

Ćwiczenie „counter” składało się z 3 części:

1. „SimpleCountingFacade – dostęp do zmiennej licznika jako sekcja krytyczna z wykorzystaniem słowa kluczowego „synchronized””
2. CountingFacadeWithLock – dostęp do zmiennej licznika jako sekcja krytyczna z wykorzystaniem interfejsu Lock
3. CountingFacadeWithAtomic – wyeliminowanie potrzeby sekcji krytycznej poprzez zapewnienie nierozdzielności operacji na liczniku

Ad1. Zmodyfikowałem metodę w następujący sposób:

```
public void countAndInvoke() {  
    synchronized (businessService) {  
        invocationCounter++;  
    }  
    businessService.executeAction();  
}
```

Ad2. Zainicjalizowałem pola:

```
private int invocationCounter;  
private Lock lock;
```

Następnie zmodyfikowałem metodę w następujący sposób:

```
try{  
    lock.lock();  
    invocationCounter++;  
}finally{  
    lock.unlock();  
}  
businessService.executeAction();
```

Ad3. Zainicjalizowałem pola:

```
private AtomicInteger invocationCounter;
```

Przypisałem :

```
invocationCounter = new AtomicInteger();
```

Zmodyfikowałem metody :

```
public void countAndInvoke() {  
    invocationCounter.incrementAndGet();  
    businessService.executeAction();  
}  
  
public int getCount() {  
    return invocationCounter.get();  
}
```

