

## Multithreading:

### Multithreading

```
class bms implements Runnable {  
    Thread t1;  
    bms() {  
        t1 = new Thread(this, "bms");  
    }  
    public void run() {  
        try {  
            for (int i = 0; i < 5; i++) {  
                System.out.println("BMSCS" + i);  
                Thread.sleep(10000);  
            }  
        } catch (InterruptedException ie) {  
            System.out.println("BMS interrupted");  
        }  
        System.out.println("Exiting BMS. " + t1);  
    }  
}
```

```
class cse implements Runnable {  
    Thread t2;  
    cse() {  
        t2 = new Thread(this, "cse");  
    }  
    public void run() {  
        try {  
            for (int i = 0; i < 5; i++) {  
                System.out.println("CSE");  
                Thread.sleep(2000);  
            }  
        } catch (InterruptedException ie) {  
            System.out.println("CSE interrupted");  
        }  
    }  
}
```

} System.out.println("CSE interrupted");  
 }  
 System.out.println("Exiting: " + t2);  
 }  
 }

class Multi9

public static void main (String [] args) {  
 bms thread1 = new bms();  
 cse thread2 = new cse();  
 thread1.t1.start();  
 thread2.t2.start();  
 }  
 }

Output:

BMSCE

CSE

CSE

CSE

CSE

CSE

BMSCE

Exiting cse : Thread [cse, 5, main] ✓

BMSCE

BMSCE

BMSCE

Exiting bms : Thread [bms, 5, main] ✓

~~06/01/2023~~

```
PS C:\Users\bmsce\Desktop\1bm21cs032> java multi
BMSCE
CSE
CSE
CSE
CSE
CSE
BMSCE
Exiting CSE: Thread[cse,5,main]
BMSCE
BMSCE
BMSCE
Exiting BMS: Thread[bms,5,main]
PS C:\Users\bmsce\Desktop\1bm21cs032> 
```