

```
1 -----DeadLockDemo.java-----
2 package com.ameya.test;
3 /*
4 C:\Users\Ameya>jps -l
5 3892 com.ameya.test.DeadLockDemo
6 6456
   org.springframework.ide.vscode.boot.app.BootLanguageServerBootApp
7 2812 Eclipse
8 4508 jdk.jcmd/sun.tools.jps.Jps
9 C:\Users\Ameya>jstack -l 3892 >
   c:\work\sapient\dumps\thread-dump01.txt
10 */
11 public class DeadLockDemo {
12
13     public static void main(String[] args) throws Exception{
14         Object resourceA=new Object();
15         Object resourceB=new Object();
16         Thread threadLockingResourceAFirst=new Thread(new
           RunnableDeadLock(resourceA,resourceB) );
17         Thread threadLockingResourceBFirst=new Thread(new
           RunnableDeadLock(resourceB,resourceA) );
18         threadLockingResourceAFirst.start();
19         Thread.sleep(500);
20         threadLockingResourceBFirst.start();
21     }
22     private static class RunnableDeadLock implements Runnable{
23         final Object firstResource;
24         final Object secondResource;
25         public RunnableDeadLock(Object firstResource,Object
           secondResource) {
26             this.firstResource=firstResource;
27             this.secondResource=secondResource;
28         }
29         public void run() {
30             try {
31                 synchronized (firstResource) {
32                     printLockedResource(firstResource);
33                     Thread.sleep(1000);
34                     synchronized (secondResource) {
35                         printLockedResource(secondResource);
36                     }

```

```
37         }
38     }catch(InterruptedException e) {
39         System.out.println("Exception Occured : "+e);
40     }
41 }
42 private static void printLockedResource(Object resource) {
43     System.out.println(Thread.currentThread().getName()+" :
44     locked resource - > "+resource);
45 }
46 }
47
```