ENROLLMENT NUMBER – EC2332251010074

NAME OF THE STUDENT – ANKUR CHAUDHARY

DATE – 11-07-2023

SUBJECT CODE- **V20PC101**

SUBJECT NAME - **PROGRAMMING IN JAVA**

COURSE NAME - MCA

**NAME OF THE PROGRAM / PROBLEM :**

**2.**    **Demonstrate the concept of Threads to achieve multitasking program using Thread class and Runnable Interface (separately) for the below scenario:**

**a.**    **Create an Array of 9 numbers. And create three Threads to split the task evenly among the three threads. And each thread has to add up and report the answer to the main thread where the main thread waits for the 3 threads and computes the summation of all the three threads. Note: Assign names to the threads as well.**

**AIM OF THE PROGRAM / PROBLEM :**

**The concept of Threads to achieve multitasking program using Thread class and Runnable Interface (separately).**

**PROCEDURE FOR PROGRAM / ALGORITHM:**

1. START THE PROGRAM.

2. Define a class that implements Runnable and implements the logic for adding a part of the array.

3. Declare instance variables for the array, the start index, the end index and the sum.

4. Constructor with three parameters.

5. Assign the parameters to the instance variables.

6. Initialize the sum to zero.

7. Getter method for sum.

8. Override the run method to perform the addition.

9. Define a main class to test the program.

10. Create an array of nine numbers.

11. Create three AdderRunnable objects with different parts of the array.

12. Create three Thread objects with the AdderRunnable objects as the target.

13. Wait for the threads to finish using join method.

14. Compute the total sum by adding the sums of each thread.

**PROGRAM:**

class AdderRunnable implements Runnable {

private int[] array;

private int start;

private int end;

private int sum;

public AdderRunnable(int[] array, int start, int end) {

this.array = array;

this.start = start;

this.end = end;

this.sum = 0;

}

public int getSum() {

return sum;

}

@Override

public void run() {

for (int i = start; i <= end; i++) {

sum += array[i];

}

System.out.println(Thread.currentThread().getName() + " computed sum: " + sum);

}

}

public class Main1 {

public static void main(String[] args) throws InterruptedException {

int[] array = {1, 2, 3, 4, 5, 6, 7, 8, 9};

AdderRunnable r1 = new AdderRunnable(array, 0, 2);

AdderRunnable r2 = new AdderRunnable(array, 3, 5);

AdderRunnable r3 = new AdderRunnable(array, 6, 8);

Thread t1 = new Thread(r1);

Thread t2 = new Thread(r2);

Thread t3 = new Thread(r3);

t1.setName("Thread-1");

t2.setName("Thread-2");

t3.setName("Thread-3");

t1.start();

t2.start();

t3.start();

t1.join();

t2.join();

t3.join();

int totalSum = r1.getSum() + r2.getSum() + r3.getSum();

System.out.println("Total sum: " + totalSum);

}

}

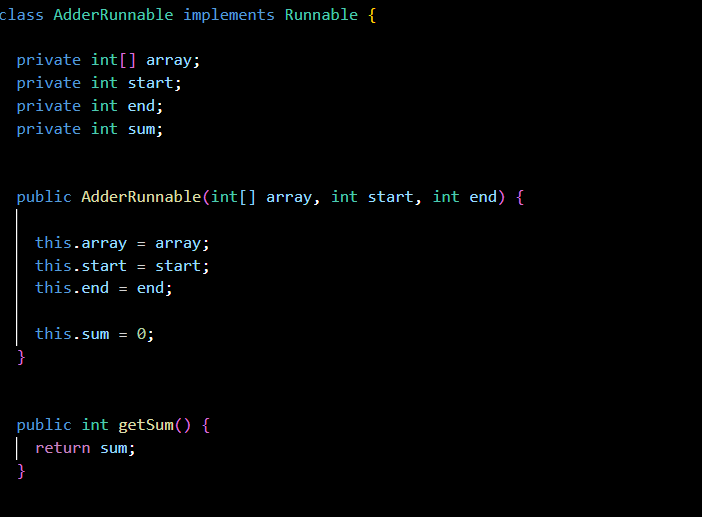
**OUTPUT OF THE PROGRAM:**

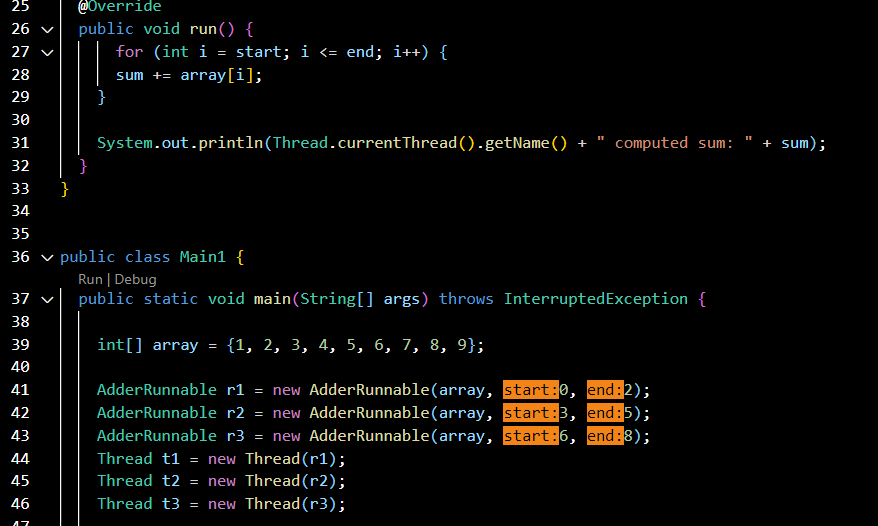
**Thread-1 computed sum: 6**

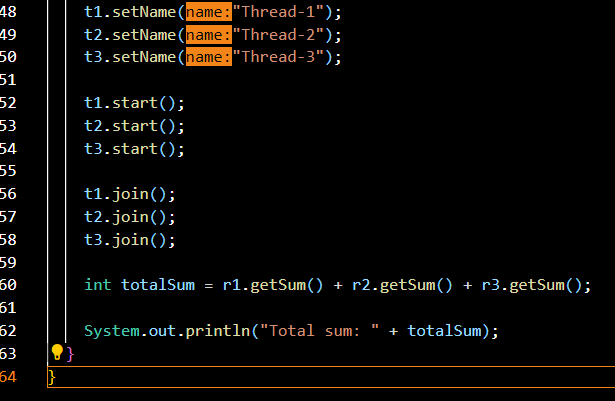
**Thread-3 computed sum: 24**

**Thread-2 computed sum: 15**

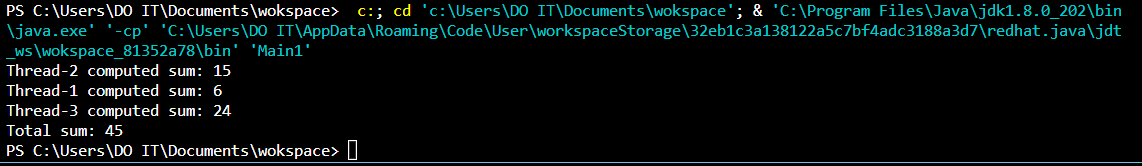
**Total sum: 45**

****

****

****

**OUTPUT :**

****

**RESULT:** Thus the program to perform **he concept of Threads to achieve multitasking program using Thread class and Runnable Interface (separately) and executed successfully.**