

配置Neo4j

pom.xml

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-neo4j</artifactId>
</dependency>

<dependency>
    <groupId>org.neo4j</groupId>
    <artifactId>neo4j-ogm-bolt-driver</artifactId>
</dependency>
```

application.properties

```
server.port=8080
spring.data.neo4j.username=neo4j
spring.data.neo4j.password=0000
spring.data.neo4j.uri=bolt://localhost:7687
```

Entity、Dao、Service、Controller

1.Entity层：实体层，数据库在项目中的类

主要用于定义与数据库对象应的属性，提供get/set方法,tostring方法,有参无参构造函数。

2.Dao层：持久层，主要与数据库交互

DAO层首先会创建Dao接口，接着就可以在配置文件中定义该接口的实现类；接着就可以在模块中调用Dao的接口进行数据业务的处理，而不用关注此接口的具体实现类是哪一个类，Dao层的数据源和数据库连接的参数都是在配置文件中配置。

3.Service层：业务层 控制业务

业务模块的逻辑应用设计，和DAO层一样都是先设计接口，再创建要实现的类，然后在配置文件中配置其实现的关联。接下来就可以在service层调用接口进行业务逻辑应用的处理。

好处：封装Service层的业务逻辑有利于业务逻辑的独立性和重复利用性。

4.Controller层：控制层 控制业务逻辑

具体的业务模块流程的控制，controller层主要调用Service层里面的接口控制具体的业务流程

Controller和Service的区别是：Controller负责具体的业务模块流程的控制；Service层负责业务模块的逻辑应用设计

总结：具体的一个项目中有：controller层调用了Service层的接口方法，Service层调用Dao层的方法，其中调用的参数是使用Entity层进行传递的。