

15-0 sequelize 数据库配置

安装并配置 egg-sequelize 插件（它会辅助我们将定义好到 Model 对象加载到 app 和 ctx 上）和 mysql2 插件

```
1 npm install --save egg-sequelize mysql2
```

然后在 [config/plugin.js](#) 中引入 egg-sequelize 插件

```
1 'use strict';
2
3 /** @type Egg.EggPlugin */
4 module.exports = {
5   cors: {
6     enable: true,
7     package: 'egg-cors',
8   },
9   sequelize: {
10     enable: true,
11     package: 'egg-sequelize',
12   },
13 };
14
```

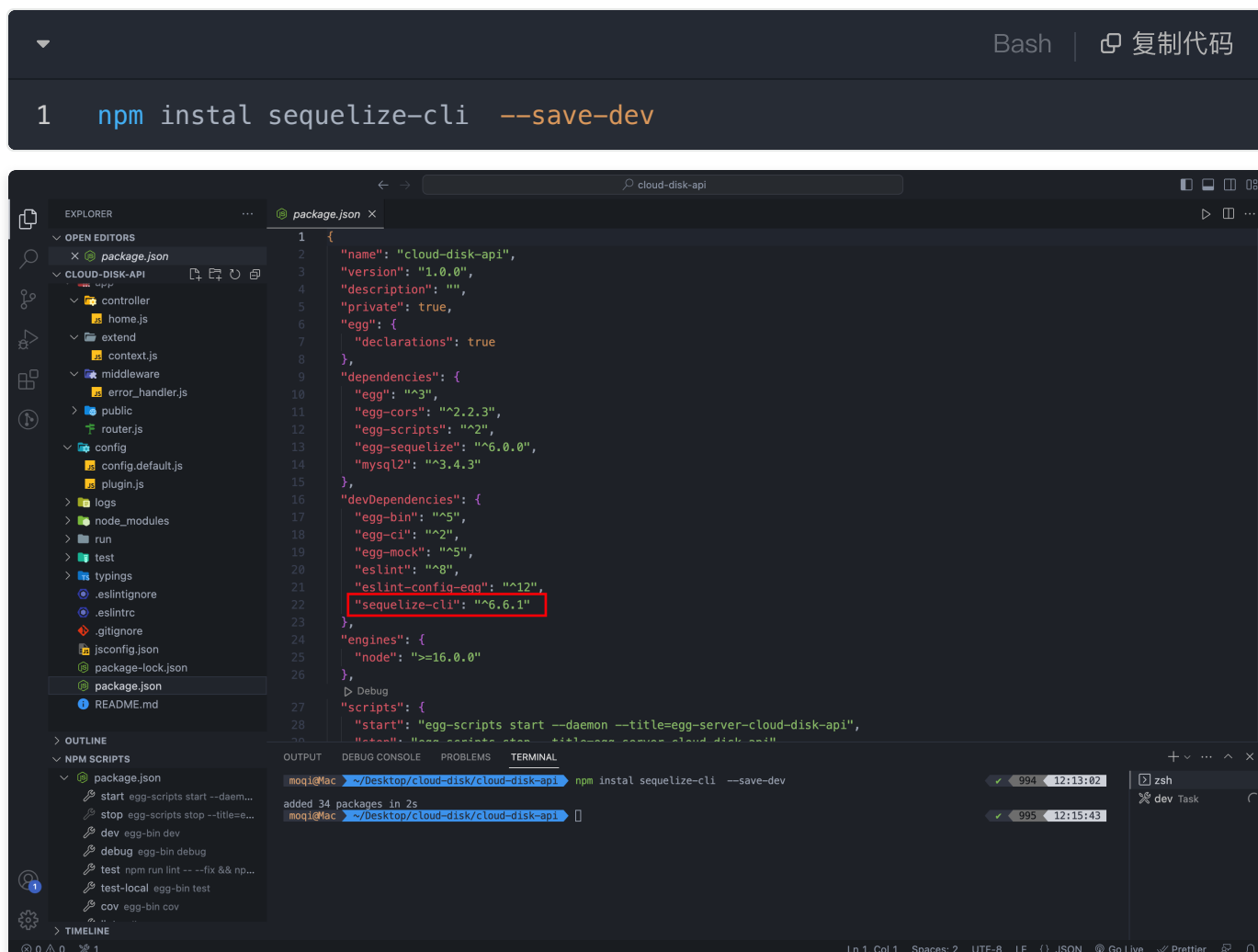
在 [config/config.default.js](#) 中加入 sequelize 的配置，注意根据你的实际情况修改数据库连接信息（不用自己创建数据库）

```
1 config.sequelize = {
2   dialect: 'mysql',
3   host: '127.0.0.1',
4   username: 'root',
5   password: '123456',
6   port: 3306,
7   database: 'cloud_disk',
8   // 中国时区
9   timezone: '+08:00',
10  define: {
11    // 取消数据表名复数
12    freezeTableName: true,
13    // 自动写入时间戳 created_at updated_at
14    timestamps: true,
15    // 字段生成软删除时间戳 deleted_at
16    // paranoid: true,
17    createdAt: 'created_time',
18    updatedAt: 'updated_time',
19    // deletedAt: 'deleted_time',
20    // 所有驼峰命名格式化
21    underscored: true
22  }
23 };
24
```



- 数据库迁移配置

1. sequelize 提供了 [sequelize-cli](#) 工具来实现 [Migrations](#)，我们可以在 egg 项目中引入 sequelize-cli



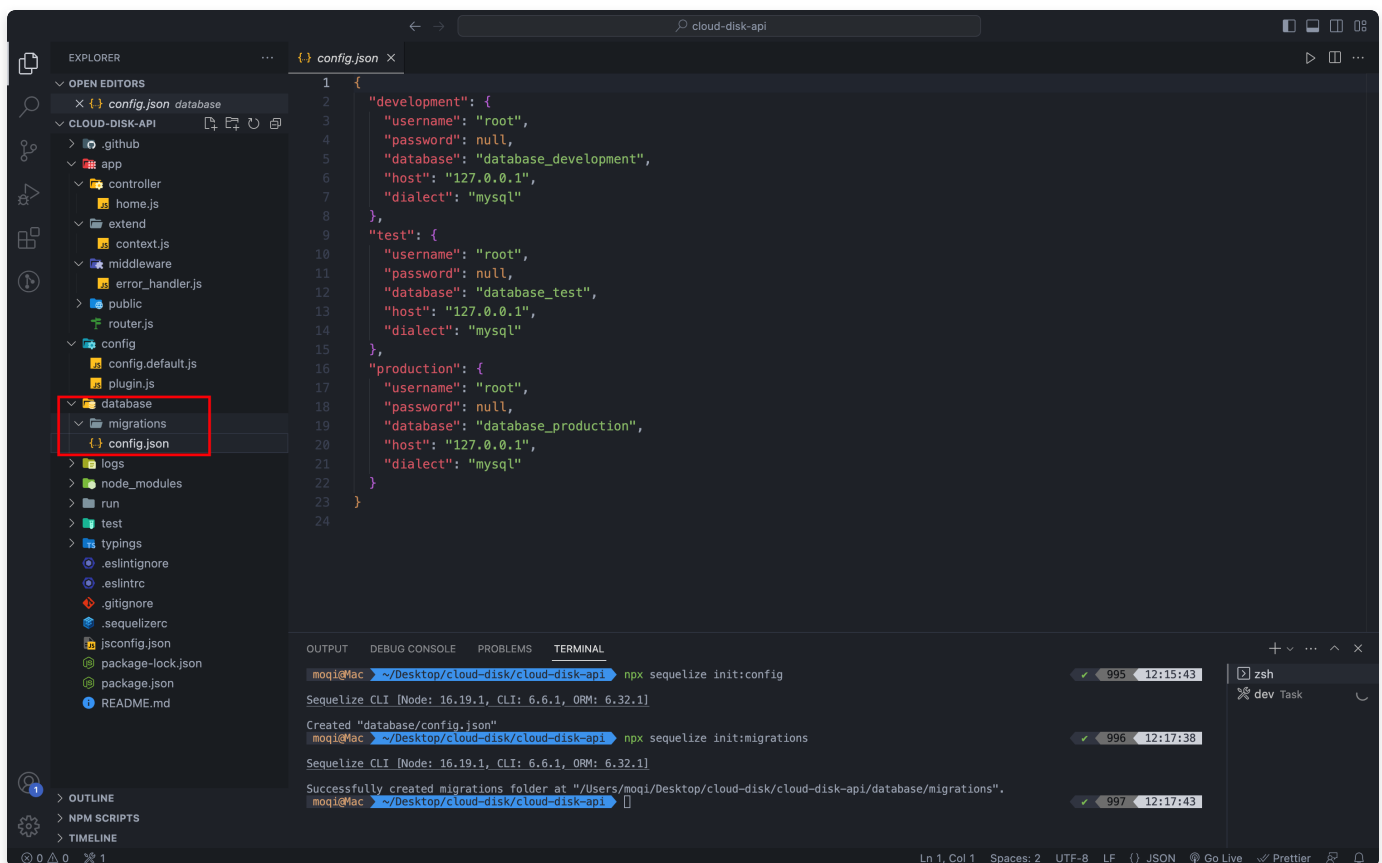
2. 我们希望将所有数据库 Migrations 相关的内容都放在 `database` 目录下，所以在项目根目录下新建一个 `.sequelizerc` 配置文件

```
1 'use strict';
2
3 const path = require('path');
4
5 module.exports = {
6   config: path.join(__dirname, 'database/config.json'),
7   'migrations-path': path.join(__dirname, 'database/migrations'),
8   'seeders-path': path.join(__dirname, 'database/seeders'),
9   'models-path': path.join(__dirname, 'app/model'),
10 };
```

3. 初始化 Migrations 配置文件和目录，在根目录分别运行以下命令

```
1 npx sequelize init:config
2 npx sequelize init:migrations
```

运行结果如图



会生成 database/config.json 文件和 database/migrations 目录，我们修改一下

database/config.json 中 **development** 开发环境配置的内容，将其改成项目中使用的数据库配置：

```
config.json ×
1 {
2   "development": {
3     "username": "root",
4     "password": "123456",
5     "database": "cloud_disk",
6     "host": "127.0.0.1",
7     "dialect": "mysql",
8     "timezone": "+08:00"
9   },
10  "test": {
11    "username": "root",
12    "password": null,
13    "database": "database_test",
14    "host": "127.0.0.1",
15    "dialect": "mysql"
16  },
17  "production": {
18    "username": "root",
19    "password": null,
20    "database": "database_production",
21    "host": "127.0.0.1",
22    "dialect": "mysql"
23  }
24 }
```

4. 创建数据库

```
▼ Bash | 复制代码
1 npx sequelize db:create
```

创建成功

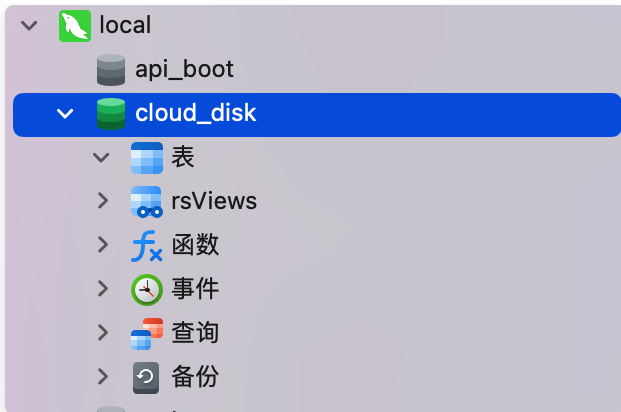
```
OUTPUT  DEBUG CONSOLE  PROBLEMS  TERMINAL

moqi@Mac > ~/Desktop/cloud-disk/cloud-disk-api npx sequelize db:create

Sequelize CLI [Node: 16.19.1, CLI: 6.6.1, ORM: 6.32.1]

Loaded configuration file "database/config.json".
Using environment "development".
Database cloud_disk created.
moqi@Mac > ~/Desktop/cloud-disk/cloud-disk-api
```

查看数据库



5. 升级数据库

```
▼ Bash | 复制代码

1 npx sequelize db:migrate
```

- 模型关联
 - 后面会用到对象之间一对一、一对多、多对多会需要
- 判断是否是移动端

在扩展文件：app/extend/context.js，顺着前面内容加入，注意每组配置之间的逗号

```
▼ JavaScript | 复制代码

1 ismobile(ctx){
2     let userAgent = this.request.header['user-agent'].toLowerCase();
3     let pat_phone = /ipad|iphone os|midp|rv:1.2.3.4|ucweb|android|windows c
e|windows mobile/;
4     return pat_phone.test(userAgent);
5 }
```

