

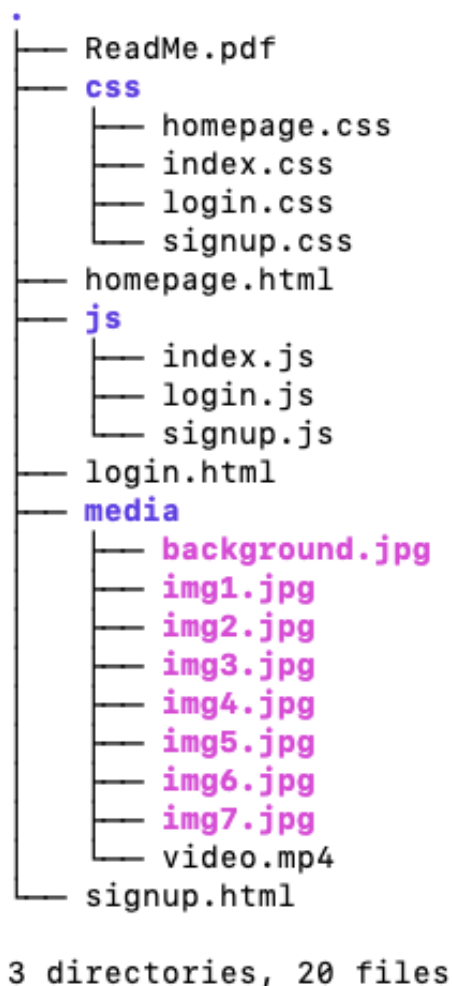
HomeWork2

实现登录与注册

HomeWork2

- 一、文件目录
- 二、数据库设计
 - 1. 数据库设计格式
 - 2. 数据库内容展示
- 三、功能介绍
 - 1. 登录注册错误提示
 - 1.1 功能实现
 - 1.2 页面展示
 - 2. 密码强度检测
 - 2.1 功能实现
 - 2.2 页面展示
 - 3. 数据库连接
 - 3.1 功能实现
 - 3.2 页面展示
 - 4. 密码加盐
 - 4.1 功能实现
 - 4.2 页面展示

一、文件目录



index.js: 实现密码加密和数据库连接

signup.js: 注册页面输入判断

login.js: 登录页面输入判断

二、数据库设计

1. 数据库设计格式

共username, password 两列, 考虑到password需经md5加密, 长度设置较长, username为主键

```
mysql> desc users;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| username | varchar(20) | NO   | PRI | NULL    |       |
| password | varchar(100) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

2. 数据库内容展示

见文件mysql.sql, 此处密码已经过加密

```
mysql> select * from users;
+-----+-----+
| username | password |
+-----+-----+
| test1    | 0512dda5f96fe87bb3a8f8615b32c1e2 |
| test2    | e29e4ac5f4264c520a746406d6d807f5 |
+-----+-----+
2 rows in set (0.00 sec)
```

三、功能介绍

1. 登录注册错误提示

1.1 功能实现

通过依次判断输入值实现，用 innerHTML 方法添加错误输出

注册页面用户名要求：提交时不能为空长度在 4-12 个字符，只能由字母或数字组成

注册页面密码要求：提交时不能为空，密码不能含有非法字符（<>{}~`!@#\$%^&*()[]?/）

注册页面确认密码要求：提交时不能为空，确认密码需要和密码一致

登录页面用户要求：提交时不能为空

注册页面密码要求：提交时不能为空

1.2 页面展示

1. 密码包含非法字符

注册

密码中含有非法字符

* 用户名: user123

* 密码:

密码强度: 弱 中 强

* 确认密码: 请再次输入密码

注册

登录账户

2. 用户名长度不符合要求

注册

用户名长度必须在 4-12 个字符

* 用户名: use

* 密码:

密码强度: 弱 中 强

* 确认密码:

注册

登录账户

2. 密码强度检测

2.1 功能实现

自定义密码强度，遍历密码确定密码所具类型，根据强度档次更改密码强度框颜色。

共四种类型数字、小写、大写、符号，数字设为0B0001，小写设为 0B0010，大写设为 0B0100，符号设为 0B1000，一位一位遍历密码进行或运算，所有密码遍历完后将结果和 1 进行与运算确定密码包括的类型，有1个类型代表低强度，2 种类型中强度，三个类型及以上为高强度，长度小于 6 个字符时不检测级别。

2.2 页面展示

1. 密码输入 test123123



The image shows a registration form titled "注册" (Register). It contains the following fields and elements:

- 用户名:** (Username) with the value "user123".
- 密码:** (Password) with masked input ".....".
- 密码强度:** (Password Strength) indicator showing three buttons: "弱" (Weak), "中" (Medium), and "强" (Strong). The "中" button is highlighted in orange, indicating the detected strength for the password "test123123".
- 确认密码:** (Confirm Password) with the placeholder text "请再次输入密码" (Please re-enter the password).
- 注册** (Register) button.
- 登录账户** (Login Account) link.

2. 密码输入 test.123123

注册

* 用户名: user123

* 密 码:

密码强度: 弱 中 强

* 确认密码:|

注册

登录账户

3. 数据库连接

3.1 功能实现

nodejs+express 实现，在控制台运行 nodejs 文件 index.js，页面进行表单提交，通过 ajax 传输数据并写入 mysql，注册时当用户名与数据库重复时输出报错，登录时和数据库内已有数据查找匹配

3.2 页面展示

1. 用户名已存在

注册

用户名已存在

* 用户名: user123

* 密码:

密码强度: 弱 中 强

* 确认密码:

注册

登录账户

2. 控制台页面

存在问题：数据提交时不知道为什么总会提交两次，第一次为空值，第二次才为正确数据，没有找到错误原因，但是因为是空值不会被写入数据库，功能实现上没有影响

```
js — node sql.js — 80x24
{ names: undefined, passwords: undefined }
{ names: 'user123', passwords: '123456' }
与mysql数据库建立连接成功
[insert error]- ER_BAD_NULL_ERROR: Column 'Username' cannot be null
与mysql数据库建立连接成功
-----插入数据-----
OkPacket {
  fieldCount: 0,
  affectedRows: 1,
  insertId: 0,
  serverStatus: 2,
  warningCount: 0,
  message: '',
  protocol41: true,
  changedRows: 0
}
-----插入数据成功-----
{ names: undefined, passwords: undefined }
{ names: 'user123', passwords: '123456' }
与mysql数据库建立连接成功
[insert error]- ER_BAD_NULL_ERROR: Column 'Username' cannot be null
与mysql数据库建立连接成功
[insert error]- ER_DUP_ENTRY: Duplicate entry 'user123' for key 'PRIMARY'
```

4. 密码加盐

4.1 功能实现

使用nodejs-crypto加密工具，注册时通过 MD5 多次加密

参考博客: <https://blog.csdn.net/shanyanwt/article/details/97101369>

4.2 页面展示

test1 的原密码为 123456，test2 的原密码为 abcdef，加密后如图

```
mysql> select * from users;
```

username	password
test1	0512dda5f96fe87bb3a8f8615b32c1e2
test2	e29e4ac5f4264c520a746406d6d807f5

```
2 rows in set (0.00 sec)
```