

```
Reader --[reader++] -----[reader_count--, notifyAll()] ---->
Writer -----[wait() and release lock] ----->
```

```
public class ReadWriterLock {
    int writer_count = 0;
    int reader_count = 0;
    int writer_request = 0;

    public synchronized void lockReader() throws InterruptedException{
        // always wait if some thread wants to write
        // or some thread is writing
        while(writer_count>0 || writer_request>0){
            wait();
        }
        reader_count++;
        notifyAll();
    }

    public synchronized void unlockReader() throws InterruptedException{
        reader_count--;
        notifyAll();
    }

    public synchronized void lockWriter() throws InterruptedException{
        // requesting write access, this will pre-lock it
        writer_request++;
        while(reader_count>0 || writer_count>0){
            wait();
        }
        writer_count++;
        writer_request--;
        notifyAll();
    }

    public synchronized void unlockWriter() throws InterruptedException{
        writer_count--;
        notifyAll();
    }
}
```