

Q. 1) Write a Java program that creates three threads: A, B, and C. Thread A should print "Hello," Thread B should print "World," and Thread C should wait for Threads A and B to finish and then print "Done."

```
package Packgetype;

//Q.1) Write a Java program that creates three threads: A, B, and C.
//Thread A should print "Hello," Thread B should print "World," and
//Thread C should wait for Threads A and B to finish and then print "Done."

public class A {    // created a thread A

    public static void main(String[] args) {
        B mt=new B();    //Created object for thread B
        mt.start();    //start method for thread B after printing Thread A
        System.out.print("Hello ");
    }
}

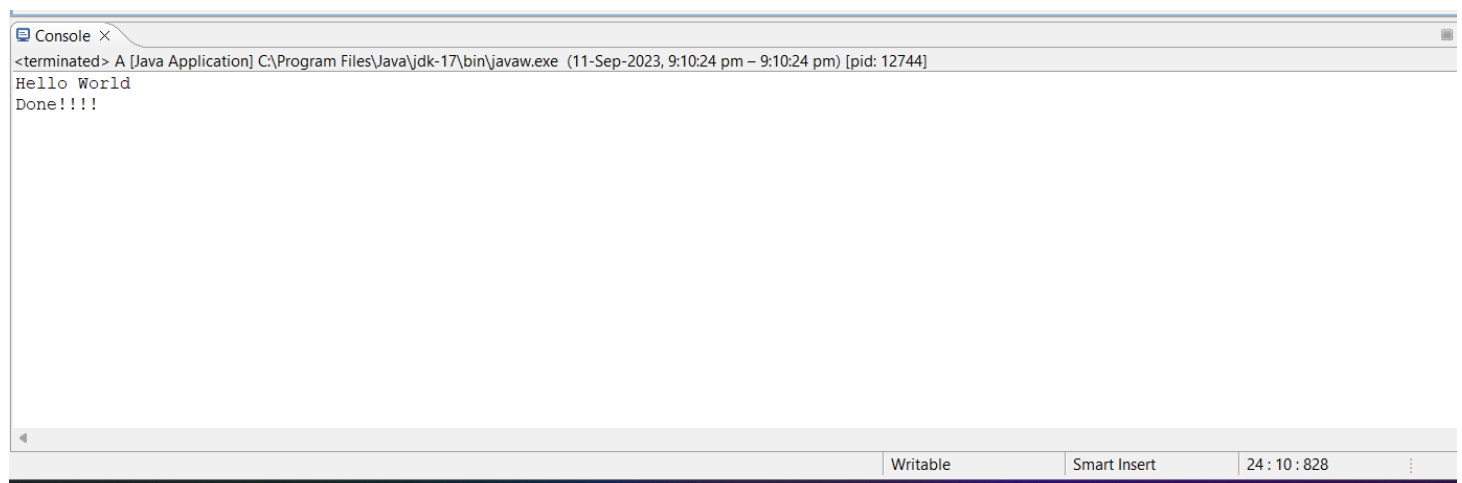
class B extends Thread{    //Created thread B and extends thread class

    public void run(){
        C pr=new C();    //Created object for thread C
        pr.start();    //start method for thread C after completion of Thread A and Thread B
        System.out.println("World");
    }
}

class C extends Thread{    ////Created thread C and extends thread class

    public void run(){
        System.out.println("Done!!!!");
    }
}
```

Output



The screenshot shows a console window titled "Console x" with the following content:

```
<terminated> A [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (11-Sep-2023, 9:10:24 pm - 9:10:24 pm) [pid: 12744]
Hello World
Done!!!!
```

At the bottom of the console window, there is a status bar with the following information:

Writable	Smart Insert	24 : 10 : 828	⋮
----------	--------------	---------------	---

Q.2) Create three threads for Gmail, Instagram and Facebook. Execute threads in sequence first Gmail, next Facebook and at last Instagram. In output, they should be in this order only.

```

package Packgetype;

public class MyThread {
    //Q.2)Create three threads for Gmail, Instagram and Facebook.
    //Execute threads in sequence first Gmail, next Facebook and
    //at last Instagram. In output, they should be in this order only.

    public static void main(String[]args) {

        Gmail g=new Gmail(); //Created a Object for Gmail thread
        g.start(); //Start() method for execute Gmail Thread

        Facebook fb=new Facebook();//Created a Object for Facebook thread
        fb.start(); //Start() method for execute Facebook Thread

        Instagram in=new Instagram();//Created a Object for Instagram thread
        in.start(); //Start() method for execute Instagram Thread
    }

    class Gmail extends Thread{ //Gmail thread 1

        public void run(){ // run() method for Gmail
            System.out.println("This is Gmail Account..");
        }

    }

    class Instagram extends Thread{ //Instagram thread 2

        public void run(){ // run() method for Instagram
            System.out.println("This is Instagram Account..");
        }

    }

    class Facebook extends Thread{ //Facebook Thread 3

        public void run(){ // run() method for Facebook
            System.out.println("This is Facebook Account..");
        }

    }
}

```

OUTPUT

```

Console X
<terminated> MyThread [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (11-Sep-2023, 9:23:52 pm - 9:23:52 pm) [pid: 8676]
This is Gmail Account..
This is Facebook Account..
This is Instagram Account..

```

Q.3) Write a Java program that creates two threads to find and print even and odd numbers from 1 to 20.

```
package Packgetype;

public class EvenOdd {
    //Q.3) Write a Java program that creates two threads to find and print
    //even and odd numbers from 1 to 20.

    public static void main(String[]args) { //main() thread

        Thread even = new Thread(); //Created a thread for finding and printing Even number
        for (int i = 2; i <=20; i+=2) { //for loop for find even number

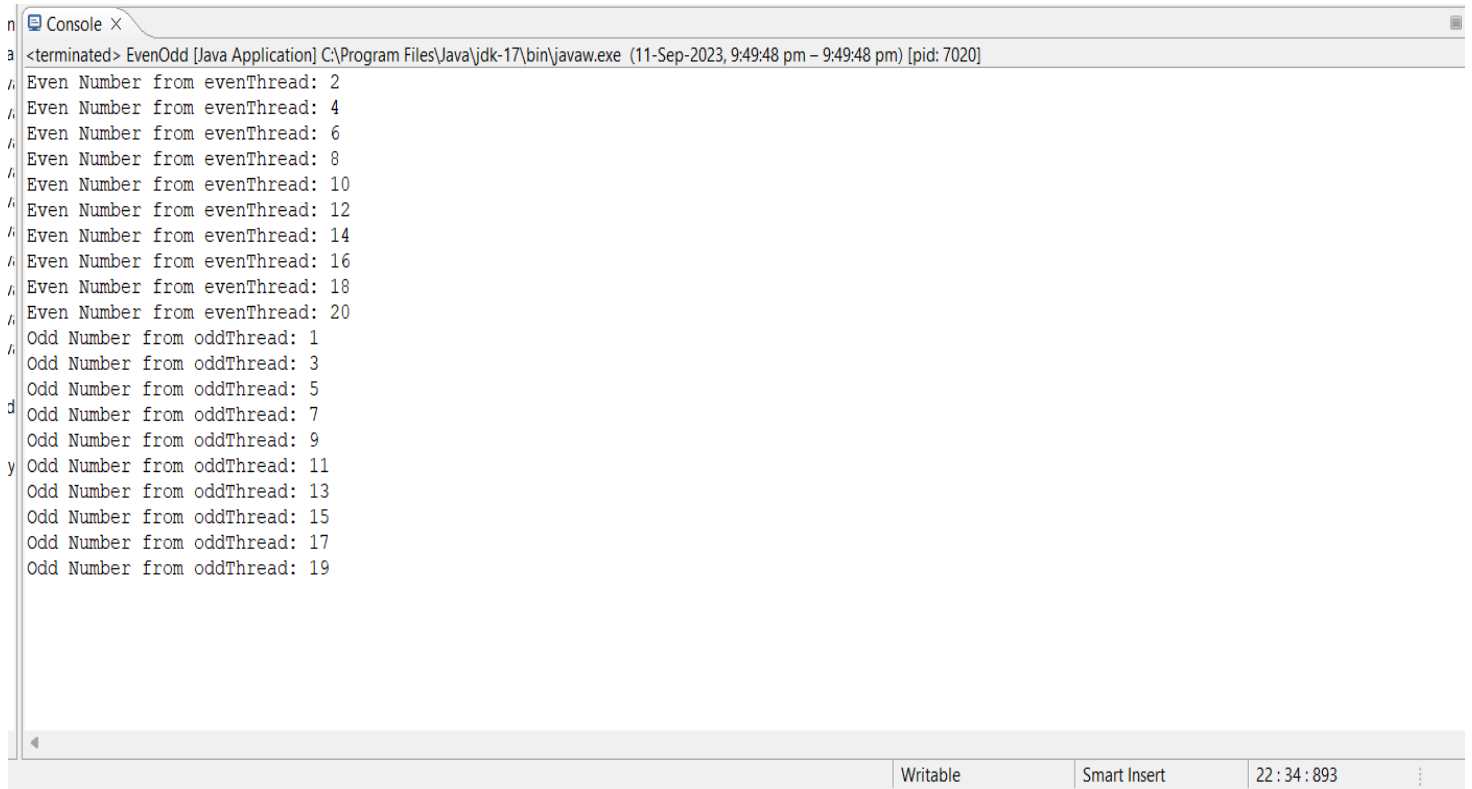
            System.out.println("Even Number from evenThread: " + i); //printing i for
even numbers
        }

        Thread odd = new Thread(); //Created a thread for finding and printing odd number
        for (int i = 1; i <= 20; i+=2) { //for loop for find odd number

            System.out.println("Odd Number from oddThread: " + i); //printing i for odd
numbers
        }

        even.start(); //start() method for even numbers
        odd.start(); //start() method for odd numbers
    }
}
```

Output



```
n Console X
a <terminated> EvenOdd [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (11-Sep-2023, 9:49:48 pm - 9:49:48 pm) [pid: 7020]
// Even Number from evenThread: 2
// Even Number from evenThread: 4
// Even Number from evenThread: 6
// Even Number from evenThread: 8
// Even Number from evenThread: 10
// Even Number from evenThread: 12
// Even Number from evenThread: 14
// Even Number from evenThread: 16
// Even Number from evenThread: 18
// Even Number from evenThread: 20
// Odd Number from oddThread: 1
// Odd Number from oddThread: 3
// Odd Number from oddThread: 5
// Odd Number from oddThread: 7
// Odd Number from oddThread: 9
// Odd Number from oddThread: 11
// Odd Number from oddThread: 13
// Odd Number from oddThread: 15
// Odd Number from oddThread: 17
// Odd Number from oddThread: 19
```

Writable	Smart Insert	22:34:893	⋮
----------	--------------	-----------	---