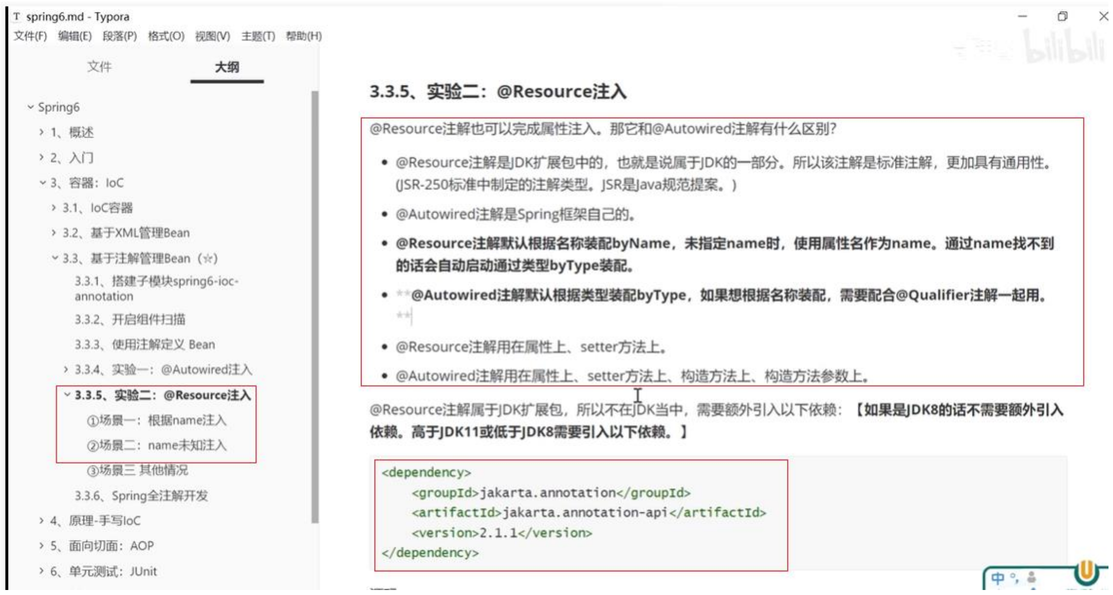

3.3.5、实验二：@Resource注入

@Resource注解也可以完成属性注入。那它和@Autowired注解有什么区别?

- @Resource注解是JDK扩展包中的，也就是说属于JDK的一部分。所以该注解是标准注解，更加具有通用性。(JSR-250标准中制定的注解类型。JSR是Java规范提案。)
- @Autowired注解是Spring框架自己的。
- **@Resource注解默认根据名称装配byName，未指定name时，使用属性名作为name。通过name找不到的话会自动启动通过类型byType装配。**
- **@Autowired注解默认根据类型装配byType，如果想根据名称装配，需要配合@Qualifier注解一起用。**
- @Resource注解用在属性上、setter方法上。
- @Autowired注解用在属性上、setter方法上、构造方法上、构造方法参数上。

@Resource注解属于JDK扩展包，所以不在JDK当中，需要额外引入以下依赖：**【如果是JDK8的话不需要额外引入依赖。高于JDK11或低于JDK8需要引入以下依赖。】**

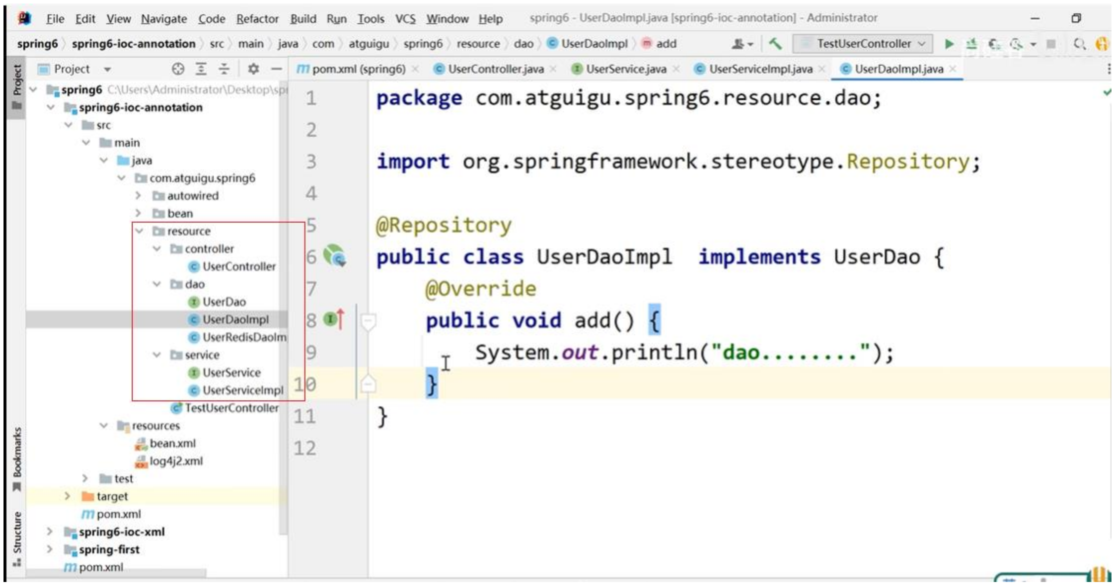
```xml
<dependency>
    <groupId>jakarta.annotation</groupId>
    <artifactId>jakarta.annotation-api</artifactId>
    <version>2.1.1</version>
</dependency>
```

注意两者的区别和在父工程中引入依赖的包。

环境准备：

1、引入依赖的包，上图有内容

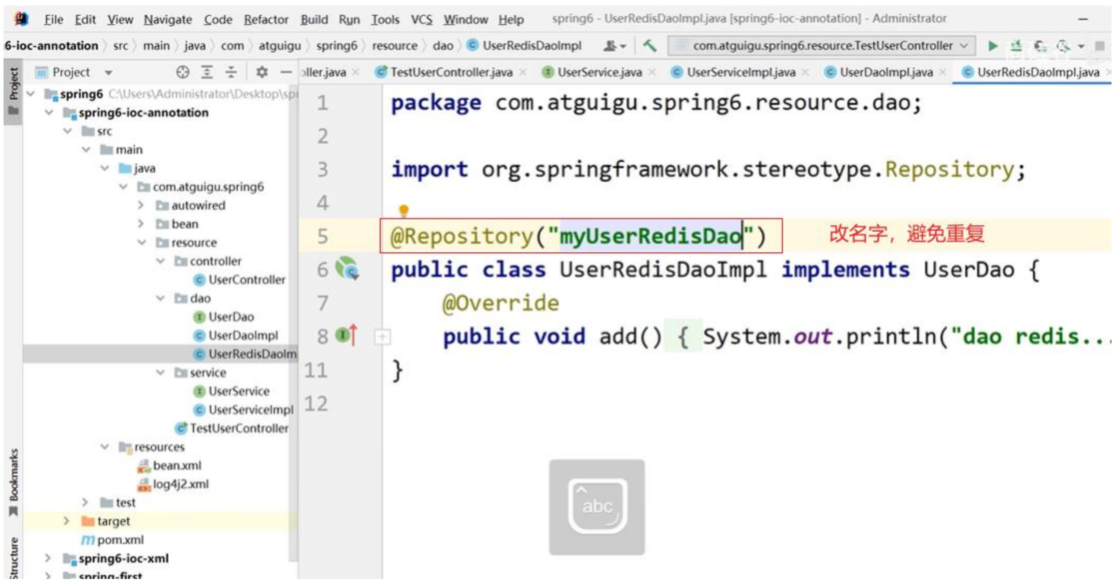
2、复制一份相同的结构，删除之前的包：


```java
package com.atguigu.spring6.resource.dao;

import org.springframework.stereotype.Repository;

@Repository
public class UserDaoImpl  implements UserDao {
    @Override
    public void add() {
        System.out.println("dao........");
    }
}
```

根据名称注入：

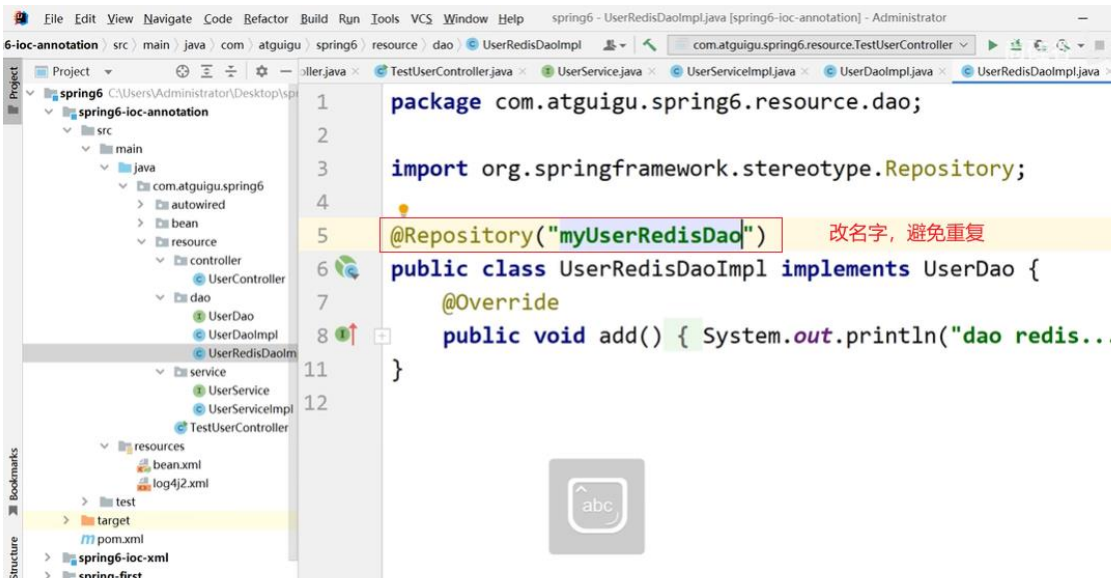


```java
package com.atguigu.spring6.resource.dao;

import org.springframework.stereotype.Repository;

@Repository("myUserRedisDao")
public class UserRedisDaoImpl implements UserDao {
    @Override
    public void add() { System.out.println("dao redis...
}
```

改名字，避免重复

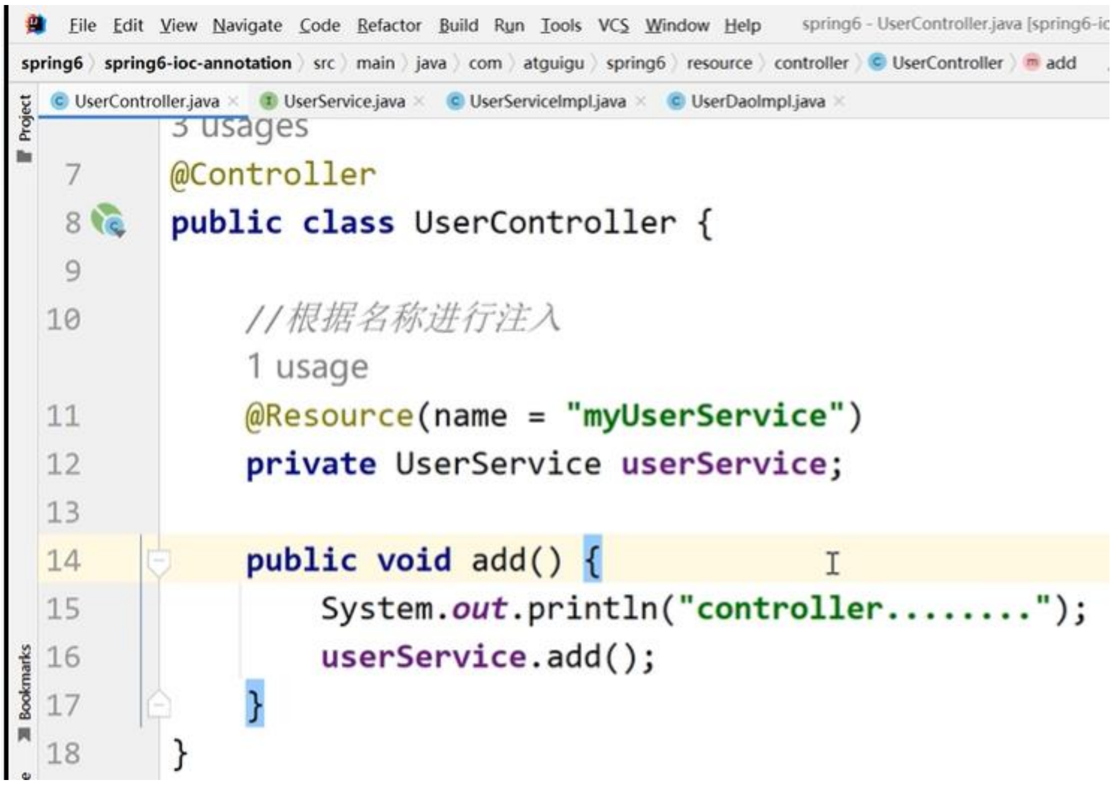


```java
package com.atguigu.spring6.resource.dao;

import org.springframework.stereotype.Repository;

@Repository("myUserRedisDao")
public class UserRedisDaoImpl implements UserDao {
    @Override
    public void add() { System.out.println("dao redis...
}
```

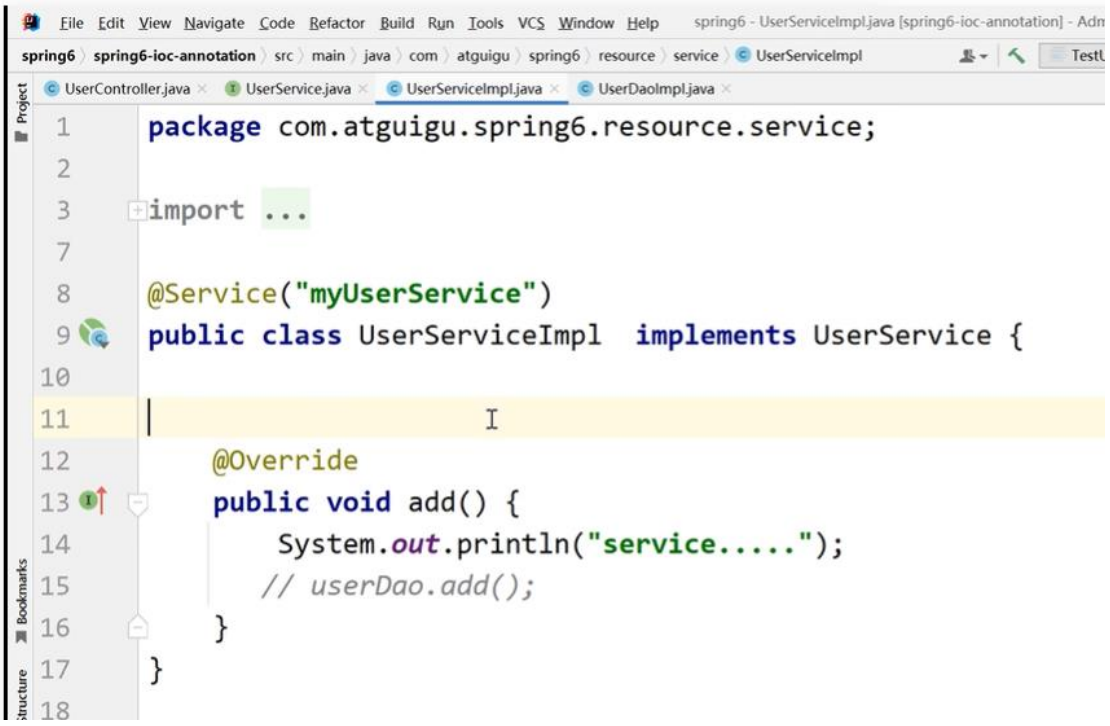
改名字，避免重复


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
spring6 - UserController.java [spring6-ic
spring6 › spring6-ioc-annotation › src › main › java › com › atguigu › spring6 › resource › controller › UserController › add
UserController.java
UserService.java
UserServiceImpl.java
UserDaoImpl.java
Project
Bookmarks

```java
3 usages
@Controller
public class UserController {

    //根据名称进行注入
    1 usage
    @Resource(name = "myUserService")
    private UserService userService;

    public void add() {
        System.out.println("controller........");
        userService.add();
    }
}
```


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
spring6 - UserServiceImpl.java [spring6-ioc-annotation] - Adm
spring6 › spring6-ioc-annotation › src › main › java › com › atguigu › spring6 › resource › service › UserServiceImpl
TestU
UserController.java
UserService.java
UserServiceImpl.java
UserDaoImpl.java
Project
Bookmarks
Structure

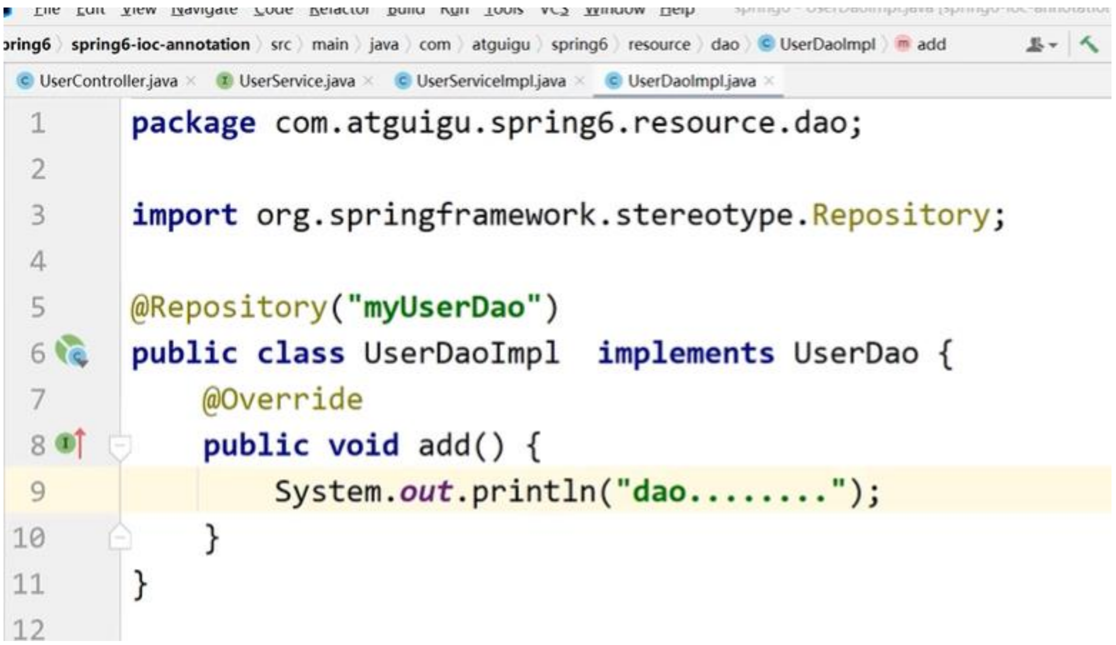
```java
package com.atguigu.spring6.resource.service;

import ...

@Service("myUserService")
public class UserServiceImpl  implements UserService {


    @Override
    public void add() {
        System.out.println("service.....");
    //  userDao.add();
    }
}
```

根据属性名称注入:

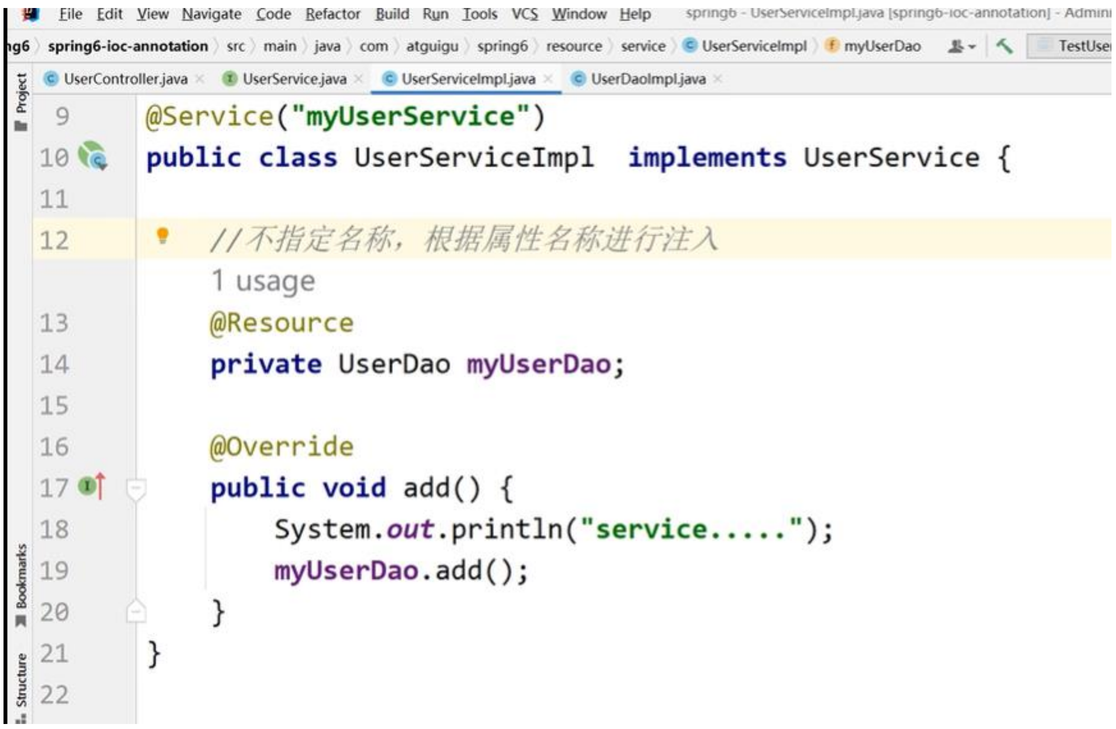


spring6 › spring6-ioc-annotation › src › main › java › com › atguigu › spring6 › resource › dao › UserDaoImpl › add
UserController.java
UserService.java
UserServiceImpl.java
UserDaoImpl.java

```java
package com.atguigu.spring6.resource.dao;

import org.springframework.stereotype.Repository;

@Repository("myUserDao")
public class UserDaoImpl  implements UserDao {
    @Override
    public void add() {
        System.out.println("dao........");
    }
}
```



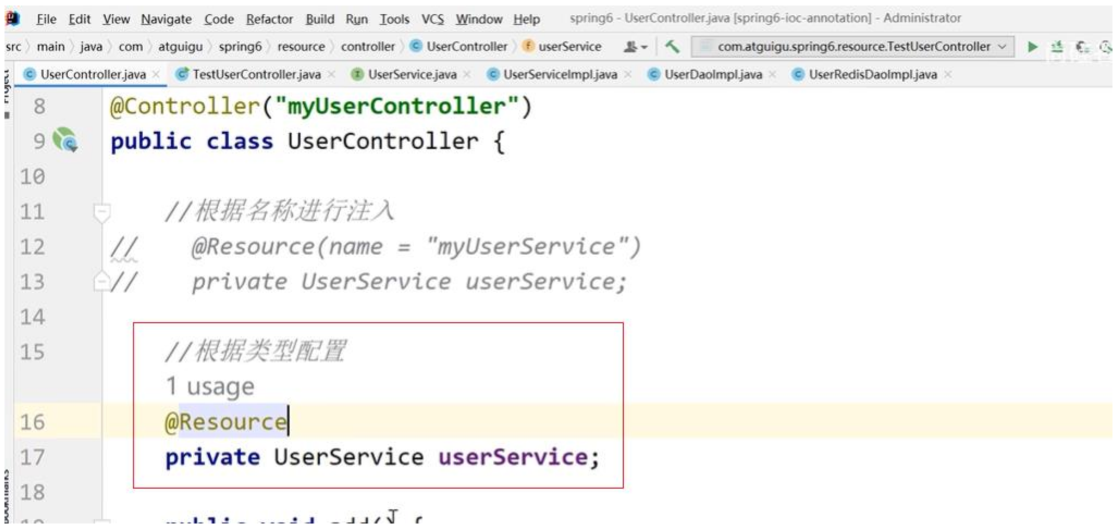
File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
spring6 - UserServiceImpl.java [spring6-ioc-annotation] - Admini
spring6-ioc-annotation › src › main › java › com › atguigu › spring6 › resource › service › UserServiceImpl › myUserDao
UserController.java
UserService.java
UserServiceImpl.java
UserDaoImpl.java
Project
Bookmarks
Structure

```java
@Service("myUserService")
public class UserServiceImpl  implements UserService {

    //不指定名称，根据属性名称进行注入
    1 usage
    @Resource
    private UserDao myUserDao;

    @Override
    public void add() {
        System.out.println("service.....");
        myUserDao.add();
    }
}
```

根据类型进行匹配:



File Edit View Navigate Code Refactor Build Run Tools VCS Window Help spring6 - UserController.java [spring6-ioc-annotation] - Administrator
src 〉 main 〉 java 〉 com 〉 atguigu 〉 spring6 〉 resource 〉 controller 〉 UserController 〉 userService
com.atguigu.spring6.resource.TestUserController
UserController.java TestUserController.java UserService.java UserServiceImpl.java UserDaoImpl.java UserRedisDaoImpl.java

```java
@Controller("myUserController")
public class UserController {

    //根据名称进行注入
//    @Resource(name = "myUserService")
//    private UserService userService;

    //根据类型配置
    1 usage
    @Resource
    private UserService userService;

```

总结:

@Resource注解：默认byName注入，没有指定name时把属性名当做name，根据name找不到时，才会byType注入。byType注入时，某种类型的Bean只能有一个