

《Web 应用基础》课程结业报告

一、网站概要

该网站是基于 HTML 和 CSS 做出的一个《原神》游戏主题的静态网站。其名称为《原神账号查询系统》，主要包含登录界面、查询选择界面、账号角色信息查询及其各元素角色子查询、账号武器信息查询及其各类型武器子查询、圣遗物信息查询界面。

二、网站开发过程

1. 登录界面

(1) 页面展示



(2) 实现代码（部分）

<body>部分

```
<body>

  <div class="login">

    

    <h1>原神账号查询系统</h1>

    <h2>登录/注册</h2>

    <div>

      <span>米 id: </span>

      <input type="text" id="username" placeholder="请输入 uid">

      <br>

      <span>密码: </span>

      <input type="password" id="password" placeholder="请输入密码">
```



```

        background-color: rgba(42, 172, 205, 0.2);
        border-radius: 60px; /* 圆角 */
        box-shadow: 5px 2px 35px -7px #90bcd4;
    }
    .login h1{
        margin-top: 0px;          /* 高度 */
        color: rgb(65, 141, 203);
        font-weight: 100;
        font-size: 50px;
    }
    .login_btn:hover{
        box-shadow: 2px 2px 15px 2px rgb(190,225,225);
        background-color :transparent;
        color: white;
        /* 选择动画 */
        animation: login_mation 0.5s;
    }
    /* 定义动画 */
    @keyframes login_mation{
        from{
            background-color: rgba(255, 255, 255, 0.582);
            box-shadow: 0px 0px 15px 2px rgb(190, 225, 225);
        }
        to{
            background-color: transparent;
            color: white;
            box-shadow: 2px 2px 15px 2px rgb(190, 225, 225);
        }
    }
}

```

(3) 实现过程

作为最基础的登录界面，由于框架比较简单，实现过程重点都放在如何将界面变得更加美观。即在css部分的设计上花了些心思。用<div>标签将该界面做层次化处理，通过定义每一个div块的类便于后期对其<style>进行编辑。

Css 部分主要对以下部分进行设定：

1. 按钮、字体、景颜色、尺寸及样式属性
2. 外边距、内边距及显示形式属性
3. 主轴、交叉轴的对齐方式属性

Js 部分是用 JavaScript 写的一个登录界面函数

用关键词 var 进行账号、密码两个变量的申明，其作用是影响变量的作用域，即申明局部变量。其中 document 对象可以从脚本中对 html 页面中元素进行访问。我用到的是：document.getElementById()，用于返回对拥有指定 id 的对象的引用

再通过 if 语句实现一个错误处理和界面的跳转：

```
if ( ) { }else if ( ) { }...else{ }
```

若账号或密码输入错误或为空，则会让浏览器弹出警告框提示请输入正确的用户名及密码；

若输入正确则跳转到下一界面。

(4) 在实现过程中遇到的一些问题

1. 为了丰富界面，用<div>将界面分为较多层次，因此各页面的逻辑关系需要较为清晰的界定，否则做出的界面就是比较混乱的，或是得不到自己想要的效果。

解决方式：写代码的时候就通过<div>标签的类分清楚该部分模块层次，弄清其逻辑关系，随时调试并且运行查看效果。

2. 选择界面

(1) 页面展示



(2) 实现代码（部分）

<body>部分

```
<body>

    <div class="select">

        <div class="select1">

            

            <!-- 轮播 Carousel -->
<div id="demo" class="carousel slide" data-bs-ride="carousel">
    <!-- 指示符 Indicators/dots -->
    <div class="carousel-indicators">
```

```

        <button type="button" data-bs-target="#demo" data-bs-slide-to="0" class="active"></button>
        <button type="button" data-bs-target="#demo" data-bs-slide-to="1"></button>
        <button type="button" data-bs-target="#demo" data-bs-slide-to="2"></button>
        <button type="button" data-bs-target="#demo" data-bs-slide-to="3"></button>
        <button type="button" data-bs-target="#demo" data-bs-slide-to="4"></button>
    </div>
    <!-- 轮播图片 The slideshow/carousel -->
    <div class="carousel-inner">
        <div class="carousel-item active">
            
            <!-- <div class="carousel-caption">
                <h3>第一张图片描述标题</h3>
                <p>第一张图片描述内容显示在这里!!!</p>
            </div> -->
        </div>
        <div class="carousel-item">
            
        </div>
        <div class="carousel-item">
            
        </div>
        <div class="carousel-item">
            
        </div>
        <div class="carousel-item">
            
        </div>
    </div>
    <!-- 左右切换按钮 Left and right controls/icons -->
    <button class="carousel-control-prev" type="button" data-bs-target="#demo"
data-bs-slide="prev">
        <span class="carousel-control-prev-icon"></span>
    </button>
    <button class="carousel-control-next" type="button" data-bs-target="#demo"
data-bs-slide="next">
        <span class="carousel-control-next-icon"></span>
    </button>
</div>
<div class="select2">
    <br>
    <button class="select1_btn" onclick="get1()">
</button>

```

```

        <button class="select1_btn" onclick="get2()"> </button>
        <button class="select1_btn" onclick="get3()"> </button>
        <br>
        <button class="select1_btn" onclick="log_out()"> </button>
    </div>
</div>
</body>

```

js 部分

```

<script type="application/javascript">
    function get1()
    {
        location.replace("web_blank1.html")
    }
    function get2()
    {
        location.replace("web_blank2.html")
    }
    function get3()
    {
        location.replace("web_blank3.html")
    }
    function log_out()
    {
        location.replace("web_login.html")
    }
</script>

```

部分 css 代码

```

<style>
    body{
        padding: 0;
        margin: 0;
        height: 100vh;
        display: flex;
        justify-content: center;
        background-image: linear-gradient(#fffdfd 0%, #a4d5e9 100%);
        background-size: cover;
        flex: 1;
        align-items: center; /*居中*/
        background-image: url("img/touxiangkuang.png");
    }

```

```
.select{
    margin-top: 0%;
}

.select1{
    text-align: center;
    margin: 0 auto;
    width: 500px;
    align-items: center;
}
```

(3) 实现过程

该部分主要由一个标题、轮播图和四个页面跳转的按钮组成。

标题是由官方 logo 图片导入。

轮播图主要分为三个部分：轮播图片部分、指示符部分以及左右切换按钮部分，而轮播图片部分又包括图片和文字描述，其中文字描述可省略。这部分代码一开始准备添加描述，但是鉴于我选了几张图片上有文字，就不再赘述。这三个主要部分也是用<div>分隔。

1) 指示符部分由按钮构成，有几张图就有几个按钮，通过 data-bs-slide-to="a"来跳转到 a 图片；

2) 轮播图片部分，通过 carousel-caption 类来添加描述，同样也是 h、p 标签来添加标题、内容；

3) 左右切换按钮部分也是由两个 button 构成，用标签标记显示按钮图标，用.carousel-control-prev 类定义返回上一张 .carousel-control-next 切换下一张。

页面的跳转主要通过 js 中的事件:onclick 实现,js 中构造函数通过 location.replace 引用跳转页面的链接。

(4) 在实现过程中遇到的一些问题

在写轮播图的过程中，遇到了以下问题：

1. 将图片缩小之后切换按钮和指示符按钮跑到图片外部
2. 左右切换按钮无法控制图片的切换
3. 指示符按钮无法控制图片的切换
4. 图片切换过程中整个模块卡顿不流畅

解决方式：

1、4 问题都是<div>标签的模块之间逻辑连接的问题，将图片部分和按钮部分分别封装在不同的 div 下再用主标签包起来。由于该部分用到的<div>标签分层较多，应当注意相应部分 div 包含其对应部分语句。

2、3 需要注意的是图片数量增加的同时，指示符按钮也要进行增加并修改其对应 data-bs-slide-to 属性

3. 查询界面

(1) 页面展示

| 账号角色信息查询 | | | | | | | | | |
|---|------|------|-----|-----|-------|---------|---------|----|--------|
| 照片 | 角色名 | 角色等级 | 神之眼 | 武器 | 武器名 | 命之座 | 圣遗物 | 国家 | 所属 |
|  | 神里绫华 | 90 | 冰系 | 单手剑 | 雾切之回光 | 雪鹤座-1 | 冰风迷途的勇士 | 稻妻 | 社奉行 |
|  | 魈 | 90 | 风系 | 长枪 | 和璞鸢 | 金翅鹏王座-0 | 辰砂往生录 | 璃月 | 璃月仙人 |
|  | 达达利亚 | 90 | 水系 | 弓 | 冬极白星 | 鲸天座-0 | 沉沦之心 | 至冬 | 愚人众执行官 |

| 账号武器信息查询 | | | | | |
|---|-------|------|-------------|-----------|---|
| 照片 | 武器名称 | 武器类型 | 副属性 | 武器描述 | 故事背景 |
|  | 和璞鸢 | 长枪 | 22.1%暴击率 | 昭理的鸢之枪 | 魔神翡翠化作的长枪，拥有刺杀远古海兽的沉重与锋利。 |
|  | 护摩之杖 | 长枪 | 66.2%暴击伤害 | 无羁的朱赤之蝶 | 在早已失落的古老祭祀中，使用的朱赤【举火杖】。 |
|  | 赤沙之杖 | 长枪 | 44.1%暴击率 | 鹰气尽头的热梦 | 黑曜石塑造的一对权杖中的一件。在传说中，具有引领沙之民的权威，但如今已经没有能辨别出这件武器形象的匠人了。 |
|  | 雍草之稻光 | 长枪 | 55.1%元素充能效率 | 非时之梦·常世灶食 | 用于【斩草】的随刀，对向此物之军势，也会如革草般倒下吧... |



(2) 实现代码（部分）

由于三个界面形式相同，只展示一部分代码：

body 导航栏部分：

```
<body>

  <ul>

    <li><div></div></li>

    <li><a href="web_login.html"></a></li>

    <div class="nav1"><a href="web_blank1.html"></a>

      <div class="nav2">

        <a href="web_ice.html">冰系角色</a>

        <a href="web_fire.html">火系角色</a>

        <a href="web_water.html">水系角色</a>

        <a href="web_thunder.html">雷系角色</a>

        <a href="web_winder.html">风系角色</a>

        <a href="web_rock.html">岩系角色</a>

        <a href="web_grass.html">草系角色</a>

      </div>

    </div>

    <div class="nav3"><a href="web_blank2.html"></i></a>

    <div class="nav4">

      <a href="web_sword.html">单手剑</a>

      <a href="web_epee.html">双手剑</a>

      <a href="web_bow.html">弓</a>

      <a href="web_spear.html">长枪</a>

      <a href="web_magic.html">法器</a>

    </div>

  </ul>
```

```

        </div>
    </div>
    <div class="nav5"><a href="web_blank3.html"></i></a>
    </div>
</ul>

```

表格部分

```

<table border="1px" ><!-- 表格 table//行 tr//列 td -->
    <tr>
        <td colspan="10"align="center">账号角色信息查询</td>
    </tr>
    <tr>
        <td>照片</td><td>角色名</td><td>角色等级</td><td>神之眼</td><td>武器</td><td>武器
名</td><td>命之座</td><td>圣遗物</td><td>国家</td><td>所属</td>
    </tr>
    <tr>
        <td></td><td>神里绫华</td><td>90</td><td>
冰系</td><td>单手剑</td><td>雾切之回光</td><td>雪鹤座-1</td><td>冰风迷途的勇士</td><td>稻妻
</td><td>社奉行</td>
    </tr>
    <tr>
        <td></td><td>魈</td><td>90</td><td>风系
</td><td>长枪</td><td>和璞鸢</td><td>金翅鹏王座-0</td><td>辰砂往生录</td><td>璃月</td><td>璃月仙人
</td>
    </tr>
    <tr>
        <td></td><td>达达利亚
</td><td>90</td><td>水系</td><td>弓</td><td>冬极白星</td><td>鲸天座-0</td><td>沉沦之心</td><td>至
冬</td><td>愚人众执行官</td>
    </tr>
    <tr>
        <td></td><td>阿贝多</td><td>90</td><td>岩
系</td><td>单手剑</td><td>辰砂之纺锤</td><td>白垩之子座-0</td><td>华馆梦醒形骸记</td><td>蒙德
</td><td>西风骑士团</td>
    </tr>
    <tr>
        <td></td><td>胡桃</td><td>90</td><td>火系
+</td>
    </tr>

```

css 部分

```

<style>

```

```

    body{
        padding: 0;
        margin: 0;
        height: 100vh;
        /* display: flex; */
        justify-content: center;
        background-image: linear-gradient(#fffdfd 0%, #a4d5e9 100%);
        background-size: cover;
        flex: 1;
        align-items: center;    /*居中*/
        background-attachment: fixed;    /* 背景颜色全覆盖 */
    }
table, td, th
    {border:1px solid black;}
    td
    {padding:15px;}
    table
    { width:100%;
      text-align: center;
    }
    th
    {height: 50px;}
    input {
    width: fit-content;
    height: 15px;
    }
ul{
    width: 600px;
    height:40px;
    background-color: #ffffff;
    width: 100%;
}
ul li{
    list-style: none;
    float: left;
}
a{
    color: #8be8f8;
    padding: 0px;
    margin: 0px;
    text-align: center;
    text-decoration: none;
}

```

```
.nav1,.nav3,.nav5{
    position: relative; /* 设置相对定位 */
    display: inline-block; /* 让元素具有块级元素和行内元素的特性,即将块级元素转化为内联元素 */
}
.nav2{
    display: none;
    position: absolute;
    background-color: #8be8f8;
    min-width: 148px;
} /* 将 a 标签设置为块级标签 */
.nav4{
    display: none;
    position: absolute;
    background-color: #8be8f8;
    min-width: 148px;
} /* 将 a 标签设置为块级标签 */
.nav2 a{
    color: white;
    padding:12px 35px;
    display: block;
}
.nav4 a{
    color: white;
    padding:12px 30px;
    display: block;
}
.nav2 a:hover{
    background-color: #1ea3bb;
}
.nav4 a:hover{
    background-color: #1ea3bb;
}
.nav1:hover .nav2{
    display: block;
}
.nav3:hover .nav4{
    display: block;
}
.nav1:hover,li:hover{
    background-color: #8be8f8;
}
.nav3:hover,li:hover{
    background-color: #8be8f8;
}
```

```
.nav5:hover,li:hover{  
    background-color: #8be8f8;  
}
```

(3) 实现过程

在该页面中主要由两个部分构成：导航栏部分和表格部分。

1. 本导航栏中用到的是无序列表。

用<div>定义块级标签属性，便于后续css中写出两部分标签之间的交互动画。同时可以看到，我们每一块也是通过<div>来将其隔开的。

其中可以看到给按钮赋予了超链接，用<a>来定义，在href属性中指定链接的地址。

导航栏的主要目的是能够在多个查询之间切换，或是便于在主查询中作子查询。

2. 表格

没有用网格布局主要是因为表格内容较为简单，并且整个页面部分也没有太多元素，因此选择了table来进行表格的绘制。

(4) 在实现过程中遇到的一些问题

表格部分没有做成动态过程，仅仅只是在展示一些内容。感觉还是时间比较紧张，毕竟期末课设也比较多，总之这部分内容还有比较大的提升空间，比方说连接数据库，将网页做成动态页面，或者是做到能够保存到文件中也是比较不错的。

三、项目总结

在学习这门课之前就听过一句话：“当你熟练掌握HTML、CSS、JavaScript这三门技术时，你能够解决网页设计99%的问题。”这三门技术是构建互联网网站所需的基本工具。HTML是超文本标记语言，它用于定义网页的结构和内容。CSS是层叠样式表，它负责为网页添加样式和布局。而JavaScript是用来增强网页的互动性和动态效果的脚本语言。

在实际编写网页时，通常首先需要使用HTML来定义页面的结构，包括标签、段落、标题、图象、列表等。之后，你可以使用CSS来为这些元素添加样式，改变它们的外观和布局。通过合理地使用CSS，可以使网页看起来更加吸引人、易于阅读和导航。

最后，在完成页面结构和样式设计后，你可以使用JavaScript来增加网页的交互性和动态效果。JavaScript可以处理用户输入、响应点击事件、实现动画效果、进行页面动态更新等操作。

总之，HTML、CSS、JavaScript这三门技术的学习，让我收获颇丰。如果时间充裕，我希望挑战自己，做出更棒更完美的网页，为自己喜欢的游戏、漫画做出更好的宣传！