

```
1  -----TestReadWriteLock.java-----
2  package com.ameya.test;
3
4  import java.util.HashMap;
5  import java.util.Map;
6  import java.util.concurrent.locks.ReentrantReadWriteLock;
7
8  import com.ameya.tasks.ReaderThread;
9  import com.ameya.tasks.WriterThread;
10
11 public class TestReadWriteLock {
12
13     private final Map<String, String> numberMap=new HashMap<String,
14     String>();
15     private final ReentrantReadWriteLock rwLock=new
16     ReentrantReadWriteLock();
17
18     public String get(String key) {
19         System.out.println("Waiting to acquire lock in get()....");
20         rwLock.readLock().lock();
21         System.out.println("Acquired read lock in get()....");
22         try {
23             try {
24                 Thread.sleep(500);
25             }catch(InterruptedException e) {
26                 e.printStackTrace();
27             }
28             return numberMap.get(key);
29         }finally {
30             System.out.println("releasing read lock in get().....");
31             rwLock.readLock().unlock();
32         }
33     }
34
35     public String put(String key,String value) {
36         System.out.println("Waiting to acquire lock in put().....");
37         rwLock.writeLock().lock();
38         System.out.println("Acquired write lock in put().....");
39         try {
40             try {
41                 Thread.sleep(1000);
42             }catch(InterruptedException e) {
```

```

40         e.printStackTrace();
41     }
42     return numberMap.put(key, value);
43 }finally {
44     System.out.println("Releasing write lock in put().....");
45     rwLock.writeLock().unlock();
46 }
47
48 }
49 public static void main(String[] args) {
50     TestReadWriteLock obj=new TestReadWriteLock();
51     obj.initialValues();
52     Thread wThread1=new Thread(new WriterThread(obj, "3",
53         "THREE"));
54     Thread rThread1=new Thread(new ReaderThread(obj, "1"));
55     Thread rThread2=new Thread(new ReaderThread(obj, "1"));
56     Thread wThread2=new Thread(new WriterThread(obj, "4",
57         "FOUR"));
58     Thread rThread3=new Thread(new ReaderThread(obj, "2"));
59     wThread1.start();
60     rThread1.start();
61     rThread2.start();
62     rThread3.start();
63     wThread2.start();
64 }
65 private void initialValues() {
66     numberMap.put("1", "ONE");
67     numberMap.put("2", "TWO");
68 }
69 }

```

```

68 -----ReaderThread.java-----

```

```

69 package com.ameya.tasks;
70
71 import com.ameya.test.TestReadWriteLock;
72
73 public class ReaderThread implements Runnable {
74
75     private TestReadWriteLock obj;
76     private String key;
77     private String value;
78     public ReaderThread(TestReadWriteLock obj,String key) {

```

```

79         this.obj=obj;
80         this.key=key;
81     }
82     @Override
83     public void run() {
84         System.out.println("VALUE => "+obj.get(key));
85     }
86 }
87 -----WriterThread.java-----
88 package com.ameya.tasks;
89
90 import com.ameya.test.TestReadWriteLock;
91
92 public class WriterThread implements Runnable {
93
94     private TestReadWriteLock obj;
95     private String key;
96     private String value;
97     public WriterThread(TestReadWriteLock obj,String key,String value) {
98         this.obj=obj;
99         this.key=key;
100        this.value=value;
101    }
102    @Override
103    public void run() {
104        obj.put(key, value);
105    }
106 }
107
108 }
109

```