

Week 10 ➤

16/08/2024

classmate

Date

Page

// demonstrate inter process communication
and deadlock

```
public class DeadlockExample {
```

```
    public static void main(String[] args) {
```

```
        final SharedResource sharedResource = new  
            SharedResource();
```

```
        Thread process1 = new Thread(() -> {  
            try {
```

```
                sharedResource.method1();
```

```
            } catch (InterruptedException e) {
```

```
                e.printStackTrace();
```

```
            }
```

```
        });
```

```
        Thread process2 = new Thread(() -> {
```

```
            try {
```

```
                sharedResource.method2();
```

```
            } catch (InterruptedException e) {
```

```
                e.printStackTrace();
```

```
            }
```

```
        });
```

```
        process1.start();
```

```
        process2.start();
```

```
    }
```

```
}
```

```
class SharedResource {
```

```
    private final Object lock1 = new Object();
```

```
    private final Object lock2 = new Object();
```


public void method() throws InterruptedException
Exception?

synchronized block?

```
System.out.println("method  
acquired lock");  
Thread.sleep(1000);
```

synchronized block?

```
System.out.println("method  
acquired lock");
```

```
}  
}  
}
```

Sample output:

main thread entered A. log
other thread entered B. log
main thread trying to call B. log
inside A. log
Back in main thread
other thread trying to call A. log
inside A. log
Back in other thread

8/2/24