Extra Programs -

1) WAP to create a thread and find the sum of odd no.s from 1 to 100 in this thread. Find the sum of even numbers for the same range in the main thread.

```
class NewThread implements Runnable {
Thread t;
 NewThread () {
t=new Thread (this, "SUM");
System.out.println ("CT: "+t);
 t.start();
}


public void run () {
 int sum =0;
 try {
  for (int n=1 ; n<=100; n++)
 { if (n %. 2! =0)
 { sum = sum+n; }
   else Thread.sleep (500);
 }
System.out.println ("Sum of all odd numbers between 1-100 \n "+sum);
 }
 catch (InterruptException ie) {
 System.out.println ("Odd Thread Interrupted");
 }
  System.out.println ("Odd Thread quitting");
   }
}
```

```
class Main {
public static void main (String ss[]) {
int sum = 0;
NewThread n1 = new NewThread();
try {
for (int n = 1; n <= 100; n++) {
  if (n % 2 == 0)
  {
   sum = sum + n;
  }
 else
  Thread.sleep (500);
  }
System.out.println ("Sum of all even numbers between 1-100 : In " + sum);
}
catch (InterruptException ie)
{
 System.out.println ("Even Main Thread interrupted");
}
System.out.println ("Even Main Thread quitting");
  }
}
```

2) Develop a multithreaded java program to create 3 threads. First thread generates random integer for every second and if the value is even, second thread computes the square of no. and prints. If the value is odd, the third thread will print the value of cube of number.

```
import java.util.Random;
class Square extends Thread {
int x;
Square (int n) {
x=n; }
```

```
public void run() {
 int sqr = x * x;
 System.out.println("Square of " + x + " = " + sqr);
  }
}

class Cube extends Thread {
 int x;
 Cube (int n) {
 x = n; }
 public void run() {
 int cub = x * x * x;
 System.out.println("Cube of " + x + " = " + cub);
  }
}

class Number extends Thread {
 public void run () {
 Random rand = new Random();
 for (int i = 0; i < 8; i++) {
  int randominteger = rand.nextInt(100);
 System.out.println("Random Integer : " + randominteger);
 if (randominteger % 2 == 0)
  {
```

```
System.out.println("Even value");
Square s = new Square(randominteger);
s.start();
}
else {
System.out.println("Odd value");
Cube c = new Cube(randominteger);
c.start();
}
try {
Thread.sleep(10000);
}
catch (InterruptException ie) {
System.out.println(ie); }
}
}
}

class Main {
public static void main(String args[]) {
Number n = new Number();
n.start();
}
}
```