# 云原生 第二次作业

## 211250175 王艺羲

## 一.两个微服务启动时都注册到 Eureka 服务

1. 先配置数据库，更改`mysql-pv.yaml`中的hostPath，然后运行客户端
   2.使用 `mysql-pv.yaml` 文件中定义的配置来创建一个 Kubernetes 持久卷

部署 MySQL,然后运行客户端

```
kubectl apply -f mysql-pv.yaml
kubectl apply -f mysql-deployment.yaml
kubectl run -it --rm --image=mysql:8.0.33 --restart=Never mysql-client -- mysql -
h mysql -pdangerous
```

```
Last login: Sat Jul  8 23:37:10 on ttys003
ethylene@wangyixideMacBook-Pro 单机 MySQL % kubectl apply -f mysql-pv.yaml

persistentvolume/mysql-pv-volume created
persistentvolumeclaim/mysql-pv-claim created
ethylene@wangyixideMacBook-Pro 单机 MySQL % kubectl apply -f mysql-deployment.ya
ml

service/mysql created
deployment.apps/mysql created
```

初步创建数据库，验证其功能

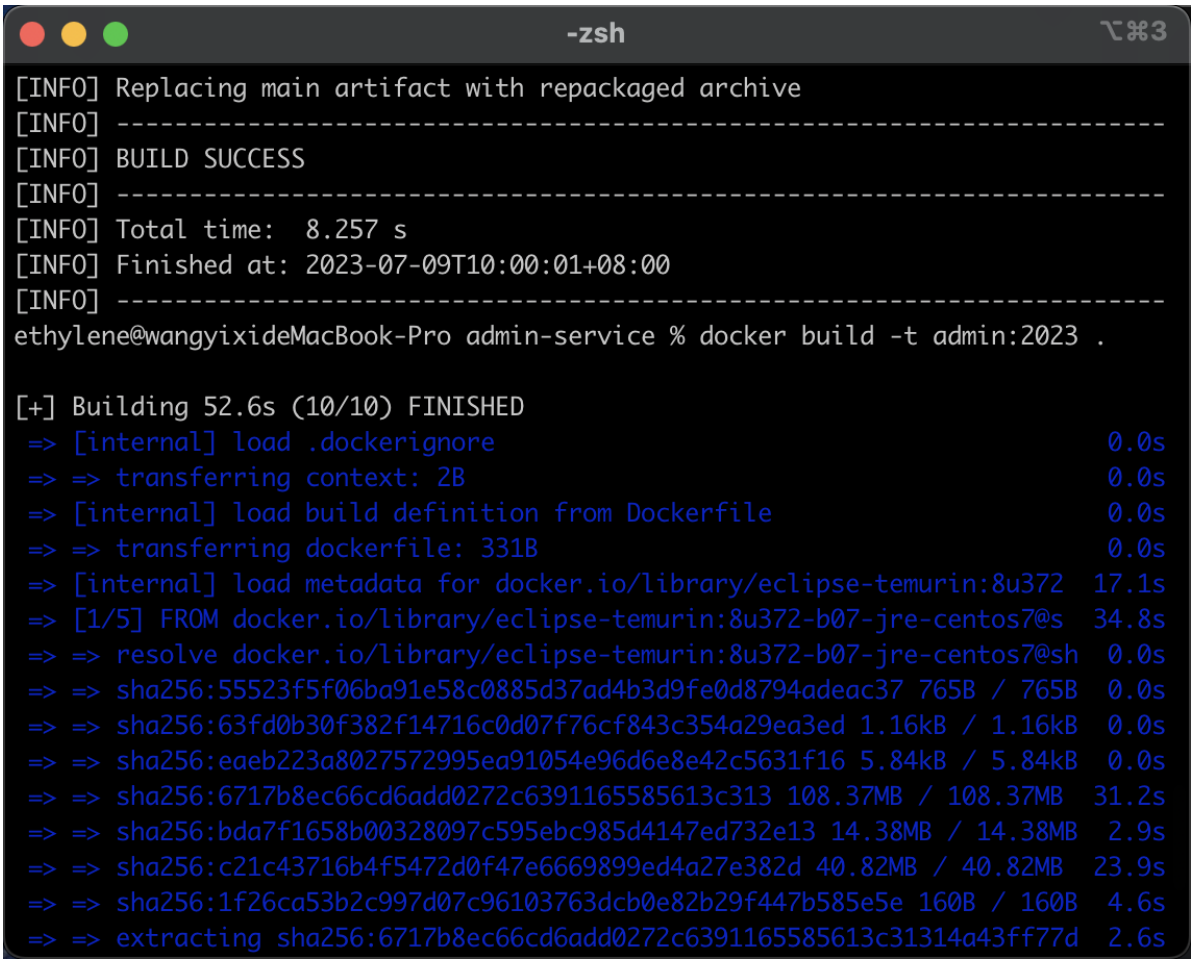
```
ethylene@wangyixideMacBook-Pro hwZ % kubectl run -it --rm --image=mysql:8.0.33 --restart=Never mysql-client -- mysql -h mysql -pdangerous

If you don't see a command prompt, try pressing enter.

mysql> SHOW databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| user               |
+--------------------+
5 rows in set (0.02 sec)

mysql> USER user
    -> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'USER user' at line 1
mysql> USE user
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SHOW TABLES;
+--------------------+
| Tables_in_user     |
+--------------------+
| hibernate_sequence |
| user               |
+--------------------+
2 rows in set (0.00 sec)

mysql> select * from user;
Empty set (0.00 sec)

mysql> CREATE TABLE usr(id INT NOT NULL, name VARCHAR(32) NOT NULL, pwd VARCHAR(64) NOT NULL);
Query OK, 0 rows affected (0.05 sec)

mysql> INSERT INTO usr (id,name,pwd) VALUES(1,'wyx','password');
Query OK, 1 row affected (0.00 sec)

mysql> select * from usr
    -> ;
+----+------+----------+
| id | name | pwd      |
+----+------+----------+
|  1 | wyx  | password |
+----+------+----------+
1 row in set (0.00 sec)

mysql>
```

## 二. 构建镜像并配置

构建Eureka镜像并配置

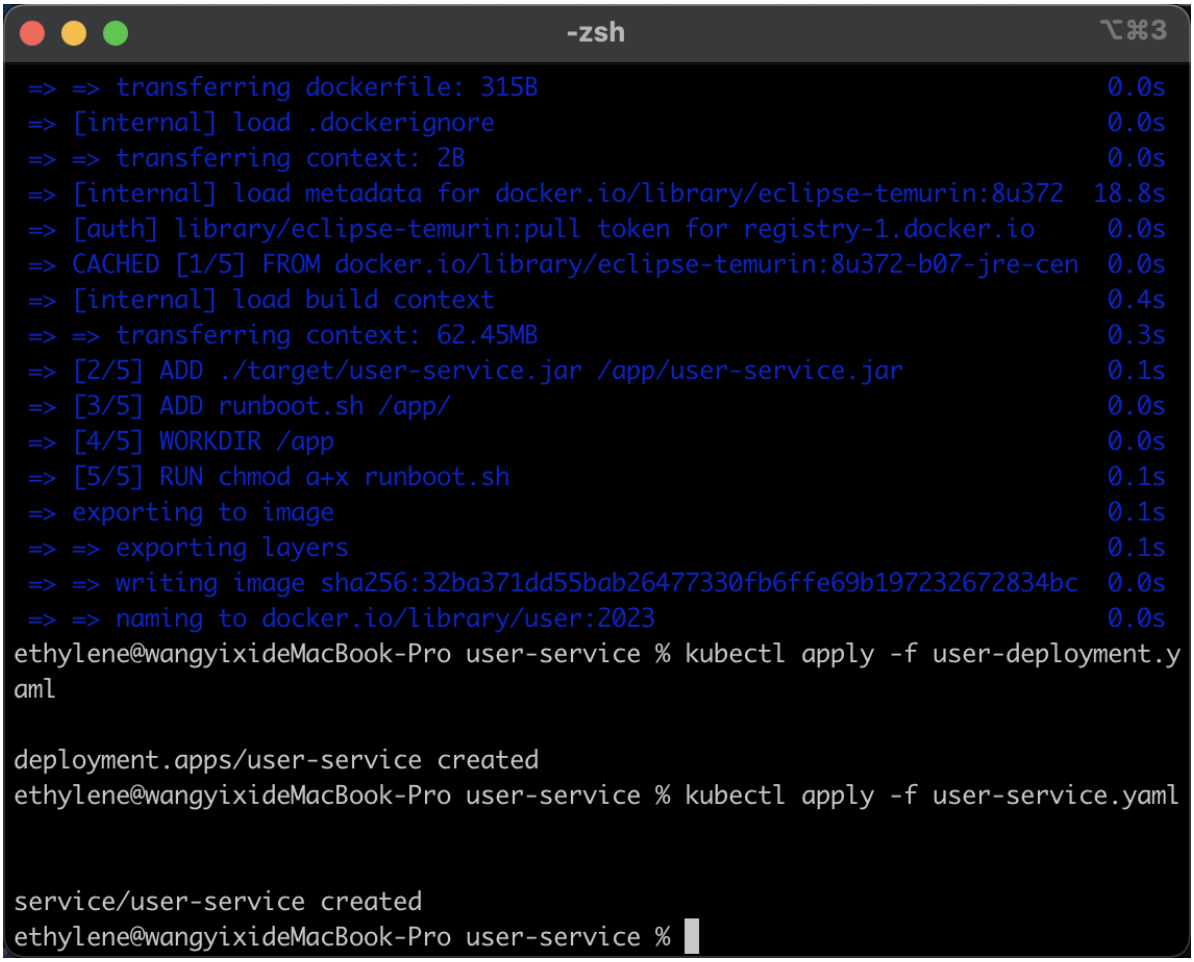
### 配置admin镜像

```
mvn -B -Dmaven.test.skip clean package
docker build -t admin-service:2023 .
kubectl apply -f admin-deployment.yaml
kubectl apply -f admin-service.yaml
```




### 配置user镜像

```
mvn -B -Dmaven.test.skip clean package
docker build -t user-service:2023 .
kubectl apply -f user-deployment.yaml
kubectl apply -f user-service.yaml
```



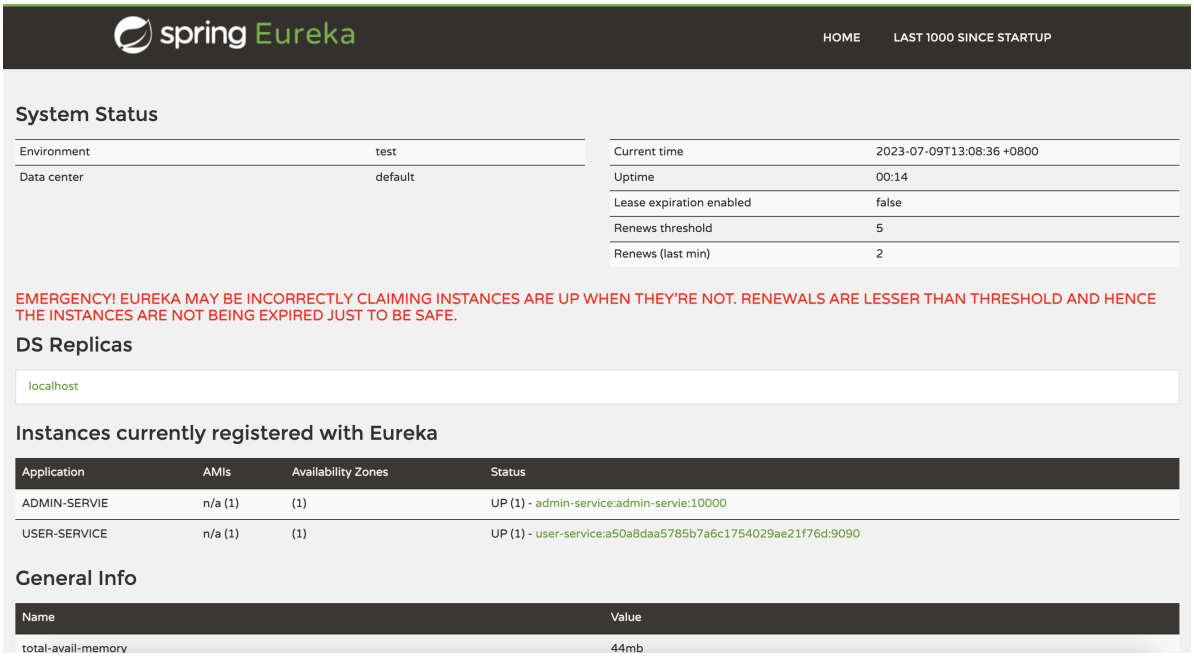

### 配置eureka镜像

```
mvn -B -Dmaven.test.skip clean package
docker build -t eureka:2023 .
kubectl apply -f eureka-deployment.yaml
kubectl apply -f eureka-service.yaml
```

```
-zsh
[INFO]
[INFO] --- compiler:3.8.1:testCompile (default-testCompile) @ eureka-server ---
[INFO] Not compiling test sources
[INFO]
[INFO] --- surefire:2.22.2:test (default-test) @ eureka-server ---
[INFO] Tests are skipped.
[INFO]
[INFO] --- jar:3.2.0:jar (default-jar) @ eureka-server ---
[INFO] Building jar: /Users/ethylene/Learning/2023Spring/cloudNative/hw/hw2/eureka-server/target/eureka-server.jar
[INFO]
[INFO] --- spring-boot:2.3.1.RELEASE:repackage (repackage) @ eureka-server ---
[INFO] Replacing main artifact with repackaged archive
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  24.346 s
[INFO] Finished at: 2023-07-09T10:24:24+08:00
[INFO] ------------------------------------------------------------------------
ethylene@wangyixideMacBook-Pro eureka-server % docker build -t eureka:2023 .

[+] Building 25.7s (11/11) FINISHED
 => [internal] load build definition from Dockerfile                                                                                                0.0s
 => => transferring dockerfile: 378B                                                                                                                0.0s
 => [internal] load .dockerignore                                                                                                                   0.0s
 => => transferring context: 2B                                                                                                                     0.0s
 => [internal] load metadata for docker.io/library/eclipse-temurin:8u372-b07-jre-centos7                                                           25.0s
 => [auth] library/eclipse-temurin:pull token for registry-1.docker.io                                                                              0.0s
 => CACHED [1/5] FROM docker.io/library/eclipse-temurin:8u372-b07-jre-centos7@sha256:55523f5f06ba91e58c0885d37ad4b3d9fe0d8794adeac375f691472c2d9e0a32  0.0s
 => [internal] load build context                                                                                                                   0.3s
 => => transferring context: 49.82MB                                                                                                                0.3s
 => [2/5] ADD ./target/eureka-server.jar /app/eureka-server.jar                                                                                     0.1s
 => [3/5] ADD runboot.sh /app/                                                                                                                      0.0s
 => [4/5] WORKDIR /app                                                                                                                              0.0s
 => [5/5] RUN chmod a+x runboot.sh                                                                                                                  0.1s
 => exporting to image                                                                                                                              0.1s
 => => exporting layers                                                                                                                             0.1s
 => => writing image sha256:e0506500483b681c47dbed65fca93edccb1d24c38e6e5156a21d69ea8d1ee30b                                                        0.0s
 => => naming to docker.io/library/eureka:2023                                                                                                      0.0s
ethylene@wangyixideMacBook-Pro eureka-server % kubectl apply -f eureka-deployment.yaml

deployment.apps/eurkea created
ethylene@wangyixideMacBook-Pro eureka-server % kubectl apply -f eureka-service.yaml

service/eureka created
ethylene@wangyixideMacBook-Pro eureka-server %
```

```
ethylene@wangyixideMacBook-Pro user-service % kubectl get svc
NAME              TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)           AGE
admin-service     NodePort       10.111.71.52    <none>        10000:30904/TCP   114m
eureka            NodePort       10.97.138.7     <none>        8080:30308/TCP    90m
kubernetes        ClusterIP      10.96.0.1       <none>        443/TCP           11d
mysql             ClusterIP      None            <none>        3306/TCP          12h
nginx-service     LoadBalancer   10.107.67.12    localhost     80:31939/TCP      9d
python            NodePort       10.102.52.217   <none>        8000:31945/TCP    6d20h
user-service      NodePort       10.107.34.76    <none>        9090:31053/TCP    112m
ethylene@wangyixideMacBook-Pro user-service %
```

在Eureka-Server上面成功运行




# 三.通过curl指令进行添加用户的尝试

```
ethylene@wangyixideMacBook-Pro user-service % kubectl get pods
NAME                               READY   STATUS    RESTARTS      AGE
admin-service-84444789db-fs64n     1/1     Running   0             14m
eurkea-76cff4d569-mgcjc            1/1     Running   0             13m
user-service-7cb8755bd7-9n8x4      1/1     Running   1 (4s ago)    16m
ethylene@wangyixideMacBook-Pro user-service % kubectl get svc
NAME             TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)           AGE
admin-service    NodePort       10.111.71.52    <none>        10000:30904/TCP   3h8m
eureka           NodePort       10.97.138.7     <none>        8080:30308/TCP    164m
kubernetes       ClusterIP      10.96.0.1       <none>        443/TCP           12d
mysql            ClusterIP      None            <none>        3306/TCP          13h
nginx-service    LoadBalancer   10.107.67.12    localhost     80:31939/TCP      9d
python           NodePort       10.102.52.217   <none>        8000:31945/TCP    6d21h
user-service     NodePort       10.107.34.76    <none>        9090:31053/TCP    3h5m
ethylene@wangyixideMacBook-Pro user-service %
```

```
curl -H "Content-Type: application/json"  -X POST --data '{"name": "grissom", "pwd": "nju2023"}' http://127.0.0.1:30904/user
```

```
ethylene@wangyixideMacBook-Pro hw2 % cd ./user-service
ethylene@wangyixideMacBook-Pro user-service % curl -H "Content-Type: application/json"  -X POST --data '{"name": "grissom", "pwd": "nju2023"}' http://127.0.0.1:30904/user
{"id":1,"name":"grissom","pwd":"nju2023"}%
ethylene@wangyixideMacBook-Pro user-service % curl -H "Content-Type: application/json"  -X POST --data '{"name": "c2h4", "pwd": "nju2023"}' http://127.0.0.1:30904/user
{"id":2,"name":"c2h4","pwd":"nju2023"}%
ethylene@wangyixideMacBook-Pro user-service % curl -H "Content-Type: application/json"  -X POST --data '{"name": "c2h4", "pwd": "nju2023"}' http://127.0.0.1:31053/user
{"id":3,"name":"c2h4","pwd":"nju2023"}%
ethylene@wangyixideMacBook-Pro user-service %
```

kubectl ⌥⌘3

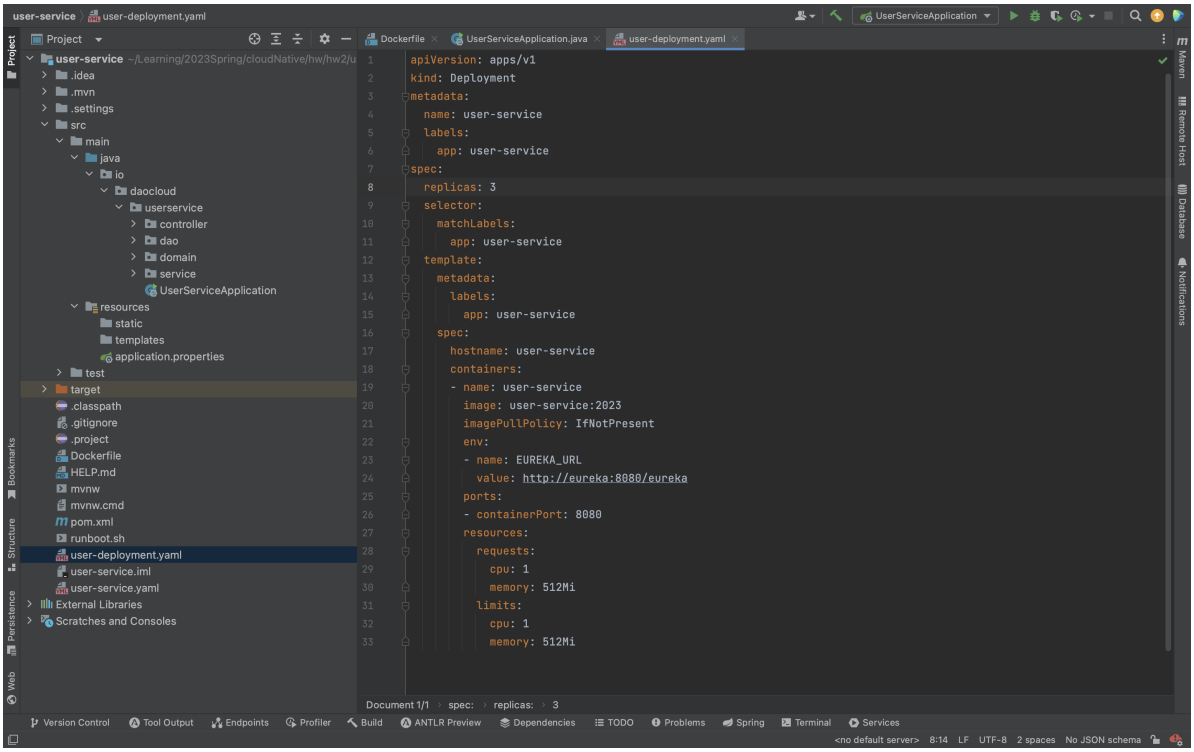
```
mysql> select * from usr
    -> ;
+----+------+----------+
| id | name | pwd      |
+----+------+----------+
|  1 | wyx  | password |
+----+------+----------+
1 row in set (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| user               |
+--------------------+
5 rows in set (0.01 sec)

mysql> USE user
Database changed
mysql> SHOW TABLES
    -> ;
+--------------------+
| Tables_in_user     |
+--------------------+
| hibernate_sequence |
| user               |
| usr                |
+--------------------+
3 rows in set (0.01 sec)

mysql> SELECT * FROM user;
+----+---------+---------+
| id | name    | pwd     |
+----+---------+---------+
|  1 | grissom | nju2023 |
|  2 | c2h4    | nju2023 |
|  3 | c2h4    | nju2023 |
+----+---------+---------+
3 rows in set (0.01 sec)

mysql>
```

尝试输入空密码或空用户名发现无法插入

```
ethylene@wangyixideMacBook-Pro user-service % curl -H "Content-Type: application/json"  -X POST --data '{"name": "grissom", "pwd": ""}' http://127.0.0.1:30904/user

{"error":"密码不能为空"}%
ethylene@wangyixideMacBook-Pro user-service % curl -H "Content-Type: application/json"  -X POST --data '{"name": "", "pwd": "nju2023"}' http://127.0.0.1:30904/user

{"error":"用户名不能为空"}%
ethylene@wangyixideMacBook-Pro user-service %
```

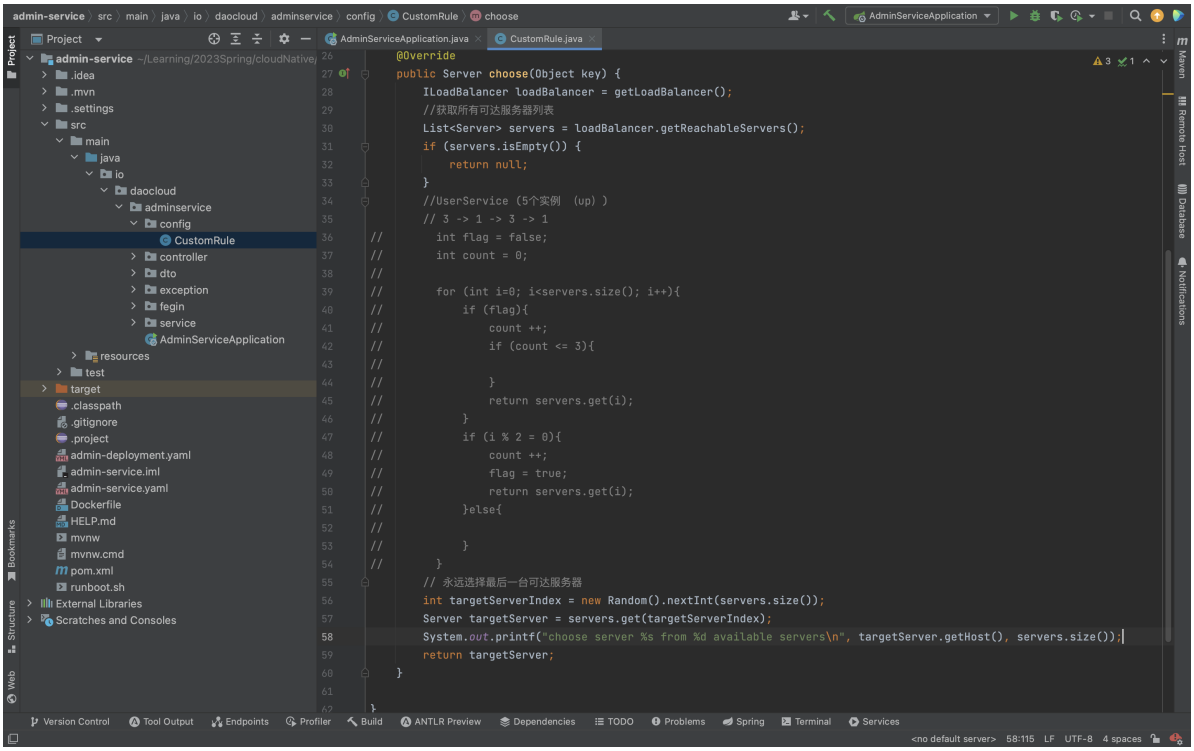
# 四.实现负载均衡

修改user-deployment，Deployment 配置会启动 3 个 Pod 副本，并将它们标记为 `app=user-service`，以便实现负载均衡




修改src/main/java/io/daocloud/adminservice/config/CustomRule.java中的choose方法

```
@Override
    public Server choose(Object key) {

        ILoadBalancer loadBalancer = getLoadBalancer();

        // 获取所有可达服务器列表
        List<Server> servers = loadBalancer.getReachableServers();
        if (servers.isEmpty()) {
            return null;
        }

        //用一个随机函数，随机访问实现负载均衡
        int targetServerIndex = new Random().nextInt(servers.size());
        Server targetServer = servers.get(targetServerIndex);

        System.out.printf("choose server %s from %d available servers\n",
targetServer.getHost(), servers.size());

        return targetServer;
    }
```




修改后在admin-service的日志中发现每次插入都可能使用不同的userservice