**Lab Sheet3**. Create a single thread by implementing Runnable interface.

### Algorithm:

Step 1: Start

Step 2: Create a class Task that implements builtin interface Runnable

Step 3: Override run() to define work of thread. Use Thread.CurrentThread() to print the thread is running currently

Step 4: Create object task of Task.

Step 5: Create object of Thread by passing task as parameter

Step 5: Assign a name to thread using setName(“Name”)

Step 6: Call start() to start the thread

Step 7: Stop

Program:

import java.lang.\*; //optional

class Task implements Runnable {

public void run() {

System.out.println(Thread.currentThread()+" printing ");

System.out.println("Welcome");

}

}

public class TestThread {

public static void main(String[] args) {

Task task = new Task();

Thread t1= new Thread(task);

t1.setName("first");

t1.start();

System.out.println(Thread.currentThread()+" printing ");

System.out.println("to Java");

}

}

Output

**Q2**. Create a single thread by extending Thread class

### Algorithm:

Step 1: Start

Step 2: Create a class MyThread that extends builtin class Thread

Step 3: Override run() to define work of thread. Use Thread.CurrentThread() to print the thread is running currently

Step 4: Create object of MyThread

Step 5: Assign a name to thread using setName(“Name”)

Step 6: Call start() to start the thread

Step 7: Stop

Program:

class MyThread extends Thread {

// run() method to perform action for thread.

public void run()

{

int a= 10;

int b=12;

int result = a+b;

System.out.println(Thread.currentThread()+" started running..");

System.out.println("Sum of two numbers is: "+ result);

System.out.println(Thread.currentThread()+" completed..");

}

}

public class TestThread {

public static void main( String args[] )

{

System.out.println(Thread.currentThread()+" started");

// Creating instance of the class extend Thread class

MyThread t = new MyThread();

t.setName("first");

//calling start method to execute the run() method of the Thread class

t.start();

System.out.println(Thread.currentThread()+" completed");

}

}

Output: