 and fetch the
	names of the thread using getName() method.
	
	Sample Output:
	Thread names are following:
	Thread A
	Thread B
	*/
	
	class ThreadExample1 extends Thread implements Runnable {
	  public void run() {
	  System.out.println("Thread A");
	  }
	}

```
class ThreadExample2 implements Runnable {
    public void run() {
        System.out.println("Thread B");
    }
}
public class Thread1 {
    public static void main(String[] args) {
        Runnable ex1 = new ThreadExample1();
        Runnable ex2 = new ThreadExample2();

        Thread t1 = new Thread(ex1);
        Thread t2 = new Thread(ex2);

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

}
```

RESULT:

```
PS C:\Users\DEBJIT GHOSAL\Desktop\SPIT_CODING\PSOOP\PSOOP>  & 'C:\Program Files\Java\j
ge\9e74b28faa244b7149de54e209cd65d5\redhat.java\jdt_ws\PSOOP_797bc398\bin' 'Thread1'
Thread A
Thread B
PS C:\Users\DEBJIT GHOSAL\Desktop\SPIT_CODING\PSOOP\PSOOP>
```

Program 2

PROBLEM STATEMENT :

Write a program to print "Welcome to SPIT" and "Computer Engineering Department" continuously on the screen in Java using threads. Add a sleep method in the welcome thread to delay its execution for 500ms.

(can use wait(), notify())

PROGRAM:

```
/*
Write a java program to print "Welcome to SPIT" and "Computer Engineering Department" continuously on the screen in Java using threads. Add a sleep method in the welcome thread to delay its execution for 200ms.
```

```
Also use can use wait(), notify() operators.
*/
/*
Flow:
extend a class
again extend second class
we use synchronized class for  a proper output in a proper order
create a void method
create the main method
*/
import java.util.Scanner;
class WelcomeThread extends Thread {
   private final Object lock;

   public WelcomeThread(Object lock) {
      this.lock = lock;
   }

   public void run() {
      while (true) {
         try {
            synchronized (lock) {
               System.out.println("Welcome to SPIT");
               lock.notify();
               lock.wait(500);
            }
         } catch (InterruptedException e) {
            e.printStackTrace();
         }
      }
   }
}
class DepartmentThread extends Thread {
   private final Object lock;
   public DepartmentThread(Object lock) {
      this.lock = lock;
   }
   public void run() {
      while (true) {
```

```
            synchronized (lock) {
                try {
                    lock.wait();
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
                System.out.println("Computer Engineering
Department");
            }
        }
    }
}

public class Thread2 {
    public static void main(String[] args) {
        Object lock = new Object();

        WelcomeThread welcomeThread = new
WelcomeThread(lock);
        DepartmentThread departmentThread = new
DepartmentThread(lock);

        welcomeThread.start();
        departmentThread.start();
    }
}
```

RESULT:

```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Thread2.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Thread2
Welcome to SPIT
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
Welcome to SPIT
Computer Engineering Department
```

Program 3	
PROBLEM STATEMENT:	Print even numbers by one thread and odd numbers by another thread and print the sum of even and odd numbers by third thread. Use the Thread control function and its method (atleast one)

PROGRAM:	/*
	Write a java program to print even numbers by one thread and odd numbers by another thread and
	print the sum of even and odd numbers by third thread.
	Use the Thread control function and its method (atleast one)
	*/
	
	import java.util.Scanner;
	
	class EvenThread extends Thread {
	public void run() {
	for (int i = 2; i <= 10; i += 2) {
	System.out.println("Even Thread: " + i);
	try {
	Thread.sleep(500);
	} catch (InterruptedException e) {
	e.printStackTrace();
	}
	}
	}
	}
	
	class OddThread extends Thread {
	public void run() {
	for (int i = 1; i <= 10; i += 2) {
	System.out.println("Odd Thread: " + i);
	try {
	Thread.sleep(500);
	} catch (InterruptedException e) {
	e.printStackTrace();
	}
	}
	}
	}

```
class SumThread extends Thread {
    private int sum;

    public void run() {
        sum = 0;
        for (int i = 1; i <= 10; i++) {
            sum += i;
        }
        System.out.println("Sum Thread: Sum of numbers from
1 to 10 is " + sum);
    }
}

public class Thread3 {
    public static void main(String[] args) {
        EvenThread evenThread = new EvenThread();
        OddThread oddThread = new OddThread();
        SumThread sumThread = new SumThread();

        evenThread.start();
        oddThread.start();

        try {
            evenThread.join();
            oddThread.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }

        sumThread.start();
    }
}
```

RESULT:

```
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ javac Thread3.java
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$ java Thread3
Odd Thread: 1
Even Thread: 2
Odd Thread: 3
Even Thread: 4
Odd Thread: 5
Even Thread: 6
Odd Thread: 7
Even Thread: 8
Odd Thread: 9
Even Thread: 10
Sum Thread: Sum of numbers from 1 to 10 is 55
lenovo@lenovo-ThinkCentre-neo-50s-Gen-3:~/Desktop/2023300065$
```

CONCLUSION:	I have learnt about threads, threading and multithreading