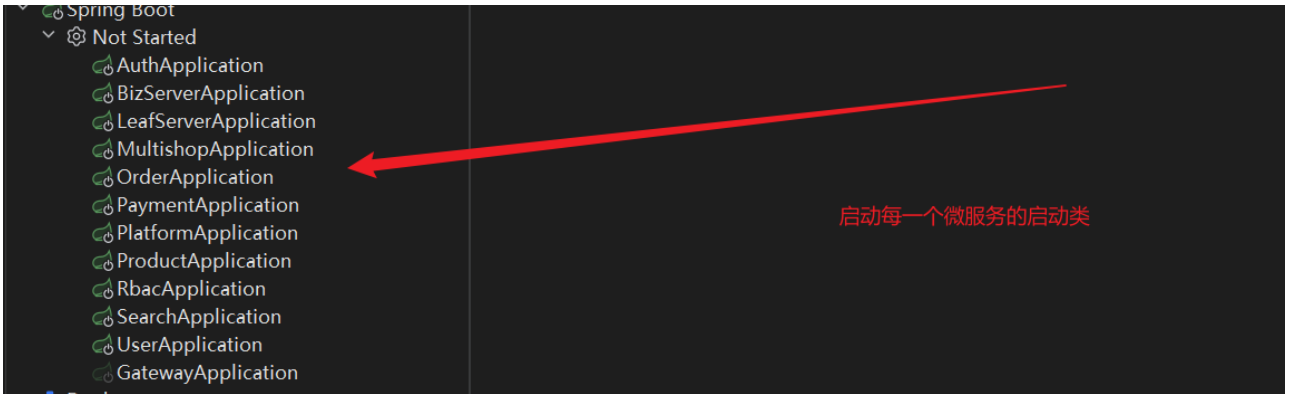


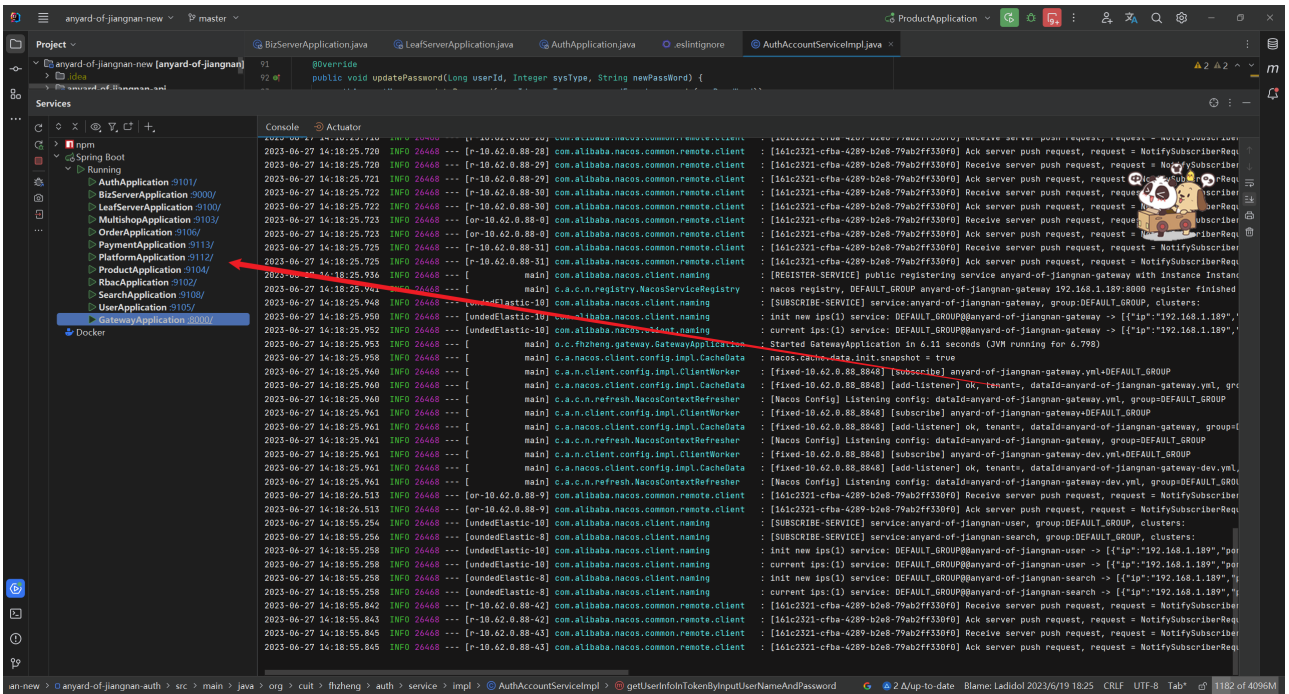
## 一、本地运行

本地运行是在线上后端环境已经部署的情况下进行的。后端环境部署详细可以参考[线上后端环境部署](#)

## 后端微服务项目运行:



### 运行成功:



NACOS

首页 文档 博客 社区 Nacos企业版 En nacos

NACOS 2.2.0

配置管理 服务管理 服务列表 订阅者列表 命名空间 集群管理

服务列表

public | seata

服务名称 请输入服务名称 分组名称 请输入分组名称 隐藏空服务 查询 创建服务

服务名	分组名称	集群数目	实例数	健康实例数	触发保护阈值	操作
anyard-of-jiangnan-payment	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-order	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-platform	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-tbac	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-user	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-multishop	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-auth	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-leaf	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-search	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除
anyard-of-jiangnan-biz	DEFAULT_GROUP	1	1	1	false	详情   示例代码   订阅者   删除

每页显示: 10 < 上一页 1 2 下一页 >

等待nacos上面的全部服务都运行，并注册成功

前端项目运行：

第一次拉项目下来需要 `npm install`

1）运行商家端：

到指定目录下运行 `npm run dev`

```
Terminal Local x Local (2) x Local (3) x + v
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

安装最新的 PowerShell。了解新功能和改进！ https://aka.ms/PSWindows

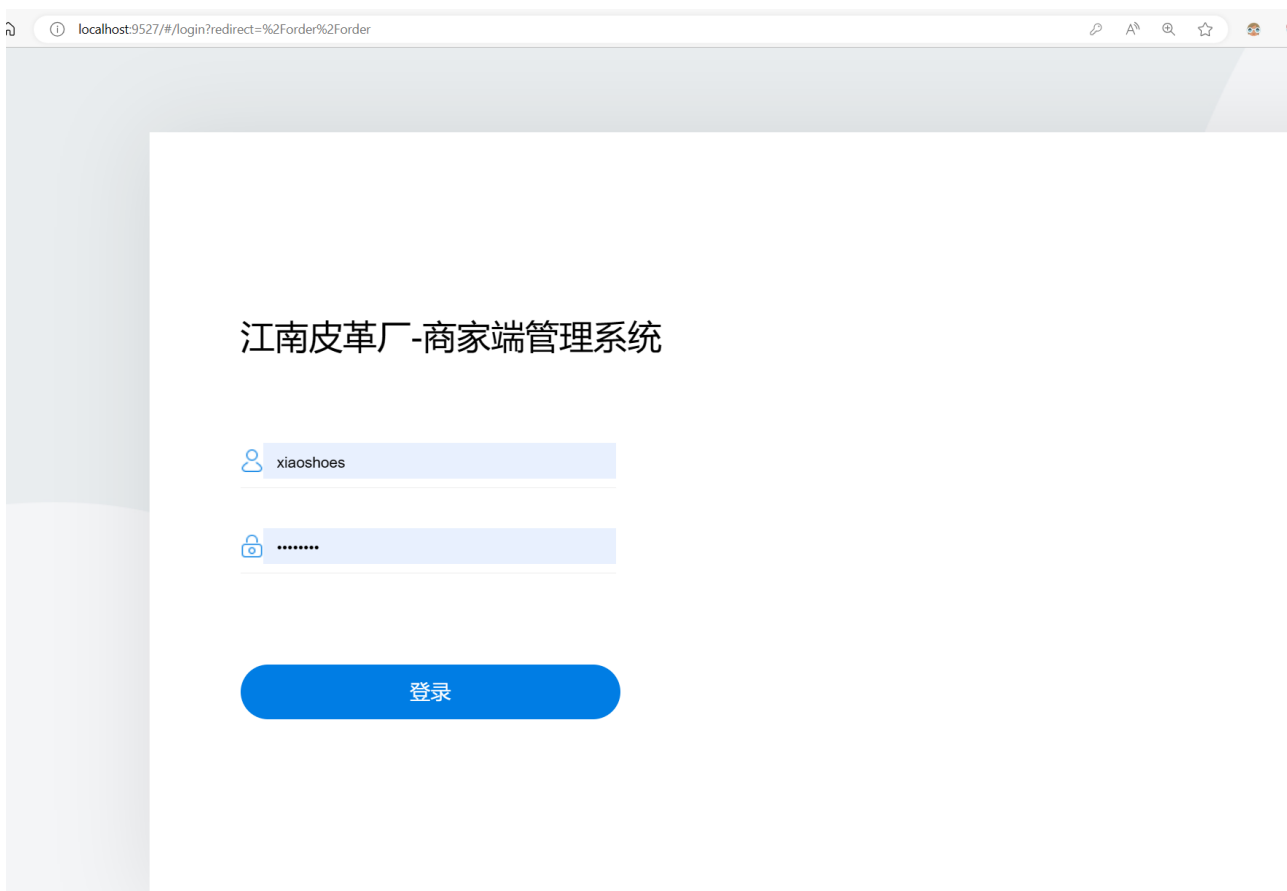
PS D:\study\projects\Java\anyard-of-jiangnan-new> cd .\front-end\
PS D:\study\projects\Java\anyard-of-jiangnan-new\front-end> cd .\anyard-of-jiangnan-multishop-master\
PS D:\study\projects\Java\anyard-of-jiangnan-new\front-end\anyard-of-jiangnan-multishop-master> npm run dev

> vue-admin-template@4.4.0 dev D:\study\projects\Java\anyard-of-jiangnan-new\front-end\anyard-of-jiangnan-multishop-master
> vue-cli-service serve

INFO Starting development server...
48% building 105/118 modules 13 active D:\study\projects\Java\anyard-of-jiangnan-new\front-end\anyard-of-jiangnan-multishop-master\node_modules\core-js\modules\es.number.constructor.js
```

运行成功：

```
Terminal  Local ×  Local (2) ×  Local (3) ×  +  ▾  
  
DONE Compiled successfully in 9828ms  
  
App running at:  
- Local: http://localhost:9527/  
- Network: http://192.168.1.189:9527/  
  
Note that the development build is not optimized.  
To create a production build, run npm run build.  
  
█
```



## 2) 运行平台端：

到指定目录下运行 `npm run dev`

```
安装最新的 PowerShell, 了解新功能和改进! https://aka.ms/PSWindows
Windows PowerShell
PS D:\study\projects\Java\anyard-of-jiangnan-new> cd .\front-end\
PS D:\study\projects\Java\anyard-of-jiangnan-new\front-end> cd .\anyard-of-jiangnan-platform-master\
PS D:\study\projects\Java\anyard-of-jiangnan-new\front-end\anyard-of-jiangnan-platform-master> npm run dev

> vue-admin-template@4.4.0 dev D:\study\projects\Java\anyard-of-jiangnan-new\front-end\anyard-of-jiangnan-platform-master
> vue-cli-service serve

[INFO] Starting development server...
67% building 997/1035 modules 38 active D:\study\projects\Java\anyard-of-jiangnan-new\front-end\anyard-of-jiangnan-platform-master\node_modules
d and will be removed in Dart Sass 2.0.0.

Recommendation: math.div(1, 5) or calc(1 / 5)

More info and automated migrator: https://sass-lang.com/d/slash-div
```

运行成功:

```
Terminal  Local × Local (2) × Local (3) × + ∨

DONE Compiled successfully in 8046ms

App running at:
- Local:  http://localhost:9528/
- Network: http://192.168.1.189:9528/

Note that the development build is not optimized.
To create a production build, run npm run build.
```

localhost:9528/#/login?redirect=%2Fproduct%2Flist

系统登录



admin

.....

登录

### 3) 运行买家端:

到指定目录下运行 `npm run dev:h5`

```
安装最新的 PowerShell，了解新功能和改进！ https://aka.ms/PSWindows
Windows PowerShell
PS D:\study\projects\Java\anyard-of-jiangnan-new> cd .\front-end\
PS D:\study\projects\Java\anyard-of-jiangnan-new\front-end> cd .\anyard-of-jiangnan-uniapp-master\
PS D:\study\projects\Java\anyard-of-jiangnan-new\front-end\anyard-of-jiangnan-uniapp-master> npm run dev:h5

> my-project@0.1.0 dev:h5 D:\study\projects\Java\anyard-of-jiangnan-new\front-end\anyard-of-jiangnan-uniapp-master
> cross-env NODE_ENV=development UNI_PLATFORM=h5 vue-cli-service uni-serve

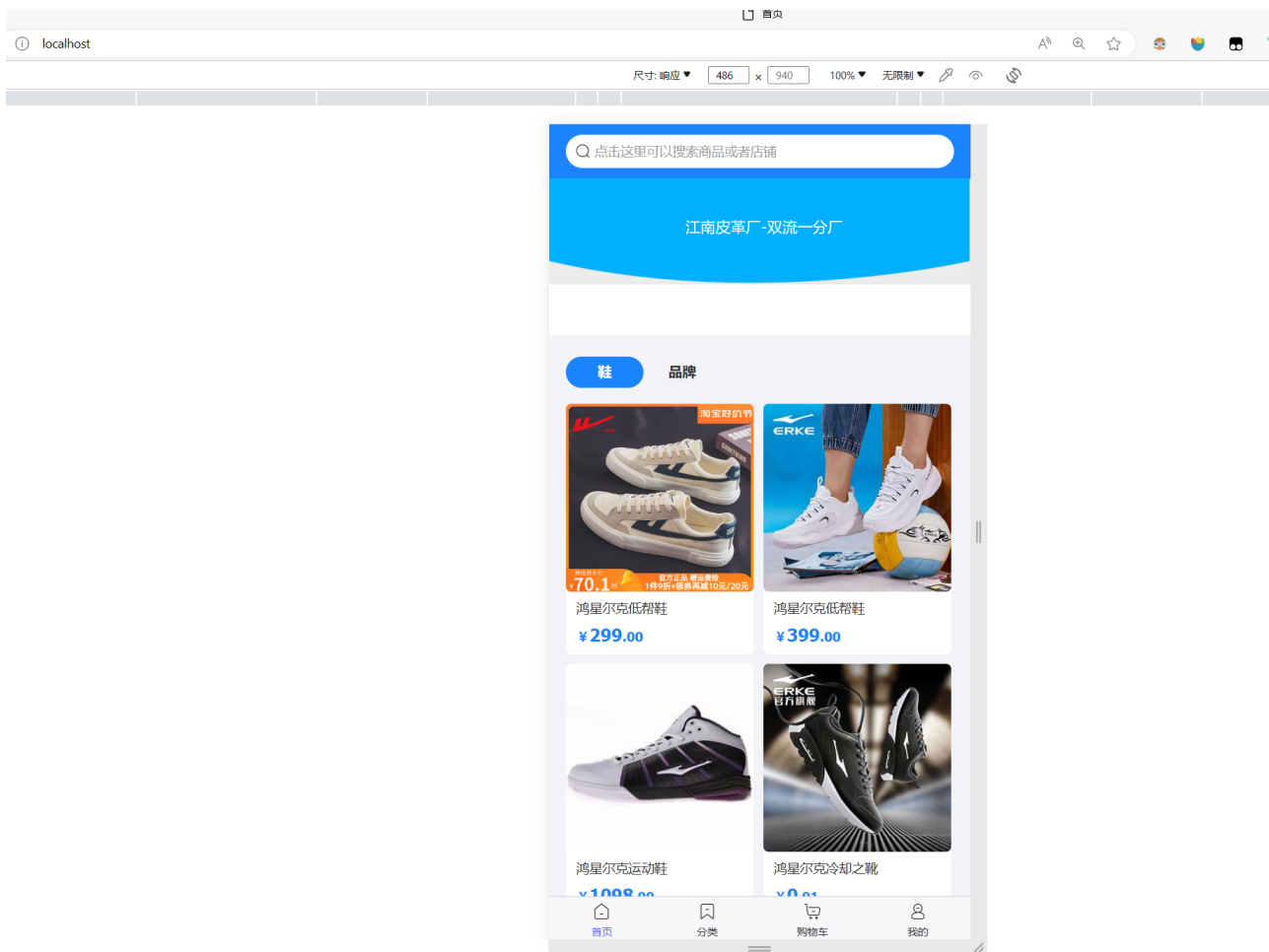
请注意运行模式下，因日志输出、sourcemap以及未压缩源码等原因，性能和包体积，均不及发行模式。
Browserslist: caniuse-lite is outdated. Please run:
  npx browserslist@latest --update-db

Why you should do it regularly:
https://github.com/browserslist/browserslist#browsers-data-updating
[INFO] Starting development server...
```

运行成功:

```
Scratches and Consoles
Terminal  Local x Local (2) x Local (3) x + v
DONE Compiled successfully in 7709ms

App running at:
- Local: http://localhost:80/
- Network: http://192.168.1.189:80/
```



## 二、线上部署并运行

### 1) 后端环境部署

我们采用`docker-compose`一键部署后端需要用的相关中间件

`docker-compose.yml`

```
version: "3.5"

services:
  anyard-of-jiangnan-mysql:
    image: mysql:8.0
    container_name: anyard-of-jiangnan-mysql
    restart: always
    environment:
```

- MYSQL\_ROOT\_PASSWORD=root

ports:

- 3306:3306

volumes:

- ./mysql/data:/var/lib/mysql
- ./mysql/conf.d:/etc/mysql/conf.d
- ./mysql/initdb:/docker-entrypoint-initdb.d

anyard-of-jiangnan-minio:

image: minio/minio:RELEASE.2021-06-17T00-10-46Z

container\_name: anyard-of-jiangnan-minio

restart: always

command: server /data

ports:

- 9000:9000

volumes:

- ./minio/data:/data

environment:

- MINIO\_ROOT\_USER=admin
- MINIO\_ROOT\_PASSWORD=admin123456

anyard-of-jiangnan-redis:

image: redis:6.2

container\_name: anyard-of-jiangnan-redis

restart: always

ports:

- 6379:6379

anyard-of-jiangnan-nacos:

image: nacos/nacos-server:v2.2.0-slim

container\_name: anyard-of-jiangnan-nacos

restart: always

depends\_on:

- anyard-of-jiangnan-mysql

ports:

- 8848:8848
- 9848:9848
- 9849:9849

environment:

- JVM\_XMS=256m
- JVM\_XMX=256m
- MODE=standalone

- PREFER\_HOST\_MODE=hostname
- SPRING\_DATASOURCE\_PLATFORM=mysql
- MYSQL\_SERVICE\_HOST=10.62.0.88
- MYSQL\_SERVICE\_DB\_NAME=anyard-of-jiangnan-nacos
- MYSQL\_SERVICE\_USER=root
- MYSQL\_SERVICE\_PASSWORD=root

volumes:

- ./nacos/logs:/home/nacos/logs
- ./nacos/conf:/home/nacos/conf

anyard-of-jiangnan-seata:

image: seataio/seata-server:1.6.1

container\_name: anyard-of-jiangnan-seata

restart: always

depends\_on:

- anyard-of-jiangnan-mysql
- anyard-of-jiangnan-nacos

ports:

- 8091:8091
- 7091:7091

environment:

- SEATA\_IP=10.62.0.88

volumes:

- ./seata/application.yml:/seata-server/resources/application.yml

anyard-of-jiangnan-elasticsearch:

image: elasticsearch:7.17.5

container\_name: anyard-of-jiangnan-elasticsearch

restart: always

ports:

- 9200:9200
- 9300:9300

environment:

- discovery.type=single-node
- ES\_JAVA\_OPTS=-Xms512m -Xmx512m

volumes:

- 
- ./elasticsearch/config/elasticsearch.yml:/usr/share/elasticsearch/config/elasticsearch.yml
- ./elasticsearch/data:/usr/share/elasticsearch/data
- ./elasticsearch/plugins:/usr/share/elasticsearch/plugins



anyard-of-jiangnan-kibana:

image: kibana:7.17.5

container\_name: anyard-of-jiangnan-kibana

restart: always

ports:

- 5601:5601

links:

- anyard-of-jiangnan-elasticsearch:elasticsearch

depends\_on:

- anyard-of-jiangnan-elasticsearch

anyard-of-jiangnan-canal:

image: canal/canal-server:v1.1.6

container\_name: anyard-of-jiangnan-canal

restart: always

ports:

- 11111:11111

volumes:

- ./canal/conf/example:/home/admin/canal-server/conf/example

- ./canal/conf/canal.properties:/home/admin/canal-server/conf/canal.properties

- ./canal/logs:/home/admin/canal-server/logs

anyard-of-jiangnan-rocketmq-namesrv:

image: apache/rocketmq:4.9.4

container\_name: anyard-of-jiangnan-rocketmq-namesrv

restart: always

ports:

- 9876:9876

volumes:

- ./rocketmq/namesrv/logs:/home/rocketmq/logs

- ./rocketmq/namesrv/store:/home/rocketmq/store

environment:

JAVA\_OPT\_EXT: "-Duser.home=/home/rocketmq -Xms512M -Xmx512M -Xmn128M"

command: ["sh", "mqnamesrv"]

networks:

rocketmq:

aliases:

- anyard-of-jiangnan-rocketmq-namesrv

```

anyard-of-jiangnan-rocketmq-broker:
  image: apache/rocketmq:4.9.4
  container_name: anyard-of-jiangnan-rocketmq-broker
  restart: always
  ports:
    - 10909:10909
    - 10911:10911
  volumes:
    - ./rocketmq/broker/logs:/home/rocketmq/logs
    - ./rocketmq/broker/store:/home/rocketmq/store
    -
  ./rocketmq/broker/conf/broker.conf:/etc/rocketmq/broker.conf
  environment:
    JAVA_OPT_EXT: "-Duser.home=/home/rocketmq -Xms512M -Xmx512M -Xmn128M -XX:-AssumeMP"
    command: ["sh", "mqbroker", "-c", "/etc/rocketmq/broker.conf", "-n", "anyard-of-jiangnan-rocketmq-namesrv:9876", "autoCreateTopicEnable=true"]
  depends_on:
    - anyard-of-jiangnan-rocketmq-namesrv
  networks:
    rocketmq:
      aliases:
        - anyard-of-jiangnan-rocketmq-broker

anyard-of-jiangnan-rocketmq-dashboard:
  image: apacherocketmq/rocketmq-dashboard:1.0.0
  container_name: anyard-of-jiangnan-rocketmq-dashboard
  restart: always
  ports:
    - 8180:8080
  environment:
    JAVA_OPTS: "-Drocketmq.namesrv.addr=anyard-of-jiangnan-rocketmq-namesrv:9876 -Dcom.rocketmq.sendMessageWithVIPChannel=false"
  depends_on:
    - anyard-of-jiangnan-rocketmq-namesrv
  networks:
    rocketmq:
      aliases:
        - anyard-of-jiangnan-rocketmq-console

```

```
networks:
  rocketmq:
    name: rocketmq
    driver: bridge
```

现在服务器上安装docker和docker-compose，然后运行上面文件就行了。详细教程请谷歌。

部署成功如下：

Containers

Container list [查看docker, 全部容器都启动成功](#)

Containers

Q Search...

Start Stop Kill Restart Pause Resume Remove Add container

Name	State	Quick Actions	Stack	Image	Created	IP Address	GPUs	Published Ports	Ownership
anyard-of-jiangnan-canal	running		mall4cloud	canal/canal-server:v1.1.6	2023-06-11 15:51:14	172.20.0.6	none	8089:11110 11111:11111	administrators
anyard-of-jiangnan-elasticsearch	running		mall4cloud	elasticsearch:7.17.5	2023-06-08 21:56:15	172.20.0.9	none	9200:9200 9300:9300	administrators
anyard-of-jiangnan-kibana	running		mall4cloud	kibana:7.17.5	2023-06-10 13:39:45	172.20.0.8	none	5601:5601	administrators
anyard-of-jiangnan-minio	running		mall4cloud	minio/minio:RELEASE.2021-06-17T00-10-46Z	2023-06-08 21:56:15	172.20.0.3	none	9000:9000	administrators
anyard-of-jiangnan-mysql	running		mall4cloud	mysql:8.0	2023-06-08 21:56:15	172.20.0.5	none	3306:3306	administrators
anyard-of-jiangnan-nacos	running		mall4cloud	nacos/nacos-server:v2.2.0-slim	2023-06-09 22:51:47	172.20.0.7	none	8848:8848 9848:9848 9849:9849	administrators
anyard-of-jiangnan-redis	running		mall4cloud	redis:6.2	2023-06-08 21:56:15	172.20.0.2	none	6379:6379	administrators
anyard-of-jiangnan-rocketmq-b...	running		mall4cloud	apache/rocketmq:4.9.4	2023-06-08 21:56:20	172.21.0.3	none	10909:10909 10911:10911	administrators
anyard-of-jiangnan-rocketmq-d...	running		mall4cloud	apacherocketmq/rocketmq-dashboard:1.0.0	2023-06-08 21:56:20	172.21.0.4	none	8180:8080	administrators
anyard-of-jiangnan-rocketmq-n...	running		mall4cloud	apache/rocketmq:4.9.4	2023-06-08 21:56:15	172.21.0.2	none	9876:9876	administrators
anyard-of-jiangnan-seata	running		mall4cloud	seata/seata-server:1.6.1	2023-06-09 22:51:48	172.20.0.4	none	7091:7091 8091:8091	administrators
portainer	running		-	portainer/portainer	2023-06-09 19:24:31	172.17.0.2	none	8000:8000 9999:9000	administrators

Items per page 25

## 2) 前端环境部署

先用部署nginx容器，并挂载目录/usr/local/vue

```
docker pull nginx #下载nginx镜像
```

```
docker run --name nginx --restart=always -p 80:80 -p 443:443 -d -v
/usr/local/nginx/nginx.conf:/etc/nginx/nginx.conf -v
/usr/local/vue:/usr/local/vue -v
/usr/local/upload:/usr/local/upload nginx
#启动nginx，映射本地配置文件，
```

nginx部署成功

<div> <div>▶ Start</div> <div>■ Stop</div> <div>🔪 Kill</div> <div>🔄 Restart</div> <div>⏸ Pause</div> <div>▶ Resume</div> <div>🗑 Remove</div> <div>+ Add container</div> </div>							
Q Search...							
<input type="checkbox"/> Name	State	Quick actions	Stack	Image	Created	Published Ports	Ownership
<input type="checkbox"/> anyard-of-jiangnan-new	starting		-	anyard-of-jiangnan-new:0.0.1-jar-jar-jar	2023-06-27 16:48	80:80	administrat
<input type="checkbox"/> anyard-of-jiangnan-api	building		-	anyard-of-jiangnan-api	2023-06-27 16:48	80:80	administrat
<input type="checkbox"/> anyard-of-jiangnan-auth	running		-	anyard-of-jiangnan-auth	2023-06-27 16:48	80:80	administrat
<input type="checkbox"/> nginx	running		-	nginx	2022-06-17 01:52:26	80:80	administrat
<input type="checkbox"/> redis	running		-	redis	2022-06-16 01:00:15	6379:6379	administrat
<input type="checkbox"/> anyard-of-jiangnan-gateway	running		-	anyard-of-jiangnan-gateway	2022-06-12 16:54:49	80:80	administrat
<input type="checkbox"/> anyard-of-jiangnan-payment	stopped		-	anyard-of-jiangnan-payment	2023-06-22 15:06:06	-	administrat
<input type="checkbox"/> anyard-of-jiangnan-platform	stopped		-	anyard-of-jiangnan-platform	2023-06-03 20:56:57	-	administrat
<input type="checkbox"/> anyard-of-jiangnan-product	stopped		-	anyard-of-jiangnan-product	2023-06-03 20:23:32	-	administrat
<input type="checkbox"/> anyard-of-jiangnan-order	stopped		-	anyard-of-jiangnan-order	2023-03-04 21:38:15	-	administrat
						Items per page 10	1 2

### 3) 后端项目部署

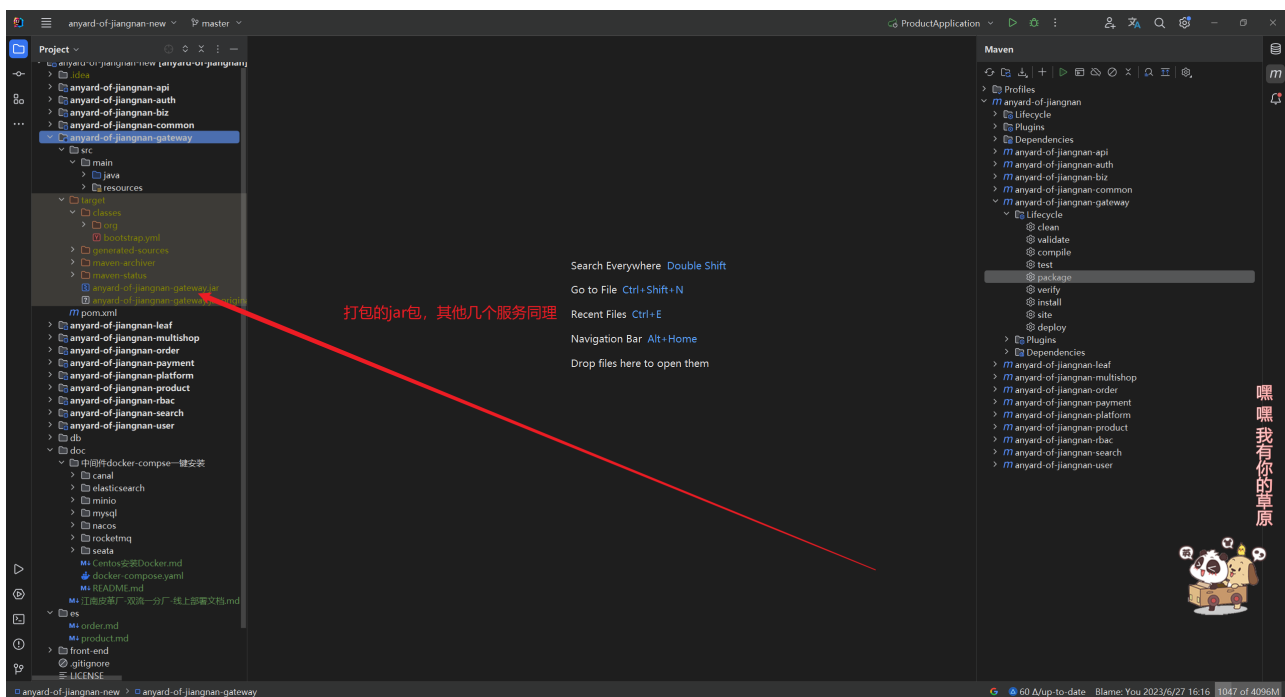
对每一个微服务进行maven打包

The screenshot shows the IntelliJ IDEA IDE with the 'anyard-of-jiangnan-gateway' module selected in the Project view. The Maven tool window on the right shows the 'package' goal being executed. The Run window at the bottom displays the build output, which includes the following information:

```

[INFO] --- maven-compiler-plugin:3.8.1:testCompile (default-testCompile) @ anyard-of-jiangnan-gateway ---
[INFO] No sources to compile
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ anyard-of-jiangnan-gateway ---
[INFO] No tests to run.
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ anyard-of-jiangnan-gateway ---
[INFO] Building jar: D:\study\projects\Java\anyard-of-jiangnan-new\anyard-of-jiangnan-gateway\target\anyard-of-jiangnan-gateway.jar
[INFO] --- spring-boot-maven-plugin:2.7.7:repackage (default) @ anyard-of-jiangnan-gateway ---
[INFO] Replicating main artifact with repackaged archive
[INFO] BUILD SUCCESS
[INFO] Total time: 2.582 s
[INFO] Finished at: 2023-06-27T16:48:49+08:00
[INFO]
Process finished with exit code 0

```



## 写dockerfile

```
FROM java:8
ADD anyard-of-jiangnan-gateway.jar anyard-of-jiangnan-gateway.jar

ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/anyard-of-jiangnan-gateway.jar"]
```

然后构建镜像 `docker build -t anyard-of-jiangnan-gateway .`

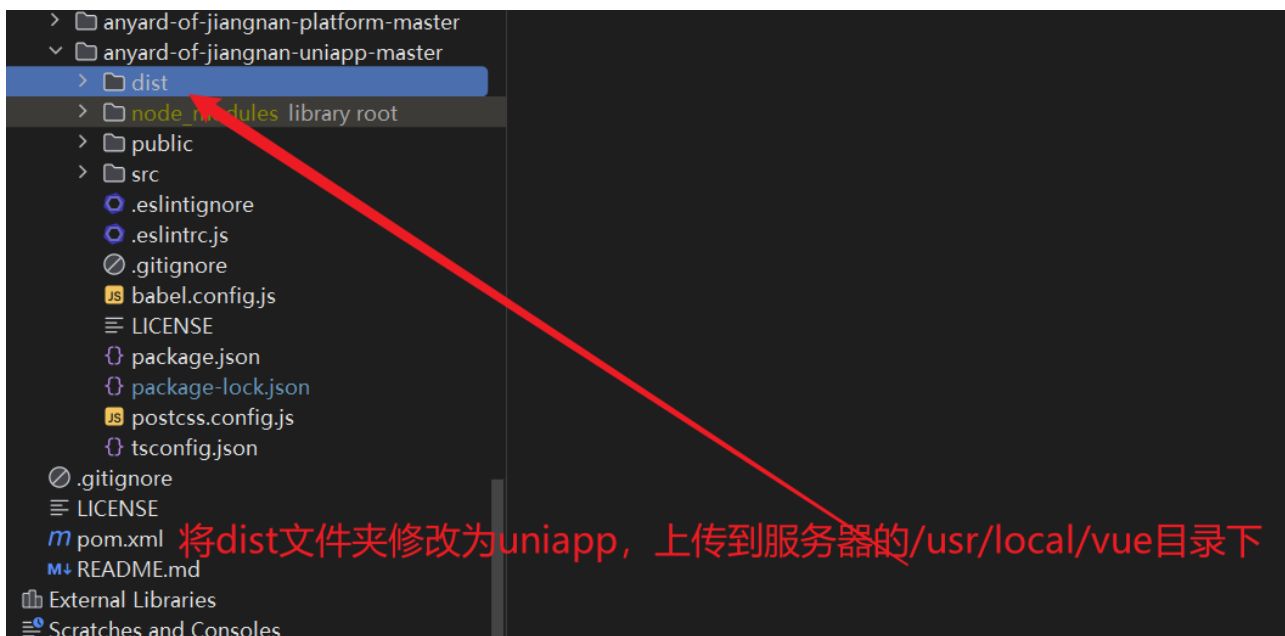
再运行就行了 `docker run --restart=on-failure --name anyard-of-jiangnan-gateway -d -p 8000:8000 anyard-of-jiangnan-gateway`

其他服务同理；

## 4) 前端项目部署

本地打包前端项目 `npm run build`，生成dist文件夹

例如uniapp端：



在nginx的配置文件中增加一个服务

```
server {  
    listen      80;  
  
    location / {  
        root    /usr/local/vue/uniapp;  
        index   index.html index.htm;  
        try_files $uri $uri/ /index.html;  
    }  
  
    location ^~ /api/ {  
        proxy_pass http://后端服务器ip:8000/;  
        proxy_set_header    Host            $host;  
        proxy_set_header    X-Real-IP       $remote_addr;  
        proxy_set_header    X-Forwarded-For  
$proxy_add_x_forwarded_for;  
    }  
}
```

其他模块同理;