



《GIS 应用开发》

课内实习一

班级序号：_____08_____

学 号：_____20221003811_____

姓 名：_____张怡静_____

指导教师：_____樊文有_____

中国地质大学（武汉）地理与信息工程学院

2025 年 3 月

目录

一、配置前端开发的编程环境	3
1. 配置前端开发的工具	3
2. 常用快捷键（Windows）	3
二、调试前端网页	3
1. 使用 VSCode 的内置终端和浏览器	3
2. 使用 VSCode 调试器结合 Chrome 或 Edge 浏览器	4
3. 使用浏览器的开发者工具	4
4. 调试 CSS 和 HTML	4
三、设计静态网页	5
1. 网页设计	5
2. 网页展示	10

一、配置前端开发的编程环境

1. 配置前端开发的工具

- (1) 本次实习使用的是 VSCode，首先打开 VSCode 官方网站：
<https://code.visualstudio.com/>
- (2) 在网站首页，可以找到下载按钮，根据个人的操作系统（Windows、macOS 或 Linux）选择合适的版本进行下载。
- (3) 下载好后打开安装包根据提示进行安装操作，选择下载位置和不同的下载要求。

2. 常用快捷键（Windows）

新建文件：Ctrl + N
打开文件：Ctrl + O
保存文件：Ctrl + S
另存为：Ctrl + Shift + S
关闭文件：Ctrl + W
撤销：Ctrl + Z
重做：Ctrl + Shift + Z
剪切：Ctrl + X
复制：Ctrl + C
粘贴：Ctrl + V
全选：Ctrl + A
查找：Ctrl + F
替换：Ctrl + H
格式化代码：Shift + Alt + F

二、调试前端网页

1. 使用 VSCode 的内置终端和浏览器

- (1) 安装 Live Server 插件。打开 VSCode，点击左侧扩展图标，搜索 Live Server 插件，点击安装。该插件允许启动一个本地开发服务器，实时预览 HTML 文件的更改。
- (2) 使用 Live Server 运行网页。在项目文件夹中，右键点击你要预览的 HTML 文件，选择 "Open with Live Server"。Live Server 会在浏览器中打开该网页，并且会自动监控你所做的更改，实时更新浏览器页面。

(3)调试网页。在浏览器中打开网页时，可以使用浏览器的开发者工具（F12）进行调试，查看控制台、元素、网络请求等信息。

2. 使用 VSCode 调试器结合 Chrome 或 Edge 浏览器

(1)安装调试扩展

A.Debugger for Chrome。打开 VSCode，点击左侧扩展图标，搜索 Debugger for Chrome 插件并安装。

(2)配置调试环境

A.安装插件后，点击左侧调试图标（或按 Ctrl + Shift + D），然后点击 "create a launch.json file" 来创建调试配置。

B.选择 Chrome，这会在项目的 .vscode 文件夹下生成一个 launch.json 文件。

(3)启动调试

A.确保 Live Server 正在运行（或者你可以使用其他本地服务器），然后点击调试视图中的绿色播放按钮，选择 Launch Chrome。

B.VSCode 会启动 Chrome，并自动打开你的网页，进入调试模式。

(4)设置断点和调试

A.在 VSCode 中打开 JavaScript 文件，在想要停止的地方设置断点（点击行号旁边的灰色区域）。

B.当浏览器执行到这个位置时，程序会暂停，可以查看变量的值、调用栈等信息。

3. 使用浏览器的开发者工具

(1)Chrome 和 Edge 都有强大的开发者工具，按 F12 或右键点击页面元素并选择 "Inspect" 打开。

(2)可以使用 Console 面板查看 console.log 输出，Elements 面板检查 HTML 结构，Network 面板查看请求和响应，Sources 面板调试 JavaScript。

4. 调试 CSS 和 HTML

(1)使用开发者工具中的 Elements 面板，可以直接修改 HTML 和 CSS，并且实时看到页面效果。

(2)在 CSS 编辑区域，改变样式后，浏览器会即时反映出效果。

三、设计静态网页

1.网页设计

html 文件编写

(1)HTML5 语法

此网页使用了 HTML5 作为基础标记语言。HTML5 是最新版本的 HTML，提供了更多语义化的元素（如 <header>, <section>, <footer>），使得网页的结构更加清晰且具有可访问性。

主要特点：

DOCTYPE 声明：<!DOCTYPE html> 声明了该文件为 HTML5 格式。

语言设置：使用 <html lang="en"> 设置网页语言为英文，利于搜索引擎优化（SEO）和更好的用户体验。

字符集：<meta charset="UTF-8"> 设置字符编码为 UTF-8，确保网页内容能够正确显示国际字符。

(2)网页结构

页眉 (<header>)：用于网页的顶部内容，通常包含网站名称、logo 或导航链接。在本页面中，页眉包括了一个主要标题 (<h1>) 和描述性的副标题 (<p>)，介绍了网站内容。

主要内容区 (<section>)：使用 <section> 标签将不同的内容部分组织起来，如个人介绍、个人信息、照片与音乐、兴趣爱好等。每个部分都通过语义化标签来标识，帮助提高内容的可读性和可访问性。

表格 (<table>)：用于展示基本技能的表格，表格内包含了表头 (<thead>) 和表格内容 (<tbody>)。每一列对应一个技能、水平和经验，帮助清晰展示信息。

页脚 (<footer>)：包含版权声明 (<p>) 和网站的版权所有信息，通常出现在页面的底部。

(3)多媒体元素

图片： 标签用于嵌入图片，src 属性指向图片的文件路径，alt 属性提供图片的替代文本，确保无论图片是否加载成功，用户都能理解图片内容。

音频播放器：<audio> 标签用于嵌入音频文件，controls 属性提供浏览器自带的音频控制功能（如播放、暂停、音量控制等）。音频文件的源通过 <source> 标签指定，确保音频能够被兼容的浏览器播放。

css 文件编写

(1)基本样式

```
body {  
    font-family: Arial, sans-serif;  
    margin: 0;  
    padding: 0;  
    background-color: #f0f8ff; /* 浅蓝色背景 */  
    color: #333;  
}
```

字体设置：Arial 字体，确保文本清晰易读。

背景颜色：使用浅蓝色背景（#f0f8ff），给网页增添温暖、轻松的氛围。

文本颜色：设置为深灰色（#333），与背景色形成良好的对比，增加可读性。

(2) 页眉设计

```
header {  
    background: linear-gradient(135deg, #0eddeb, #008cba); /* 渐变背景 */  
    color: white;  
    padding: 20px;  
    text-align: center;  
    border-radius: 15px; /* 圆角效果 */  
    margin: 20px;  
}
```

渐变背景：页眉使用了一个线性渐变色背景（从浅蓝到深蓝），使其看起来更加现代且引人注目。

白色文本：为了在深色背景上提供更高的对比，文字采用白色。

圆角：页眉边角做了圆角处理，增加页面的柔和感。

内外边距：padding 和 margin 增加了空间，使页面内容不会贴边。

(3) 个人信息区设计

```
section {  
    margin: 20px;  
    padding: 20px;  
    background: #fff;  
    border-radius: 10px; /* 圆角 */  
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1); /* 阴影效果 */  
}
```

背景：每个部分（如个人信息区）的背景为白色，与整体色调保持一致。

圆角和阴影：使用圆角和轻微阴影来提高视觉层次感，使页面看起来更有深度。

(4)信息盒子布局

```
.info-container {  
    display: flex;
```

```

        justify-content: space-around;
        margin: 20px 0;
    }

    .info-box {
        background-color: #f0f8ff; /* 浅蓝色背景 */
        border: 2px solid #87ceeb; /* 深蓝色边框 */
        border-radius: 8px;
        padding: 20px;
        width: 30%; /* 定义盒子的宽度 */
        text-align: center;
        box-shadow: 0 4px 8px rgba(0,0,0,0.2);
        transition: transform 0.3s ease; /* 添加平滑效果 */
    }

    .info-box:hover {
        transform: translateY(-5px); /* 鼠标悬停时的效果 */
    }

```

Flexbox 布局：使用 flexbox 来布局信息盒子，justify-content: space-around 使盒子均匀分布，增加美观。

信息盒子样式：每个盒子使用浅蓝色背景和深蓝色边框，设计简洁且具有现代感。

平滑悬停效果：通过 transition 和 transform，当鼠标悬停时，盒子会向上轻微浮动，提供交互感。

(5)响应式设计

```

@media (max-width: 768px) {
    .info-container {
        flex-direction: column;
        align-items: center;
    }
    .info-box {
        width: 80%; /* 在小屏幕上占据更大宽度 */
        margin-bottom: 20px;
    }
}

```

小屏幕适配：当浏览器宽度小于 768px 时，flexbox 的布局会变为纵向排列，确保内容在手机等小屏设备上能良好显示。

宽度调整：盒子在小屏幕上宽度变为 80%，确保元素不会显得过于紧凑。

(6)照片展示样式

```

.photo-container {
    text-align: center;
    margin: 20px 0;
}

```

```
.photo {
  width: 30%;
  border-radius: 10px;
  border: 5px solid #0eddeb;
}
```

图片居中：text-align: center 确保照片居中显示。

边框和圆角：为图片添加浅蓝色边框和圆角，增加视觉效果。

(7)音乐播放器样式

```
.music-container {
  text-align: center;
  margin: 20px 0;
}audio {
  border-radius: 10px;
  border: 2px solid #008cba;
  padding: 5px;
}
```

音频播放器样式：为 <audio> 标签添加了边框、圆角和内边距，使播放器看起来更加精致。

(8)兴趣爱好卡片展示

```
.card-container {
  display: flex;
  justify-content: space-between;
  gap: 20px;
}
.card {
  background-color: #f9f9f9;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
  width: 30%;
  transition: transform 0.3s ease, box-shadow 0.3s ease;
}
.card:hover {
  transform: scale(1.05); /* 鼠标悬停时放大效果 */
  box-shadow: 0 6px 10px rgba(0, 0, 0, 0.2);
}
.card h3 {
  color: #0eddeb;
}
```

卡片布局：使用 flexbox 将卡片均匀排列，确保内容不会挤在一起。

悬停效果：卡片在鼠标悬停时会放大，增加互动感，且阴影加深，给用户反馈。

(9)表格展示样式

```
table {  
    width: 100%;  
    margin: 20px 0;  
    border-collapse: collapse;  
    background-color: #fff;  
    border-radius: 10px;  
    margin-bottom: 40px;  
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);  
}  
th, td {  
    padding: 12px;  
    text-align: center;  
    border: 1px solid #ddd;  
}  
caption {  
    font-weight: bold;  
    color: #0eddeb;  
}
```

表格样式：表格宽度为 100%，并且没有多余的空白。

单元格样式：th 和 td 标签有适当的内边距，确保内容不至于太拥挤。

标题样式：表格的标题加粗并设置为蓝色，突出表格的内容。

(10)页脚样式

```
footer {  
    text-align: center;  
    padding: 15px;  
    background-color: #008cba;  
    color: white;  
    border-radius: 10px;  
    margin-top: 30px;  
}
```

页脚样式：页脚有深蓝色背景和白色字体，居中显示，且增加了内外边距与圆角，使其与其他部分的设计风格保持一致。

2.网页展示



