

如何部署 OnlineJudge

何雨菁

时间管理超级大师组

版本：1.0

更新：2020 年 9 月 22 日

1 查看容器

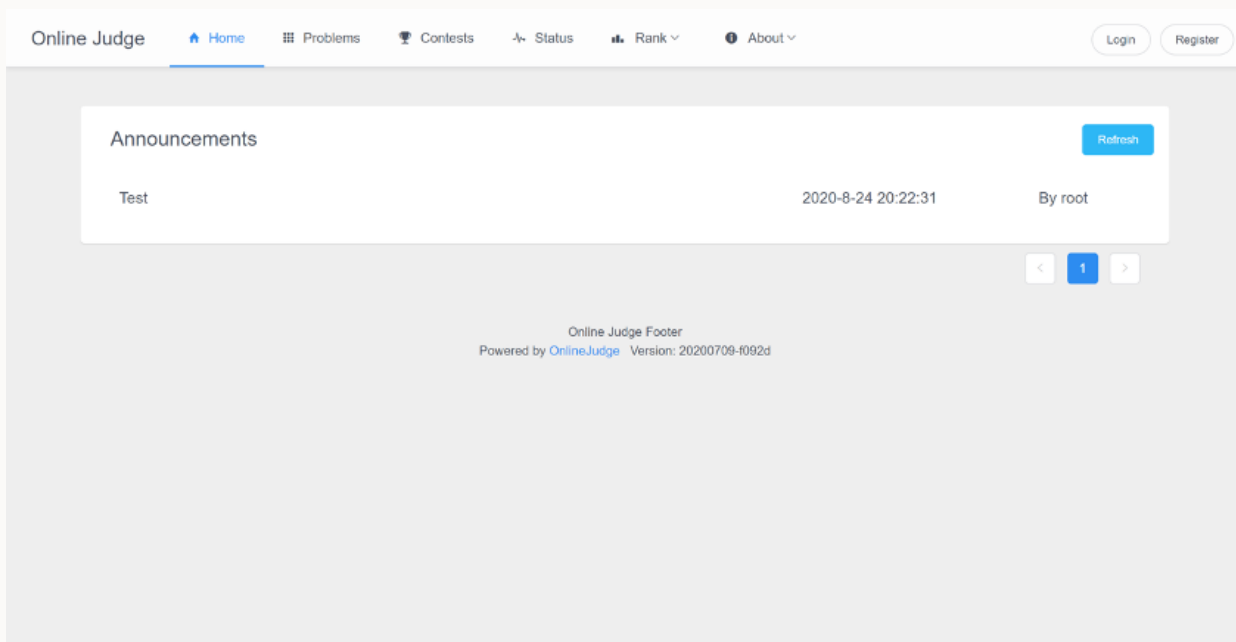
使用 docker 命令查看容器的状态（Healthy 表示运行正常）

```
docker ps -a
```

ref:<https://github.com/QingdaoU/OnlineJudgeDeploy>

| CONTAINER ID | IMAGE | NAMES | COMMAND | CREATED | STATUS | PORTS |
|-------------------|--|--------------|--------------------------|-------------|-------------|------------------|
| 2fe908b8c09f | registry.cn-hangzhou.aliyuncs.com/onlinejudge/oj_backend | | "/bin/sh -c /app/dep..." | 2 weeks ago | Up 24 hours | (healthy) 0.0.0. |
| 0:443->1443/tcp | 0.0.0.0:80->8000/tcp | oj-backend | | | | |
| 629a355e61c5 | registry.cn-hangzhou.aliyuncs.com/onlinejudge/judge_server | | "/bin/sh -c /code/en..." | 2 weeks ago | Up 24 hours | (healthy) 8080/t |
| cp | | judge-server | | | | |
| c2f9b83b6fcb | redis:4.0-alpine | oj-redis | "docker-entrypoint.s..." | 2 weeks ago | Up 24 hours | 6379/t |
| cp | | oj-redis | | | | |
| 13eff32655f1 | postgres:10-alpine | oj-postgres | "docker-entrypoint.s..." | 2 weeks ago | Up 24 hours | 0.0.0. |
| 0:12348->5432/tcp | | oj-postgres | | | | |

部署好后浏览器访问 `http://IPaddr/`，可以看到一个 naive 的 oj：



2 连接数据库

OJ 系统部署在虚拟机中的容器中，如果需要连接到容器中的数据库，则需要做一个端口映射

```

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql_psycopg2',
        # 'HOST': '127.0.0.1',
        # 'PORT': 5435,
        'HOST': '10.24.0.39',
        'PORT': 12348,
        'NAME': "onlinejudge",
        'USER': "onlinejudge",
        'PASSWORD': 'onlinejudge'
    }
}

```

注 容器端口映射到云主机端口：最后一行，由端口 12348 映射到 5432

```

10
11 oj-postgres:
12   image: mxzf0213/ecnu_oj:postgres
13   container_name: oj-postgres
14   restart: always
15   volumes:
16     - ./data/postgres:/var/lib/postgresql/data
17   environment:
18     - POSTGRES_DB=onlinejudge
19     - POSTGRES_USER=onlinejudge
20     - POSTGRES_PASSWORD=onlinejudge
21   ports:
22     - "0.0.0.0:12348:5432"
23

```






















```
heyujing@10-24-0-39: ~/OJ/OnlineJudgeDeploy
version: "2"
services:

  oj-redis:
    image: redis:4.0-alpine
    container_name: oj-redis
    restart: always
    volumes:
      - ./data/redis:/data

  oj-postgres:
    image: postgres:10-alpine
    container_name: oj-postgres
    restart: always
    volumes:
      - ./data/postgres:/var/lib/postgresql/data
    environment:
      - POSTGRES_DB=onlinejudge
      - POSTGRES_USER=onlinejudge
      - POSTGRES_PASSWORD=onlinejudge
    ports:
      - "0.0.0.0:12348:5432"
```

2.1 访问数据库

直接使用 pagadmin4 可连接到数据库，该数据库共有 20 个表，Problem 存储我们的习题

- ▼  Tables (20)
 - >  acm_contest_rank
 - >  announcement
 - >  auth_group
 - >  auth_group_permission
 - >  auth_permission
 - >  contest
 - >  contest_announcement
 - >  django_content_type
 - >  django_dramatiq_tasks
 - >  django_migrations
 - >  django_session
 - >  judge_server
 - >  oi_contest_rank
 - >  options_sysoptions
 - >  **problem**
 - >  problem_tag
 - >  problem_tags
 - >  submission
 - >  user
 - >  user_profile

3 为数据库添加字段

我们添加的属性有：

- 标签
- 难度
- 状态

```
# RS_oj
labels = models.TextField(null=True)
difficulty_ = models.IntegerField(default=0)
answer_status = models.IntegerField(default=0)
```

添加好后可以使用命令 `python manage.py makemigrations` 以及 `python manage.py migrate` 之后重新 up 容器，刷新数据库后就可看到字段已经添加成功

| Data | Output | Explain | Messages | Notifications | | | | | | |
|---------|------------|-----------|----------------|-------------------|------------------|-------------|---------------|---------------------|--|--|
| score | contest_id | is_public | spj_compile_ok | io_mode | share_submission | difficulty_ | answer_status | labels | | |
| integer | integer | boolean | boolean | jsonb | boolean | integer | integer | integer[] | | |
| 0 | [null] | false | false | {"input": "inp... | false | 1 | -1 | ,0,0,1,0,0,1,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 3 | 0 | ,0,0,0,0,0,0,1} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 2 | 0 | ,0,0,0,0,0,0,1,1} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 2 | 1 | ,0,0,1,0,0,0,0,1} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 1 | 1 | ,0,0,0,0,0,1,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 3 | -1 | ,0,0,0,0,0,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 2 | 0 | ,0,0,0,1,0,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 3 | 0 | ,0,0,0,1,0,0,0,1} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 1 | 0 | ,0,0,0,0,1,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 2 | 1 | ,0,0,0,1,1,0,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 2 | -1 | ,0,0,0,0,0,0,1,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 2 | 1 | ,0,0,0,0,1,0,1,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 1 | -1 | ,0,0,0,0,1,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 3 | -1 | ,0,0,1,0,0,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 1 | 1 | ,0,1,0,0,0,0,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 3 | 1 | ,1,0,0,0,0,0,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 2 | 0 | ,0,0,0,0,0,0,0,1} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 1 | 0 | ,0,0,1,0,0,0,0,0,0} | | |
| 0 | [null] | false | false | {"input": "inp... | false | 2 | -1 | ,0,0,0,0,1,1,1,0,0} | | |

注 蓝色标注部分为新添加的属性，用于推荐系统