Mathew Jerry Meleth
17BIT0050

JAVA LAB (CSE-1007)

ASSESSMENT - 3

SCENARIO – I

1. Develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa) , time converter (hours to minutes, seconds and vice versa) using packages.
- Create one package which contain Currency converter class
- Create Another package which contain distance converter class and time converter class
- Import this two package in new class
- Based on user option call the respective function of the package class and display the output

BRIEF ABOUT YOUR APPROACH:

In this we create two packages namely ‘package1’ in java file and ‘package2’ in another java file. ‘package1’ contains the TimeConverter class and the ‘package2’ contains the DistanceConverter and CurrencyConverter Class. Then we create a main file which contains the main function and into it we we import the 3 classes from both the packages from the two different files and call these classes from different packages in the main function.

SOURCE CODE:

TimeConverter.java

```
package package1;
public class TimeConverter
{
public double HrToMin(double Hours)
{
double Minutes = 0;
Minutes = Hours * 60;
return Minutes;
}
public double MinToHour(double Minutes)
{
double Hours = 0;
Hours = Minutes / 60;
return Hours;
}
public double HrToSec(double Hours)
{
double Seconds = 0;
Seconds = Hours * 3600;
return Seconds;
}
```

```
public double SecToHour(double Seconds)
{
double Hours = 0;
Hours = Seconds / 3600;
return Hours;
}
}
```

DistanceConverter.java

```
package package2;
public class DistanceConverter
{
public double MeterToKM(double Meters)
{
double KM = 0;
KM = Meters / 1000;
return KM;
}
public double KMToMeter(double KM)
{
double Meters = 0;
Meters = KM * 1000;
return Meters;
}
public double MileToKM(double Miles)
{
double KM = 0;
KM = Miles / 0.621371;
return KM;
}
public double KMToMile(double KM)
{
double Miles = 0;
Miles = KM * 0.621371;
return Miles;
}
}
```

CurrencyConverter.java

```
package package2;
public class CurrencyConverter
{
double ER = 0;
public CurrencyConverter(double CurrentExchange)
{
ER = CurrentExchange;
}
public double DollarToINR(double Dollars)
{
```

```
double INR = 0;
INR = Dollars * ER;
return INR;
}
public double INRToDollar(double INR)
{
double Dollars = 0;
Dollars = INR / ER;
return Dollars;
}
public double EuroToINR(double Euros)
{
double INR = 0;
INR = Euros * ER;
return INR;
}
public double INRToEuro(double INR)
{
double Euros = 0;
Euros = INR / ER;
return Euros;
}
public double YenToINR(double Yens)
{
double INR = 0;
INR = Yens * ER;
return INR;
}
public double INRToYen(double INR)
{
double Yens = 0;
Yens = INR / ER;
return Yens;
}
}
```

Converter.java (Main File)

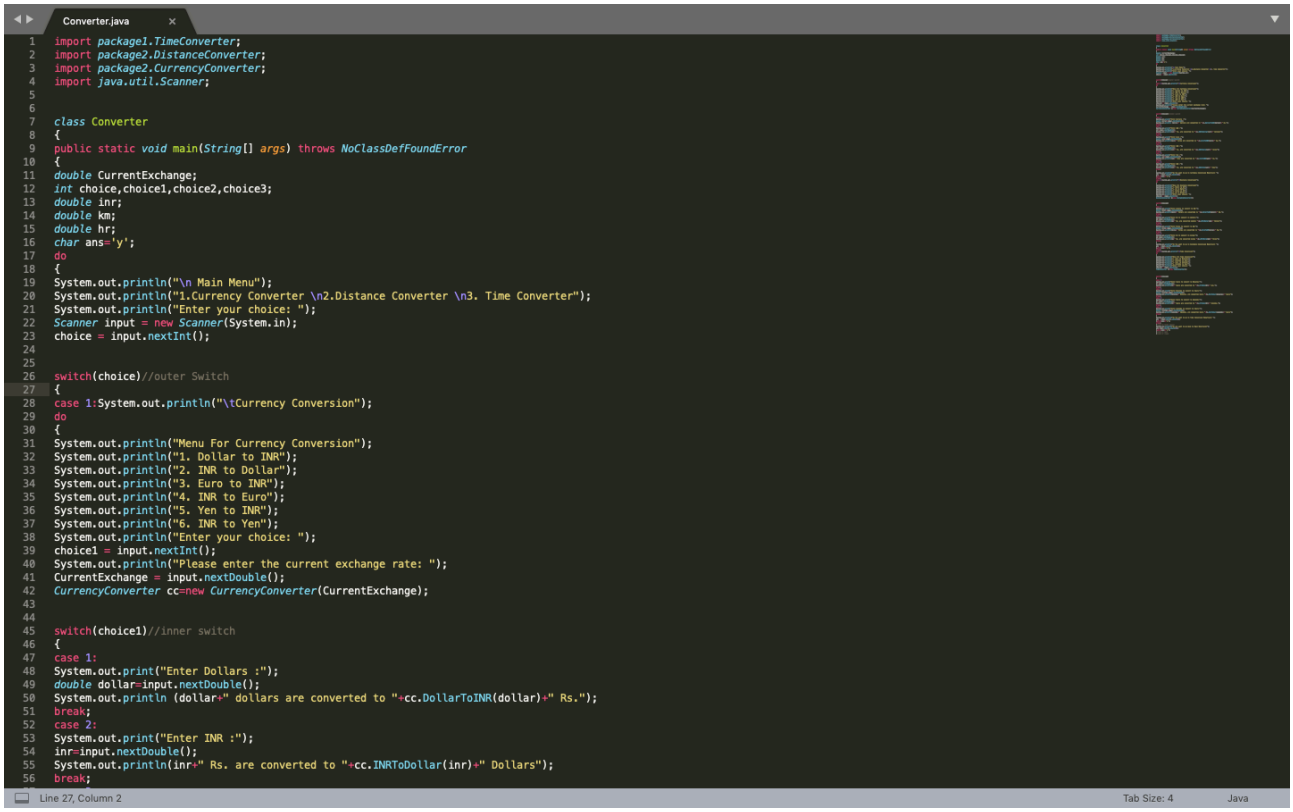
```
import package1.TimeConverter;
import package2.DistanceConverter;
import package2.CurrencyConverter;
import java.util.Scanner;


class Converter
{
public static void main(String[] args) throws NoClassDefFoundError
{
double CurrentExchange;
int choice,choice1,choice2,choice3;
double inr;
```

```
double km;
double hr;
char ans='y';
do
{
System.out.println("\n Main Menu");
System.out.println("1.Currency Converter \n2.Distance Converter \n3. Time Converter");
System.out.println("Enter your choice: ");
Scanner input = new Scanner(System.in);
choice = input.nextInt();


switch(choice)//outer Switch
{
case 1:System.out.println("\tCurrency Conversion");
do
{
System.out.println("Menu For Currency Conversion");
System.out.println("1. Dollar to INR");
System.out.println("2. INR to Dollar");
System.out.println("3. Euro to INR");
System.out.println("4. INR to Euro");
System.out.println("5. Yen to INR");
System.out.println("6. INR to Yen");
System.out.println("Enter your choice: ");
choice1 = input.nextInt();
System.out.println("Please enter the current exchange rate: ");
CurrentExchange = input.nextDouble();
CurrencyConverter cc=new CurrencyConverter(CurrentExchange);


switch(choice1)//inner switch
{
case 1:
System.out.print("Enter Dollars :");
double dollar=input.nextDouble();
System.out.println (dollar+" dollars are converted to "+cc.DollarToINR(dollar)+" Rs.");
break;
case 2:
System.out.print("Enter INR :");
inr=input.nextDouble();
System.out.println(inr+" Rs. are converted to "+cc.INRToDollar(inr)+" Dollars");
break;
case 3:
System.out.print("Enter Euro :");
double euro=input.nextDouble();
System.out.println(euro+" Euros are converted to "+cc.EuroToINR(euro)+" Rs.");
break;
case 4:
System.out.print("Enter INR :");
inr=input.nextDouble();
System.out.println(inr+" Rs. are converted to "+cc.INRToEuro(inr)+" Euros");
```

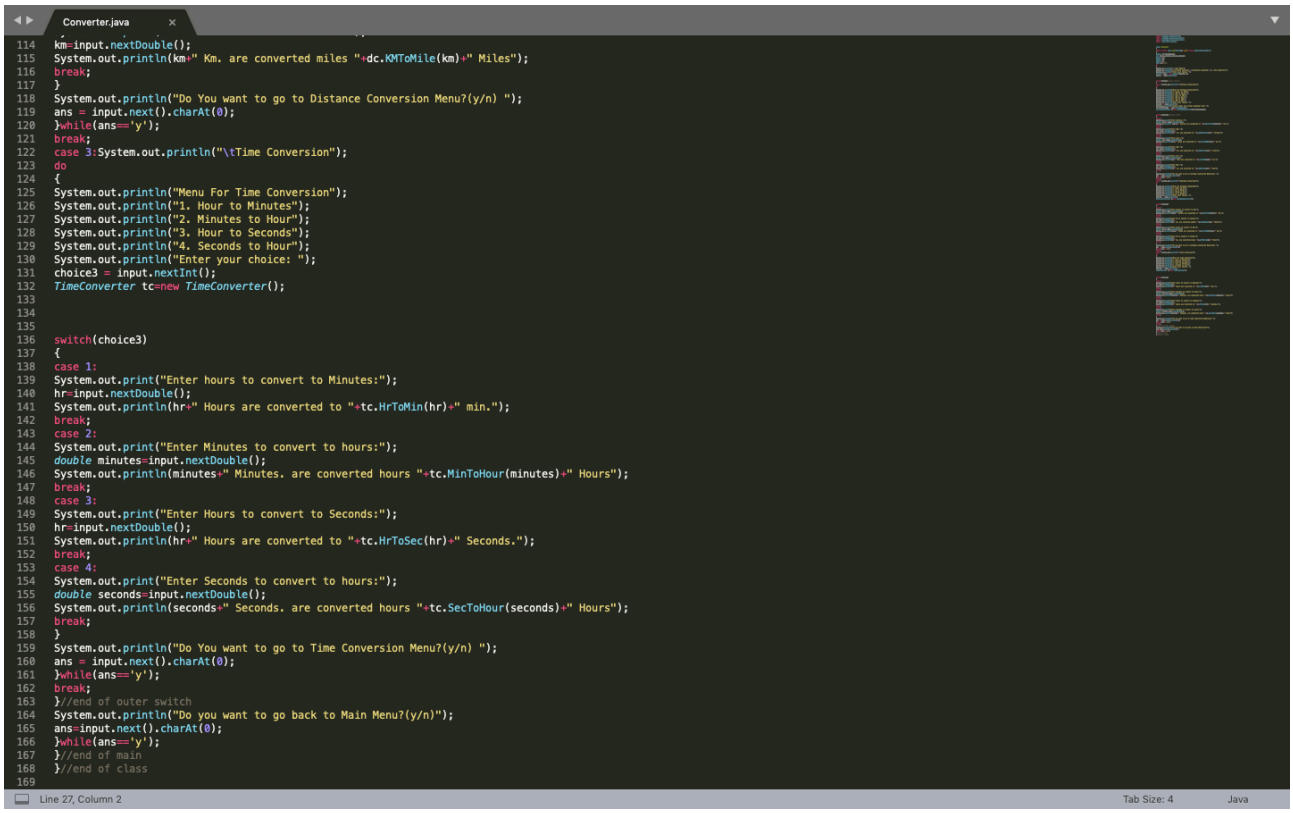
```
break;
case 5:
System.out.print("Enter Yen :");
double yen=input.nextDouble();
System.out.println(yen+" Yens are converted to "+cc.YenToINR(yen)+" Rs.");
break;
case 6:
System.out.print("Enter INR :");
inr=input.nextDouble();
System.out.println(inr+" Rs. are converted to "+cc.INRToYen(inr)+" Yens");
break;
}
System.out.println("Do You want to go to Currency Conversion Menu?(y/n) ");
ans = input.next().charAt(0);
}while(ans=='y');
break;
case 2:System.out.println("\tDistance Conversion");
do
{
System.out.println("Menu For Distance Conversion");
System.out.println("1. Meter to Km");
System.out.println("2. Km to Meter");
System.out.println("3. Miles to Km");
System.out.println("4. Km to Miles");
System.out.println("Enter your choice: ");
choice2 = input.nextInt();
DistanceConverter dc=new DistanceConverter();


switch(choice2)
{
case 1:
System.out.print("Enter meters to convert to Km:");
double meter=input.nextDouble();
System.out.println(meter+" Meters are converted to "+dc.MeterToKM(meter)+" Km.");
break;
case 2:
System.out.print("Enter Km to convert to meters:");
km=input.nextDouble();
System.out.println(km+" Km. are converted meters "+dc.KMToMeter(km)+" Meters");
break;
case 3:
System.out.print("Enter Miles to convert to Km:");
double miles=input.nextDouble();
System.out.println(miles+" Miles are converted to "+dc.MileToKM(miles)+" Km.");
break;
case 4:
System.out.print("Enter Km to convert to miles:");
km=input.nextDouble();
System.out.println(km+" Km. are converted miles "+dc.KMToMile(km)+" Miles");
break;
}
```

```
System.out.println("Do You want to go to Distance Conversion Menu?(y/n) ");
ans = input.next().charAt(0);
}while(ans=='y');
break;
case 3:System.out.println("\tTime Conversion");
do
{
System.out.println("Menu For Time Conversion");
System.out.println("1. Hour to Minutes");
System.out.println("2. Minutes to Hour");
System.out.println("3. Hour to Seconds");
System.out.println("4. Seconds to Hour");
System.out.println("Enter your choice: ");
choice3 = input.nextInt();
TimeConverter tc=new TimeConverter();



switch(choice3)
{
case 1:
System.out.print("Enter hours to convert to Minutes:");
hr=input.nextDouble();
System.out.println(hr+" Hours are converted to "+tc.HrToMin(hr)+" min.");
break;
case 2:
System.out.print("Enter Minutes to convert to hours:");
double minutes=input.nextDouble();
System.out.println(minutes+" Minutes. are converted hours "+tc.MinToHour(minutes)+" Hours");
break;
case 3:
System.out.print("Enter Hours to convert to Seconds:");
hr=input.nextDouble();
System.out.println(hr+" Hours are converted to "+tc.HrToSec(hr)+" Seconds.");
break;
case 4:
System.out.print("Enter Seconds to convert to hours:");
double seconds=input.nextDouble();
System.out.println(seconds+" Seconds. are converted hours "+tc.SecToHour(seconds)+" Hours");
break;
}
System.out.println("Do You want to go to Time Conversion Menu?(y/n) ");
ans = input.next().charAt(0);
}while(ans=='y');
break;
}//end of outer switch
System.out.println("Do you want to go back to Main Menu?(y/n)");
ans=input.next().charAt(0);
}while(ans=='y');
}//end of main
}//end of class
```

EXECUTION:



```
Converter.java
1  import package1.TimeConverter;
2  import package2.DistanceConverter;
3  import package2.CurrencyConverter;
4  import java.util.Scanner;
5
6
7  class Converter
8  {
9  public static void main(String[] args) throws NoClassDefFoundError
10 {
11 double CurrentExchange;
12 int choice,choice1,choice2,choice3;
13 double inr;
14 double km;
15 double hr;
16 char ans='y';
17 do
18 {
19 System.out.println("\n Main Menu");
20 System.out.println("1.Currency Converter \n2.Distance Converter \n3. Time Converter");
21 System.out.println("Enter your choice: ");
22 Scanner input = new Scanner(System.in);
23 choice = input.nextInt();
24
25
26 switch(choice)//outer Switch
27 {
28 case 1:System.out.println("\tCurrency Conversion");
29 do
30 {
31 System.out.println("Menu For Currency Conversion");
32 System.out.println("1. Dollar to INR");
33 System.out.println("2. INR to Dollar");
34 System.out.println("3. Euro to INR");
35 System.out.println("4. INR to Euro");
36 System.out.println("5. Yen to INR");
37 System.out.println("6. INR to Yen");
38 System.out.println("Enter your choice: ");
39 choice1 = input.nextInt();
40 System.out.println("Please enter the current exchange rate: ");
41 CurrentExchange = input.nextDouble();
42 CurrencyConverter cc=new CurrencyConverter(CurrentExchange);
43
44
45 switch(choice1)//inner switch
46 {
47 case 1:
48 System.out.print("Enter Dollars :");
49 double dollar=input.nextDouble();
50 System.out.println (dollar+" dollars are converted to "+cc.DollarToINR(dollar)+" Rs.");
51 break;
52 case 2:
53 System.out.print("Enter INR :");
54 inr=input.nextDouble();
55 System.out.println(inr+" Rs. are converted to "+cc.INRToDollar(inr)+" Dollars");
56 break;
Line 27, Column 2   Tab Size: 4   Java
```



```
Converter.java
114 km=input.nextDouble();
115 System.out.println(km+" Km. are converted miles "+dc.KMToMile(km)+" Miles");
116 break;
117 }
118 System.out.println("Do You want to go to Distance Conversion Menu?(y/n) ");
119 ans = input.next().charAt(0);
120 }while(ans=='y');
121 break;
122 case 3:System.out.println("\tTime Conversion");
123 do
124 {
125 System.out.println("Menu For Time Conversion");
126 System.out.println("1. Hour to Minutes");
127 System.out.println("2. Minutes to Hour");
128 System.out.println("3. Hour to Seconds");
129 System.out.println("4. Seconds to Hour");
130 System.out.println("Enter your choice: ");
131 choice3 = input.nextInt();
132 TimeConverter tc=new TimeConverter();
133
134
135
136 switch(choice3)
137 {
138 case 1:
139 System.out.print("Enter hours to convert to Minutes:");
140 hr=input.nextDouble();
141 System.out.println(hr+" Hours are converted to "+tc.HrToMin(hr)+" min.");
142 break;
143 case 2:
144 System.out.print("Enter Minutes to convert to hours:");
145 double minutes=input.nextDouble();
146 System.out.println(minutes+" Minutes. are converted hours "+tc.MinToHour(minutes)+" Hours");
147 break;
148 case 3:
149 System.out.print("Enter Hours to convert to Seconds:");
150 hr=input.nextDouble();
151 System.out.println(hr+" Hours are converted to "+tc.HrToSec(hr)+" Seconds.");
152 break;
153 case 4:
154 System.out.print("Enter Seconds to convert to hours:");
155 double seconds=input.nextDouble();
156 System.out.println(seconds+" Seconds. are converted hours "+tc.SecToHour(seconds)+" Hours");
157 break;
158 }
159 System.out.println("Do You want to go to Time Conversion Menu?(y/n) ");
160 ans = input.next().charAt(0);
161 }while(ans=='y');
162 break;
163 }//end of outer switch
164 System.out.println("Do you want to go back to Main Menu?(y/n)");
165 ans=input.next().charAt(0);
166 }while(ans=='y');
167 }//end of main
168 }//end of class
169
Line 27, Column 2   Tab Size: 4   Java
```

RESULT

```
D:\>cd ConversionDemo
D:\ConversionDemo>javac CurrencyConverter.java
D:\ConversionDemo>javac DistanceConverter.java
D:\ConversionDemo>javac TimeConverter.java
D:\ConversionDemo>cd..
D:\>javac Converter.java
D:\>java Converter
 Main Menu
1.Currency Converter
2.Distance Converter
3. Time Converter
Enter your choice:
1
       Currency Conversion
Menu For Currency Conversion
1. Dollar to INR
2. INR to Dollar
3. Euro to INR
4. INR to Euro
5. Yen to INR
6. INR to Yen
Enter your choice:
1
Please enter the current exchange rate:
66.21
Enter Dollars :25
25.0 dollars are converted to 1655.2499999999998 Rs.
Do You want to go to Currency Conversion Menu?(y/n)
y
Menu For Currency Conversion
1. Dollar to INR
2. INR to Dollar
3. Euro to INR
4. INR to Euro
5. Yen to INR
6. INR to Yen
Enter your choice:
4
Please enter the current exchange rate:
81.1
Enter INR :5000
5000.0 Rs. are converted to 61.652281134401974 Euros
Do You want to go to Currency Conversion Menu?(y/n)
n
Do you want to go back to Main Menu?(y/n)
y
```

```
 Main Menu
1.Currency Converter
2.Distance Converter
3. Time Converter
Enter your choice:
2
      Distance Conversion
Menu For Distance Conversion
1. Meter to Km
2. Km to Meter
3. Miles to Km
4. Km to Miles
Enter your choice:
1
Enter meters to convert to Km:4000
4000.0 Meters are converted to 4.0 Km.
Do You want to go to Distance Conversion Menu?(y/n)
n
Do you want to go back to Main Menu?(y/n)
N
```

SCENARIO – II

Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

BRIEF ABOUT YOUR APPROACH:

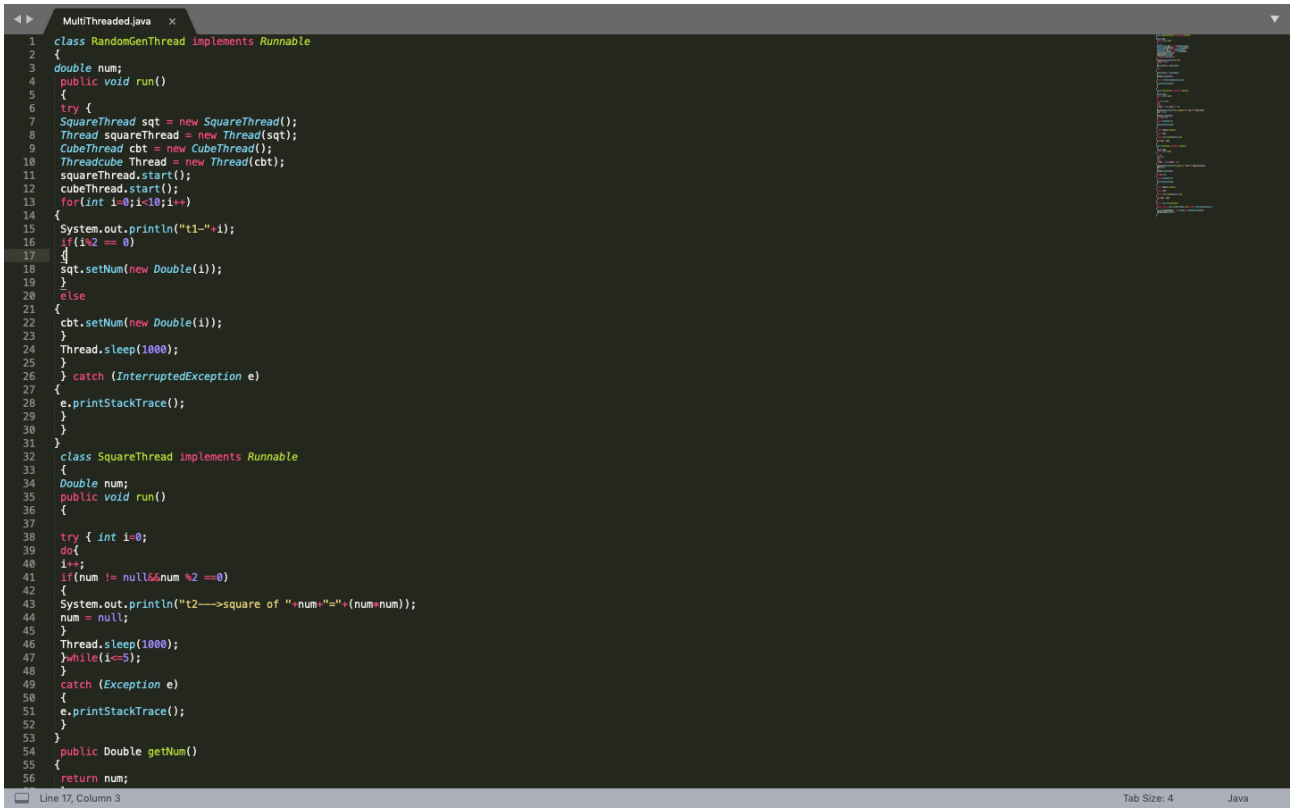
We will create three classes namely square , cube , number all extending the Thread class. The class Number creates objects of square and cube. Then according to the given conditions initializes the threads. Then we will print the output accordingly.

SOURCE CODE:

```
class RandomGenThread implements Runnable
{
double num;
 public void run()
 {
 try {
 SquareThread sqt = new SquareThread();
 Thread squareThread = new Thread(sqt);
 CubeThread cbt = new CubeThread();
 Threadcube Thread = new Thread(cbt);
 squareThread.start();
 cubeThread.start();
 for(int i=0;i<10;i++)
{
 System.out.println("t1-"+i);
 if(i%2 == 0)
 {
 sqt.setNum(new Double(i));
 }
 else
{
 cbt.setNum(new Double(i));
 }
 Thread.sleep(1000);
 }
 } catch (InterruptedException e)
{
 e.printStackTrace();
 }
 }
}
 class SquareThread implements Runnable
 {
 Double num;
 public void run()
 {

 try { int i=0;
 do{
 i++;
 if(num != null&&num %2 ==0)
 {
 System.out.println("t2--->square of "+num+"="+(num*num));
 num = null;
 }
 Thread.sleep(1000);
 }while(i<=5);
 }
 catch (Exception e)
 {
```

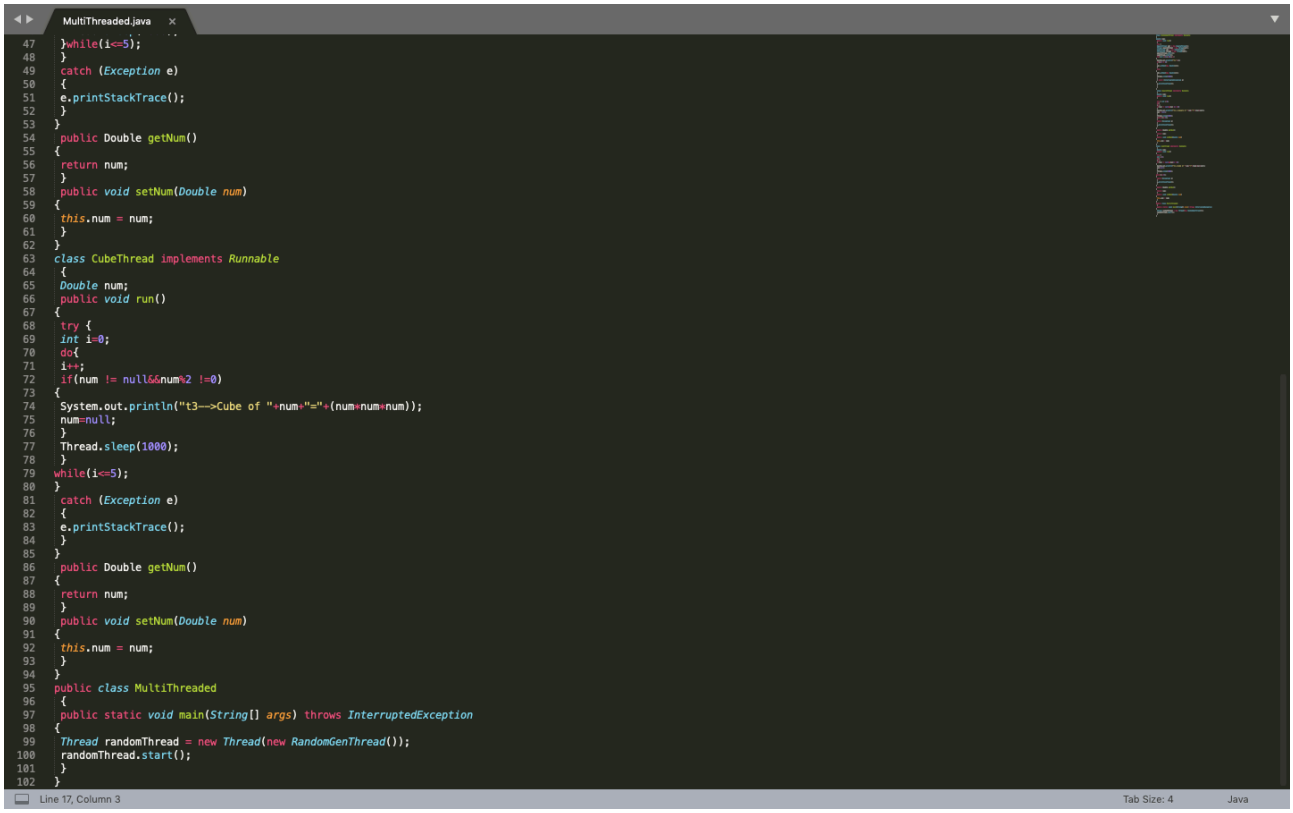
```
 e.printStackTrace();
 }
}
 public Double getNum()
{
 return num;
 }
 public void setNum(Double num)
{
 this.num = num;
 }
}
class CubeThread implements Runnable
 {
 Double num;
 public void run()
{
 try {
 int i=0;
 do{
 i++;
 if(num != null&&num%2 !=0)
{
 System.out.println("t3-->Cube of "+num+"="+(num*num*num));
 num=null;
 }
 Thread.sleep(1000);
 }
while(i<=5);
}
 catch (Exception e)
 {
 e.printStackTrace();
 }
}
 public Double getNum()
{
 return num;
 }
 public void setNum(Double num)
{
 this.num = num;
 }
}
public class MultiThreaded
 {
 public static void main(String[] args) throws InterruptedException
{
 Thread randomThread = new Thread(new RandomGenThread());
 randomThread.start();
 }
}
```

EXECUTION:



```
MultiThreaded.java
class RandomGenThread implements Runnable
{
double num;
 public void run()
 {
 try {
 SquareThread sqt = new SquareThread();
 Thread squareThread = new Thread(sqt);
 CubeThread cbt = new CubeThread();
 Threadcube Thread = new Thread(cbt);
 squareThread.start();
 cubeThread.start();
 for(int i=0;i<10;i++)
{
 System.out.println("t1-"+i);
 if(i%2 == 0)
 {
 sqt.setNum(new Double(i));
 }
 else
{
 cbt.setNum(new Double(i));
 }
 Thread.sleep(1000);
 }
 } catch (InterruptedException e)
{
 e.printStackTrace();
 }
 }
}
 class SquareThread implements Runnable
 {
 Double num;
 public void run()
 {

 try { int i=0;
 do{
 i++;
 if(num != null&&num %2 ==0)
 {
 System.out.println("t2--->square of "+num+"="+(num*num));
 num = null;
 }
 Thread.sleep(1000);
 }while(i<=5);
 }
 catch (Exception e)
 {
 e.printStackTrace();
 }
}
 public Double getNum()
 {
 return num;
```

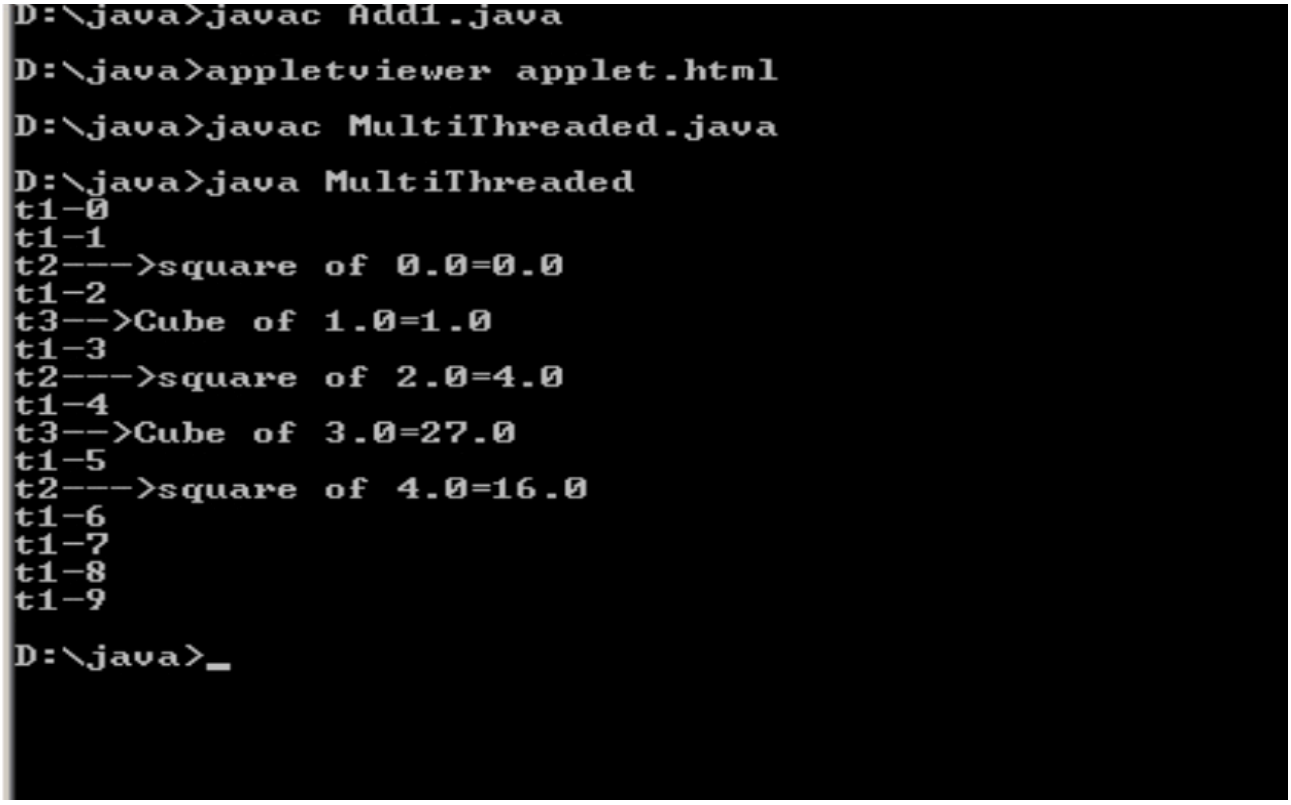
Line 17, Column 3 Tab Size: 4 Java



```
MultiThreaded.java
 }while(i<=5);
 }
 catch (Exception e)
 {
 e.printStackTrace();
 }
}
 public Double getNum()
 {
 return num;
 }
 public void setNum(Double num)
 {
 this.num = num;
 }
}
class CubeThread implements Runnable
 {
 Double num;
 public void run()
 {
 try {
 int i=0;
 do{
 i++;
 if(num != null&&num%2 !=0)
 {
 System.out.println("t3--->Cube of "+num+"="+(num*num*num));
 num=null;
 }
 Thread.sleep(1000);
 }
while(i<=5);
 }
 catch (Exception e)
 {
 e.printStackTrace();
 }
}
 public Double getNum()
 {
 return num;
 }
 public void setNum(Double num)
 {
 this.num = num;
 }
}
public class MultiThreaded
 {
 public static void main(String[] args) throws InterruptedException
 {
 Thread randomThread = new Thread(new RandomGenThread());
 randomThread.start();
 }
 }
```

Line 17, Column 3 Tab Size: 4 Java

RESULT


D:\java>javac Add1.java
D:\java>appletviewer applet.html
D:\java>javac MultiThreaded.java
D:\java>java MultiThreaded
t1-0
t1-1
t2--->square of 0.0=0.0
t1-2
t3-->Cube of 1.0=1.0
t1-3
t2--->square of 2.0=4.0
t1-4
t3-->Cube of 3.0=27.0
t1-5
t2--->square of 4.0=16.0
t1-6
t1-7
t1-8
t1-9
D:\java>

SCENARIO – III

3. Within the package named ‘primespackage’, define a class Primes which includes a method checkForPrime() for checking if the given number is prime or not. Define another class named TwinPrimes outside of this package which will display all the pairs of prime numbers whose difference is 2. (Eg, within the range 1 to 10, all possible twin prime numbers are (3,5), (5,7)). The TwinPrimes class should make use of the checkForPrime() method in the Primes class.

BRIEF ABOUT YOUR APPROACH:

We will create a package primespackage which will contain Primes class which in turn has a checkForPrime() method which checks for prime number. We will create a class TwinPrimes class outside this package in this we will include the primespackage and use the function checkForPrime() which is a part of the condition to check for twin primes. Then the program will give the necessary output according to the condition.

SOURCE CODE:

Prime.java

```
package primespackage;

public class Prime {

    public static boolean checkPrime(int num) {
        int temp;
        boolean isPrime = true;

        for (int i = 2; i <= num / 2; i++) {
            temp = num % i;
            if (temp == 0) {
                isPrime = false;
                break;
            }
        }

        return isPrime;
    }
}
```

TwinsPrime.java

```
package assignment3;

import java.util.Scanner;
import primespackage.Prime;

public class TwinsPrime {
    public static void main(String args[]) {
        int i;

        //take input
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first number: ");
        int n1 = sc.nextInt();
        System.out.print("Enter second number: ");
        int n2 = sc.nextInt();
        sc.close();

        System.out.println("Twin prime numbers are:
");
        //checking twins prime
        for(i = n1; i <= n2; i++) {
            if(Prime.checkPrime(i) &
Prime.checkPrime(i+2) & i > 1)
            {
                System.out.println("(" + i + "," +
(i+2) + ")");
            }
        }
    }
}
```

EXECUTION:

```java
package primespackage;

public class Prime {

    public static boolean checkPrime(int num) {
        int temp;
        boolean isPrime = true;

        for (int i = 2; i <= num / 2; i++) {
            temp = num % i;
            if (temp == 0) {
                isPrime = false;
                break;
            }
        }

        return isPrime;
    }
}
```

```java
package assignment3;

import java.util.Scanner;
import primespackage.Prime;

public class TwinsPrime {
    public static void main(String args[]) {
        int i;

        //take input
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first number: ");
        int n1 = sc.nextInt();
        System.out.print("Enter second number: ");
        int n2 = sc.nextInt();
        sc.close();

        System.out.println("Twin prime numbers are: ");
        //checking twins prime
        for(i = n1; i <= n2; i++) {
            if(Prime.checkPrime(i) & Prime.checkPrime(i+2) & i > 1)
            {
                System.out.println("(" + i + "," + (i+2) + ")");
            }
        }
    }
}
```

RESULT

```
<terminated> TwinPrimes [Java Application] /usr/lib/jvm/java-8-oracle/bin/java (12-Sep-2019 9:12:07 AM)
input limit to check for twin primes till:
10
3, 5
5, 7
```

SCENARIO – IV

4.Create multiple threads to access the contents of a stack. Synchronize thread to prevent simultaneous access to push and pop operations.

BRIEF ABOUT YOUR APPROACH:

We will create a class Stackimpl to access the contents of stack and do the necessary operations on it. It has 2 synchronized methods push() and pop() which are called by different threads created in the main function. The synchronized keyword enables only one thread to perform an operation at a time.

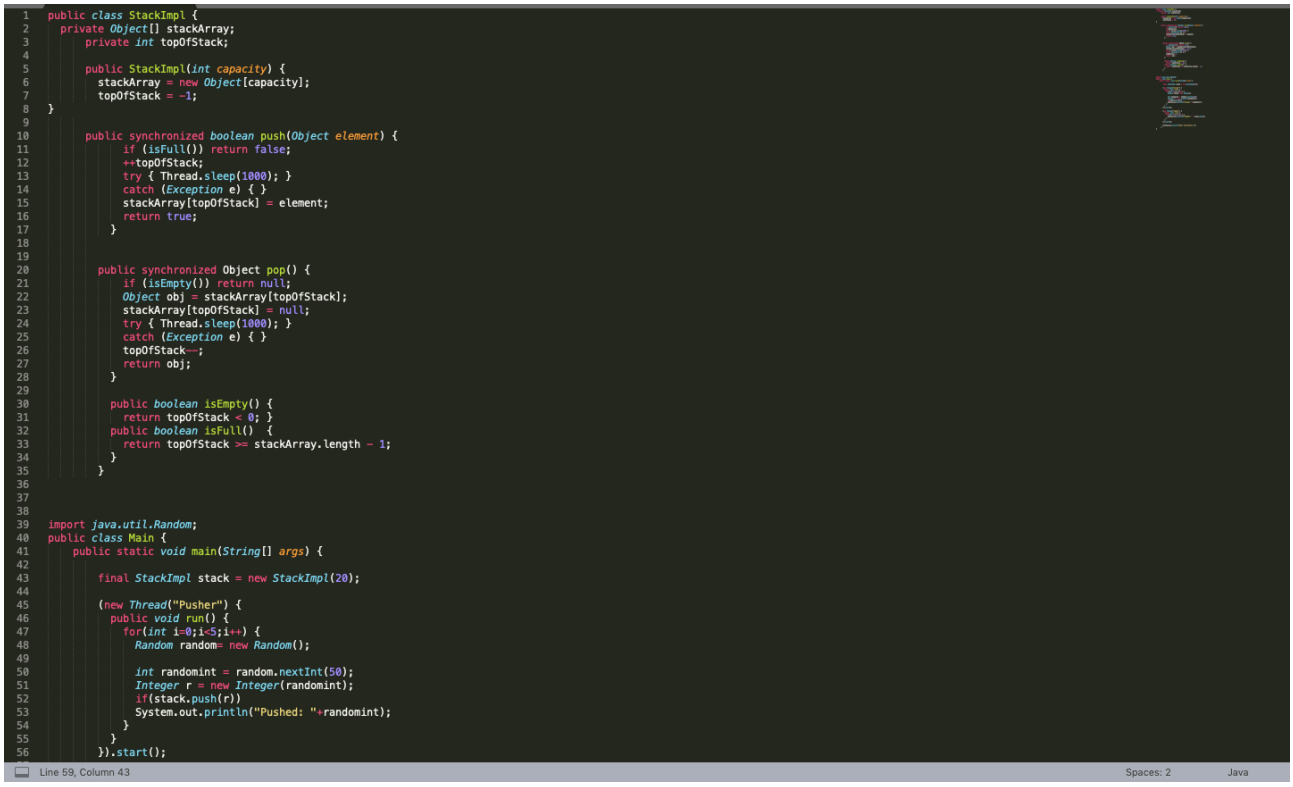
SOURCE CODE:

```
public class StackImpl {
  private Object[] stackArray;
     private int topOfStack;

     public StackImpl(int capacity) {
       stackArray = new Object[capacity];
       topOfStack = -1;
}

     public synchronized boolean push(Object element) {
          if (isFull()) return false;
          ++topOfStack;
          try { Thread.sleep(1000); }
          catch (Exception e) { }
          stackArray[topOfStack] = element;
          return true;
        }


       public synchronized Object pop() {
          if (isEmpty()) return null;
          Object obj = stackArray[topOfStack];
```

```
    stackArray[topOfStack] = null;
    try { Thread.sleep(1000); }
    catch (Exception e) { }
    topOfStack--;
    return obj;
   }

   public boolean isEmpty() {
    return topOfStack < 0; }
   public boolean isFull()  {
    return topOfStack >= stackArray.length - 1;
   }
  }



import java.util.Random;
public class Main {
   public static void main(String[] args) {

     final StackImpl stack = new StackImpl(20);

     (new Thread("Pusher") {
      public void run() {
       for(int i=0;i<5;i++) {
        Random random= new Random();

        int randomint = random.nextInt(50);
        Integer r = new Integer(randomint);
        if(stack.push(r))
        System.out.println("Pushed: "+randomint);
       }
      }
     }).start();

     (new Thread("Popper") {
      public void run() {
       for(int i=0;i<5;i++) {
        System.out.println("Popped: " + stack.pop());
       }
      }
     }).start();

     System.out.println("Exit from main().");
    }
```

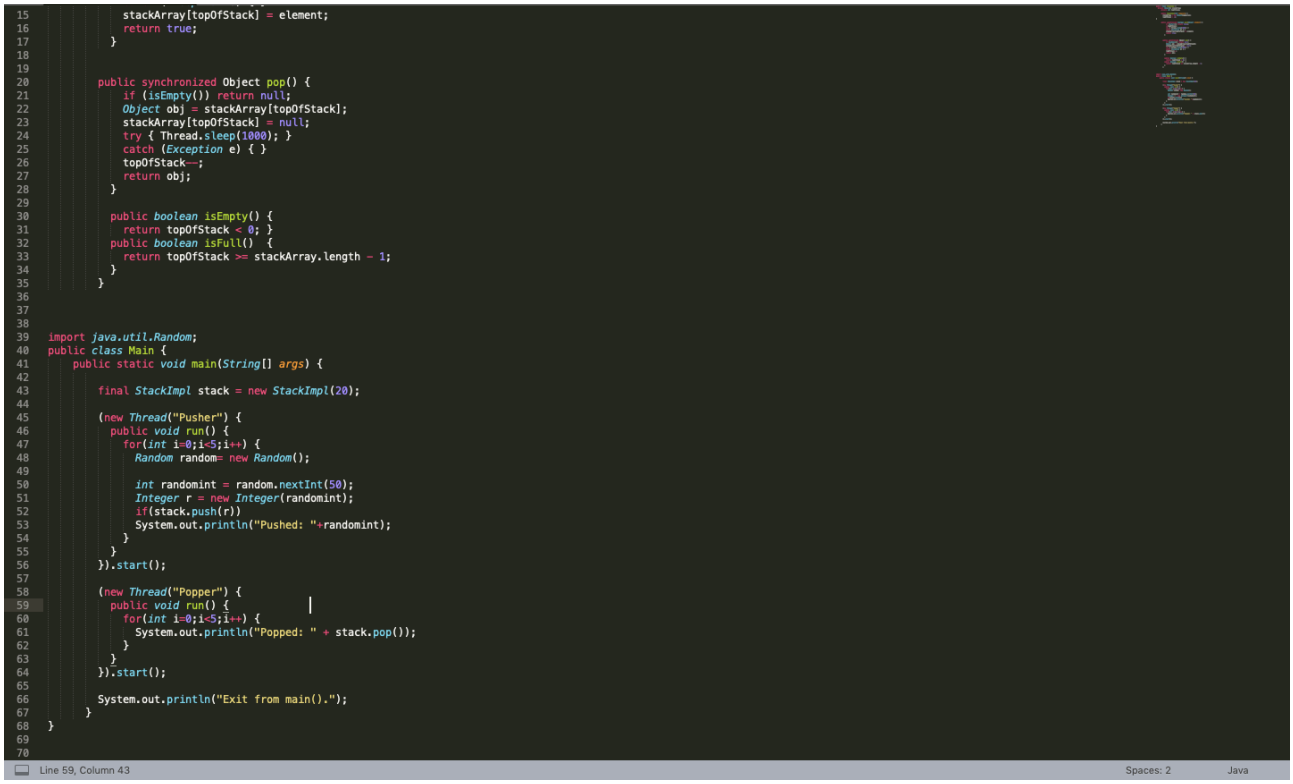
}

EXECUTION:



```
public class StackImpl {
  private Object[] stackArray;
      private int topOfStack;

      public StackImpl(int capacity) {
        stackArray = new Object[capacity];
        topOfStack = -1;
}

      public synchronized boolean push(Object element) {
           if (isFull()) return false;
           ++topOfStack;
           try { Thread.sleep(1000); }
           catch (Exception e) { }
           stackArray[topOfStack] = element;
           return true;
         }


        public synchronized Object pop() {
           if (isEmpty()) return null;
           Object obj = stackArray[topOfStack];
           stackArray[topOfStack] = null;
           try { Thread.sleep(1000); }
           catch (Exception e) { }
           topOfStack--;
           return obj;
         }

          public boolean isEmpty() {
            return topOfStack < 0; }
          public boolean isFull()  {
            return topOfStack >= stackArray.length - 1;
          }
        }


import java.util.Random;
public class Main {
    public static void main(String[] args) {

        final StackImpl stack = new StackImpl(20);

        (new Thread("Pusher") {
          public void run() {
            for(int i=0;i<5;i++) {
              Random random= new Random();

              int randomint = random.nextInt(50);
              Integer r = new Integer(randomint);
              if(stack.push(r))
              System.out.println("Pushed: "+randomint);
            }
          }
        }).start();
```

Line 59, Column 43 Spaces: 2 Java



```
            stackArray[topOfStack] = element;
            return true;
          }


        public synchronized Object pop() {
           if (isEmpty()) return null;
           Object obj = stackArray[topOfStack];
           stackArray[topOfStack] = null;
           try { Thread.sleep(1000); }
           catch (Exception e) { }
           topOfStack--;
           return obj;
         }

          public boolean isEmpty() {
            return topOfStack < 0; }
          public boolean isFull()  {
            return topOfStack >= stackArray.length - 1;
          }
        }


import java.util.Random;
public class Main {
    public static void main(String[] args) {

        final StackImpl stack = new StackImpl(20);

        (new Thread("Pusher") {
          public void run() {
            for(int i=0;i<5;i++) {
              Random random= new Random();

              int randomint = random.nextInt(50);
              Integer r = new Integer(randomint);
              if(stack.push(r))
              System.out.println("Pushed: "+randomint);
            }
          }
        }).start();

        (new Thread("Popper") {
          public void run() {
            for(int i=0;i<5;i++) {
              System.out.println("Popped: " + stack.pop());
            }
          }
        }).start();

        System.out.println("Exit from main().");
    }
}
```

Line 59, Column 43 Spaces: 2 Java

RESULT:

```
<terminated> Main (1) [Java Application] /usr/lib/jvm/java-8-oracle/bin/java (12-Sep-2019 9:27:28 AM)
Exit from main().
Pushed: 2
Popped: 2
Pushed: 18
Popped: 18
Pushed: 7
Popped: 7
Pushed: 29
Popped: 29
Pushed: 42
Popped: 42
```