

PRACTICAL 1

AIM: Demonstrate round robin scheduling using threads.

CODE:

```
import java.io.*;
```

```
class job implements Runnable {
```

```
    int process_id, no_of_instr, time_quantum;
```

```
    Thread t;
```

```
    job(int pid, int instr, int tq) {
```

```
        process_id = pid;
```

```
        no_of_instr = instr;
```

```
        time_quantum = tq;
```

```
        t = new Thread(this);
```

```
        t.start();
```

```
    }
```

```
    public void run()
```

```
    {
```

```
        try
```

```
        {
```

```
            for(int i=1;i<=no_of_instr;i++)
```

```
            {
```

```
                System.out.println("executing instr no"+i+"of process"+process_id);
```

```
                Thread.sleep(time_quantum);
```

```
            }
```

```

System.out.println("job"+process_id+"is over");
}
catch(InterruptedException e)
{
System.out.println("the job has been interrupted....");
}
}
}
}

```

```

class os {
    public static void main(String args[]) {
        try {
            int process_id = 100, time_quantum = 100;

            BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));

            System.out.println("Enter a user process starting number:");
            process_id = Integer.parseInt(br.readLine());

            System.out.println("Enter a time quantum(in millis):");
            time_quantum = Integer.parseInt(br.readLine());

            job j1 = new job(++process_id, 10, time_quantum);
            job j2 = new job(++process_id, 6, time_quantum);
            job j3 = new job(++process_id, 8, time_quantum);
        } catch (Exception e) {

            System.out.println("Some process failed to complete...");
            System.out.println("Plz contact system admin...");

        }
    }
}

```

}

OUTPUT:

```
C:\SYCS 05>java os
Enter a user process starting number:
500
Enter a time quantum(in millis):
100
executing instr no1of process501
executing instr no1of process502
executing instr no1of process503
executing instr no2of process502
executing instr no2of process501
executing instr no2of process503
executing instr no3of process501
executing instr no3of process502
executing instr no3of process503
executing instr no4of process501
executing instr no4of process502
executing instr no4of process503
executing instr no5of process502
executing instr no5of process501
executing instr no5of process503
executing instr no6of process502
executing instr no6of process501
executing instr no6of process503
job502is over
executing instr no7of process501
executing instr no7of process503
executing instr no8of process501
executing instr no8of process503
executing instr no9of process501
job503is over
executing instr no10of process501
job501is over
```