部署项目

1.打包

```
# server
server.port=8080

# datasource
spring.datasource.driverClassName=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://127.0.0.1:3306/seckill?characterEncoding=utf-8&useSSL=false&se
spring.datasource.username=root
spring.datasource.password=123456
spring.datasource.type=com.alibaba.druid.pool.DruidDataSource
spring.datasource.druid.initial-size=50
spring.datasource.druid.min-idle=50
spring.datasource.druid.max-active=200

# nowcoder
nowcoder.web.path=http://8.130.163,211

# tomcat
server.tomcat.threads.min-spare=50
server.tomcat.threads.max=300
server.tomcat.accept-count=600
```

```
mvn clean package -Dmaven.test.skip=true
```

2.上传前后台代码,以及脚本,并安装jdk,nginx

```
yum list java*
yum install java-1.8.0-openjdk.x86_64
yum list nginx*
yum install nginx.x86_64
```

3.打开mysql服务

```
systemctl start mysqld.service
```

4.运行bash

```
chmod -R 777 startup.sh
./startup.sh
```

5.将前台代码上传至 /usr/share/nginx/html,并修改common.js

```
tar -zxvf seckill-site.zip
```

6.cd 到 /etc/nginx , 修改配置文件 nginx.conf

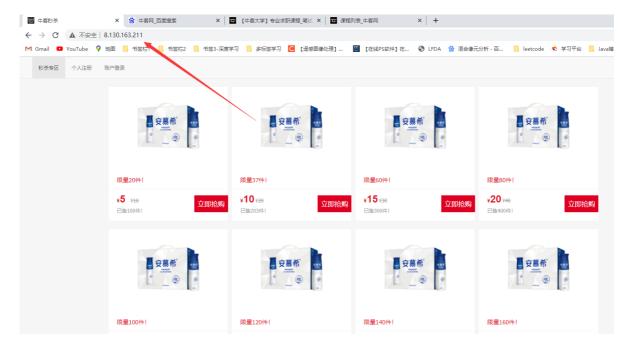
7.运行nginx

```
nginx -c /etc/nginx/nginx.conf
nginx -s reload
```

关闭nginx

```
nginx -s stop
```

8.访问地址



9.压力测试

1)修改配置文件

```
📄 jmeter.properties🖾
      # Maximum redirects to follow in a single sequence (default 20)
       #httpsampler.max_redirects=20
      # Maximum frame/iframe nesting depth (default 5)
      #httpsampler.max_frame_depth=5
      # Revert to BUG 51939 behaviour (no separate container for embedded resources) by setting the following false:
       #httpsampler.separate.container=true
      # If embedded resources download fails due to missing resources or other reasons, if this property is true
       # Parent sample will not be marked as failed
       #httpsampler.ignore_failed_embedded_resources=false
1083
       #keep alive time for the parallel download threads (in seconds)
       #httpsampler.parallel_download_thread_keepalive_inseconds=60
1086
       # Don't keep the embedded resources response data : just keep the size and the MD5
       # default to false
1088
       #httpsampler.embedded_resources_use_md5=false
1089
       # List of extra HTTP methods that should be available in select box
1091
       #httpsampler.user_defined_methods=VERSION-CONTROL, REPORT, CHECKOUT, CHECKIN, UNCHECKOUT, MKWORKSPACE, UPDATE, LABEL, MERGE, BASELINE-CONTI
1092
      # The encoding to be used if none is provided (default ISO-8859-1)
1094
       sampleresult.default.encoding=UTF-8
1095
      # CookieManager behaviour - should obtains with null/empty values be deleted?
# Default is true. Use false to revert to riginal behaviour
1097
1098
       #CookieManager.delete_null_cookies=true
       # CookieManager behaviour - should variable cookies be allowed?
       # Default is true. Use false to revert to original behaviour
       #CookieManager.allow variable cookies=true
1104
       # CookieManager behaviour - should Cookies be stored as variables?
1105
       # Default is false
```

2.查询压力测试

Test Plan 🖒 线程组	线程组
HTTP请求 《 察看结果树 《 聚合报告 计划则试 HTTP请求 《 聚合报告 《 联合报告 》 HTTP Cookie管理器	名称: 线程组
	注稿:
	在取样器描误后要执行的动作
	● 继续 ○ 启动下一进程循环 ○ 停止頻程 ○ 停止頻武 ○ 立即停止朔武
	线程属性
	线程数: 150
	Ramp-Up附间(秒): 10
	循环次数 二 永远 10
	✓ Same user on each iteration
	□ 延迟的键线程直到需要
	□ 调度器
Test Plan	HTTP请求
/ <mark>♦</mark> 线程组 // HTTP请求	名称: HTTP请求
☑ 察看结果树	注释:
■ 聚合银音	±11 ±11
	Web服务器
	协议: http://
	GET ▼ 路径: //item/detail/161 内容编码:
	自的重定 ☑ 既建重定向 ☑ 使用 KeepAlive 〗对POST使用multipart / form-data □ 与则及磁素等的头
	参数 消息体数据 文件上传
	同请求—起发送参教:
	名称: 值 编码? 内容类型 包含等于?

3.秒杀压力测试



