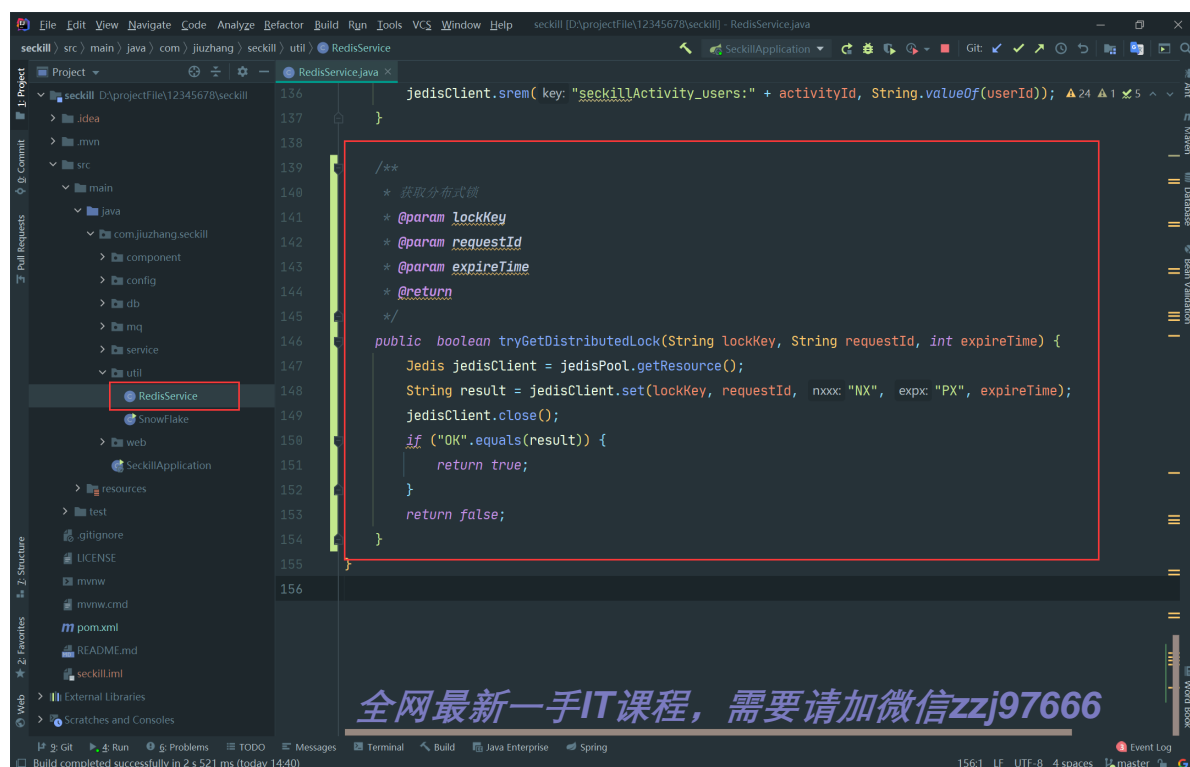


《保姆级教程》第十四集 实现基于 Redis 的分布式锁 & 简单的打包部署

1. 加锁代码



```
/**
 * 获取分布式锁
 * @param lockKey
 * @param requestId
 * @param expireTime
 * @return
 */
public boolean tryGetDistributedLock(String lockKey, String requestId, int expireTime) {
    Jedis jedisClient = jedisPool.getResource();
    String result = jedisClient.set(lockKey, requestId, "NX", "PX", expireTime);
    jedisClient.close();
    if ("OK".equals(result)) {
        return true;
    }
    return false;
}
```

nx 参数有两个值可选：

NX: not exists, 只有key不存在时才把 key value set 到 redis

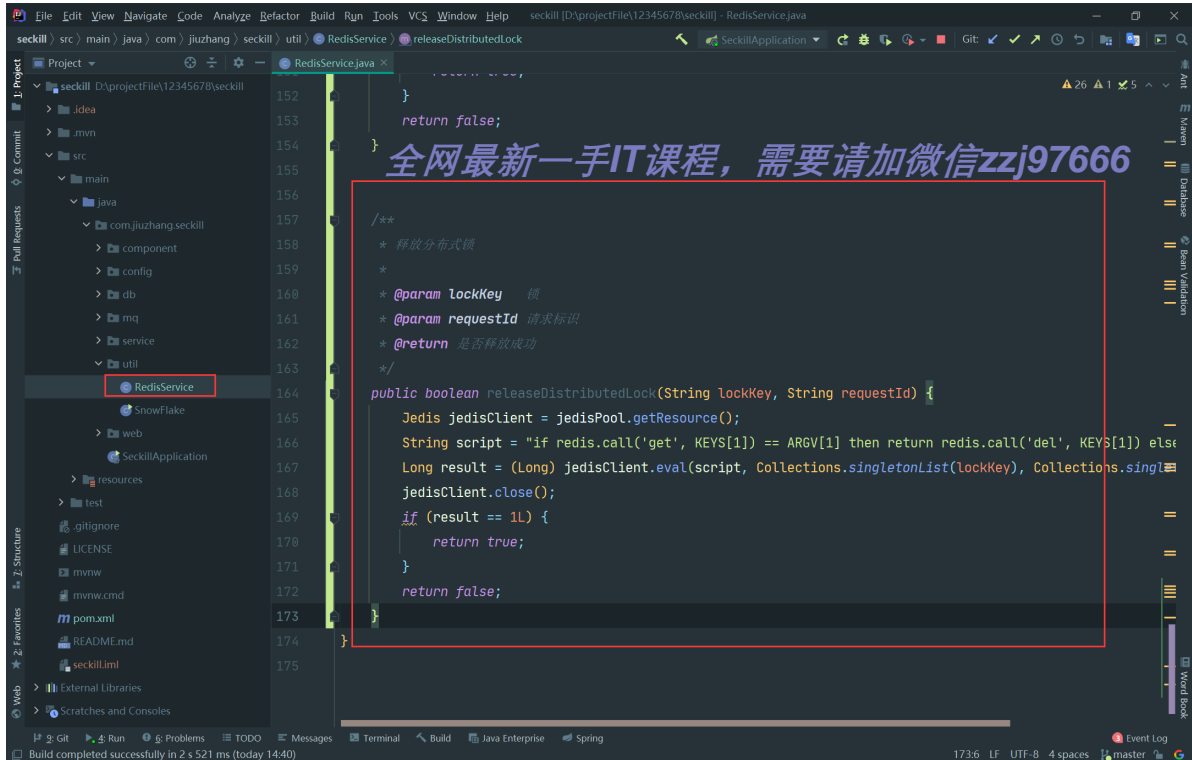
XX: is exists , 只有 key 存在是, 才把 key value set 到 redis

expx 参数有两个值可选 :

EX: seconds 秒

PX: milliseconds 毫秒

2. 解锁代码

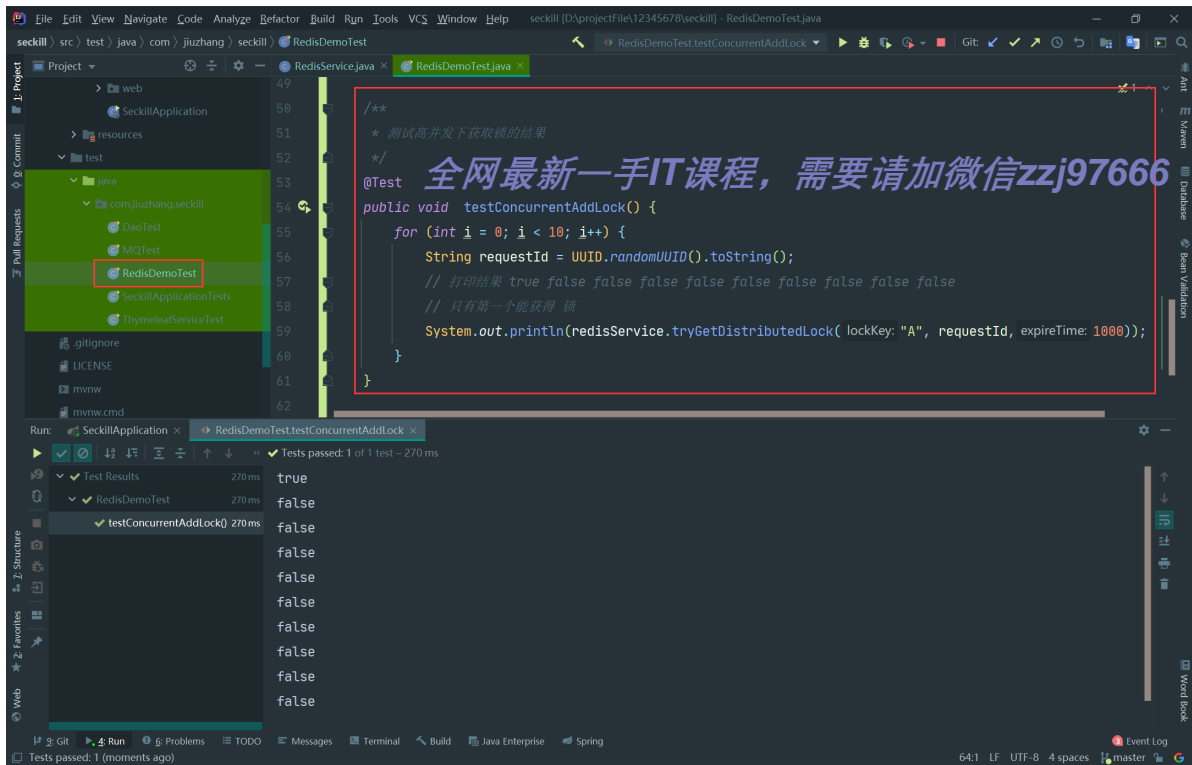


```
/**
 * 释放分布式锁
 *
 * @param lockKey 锁
 * @param requestId 请求标识
 * @return 是否释放成功
 */
public boolean releaseDistributedLock(String lockKey, String requestId) {
    Jedis jedisClient = jedisPool.getResource();
    String script = "if redis.call('get', KEYS[1]) == ARGV[1] then return
redis.call('del', KEYS[1]) else return 0 end";
    Long result = (Long) jedisClient.eval(script,
Collections.singletonList(lockKey), Collections.singletonList(requestId));
    jedisClient.close();
    if (result == 1L) {
        return true;
    }
    return false;
}
```

3. 测试一下这个分布式锁

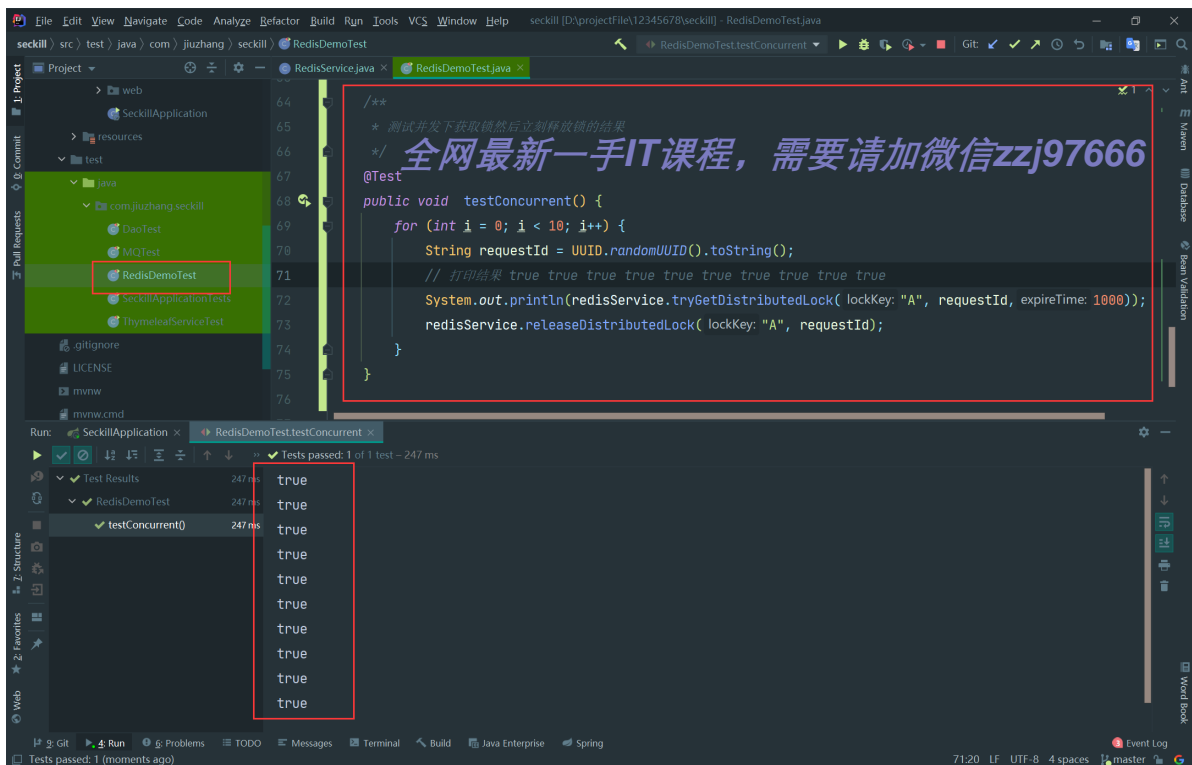
演示的是单机环境下

先演示多个请求一起去获取锁的结果



```
/**
 * 测试高并发下获取锁的结果
 */
@Test
public void testConcurrentAddLock() {
    for (int i = 0; i < 10; i++) {
        String requestId = UUID.randomUUID().toString();
        // 打印结果 true false false false false false false false false
        // 只有第一个能获得 锁
        System.out.println(redisService.tryGetDistributedLock("A",
requestId,1000));
    }
}
```

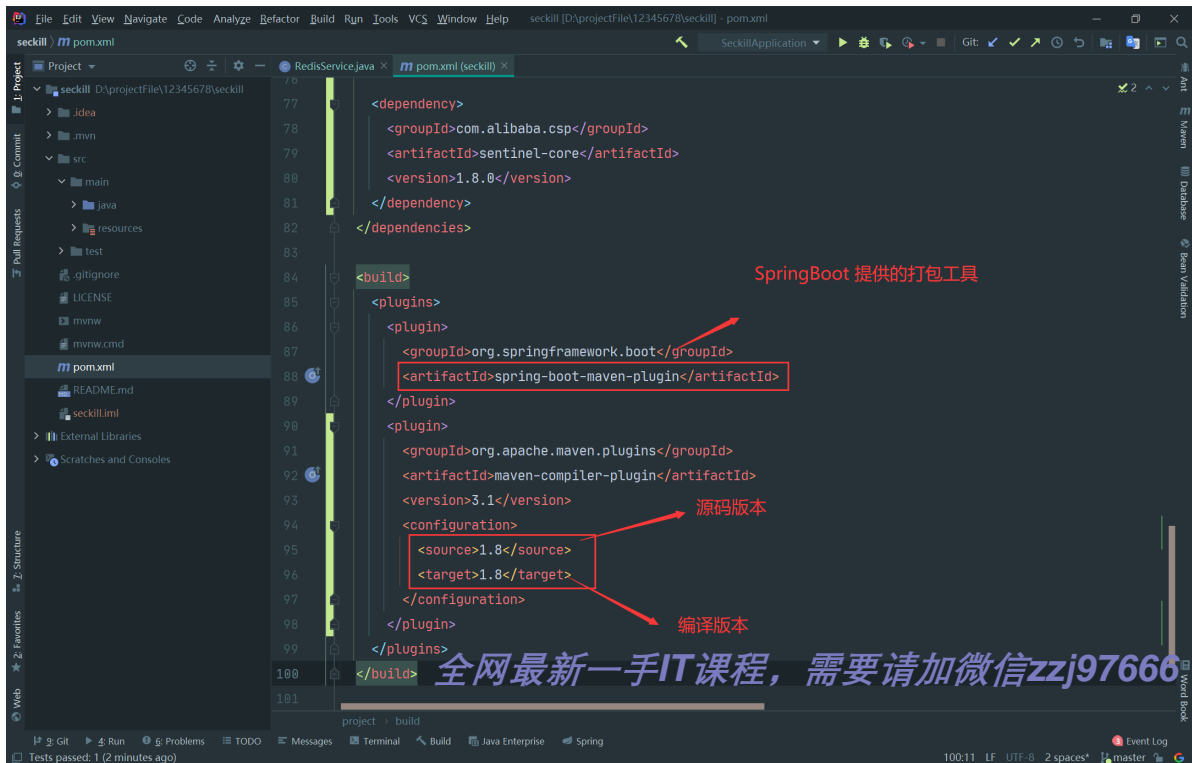
再演示获取锁然后立马再释放锁后的结果



```
/**
 * 测试并发下获取锁然后立刻释放锁的结果
 */
@Test
public void testConcurrent() {
    for (int i = 0; i < 10; i++) {
        String requestId = UUID.randomUUID().toString();
        // 打印结果 true true true true true true true true true true
        System.out.println(redisService.tryGetDistributedLock("A",
requestId,1000));
        redisService.releaseDistributedLock("A", requestId);
    }
}
```

4. 简单打包部署

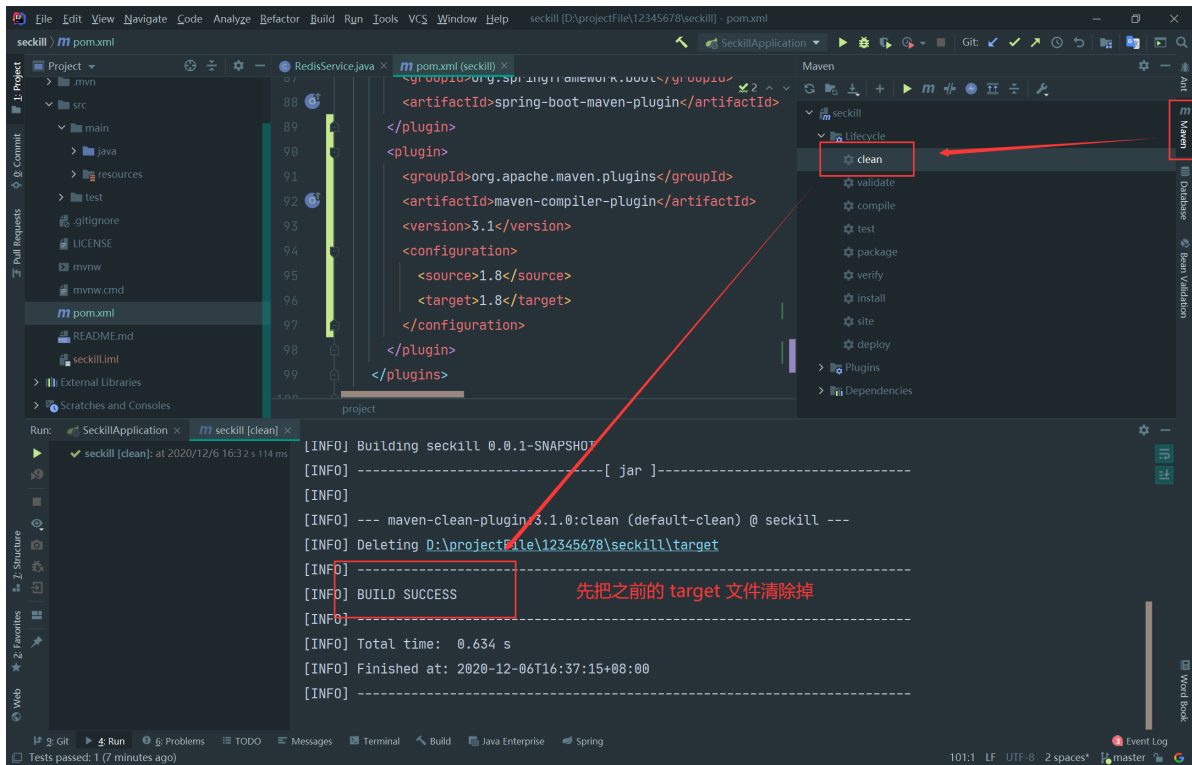
1. pom 依赖



```
<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.1</version>
      <configuration>
        <source>1.8</source>
        <target>1.8</target>
      </configuration>
    </plugin>
  </plugins>
</build>
```

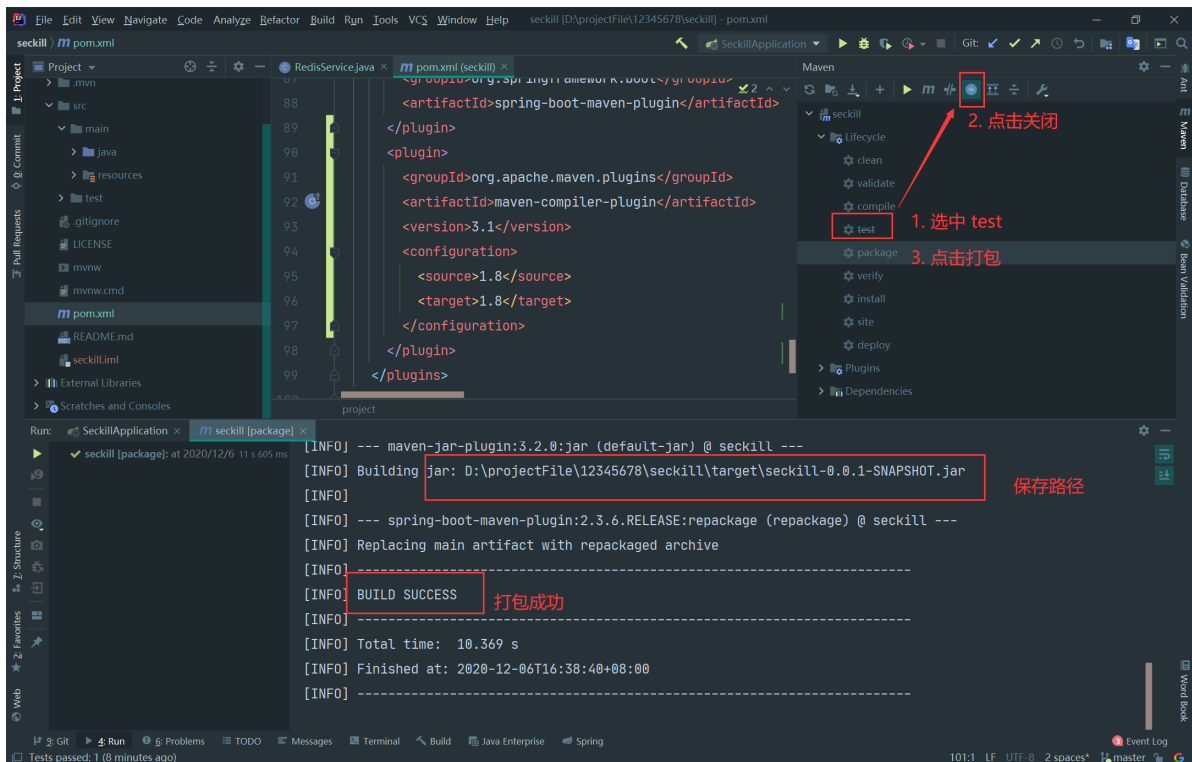
全网最新一手IT课程，需要请加微信zzj97666

2. clear

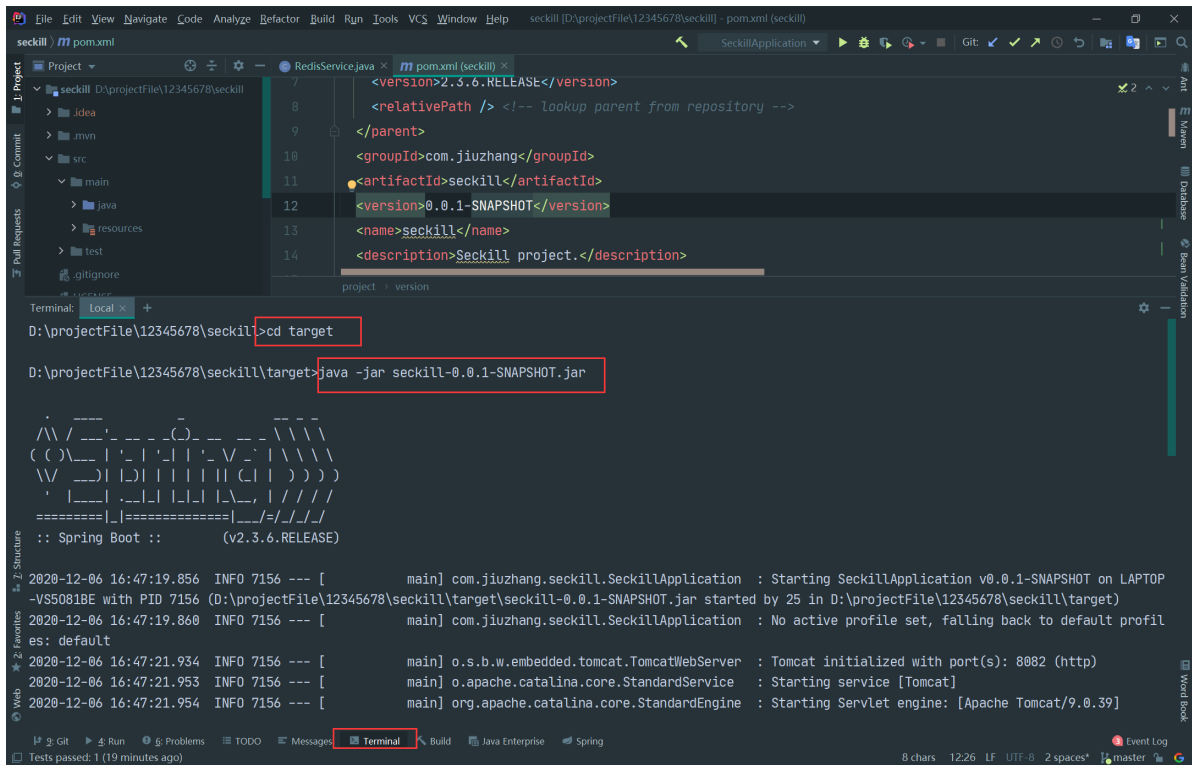


全网最新一手IT课程，需要请加微信zzj97666

3. 项目打包



4. 启动项目



```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help seckill [D:\projectFile\12345678\seckill] - pom.xml (seckill)

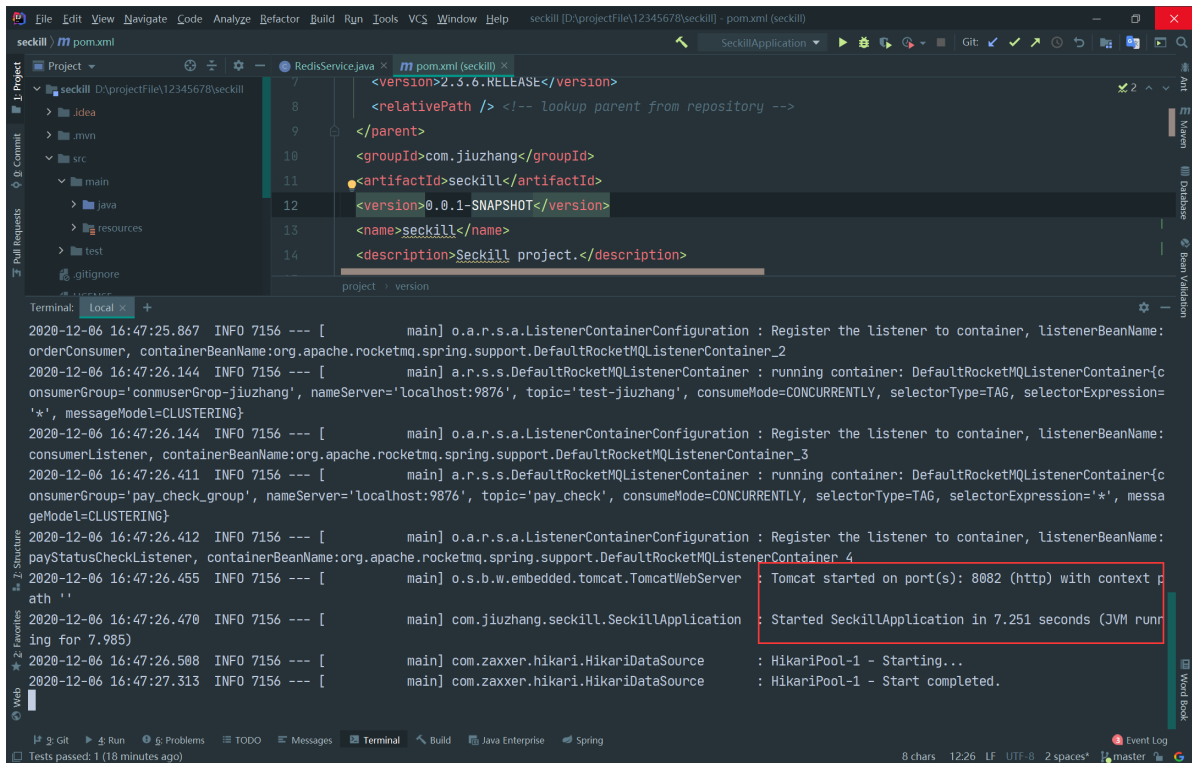
seckill pom.xml
7 <version>2.3.6.RELEASE</version>
8 <relativePath /> <!-- lookup parent from repository -->
9 </parent>
10 <groupId>com.jiuzhang</groupId>
11 <artifactId>seckill</artifactId>
12 <version>0.0.1-SNAPSHOT</version>
13 <name>seckill</name>
14 <description>Seckill project.</description>

Terminal: Local x +
D:\projectFile\12345678\seckill>cd target
D:\projectFile\12345678\seckill\target>java -jar seckill-0.0.1-SNAPSHOT.jar

  ____  _
 / ___|| | | |
| |___| |_| |
 \___ \|  _/ |
      |_| |_|

:: Spring Boot ::      (v2.3.6.RELEASE)

2020-12-06 16:47:19.856 INFO 7156 --- [main] com.jiuzhang.seckill.SeckillApplication : Starting SeckillApplication v0.0.1-SNAPSHOT on LAPTOP
-VS081BE with PID 7156 (D:\projectFile\12345678\seckill\target\seckill-0.0.1-SNAPSHOT.jar started by 25 in D:\projectFile\12345678\seckill\target)
2020-12-06 16:47:19.860 INFO 7156 --- [main] com.jiuzhang.seckill.SeckillApplication : No active profile set, falling back to default profil
es: default
2020-12-06 16:47:21.934 INFO 7156 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8082 (http)
2020-12-06 16:47:21.953 INFO 7156 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2020-12-06 16:47:21.954 INFO 7156 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.39]
```



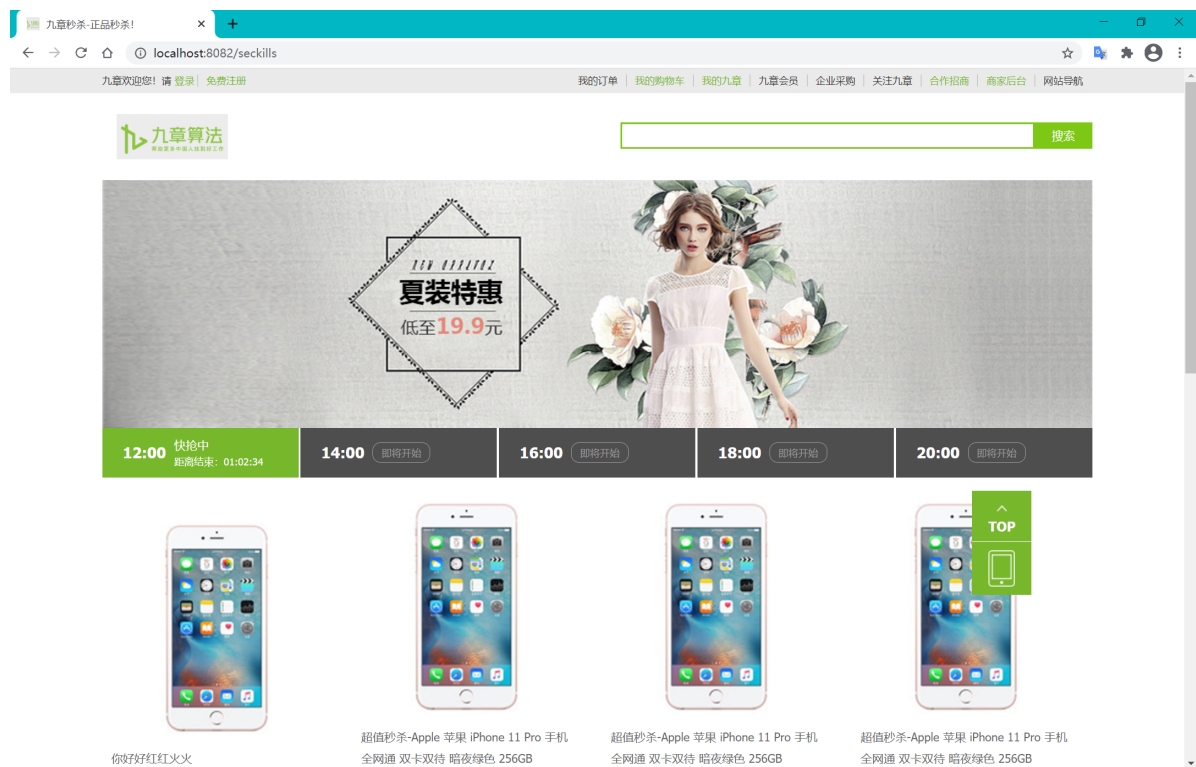
```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help seckill [D:\projectFile\12345678\seckill] - pom.xml (seckill)

seckill pom.xml
7 <version>2.3.6.RELEASE</version>
8 <relativePath /> <!-- lookup parent from repository -->
9 </parent>
10 <groupId>com.jiuzhang</groupId>
11 <artifactId>seckill</artifactId>
12 <version>0.0.1-SNAPSHOT</version>
13 <name>seckill</name>
14 <description>Seckill project.</description>

Terminal: Local x +
2020-12-06 16:47:25.867 INFO 7156 --- [main] o.a.r.s.a.ListenerContainerConfiguration : Register the listener to container, listenerBeanName:
orderConsumer, containerBeanName:org.apache.rocketmq.spring.support.DefaultRocketMQListenerContainer_2
2020-12-06 16:47:26.144 INFO 7156 --- [main] a.r.s.s.DefaultRocketMQListenerContainer : running container: DefaultRocketMQListenerContainer{c
onsumerGroup='consumerGroup-jjuzhang', nameServer='localhost:9876', topic='test-jjuzhang', consumeMode=CONCURRENTLY, selectorType=TAG, selectorExpression=
'*', messageModel=CLUSTERING}
2020-12-06 16:47:26.144 INFO 7156 --- [main] o.a.r.s.a.ListenerContainerConfiguration : Register the listener to container, listenerBeanName:
consumerListener, containerBeanName:org.apache.rocketmq.spring.support.DefaultRocketMQListenerContainer_3
2020-12-06 16:47:26.411 INFO 7156 --- [main] a.r.s.s.DefaultRocketMQListenerContainer : running container: DefaultRocketMQListenerContainer{c
onsumerGroup='pay_check_group', nameServer='localhost:9876', topic='pay_check', consumeMode=CONCURRENTLY, selectorType=TAG, selectorExpression='*', messa
geModel=CLUSTERING}
2020-12-06 16:47:26.412 INFO 7156 --- [main] o.a.r.s.a.ListenerContainerConfiguration : Register the listener to container, listenerBeanName:
payStatusCheckListener, containerBeanName:org.apache.rocketmq.spring.support.DefaultRocketMQListenerContainer_4
2020-12-06 16:47:26.455 INFO 7156 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8082 (http) with context p
ath ''
2020-12-06 16:47:26.470 INFO 7156 --- [main] com.jiuzhang.seckill.SeckillApplication : Started SeckillApplication in 7.251 seconds (JVM runn
ing for 7.985)
2020-12-06 16:47:26.508 INFO 7156 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2020-12-06 16:47:27.313 INFO 7156 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
```

5. 访问页面

全网最新一手IT课程，需要请加微信zzj97666



到这里就算成功了。

打包出来的 jar 包，可以发给别人，别人只要 配置好环境 就可以直接启动项目了。

大家启动项目也可以在 CMD 中启动项目，跟那个是一样的。