

# 知识点和总结

http  
json  
xml  
tomcat  
Servlet  
jdbc sql数据库 使用PreparedStatement实现CRUD操作  
controller--service--dao  
maven  
spring boot  
MyBatis  
postman  
lombok 会操作 Maven 的聚合，继承，了解分模块开发  
学习如何使用Spring Security来添加安全层

最开始就一个一个搜这些词语是什么意思，但是理解不了，就在b站找课看，但是又不知道看哪一个课，搜了JDBC，前后端等关键词之后，都没有找到能够有哪一门课能覆盖这些陌生的词语，也是找了好久，才找到了黑马的javaweb这门课，在搞清楚要学什么的路上花了挺长时间的，然后一直往写0101的题目努力，结果一写完，发现黑马的课基本上把0111和1000的好多内容都讲了，距离截止时间也没两天了，就索性把0101，0110，1000结合在一起，给自己所学到的做一个总结。

## 遇到的傻问题

网页打不开，发现是端口被占用了  
不知道html文件怎么放到idea中，一直放在在vscode  
不知道怎么选择浏览器打开，发现原来右上角就可以选  
浏览器打开了不能运行，原来还要先让后端代码开始运行，前端才能运行  
不知道怎么运行代码，一直在运行test，不知到去main方法  
password写成了passpport

## 前后端不分离时的开发流程

前端人员负责页面设计，后端人员除了逻辑实现还要涉及页面的渲染等，如果出现错误，维护成本较高，前后端沟通成本较高

# 前端

主要是看得见的部分，有三件套

## HTML

用于创建网页的元素内容，展示出文本，图像，链接等  
基本格式

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Title</title>
</head>
<body>

</body>
</html>
```

## CSS

对字体类型颜色，图片大小等进行美化工作

## JavaScript

给网页增加交互动态功能（如点击、鼠标移动等）

## JSON

用键值对来传递数据，便于交换数据，与xml相比，更见简洁

```
{
  "name": "John",
  "age": 30,
  "isStudent": false
}
```

# 后端

## maven

可以管理依赖

## SpringBoot工程

在创建SpringBoot的时候要选择是maven类型的，还可以添加工程所需要的依赖，如（lombok，Springweb等）  
main方法就在QuickstartApplication，主要程序在java下面，html文件放在resources中，测试文件在test中，依赖在pom.xml中，最开始没有添加的依赖也可以在这里手动添加

## HTTP

## postman

已安装，会简单使用

请求响应

简单参数传入

post/gef方法

```
public class User {
    private String name;
    private String password;
}

@PostMapping("/login")
public String login( User user) {
}
```

http://localhost:8080/login

POST http://localhost:8080/login

Send

Params Authorization Headers (9) Body Scripts Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "name": "testuser",
3   "password": "testpassword"
4 }
```

数据库

```
CREATE TABLE User (
    uid INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    password VARCHAR(50) NOT NULL
);
```

@localhost

db01

表

user

information\_schema

mysql

performance\_schema

sys

tiias

服务器对象

Mybatis和JDBC

原始jdbc操作数据库方法

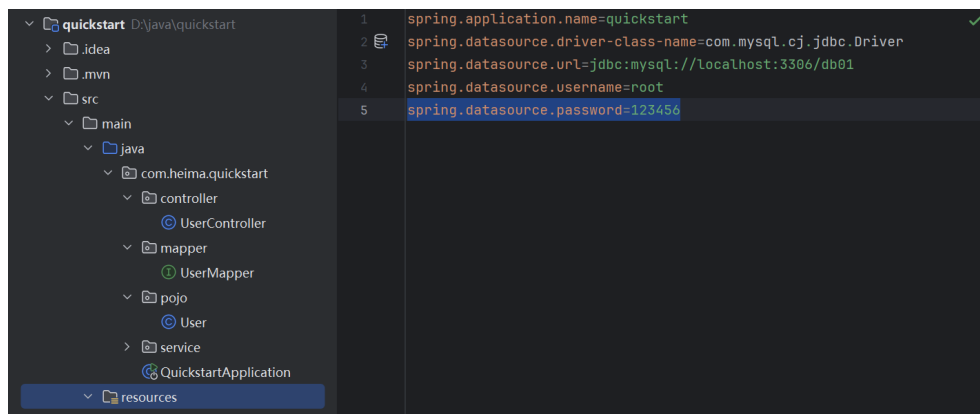
```
public void Connection() throws Exception {
    // 1. 加载配置文件
    InputStream is = ConnectionTest.class.getClassLoader().getResourceAsStream("jdbc.properties");
    Properties pros = new Properties();
    pros.load(is);

    // 2. 读取配置信息
    String user = pros.getProperty("user");
    String password = pros.getProperty("password");
    String url = pros.getProperty("url");
    String driverClass = pros.getProperty("driverClass");

    // 3. 加载驱动
    Class.forName(driverClass);

    // 4. 获取连接
    Connection conn = DriverManager.getConnection(url, user, password);
    System.out.println(conn);
}
```

相较于Mybatis较为麻烦，Mybatis则是在resource文件下的application中进行连接，可以进行多次使用



运用mybatis进行查询，添加操作

```
@Select("select * from user where name=#{name}and password=#{password}")
public User find(String name,String password);

@Insert("insert into user( name, password)\n" +
        "values (#{name},#{password})")
public void insert(User user);
```

网页登陆

{"code":200,"msg":"注册成功"}