

深圳大学实验报告

课程名称: web 开发与人机交互导论

实验项目名称: 实验 3 超链接与多媒体文件应用

学院: 计算机与软件学院

专业: 计算机科学与技术

指导教师: 李俊杰

报告人: 曹秦鲁 学号: 2023150203 班级: 国际班

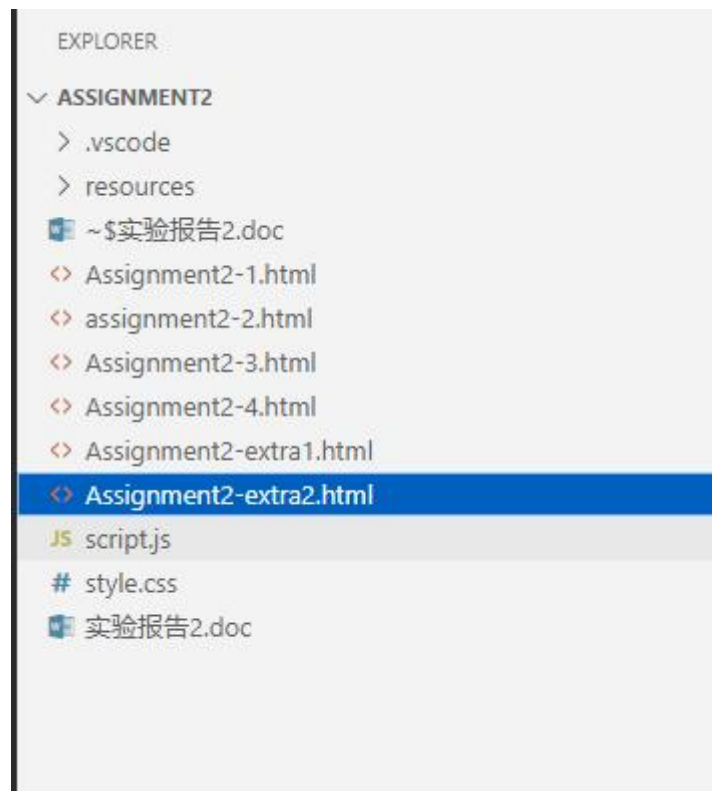
实验目的与要求：

通过本次作业，学生可以学习和掌握以下技能：

- **HTML 结构：**理解和应用 HTML 的基本结构和元素。
- **CSS 样式：**学习如何使用 CSS 定义全局样式、标题样式、页脚样式、容器样式和链接样式。
- **Flexbox 布局：**掌握 Flexbox 布局的基本用法，能够实现子元素的水平居中、换行和间距设置。
- **用户体验：**通过美化页面，提高用户体验，使页面更加美观和易于阅读。

实验过程及内容：

所有项目：



项目 1 设计简易灯箱画廊

这个项目是一个简易的灯箱画廊网页，用户可以点击缩略图来查看大图。项目由三个主要文件组成：HTML 文件、CSS 文件和 JavaScript 文件。

代码截图

```
<> assignment2-1.html • # style.css • Settings JS script.js •
<> assignment2-1.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>简易灯箱画廊</title>
7      <link rel="stylesheet" href="style.css">
8  </head>
9  <body>
10     <h1>简易灯箱画廊</h1>
11     <div class="gallery">
12         <a href="resources/image31.jpg" class="thumbnail">
13             
14         </a>
15         <a href="resources/image32.jpg" class="thumbnail">
16             
17         </a>
18         <a href="resources/image33.jpg" class="thumbnail">
19             
20         </a>
21         <!-- 添加更多缩略图 -->
22     </div>
23     <div class="lightbox" id="lightbox">
24         <img id="lightbox-img" src="" alt="Large Image">
25     </div>
26     <script src="script.js"></script>
27 </body>
28 </html>
```

代码解释：

- `<!DOCTYPE html>`: 声明文档类型为 HTML5。
- `<html lang="en">`: 定义文档的根元素，并设置语言为英语。
- `<head>`: 包含文档的元数据，如字符集、视口设置和标题。
- `<title>`: 设置网页的标题。
- `<link rel="stylesheet" href="style.css">`: 引入外部 CSS 文件 `style.css`。
- `<body>`: 包含网页的主要内容。
- `<h1>`: 定义网页的主标题。
- `<div class="gallery">`: 包含所有缩略图的容器。
- ``: 定义一个链接，点击后显示大图。
- ``: 显示缩略图。
- `<div class="lightbox" id="lightbox">`: 定义一个浮动框架，用于显示大图。
- ``: 显示大图的容器。
- `<script src="script.js"></script>`: 引入外部 JavaScript 文件 `script.js`。

CSS 文件：

```
1.  /* style.css */
2.  body {
3.      font-family: Arial, sans-serif;
4.      text-align: center; /* 使标题居中 */
5.  }
6.
7.  h1 {
8.      margin-top: 20px;
9.      font-size: 2em;
10.     color: #333;
11. }
12.
13. .gallery {
14.     display: flex;
15.     flex-wrap: wrap;
16.     gap: 10px;
17.     justify-content: center; /* 使画廊居中 */
18. }
19.
20. .thumbnail {
21.     position: relative;
22.     display: inline-block;
23. }
24.
25. .thumbnail img {
26.     width: 100px;
27.     height: 100px;
28.     object-fit: cover;
29.     cursor: pointer;
30. }
31.
32. .caption {
33.     position: absolute;
34.     bottom: 0;
35.     left: 0;
36.     width: 100%;
37.     background-color: rgba(0, 0, 0, 0.5);
38.     color: white;
39.     text-align: center;
40.     padding: 5px 0;
41.     font-size: 12px;
```

```

42. }
43.
44. .lightbox {
45.     position: fixed;
46.     bottom: 0;
47.     left: 0;
48.     width: 100%;
49.     background-color: rgba(0, 0, 0, 0.8);
50.     text-align: center;
51.     padding: 10px;
52.     display: none;
53. }
54.
55. .lightbox img {
56.     max-width: 100%;
57.     height: auto;
58. }

```

代码解释：

1. **.caption**：设置图片标题的样式，包括位置、背景颜色、文本颜色和字体大小。
2. **.lightbox**：设置浮动框架的样式，包括位置、大小、背景颜色和布局方式。默认隐藏。
3. **.lightbox img**：设置浮动框架内图片的样式，确保图片不会超出展示区域，并放大图片。

java script 代码：

```

1. // script.js
2. document.addEventListener('DOMContentLoaded', function() {
3.     const thumbnails = document.querySelectorAll('.thumbnail');
4.     const lightbox = document.getElementById('lightbox');
5.     const lightboxImg = document.getElementById('lightbox-img');
6.
7.     thumbnails.forEach(thumbnail => {
8.         thumbnail.addEventListener('click', function(event) {
9.             event.preventDefault();
10.            const largeImageSrc = this.href;
11.            lightboxImg.src = largeImageSrc;
12.            lightbox.style.display = 'flex'; // 使用 flex 布局显示
13.        });
14.    });

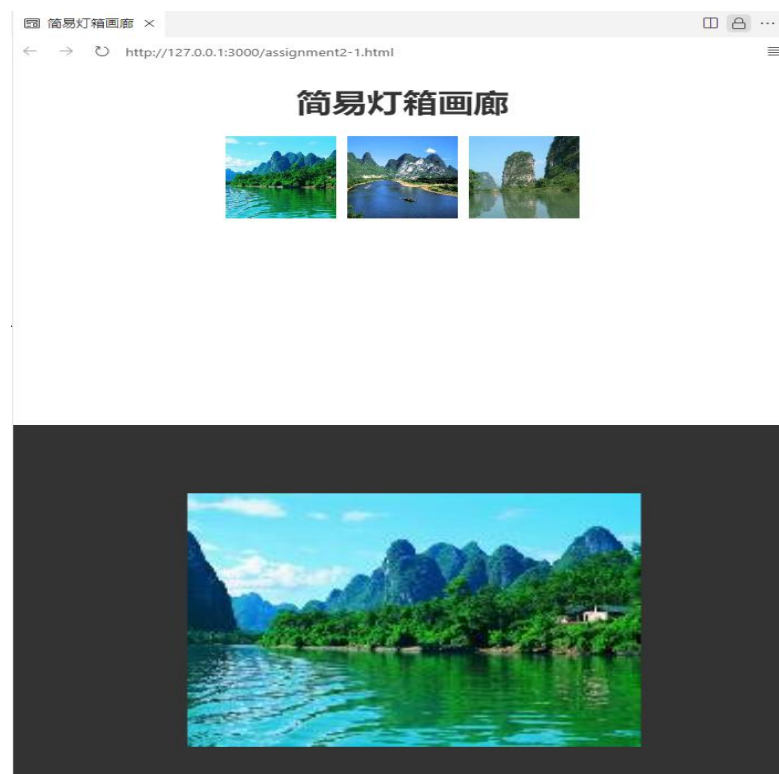
```

```
15.  
16.     lightbox.addEventListener('click', function() {  
17.         lightbox.style.display = 'none';  
18.     });  
19. });
```

代码解释：

1. `document.addEventListener('DOMContentLoaded', function() {...})`: 确保 DOM 完全加载后执行代码。
2. `const thumbnails = document.querySelectorAll('.thumbnail');`: 选择所有缩略图元素。
3. `const lightbox = document.getElementById('lightbox');`: 选择浮动框架元素。
4. `const lightboxImg = document.getElementById('lightbox-img');`: 选择浮动框架内的图片元素。
5. `thumbnails.forEach(thumbnail => {...})`: 为每个缩略图添加点击事件监听器。
6. `thumbnail.addEventListener('click', function(event) {...})`: 处理缩略图的点击事件，更新浮动框架内的图片，并显示浮动框架。
7. `lightbox.addEventListener('click', function() {...})`: 处理浮动框架的点击事件，隐藏浮动框架。

结果展示：



项目 2 设计支持音频、视频播放的网页

代码截图

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>音频和视频播放示例</title>
7      <link rel="stylesheet" href="style.css">
8  </head>
9  <body>
10     <h1>明月几时有</h1>
11
12     <h2>歌词</h2>
13     <div class="lyrics">
14         <p>明月几时有? 把酒问青天。</p>
15         <p>不知天上宫阙, 今夕是何年。</p>
16         <p>我欲乘风归去, 又恐琼楼玉宇, 高处不胜寒, 起舞弄清影, 何似在人间。</p>
17         <p>转朱阁, 抵绮户, 照无眠。</p>
18         <p>不应有恨, 何事偏向别时圆。</p>
19         <p>人有悲欢离合, 月有阴晴圆缺, 此事古难全。</p>
20         <p>但愿人长久, 千里共婵娟。</p>
21     </div>
22
23     <h2>音频播放</h2>
24     <audio controls>
25         <source src="resources/embed/蔡琴明月几时有.mp3" type="audio/mpeg">
26         您的浏览器不支持音频元素。
27     </audio>
28
29     <h2>视频播放</h2>
30     <video controls width="600">
31         <source src="resources/puppy.mp4" type="video/mp4">
32         您的浏览器不支持视频元素。
33     </video>
34
35     <script src="script.js"></script>
36 </body>
37 </html>
```

代码解释:

Lyrics css 代码

```
1.  /* 歌词样式 */
2.  .lyrics {
3.      font-family: 'KaiTi', 'STKaiti', serif; /* 设置字体为楷体 */
4.      font-size: 1.2em; /* 设置字体大小 */
5.      line-height: 1.8; /* 设置行高 */
6.      color: #555; /* 设置文本颜色 */
7.      margin: 20px auto; /* 设置上下外边距为20px, 左右自动 */
8.      max-width: 800px; /* 设置最大宽度 */
```

```
9.     padding: 20px; /* 设置内边距 */
10.    background-color: #fff; /* 设置背景颜色 */
11.    border-radius: 10px; /* 设置圆角 */
12.    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1); /* 添加阴影 */
13.    text-align: center; /* 设置文本居中 */
14. }
15.
16. .lyrics p {
17.     margin: 10px 0; /* 设置段落的上下外边距 */
18. }
```

音视频部分 html 代码

```
1.  <h2>音频播放</h2>
2.      <audio controls>
3.          <source src="resources/embed/蔡琴明月几时
   有.mp3" type="audio/mpeg">
4.          您的浏览器不支持音频元素。
5.      </audio>
6.
7.  <h2>视频播放</h2>
8.      <video controls width="600">
9.          <source src="resources/puppy.mp4" type="video/mp4">
10.         您的浏览器不支持视频元素。
11.      </video>
```

结果展示:

明月几时有

歌词

明月几时有？把酒问青天。
不知天上宫阙，今夕是何年。
我欲乘风归去，又恐琼楼玉宇，高处不胜寒，起舞弄清影，何似在人间。
转朱阁，抵绮户，照无眠。
不应有恨，何事偏向别时圆。
人有悲欢离合，月有阴晴圆缺，此事古难全。
但愿人长久，千里共婵娟。

音频播放

▶ 0:00 / 3:26 ———— 🔊 ⋮

视频播放



项目 3 设计简易网站导航

代码解释： 使用段落标记和超链接实现导航:

1. `<h2>`使用段落标记和超链接实现导航`</h2>`
2. `<header>`
3. `<nav>`
4. `<p>`
5. `百度 |`
6. `新浪 |`
7. `腾讯 |`
8. `搜狐 |`
9. `网易 |`
10. `谷歌`
11. `</p>`

```
12.     </nav>
```

```
13. </header>
```

解释：

1. `<h2>`使用段落标记和超链接实现导航`</h2>`：标题，描述导航的实现方式。
2. `<header>`：定义文档的头部区域。
3. `<nav>`：定义导航链接的区域。
4. `<p>`：使用段落标记包含多个超链接 `<a>`，每个超链接指向不同的外部网站。
5. `|` 用于分隔各个链接。

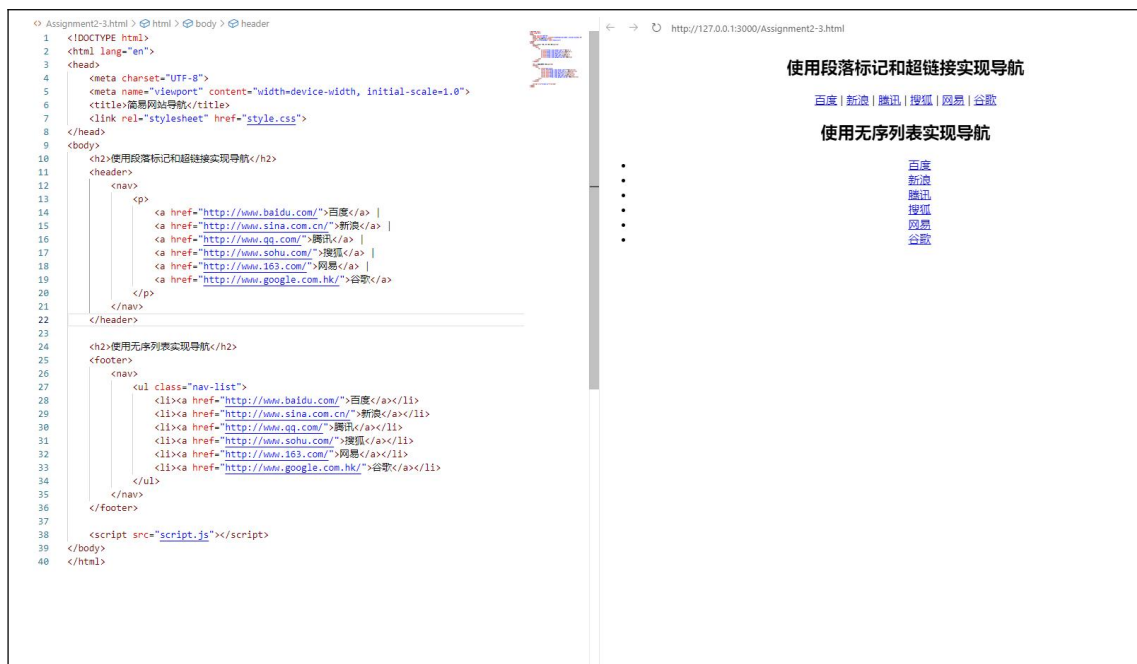
使用无序列表实现导航：

```
1. <h2>使用无序列表实现导航</h2>
2. <footer>
3.     <nav>
4.         <ul class="nav-list">
5.             <li><a href="http://www.baidu.com/">百度</a></li>
6.             <li><a href="http://www.sina.com.cn/">新浪</a></li>
7.             <li><a href="http://www.qq.com/">腾讯</a></li>
8.             <li><a href="http://www.sohu.com/">搜狐</a></li>
9.             <li><a href="http://www.163.com/">网易</a></li>
10.            <li><a href="http://www.google.com.hk/">谷歌</a></li>
11.        </ul>
12.    </nav>
13. </footer>
```

解释：

1. `<h2>`使用无序列表实现导航`</h2>`：标题，描述导航的实现方式。
2. `<footer>`：定义文档的底部区域。
3. `<nav>`：定义导航链接的区域。
4. `<ul class="nav-list">`：使用无序列表 `` 包含多个列表项 ``，每个列表项包含一个超链接 `<a>`，指向不同的外部网站。

代码截图和结果展示：



项目 4 设计新生课程简介

代码截图和结果展示



代码：

1. `</p>`
2. `返回`

3. `</section>`

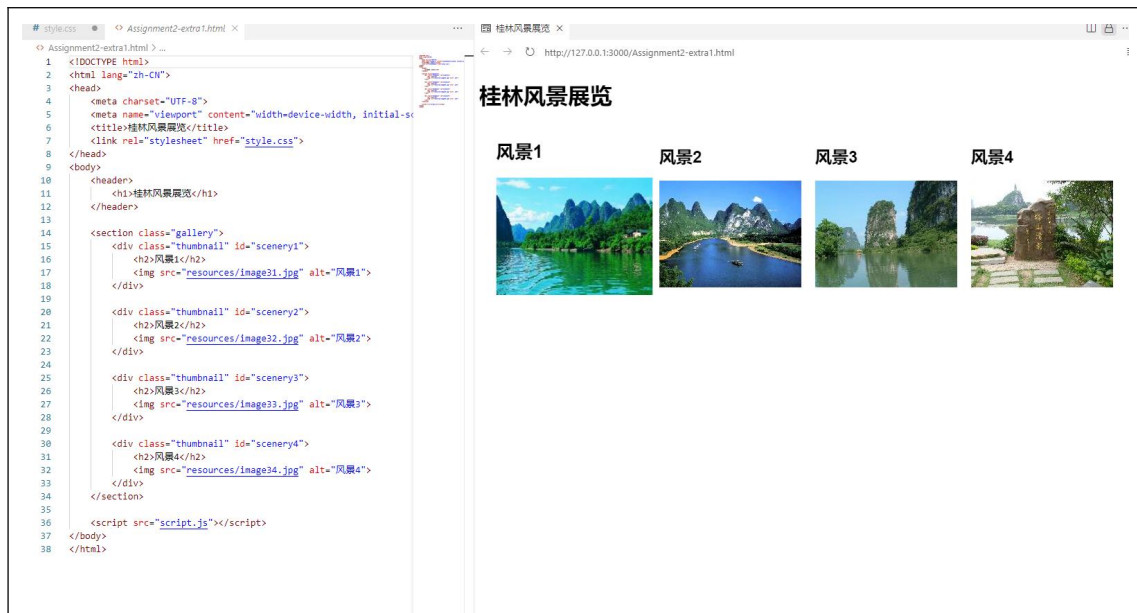
1. `</p>`: 关闭段落标签。
2. `返回`: 创建一个超链接，链接到页面顶部（通过 `href="#top"`），显示文本为“返回”。
3. `</section>`: 关闭当前的 `section` 标签。

```
1. <footer>
2.     <nav>
3.         <ul class="nav-list">
4.             <li><a href="#english-courses">英语课程</a></li>
5.             <li><a href="#math-course">高等数学</a></li>
6.             <li><a href="#physics-course">大学物理</a></li>
7.         </ul>
8.     </nav>
9. </footer>
```

1. `<footer>`: 定义文档或节的页脚部分。
2. `<nav>`: 定义导航链接的区域。
3. `<ul class="nav-list">`: 创建一个无序列表，并应用 `nav-list` 类的样式。
4. `英语课程`: 创建一个列表项，包含一个链接到 `#english-courses` 的超链接，显示文本为“英语课程”。
5. `高等数学`: 创建一个列表项，包含一个链接到 `#math-course` 的超链接，显示文本为“高等数学”。
6. `大学物理`: 创建一个列表项，包含一个链接到 `#physics-course` 的超链接，显示文本为“大学物理”。
7. `</nav>`: 关闭 `nav` 标签。
8. `</footer>`: 关闭 `footer` 标签。

课外拓展训练

1.
代码截图和结果展示:



html 文件解释

1. `<!DOCTYPE html>`
2. `<html lang="zh-CN">`
3. `<head>`
4. `<meta charset="UTF-8">` `<!-- 设置文档的字符编码为 UTF-8 -->`
5. `<meta name="viewport" content="width=device-width, initial-scale=1.0">` `<!-- 设置视口的宽度等于设备宽度，并初始缩放比例为 1 -->`
6. `<title>桂林风景展览</title>` `<!-- 设置文档的标题 -->`
7. `<link rel="stylesheet" href="style.css">` `<!-- 链接外部 CSS 文件 -->`
8. `</head>`
9. `<body>`
10. `<header>`
11. `<h1>桂林风景展览</h1>` `<!-- 页面主标题 -->`
12. `</header>`
- 13.
14. `<section class="gallery">` `<!-- 包含所有缩略图的容器 -->`
15. `<div class="thumbnail" id="scenery1">` `<!-- 缩略图容器，ID 为 scenery1 -->`
16. `<h2>风景 1</h2>` `<!-- 缩略图标题 -->`
17. `` `<!-- 缩略图图片 -->`
18. `</div>`
- 19.
20. `<div class="thumbnail" id="scenery2">` `<!-- 缩略图容器，ID 为 scenery2 -->`
21. `<h2>风景 2</h2>` `<!-- 缩略图标题 -->`
22. `` `<!-- 缩略图图片 -->`

```

23.         </div>
24.
25.         <div class="thumbnail" id="scenery3"> <!-- 缩略图容器,
           ID 为 scenery3 -->
26.             <h2>风景 3</h2> <!-- 缩略图标题 -->
27.              <!-- 缩
           略图图片 -->
28.         </div>
29.
30.         <div class="thumbnail" id="scenery4"> <!-- 缩略图容器,
           ID 为 scenery4 -->
31.             <h2>风景 4</h2> <!-- 缩略图标题 -->
32.              <!-- 缩
           略图图片 -->
33.         </div>
34.     </section>
35.
36.     <script src="script.js"></script> <!-- 链接外部 JavaScript 文
           件 -->
37. </body>
38. </html>

```

css 部分注释

```

1.  /* 缩略图样式 */
2.  .thumbnail {
3.      cursor: pointer; /* 鼠标悬停时显示为指针 */
4.      transition: transform 0.2s; /* 添加缩放过渡效果 */
5.      margin: 10px; /* 设置缩略图的外边距 */
6.  }
7.
8.  .thumbnail:hover {
9.      transform: scale(1.1); /* 鼠标悬停时缩放 10% */
10. }
11.
12. /* 画廊样式 */
13. .gallery {
14.     display: flex; /* 使用 Flexbox 布局 */
15.     justify-content: center; /* 子元素水平居中 */
16.     flex-wrap: wrap; /* 子元素在必要时换行 */
17. }
18.
19. .thumbnail img {
20.     max-width: 100%; /* 图片最大宽度为 100% */
21.     height: auto; /* 高度自动调整 */
22. }

```

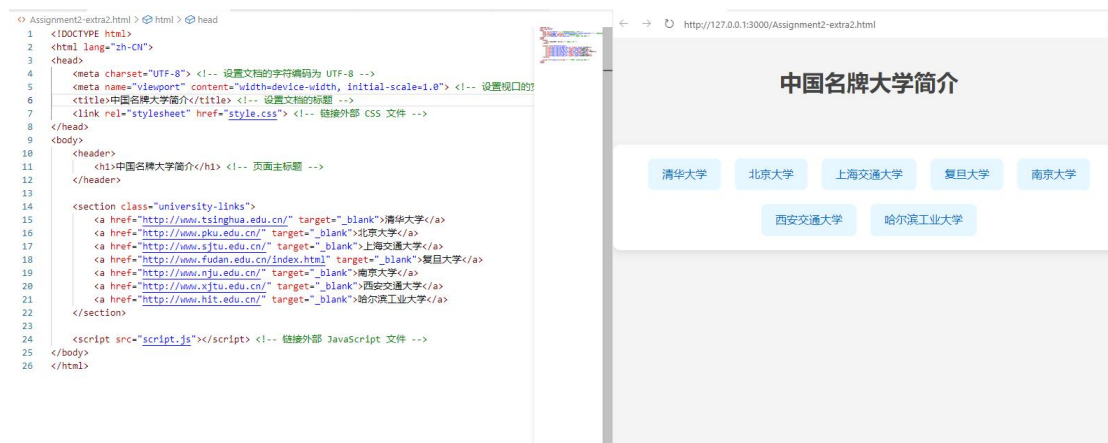
```

23.
24. /* 画廊部分样式 */
25. section.gallery {
26.     padding: 20px; /* 设置画廊部分的内边距 */
27.     background-color: #fff; /* 设置背景颜色 */
28.     border-radius: 8px; /* 设置圆角 */
29.     box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); /* 设置阴影 */
30. }

```

2.

代码截图和结果展示



代码注解：

```

1.  css 文件解释
2.  /* 大学链接容器样式 */
3.  .university-links {
4.      display: flex; /* 使用 Flexbox 布局 */
5.      justify-content: center; /* 子元素水平居中 */
6.      flex-wrap: wrap; /* 子元素在必要时换行 */
7.      gap: 20px; /* 设置子元素之间的间距 */
8.      padding: 20px; /* 设置容器的内边距 */
9.      background-color: #fff; /* 设置背景颜色 */
10.     border-radius: 12px; /* 设置圆角 */
11.     box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1); /* 添加阴影 */
12.     max-width: 800px; /* 设置最大宽度 */
13.     margin: 20px auto; /* 设置上下外边距为 20px，左右自动 */
14.     text-align: center; /* 设置文本居中 */
15. }
16.

```

```
17. /* 超链接样式 */
18. .university-links a {
19.     color: #0066cc; /* 设置链接颜色 */
20.     text-decoration: none; /* 去除链接下划线 */
21.     padding: 10px 20px; /* 设置内边距 */
22.     background-color: #e6f7ff; /* 设置背景颜色 */
23.     border-radius: 8px; /* 设置圆角 */
24.     transition: background-color 0.3s ease; /* 添加背景颜色过渡效果 */
25. }
26.
27. .university-links a:hover {
28.     background-color: #cceeff; /* 鼠标悬停时改变背景颜色 */
29.     text-decoration: underline; /* 鼠标悬停时显示下划线 */
30. }
```

心得体会：

Flexbox (Flexible Box Layout) 是一种用于布局的 CSS 模块，旨在提供一种更高效的方式来排列、对齐和分配容器中的空间，尤其是在容器的尺寸是动态变化的或未知的情况下。Flexbox 布局模型的主要目标是使容器内的子元素能够在不同的屏幕尺寸和设备上更灵活地排列。

Flexbox 的基本概念

Flex 容器 (Flex Container)：

任何应用了 `display: flex` 或 `display: inline-flex` 的元素都成为一个 Flex 容器。

Flex 容器的直接子元素称为 Flex 项目 (Flex Items)。

Flex 项目 (Flex Items)：

Flex 容器内的每个直接子元素都是一个 Flex 项目。

Flex 项目可以在主轴 (Main Axis) 和交叉轴 (Cross Axis) 上进行排列和对齐。

1. HTML 结构的掌握

在本次实验中，我深入了解了 HTML 的基本结构和标签使用。通过实际操作，我熟悉了 HTML 文档的基本结构，包括 `<!DOCTYPE html>` 声明、`<html>`、`<head>` 和 `<body>` 标签。这些标签是构建网页的基础，确保了网页的正确显示和运行。此外，我还掌握了常用的 HTML 标签，如 `<h1>`、`<p>`、``、`` 等，这些标签帮助我在网页中组织和展示内容。

2. CSS 样式的应用

CSS 用于控制网页的外观和布局。在实验中，我学习了如何使用 CSS 为网页元素设置样式。通过实践，我掌握了 CSS 的基本语法和选择器，包括类选择器和元素选择器。在实际操作中，我为网页的不同部分设置了样式，例如：

- `.header` 类用于设置标题部分的背景颜色、文本颜色、对齐方式和内边距。
- `.content` 类用于设置内容部分的内边距。
- `.footer` 类用于设置页脚部分的背景颜色、文本颜色、对齐方式、内边距，并将其固定在页面底部。

3. CSS 布局的理解

在实验中，我还学习了 CSS 布局模型，包括盒模型和固定定位。在实际操作中，我使用了固定定位将页脚部分固定在页面底部。通过设置 `position: fixed;`、`bottom: 0;` 和 `width: 100%;`，我确保了页脚部分始终显示在页面的底部，并且宽度占满整个页面。

4. 实践中的挑战与解决

在实际操作中，我遇到了一些挑战。例如，在使用 CSS 进行布局时，有时会出现元素错位或样式冲突的问题。通过查阅资料和反复调试，我逐渐掌握了 CSS 布局的技巧，能够更好地控制网页的布局和样式。此外，在编写 HTML 和 CSS 代码时，我也注意到了代码的可读性和维护性，尽量使用语义化的标签和清晰的命名。

指导教师批阅意见：

成绩评定： 分

指导教师签字：

年 月 日

备注：