

Windows Server 2012 IIS+SheetJS 实现Excel HTML5在线预览（主/子页面拓展版）详细执行方案

一、方案概述

本方案基于 Windows Server 2012 系统，通过部署 IIS（互联网信息服务）搭建本地网页服务器，结合 SheetJS（前端JavaScript库）实现 Excel 文件（.xlsx、.xls 格式）的 HTML5 在线预览。为适配后续拓展需求，优化页面结构为「1个主页面+2个子页面」，主页面作为入口提供选项跳转，两个子页面对应独立文件夹，分别存储并预览对应类型的Excel文件，结构清晰、便于后续新增页面/文件夹拓展。

核心逻辑：无需服务器端复杂转换，前端通过 SheetJS 直接解析 Excel 文件，将其渲染为 HTML5 表格；主页面（index.html）提供两个选项，点击分别跳转至对应子页面，子页面自动加载对应文件夹内的Excel文件，支持点击预览、右键下载，无需手动上传文件。

核心优势：操作简单、无复杂环境依赖、部署速度快、支持本地离线预览（仅需服务器本地资源）、兼容 Excel 常见格式，页面结构模块化，便于后续新增子页面和文件文件夹，适配长期拓展需求。

前置条件：1. Windows Server 2012 系统（标准版/数据中心版均可，无需升级至 R2）；2. 服务器可正常操作（管理员权限）；3. 浏览器（用于测试，推荐 Chrome、Edge 最新版）；4. 需预览的 Excel 文件（.xlsx 优先，.xls 兼容，数量不少于2个，用于测试两个子文件夹）。

二、详细执行步骤（共6步，全程可复制操作，重点适配主/子页面+独立文件夹结构）

步骤1：登录服务器并获取管理员权限

1. 登录 Windows Server 2012 服务器，确保使用「管理员账号」登录（右下角显示“管理员”标识，或右键“计算机”→“管理”，确认登录账号拥有管理员权限）。

2. 关闭服务器自带的“IE增强安全配置”（避免后续下载文件、访问网页受限）：

- 打开「服务器管理器」（桌面图标或通过“开始”→“管理工具”→“服务器管理器”）。
- 点击左侧「本地服务器」，右侧找到「IE增强安全配置」，点击“启用”（默认状态）。
- 在弹出的窗口中，将「管理员」和「用户」均设置为「关闭」，点击「确定」，关闭服务器管理器。

步骤2：安装 IIS（互联网信息服务）8.0

IIS 是 Windows Server 自带的网页服务器，用于托管我们的 HTML5 主/子页面和 Excel 文件，全程通过系统自带工具安装，无需额外下载。

1. 重新打开「服务器管理器」，点击顶部「添加角色和功能」。
2. 弹出“添加角色和功能向导”，点击「下一步」（默认“基于角色或基于功能的安装”）。

3. 选择「从服务器池中选择服务器」，确保当前服务器在列表中（默认已选中），点击「下一步」。
4. 角色列表中，勾选「Web 服务器（IIS）」，弹出“添加角色和功能向导”提示框，点击「添加功能」，然后点击「下一步」。
5. 功能列表无需额外勾选（默认即可），点击「下一步」。
6. 进入「Web 服务器角色（IIS）」配置页，点击「下一步」（无需修改默认角色服务）。
7. 角色服务列表保持默认（核心服务、应用程序开发、安全等均已勾选），点击「下一步」。
8. 确认安装配置，点击「安装」，等待安装完成（约5-10分钟，取决于服务器性能），安装完成后点击「关闭」。
9. 验证 IIS 是否安装成功：打开服务器自带浏览器（IE 或 Edge），输入地址「<http://localhost>」，若显示“IIS 8”默认页面（蓝色背景，显示“欢迎使用 Internet 信息服务”），则 IIS 安装成功。

步骤3：配置 IIS 网站+创建文件存储结构（重点适配拓展需求）

创建专属网站，指定主目录，同时创建「1个主文件夹+2个子文件夹」，主文件夹存放主/子页面HTML文件，两个子文件夹分别存放对应类型的Excel文件，配置访问权限，确保页面可正常加载文件夹内文件。

3.1 创建文件存储目录（严格按照以下结构，便于后续拓展）

1. 打开「计算机」，在 D 盘（或其他非系统盘，避免系统盘占用过高）创建**主文件夹**，命名为「ExcelPreview」（建议英文名称，无特殊符号，作为所有页面和文件的根目录）。
2. 打开「ExcelPreview」主文件夹，在其内部创建**两个子文件夹**，分别命名为：
 - 「系统账号权限表」：用于存放“系统账号权限表”相关的所有Excel文件，对应同名子页面。
 - 「出差申请报告表」：用于存放“出差申请报告表”相关的所有Excel文件，对应同名子页面。
3. 权限配置（主文件夹配置权限，子文件夹自动继承，无需单独配置）：
 - 右键「ExcelPreview」主文件夹→「属性」→「安全」标签页→点击「编辑」→「添加」，输入「Everyone」（允许所有用户访问，适合内部共享；若需限制权限，可输入指定用户），点击「确定」。
 - 选中「Everyone」，在下方“权限”列表中，勾选「完全控制」「修改」「读取和执行」「列出文件夹目录」「读取」「写入」，点击「确定」→「确定」，完成权限配置（确保 IIS 可读取主文件夹内的HTML文件、子文件夹内的Excel文件）。

3.2 创建 IIS 网站（托管主文件夹，适配所有页面和文件）

1. 打开「服务器管理器」→「工具」→「Internet 信息服务（IIS）管理器」，打开 IIS 管理界面。
2. 左侧导航栏，展开服务器名称（如「WIN-SERVER2012」），右键「网站」→「添加网站」。
3. 弹出“添加网站”窗口，填写以下配置（关键项必对，适配主/子页面结构）：
 - 网站名称：ExcelPreview（建议与主文件夹名称一致，便于管理，后续拓展可快速识别）。

- 物理路径：点击「浏览」，选择步骤3.1创建的「D:\ExcelPreview」主文件夹，点击「确定」（无需选择子文件夹，IIS托管主目录后，可自动识别内部的HTML文件和子文件夹）。

- 绑定：类型选择「HTTP」，IP地址选择「全部未分配」（允许局域网内所有设备访问，若需指定IP，可选择服务器本地IP），端口填写「8080」（避免与默认IIS网站的80端口冲突，若80端口未被占用，也可使用80），主机名留空。

- 勾选「立即启动网站」，点击「确定」。

3.3 配置 IIS MIME 类型（关键：确保Excel文件可被网页访问）

1. 左侧导航栏，选中刚创建的「ExcelPreview」网站，右侧「IIS」区域，找到「MIME类型」，双击打开。

2. 右侧点击「添加」，弹出“添加MIME类型”窗口，分别添加以下两种类型（适配.xlsx和.xls格式，覆盖所有Excel文件）：

1. 扩展名：.xlsx，MIME类型：application/vnd.openxmlformats-officedocument.spreadsheetml.sheet，点击「确定」。

2. 扩展名：.xls，MIME类型：application/vnd.ms-excel，点击「确定」。

3. 添加完成后，关闭「MIME类型」窗口，回到IIS管理界面。

3.4 启用 IIS 目录浏览（必选，适配子文件夹文件列表加载）

1. 选中「ExcelPreview」网站，右侧「IIS」区域，找到「目录浏览」，双击打开。

2. 右侧点击「启用」，关闭「目录浏览」窗口（必须启用，否则子页面无法加载对应子文件夹内的Excel文件，后续新增子文件夹也需依赖此配置）。

3.5 测试 IIS 网站可访问性

1. 打开服务器浏览器，输入「<http://localhost:8080>」（端口与步骤3.2一致）。

2. 若页面显示「ExcelPreview」主文件夹的目录（包含「系统账号权限表」「出差申请报告表」两个子文件夹），则网站配置成功，后续存放HTML文件后可正常访问。

步骤4：部署 SheetJS 库并编写 HTML5 页面（主页面+2个子页面，核心拓展步骤）

SheetJS 是前端开源库，可直接解析 Excel 文件并渲染为 HTML 表格，无需后端代码，我们直接引入在线库（无需下载本地文件，简化部署），编写1个主页面（index.html）和2个子页面，**全部放入「D:\ExcelPreview」主文件夹**，子页面自动关联对应子文件夹，实现“主页面选表→进入子页面→预览/下载对应Excel文件”的完整流程。

4.1 编写主页面（index.html，核心入口，显示两个表格选项）

主页面作为唯一入口，仅提供选项跳转，不涉及Excel解析，结构简洁，便于后续新增选项（拓展新子页面）。

1. 打开服务器「记事本」（开始→所有程序→附件→记事本），复制以下代码（完整复制，无需修改，跳转地址已与子页面文件名对应）：

```
1  <!DOCTYPE html>
2  <html lang="zh-CN">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Excel HTML5 在线预览（主页面）</title>
7      <!-- 简单CSS样式，优化页面显示，便于后续拓展修改 -->
8      <style>
9          body { margin: 20px; font-family: "微软雅黑", sans-serif; }
10         .container { max-width: 800px; margin: 50px auto; text-align: center; }
11         h1 { color: #333; margin-bottom: 50px; }
12         .table-list { display: flex; flex-direction: column; gap: 30px; align-
13 items: center; }
14         .table-item {
15             width: 60%;
16             padding: 30px 20px;
17             border: 2px dashed #ccc;
18             border-radius: 10px;
19             cursor: pointer;
20             transition: all 0.3s ease;
21         }
22         .table-item:hover {
23             background-color: #f5f5f5;
24             border-color: #666;
25         }
26         .table-item h2 { color: #333; margin: 0; font-size: 22px; }
27         .table-item p { color: #666; margin-top: 10px; font-size: 16px; }
28     </style>
29 </head>
30 <body>
31     <div class="container">
32         <h1>Excel 在线预览系统（主页面）</h1>
33         <div class="table-list">
34             <!-- 系统账号权限表 选项，点击跳转至对应子页面（文件名：system-
35 account.html） -->
36             <div class="table-item" onclick="window.location.href='system-
37 account.html'">
38                 <h2>系统账号权限表</h2>
39                 <p>点击进入，预览/下载「系统账号权限表」文件夹内的Excel文件</p>
40             </div>
41             <!-- 出差申请报告表 选项，点击跳转至对应子页面（文件名：business-
42 trip.html） -->
43             <div class="table-item" onclick="window.location.href='business-
44 trip.html'">
45                 <h2>出差申请报告表</h2>
46                 <p>点击进入，预览/下载「出差申请报告表」文件夹内的Excel文件</p>
47             </div>
48         </div>
49     </div>
50 </body>
51 </html>
```

```
43         </div>
44     </div>
45 </body>
46 </html>
```

2. 点击记事本「文件」→「保存」，弹出保存窗口，严格按照以下配置保存（确保主页面可被IIS识别）：

- 保存位置：选择步骤3.1创建的「D:\ExcelPreview」主文件夹（必须是主文件夹，不可放入子文件夹）。
- 文件名：输入「index.html」（必须是这个名称，作为网站默认首页，IIS会优先访问，后续拓展无需修改）。
- 保存类型：选择「所有文件」（避免保存为.txt格式，否则无法被浏览器识别）。
- 编码：选择「UTF-8」（避免中文乱码，后续所有HTML文件均需保持此编码），点击「保存」。

4.2 编写子页面1：系统账号权限表（system-account.html）

子页面1与「系统账号权限表」子文件夹绑定，自动加载该文件夹内所有Excel文件，支持点击预览、右键下载，代码可复用，后续新增子页面可参考此模板修改。

1. 重新打开一个空白记事本，复制以下代码（完整复制，无需修改，已自动关联「系统账号权限表」子文件夹）：

Code block

```
1  <!DOCTYPE html>
2  <html lang="zh-CN">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>系统账号权限表 - Excel预览</title>
7      <!-- 引入 SheetJS 在线库（无需下载，直接引用，稳定可靠） -->
8      <script
9          src="https://cdn.jsdelivr.net/npm/xlsx@0.18.5/dist/xlsx.full.min.js"></script>
10     <style>
11         body { margin: 20px; font-family: "微软雅黑", sans-serif; }
12         .container { max-width: 1200px; margin: 0 auto; }
13         .header { display: flex; align-items: center; gap: 20px; margin-
14             bottom: 30px; }
15         .back-btn { padding: 8px 16px; border: 1px solid #ccc; border-radius:
16             5px; cursor: pointer; background: #f5f5f5; }
17         .back-btn:hover { background: #eee; }
18         h1 { color: #333; margin: 0; }
19         /* 文件列表样式，便于后续拓展修改 */
20         #fileList { border: 2px dashed #ccc; border-radius: 5px; padding:
21             20px; margin-bottom: 30px; }
```

```

18     #fileList h2 { color: #555; margin-top: 0; margin-bottom: 20px; font-
size: 18px; }
19     .file-item {
20         display: flex; align-items: center; padding: 10px 15px;
21         border-bottom: 1px solid #eee; cursor: pointer;
22     }
23     .file-item:hover { background-color: #f5f5f5; }
24     .file-name { flex: 1; color: #333; }
25     .file-tip { color: #666; font-size: 14px; margin-left: 20px; }
26     /* 预览区域样式 */
27     #previewArea { border: 1px solid #eee; border-radius: 5px; padding:
20px; min-height: 300px; }
28     table { width: 100%; border-collapse: collapse; margin-top: 20px; }
29     th, td { border: 1px solid #ccc; padding: 8px 12px; text-align: left; }
30     th { background-color: #f5f5f5; font-weight: bold; }
31     tr:hover { background-color: #fafafa; }
32     /* 无文件提示 */
33     .no-file { text-align: center; color: #666; padding: 20px 0; }
34 </style>
35 </head>
36 <body>
37     <div class="container">
38         <div class="header">
39             <!-- 返回主页面按钮，跳转地址关联主页面文件名 -->
40             <div class="back-btn" onclick="window.location.href='index.html'">
返回主页面</div>
41             <h1>系统账号权限表 - Excel在线预览</h1>
42         </div>
43
44         <!-- 文件列表区域：仅展示「系统账号权限表」子文件夹下所有Excel文件 -->
45         <div id="fileList">
46             <h2>「系统账号权限表」文件夹文件列表（点击预览，右键下载）</h2>
47             <div id="fileItems">
48                 <p class="no-file">正在加载文件列表...若长时间无显示，请检查文件夹权
限或IIS配置</p>
49             </div>
50         </div>
51
52         <!-- 预览区域，用于显示解析后的Excel表格 -->
53         <div id="previewArea">
54             <p style="text-align: center; color: #666;">请点击上方文件列表中的
Excel文件，进行在线预览</p>
55         </div>
56     </div>
57
58     <!-- 核心JS代码：加载子文件夹文件列表、点击预览、右键下载（可复用） -->
59     <script>

```



```

60      // 获取页面元素
61      const fileItems = document.getElementById('fileItems');
62      const previewArea = document.getElementById('previewArea');
63      // 子文件夹路径（严格对应「系统账号权限表」子文件夹名称，不可修改，否则无法加载文
      件)
64      const folderPath = '系统账号权限表';
65
66      // 1. 加载子文件夹内所有Excel文件，生成文件列表
67      function loadFileList() {
68          // 利用IIS目录浏览的特性，请求子文件夹，解析返回的HTML获取文件列表（无需后
      端接口)
69          fetch(folderPath)
70              .then(response => {
71                  if (!response.ok) throw new Error('文件列表加载失败，请检查IIS
      目录浏览是否启用或子文件夹是否存在');
72                  return response.text();
73              })
74              .then(html => {
75                  // 解析HTML，仅提取.xlsx和.xls格式的Excel文件
76                  const parser = new DOMParser();

```

2. 点击记事本「文件」→「保存」，弹出保存窗口，严格按照以下配置保存（与主页面同目录）：

- 保存位置：选择「D:\ExcelPreview」主文件夹（与index.html同目录，不可放入子文件夹）。
- 文件名：输入「system-account.html」（必须是这个名称，与主页面跳转地址严格对应，不可修改，否则无法跳转）。
- 保存类型：选择「所有文件」，编码：选择「UTF-8」，点击「保存」。

4.3 编写子页面2：出差申请报告表（business-trip.html）

子页面2与「出差申请报告表」子文件夹绑定，逻辑、样式与子页面1完全一致，仅修改子文件夹路径和页面标题，便于后续统一维护和拓展。

1. 重新打开一个空白记事本，复制以下代码（完整复制，无需修改，已自动关联「出差申请报告表」子文件夹）：

Code block

```

1  <!DOCTYPE html>
2  <html lang="zh-CN">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>出差申请报告表 - Excel预览</title>
7      <!-- 引入 SheetJS 在线库（无需下载，直接引用，与子页面1一致） -->
8      <script
      src="https://cdn.jsdelivr.net/npm/xlsx@0.18.5/dist/xlsx.full.min.js"></script>
9      <style>

```

```

10     body { margin: 20px; font-family: "微软雅黑", sans-serif; }
11     .container { max-width: 1200px; margin: 0 auto; }
12     .header { display: flex; align-items: center; gap: 20px; margin-bottom: 30px; }
13     .back-btn { padding: 8px 16px; border: 1px solid #ccc; border-radius: 5px; cursor: pointer; background: #f5f5f5; }
14     .back-btn:hover { background: #eee; }
15     h1 { color: #333; margin: 0; }
16     /* 文件列表样式, 与子页面1一致, 便于统一维护拓展 */
17     #fileList { border: 2px dashed #ccc; border-radius: 5px; padding: 20px; margin-bottom: 30px; }
18     #fileList h2 { color: #555; margin-top: 0; margin-bottom: 20px; font-size: 18px; }
19     .file-item {
20         display: flex; align-items: center; padding: 10px 15px;
21         border-bottom: 1px solid #eee; cursor: pointer;
22     }
23     .file-item:hover { background-color: #f5f5f5; }
24     .file-name { flex: 1; color: #333; }
25     .file-tip { color: #666; font-size: 14px; margin-left: 20px; }
26     /* 预览区域样式, 与子页面1一致 */
27     #previewArea { border: 1px solid #eee; border-radius: 5px; padding: 20px; min-height: 300px; }
28     table { width: 100%; border-collapse: collapse; margin-top: 20px; }
29     th, td { border: 1px solid #ccc; padding: 8px 12px; text-align: left; }
30     th { background-color: #f5f5f5; font-weight: bold; }
31     tr:hover { background-color: #fafafa; }
32     /* 无文件提示 */
33     .no-file { text-align: center; color: #666; padding: 20px 0; }
34 </style>
35 </head>
36 <body>
37     <div class="container">
38         <div class="header">
39             <!-- 返回主页面按钮, 跳转地址关联主页面文件名, 与子页面1一致 -->
40             <div class="back-btn" onclick="window.location.href='index.html'">
返回主页面</div>
41             <h1>出差申请报告表 - Excel在线预览</h1>
42         </div>
43
44         <!-- 文件列表区域: 仅展示「出差申请报告表」子文件夹下所有Excel文件 -->
45         <div id="fileList">
46             <h2>「出差申请报告表」文件夹文件列表（点击预览, 右键下载）</h2>
47             <div id="fileItems">
48                 <p class="no-file">正在加载文件列表...若长时间无显示, 请检查文件夹权限或IIS配置</p>
49             </div>

```



```

50         </div>
51
52         <!-- 预览区域，用于显示解析后的Excel表格 -->
53         <div id="previewArea">
54             <p style="text-align: center; color: #666;">请点击上方文件列表中的
Excel文件，进行在线预览</p>
55         </div>
56     </div>
57
58     <!-- 核心JS代码：与子页面1一致，仅修改子文件夹路径，可复用 -->
59     <script>
60         // 获取页面元素
61         const fileItems = document.getElementById('fileItems');
62         const previewArea = document.getElementById('previewArea');
63         // 子文件夹路径（严格对应「出差申请报告表」子文件夹名称，不可修改）
64         const folderPath = '出差申请报告表';
65
66         // 1. 加载子文件夹内所有Excel文件，生成文件列表（与子页面1逻辑一致）
67         function loadFileList() {
68             fetch(folderPath)
69                 .then(response => {
70                     if (!response.ok) throw new Error('文件列表加载失败，请检查IIS
目录浏览是否启用或子文件夹是否存在');
71                     return response.text();
72                 })
73                 .then(html => {
74                     const parser = new DOMParser();
75                     const doc = parser.parseFromString(html, 'text/html');
76                     const links = doc.querySelectorAll('a[href$=".xlsx"],
a[href$=".xls"]');
77

```

2. 点击记事本「文件」→「保存」，弹出保存窗口，严格按照以下配置保存：

- 保存位置：选择「D:\ExcelPreview」主文件夹（与index.html、system-account.html同目录）。
- 文件名：输入「business-trip.html」（必须是这个名称，与主页面跳转地址严格对应，不可修改）。
- 保存类型：选择「所有文件」，编码：选择「UTF-8」，点击「保存」。

4.4 放置测试Excel文件（适配子文件夹结构，用于功能验证）

为测试两个子页面的功能，需将Excel文件分别放入对应子文件夹，避免放错导致无法加载：

1. 找1-2个测试用的Excel文件（.xlsx或.xls格式），复制到「D:\ExcelPreview\系统账号权限表」子文件夹（对应子页面1）。
2. 再找1-2个不同的Excel文件（.xlsx或.xls格式），复制到「D:\ExcelPreview\出差申请报告表」子文件夹（对应子页面2）。

3. 确保所有Excel文件均可正常打开（无损坏），避免测试时出现解析失败问题。

步骤5：测试主/子页面完整功能（核心验证步骤，确保符合拓展需求）

按照“主页面→子页面→预览→下载→返回”的流程测试，确保所有功能正常，页面跳转流畅，文件夹与子页面对应正确：

5.1 服务器本地测试（重点验证）

1. 打开服务器浏览器（推荐Chrome/Edge），输入地址「<http://localhost:8080>」（端口与步骤3.2一致），自动进入主页面（index.html）。

2. 主页面测试：页面显示两个选项（系统账号权限表、出差申请报告表），鼠标悬浮时样式变化，点击无报错，说明主页面正常。

3. 子页面1测试（系统账号权限表）：

- 点击主页面「系统账号权限表」选项，跳转至子页面1（system-account.html），页面顶部显示“返回主页面”按钮，下方自动加载「系统账号权限表」子文件夹内的Excel文件列表。

- 点击列表中的任意Excel文件，等待1-2秒，页面预览区域显示解析后的Excel表格（包含所有工作表），说明预览功能正常。

- 右键点击列表中的任意Excel文件，自动触发下载，下载后的文件可正常打开，说明下载功能正常。

- 点击页面顶部「返回主页面」按钮，可顺利回到主页面，说明跳转功能正常。

4. 子页面2测试（出差申请报告表）：

- 回到主页面，点击「出差申请报告表」选项，跳转至子页面2（business-trip.html），自动加载「出差申请报告表」子文件夹内的Excel文件列表（与子页面1的文件不重复）。

- 重复步骤5.1.3的预览、下载、返回操作，确保子页面2功能与子页面1一致，无异常。

5. 拓展性测试（验证后续新增便捷性）：

- 检查主页面代码，新增选项仅需复制现有「table-item」模块，修改标题、提示文字和跳转地址，即可适配新子页面。

- 检查子页面代码，新增子页面时，仅需复制子页面1代码，修改页面标题、子文件夹路径，即可关联新的子文件夹，无需修改核心解析逻辑。

5.2 局域网内其他设备测试（可选，若需局域网共享）

1. 查看服务器的本地IP地址：右键服务器桌面「网络」→「属性」→「本地连接」→「详细信息」，找到「IPv4地址」（如192.168.1.100）。

2. 打开局域网内另一台电脑的浏览器，输入地址「<http://服务器IP:8080>」（如<http://192.168.1.100:8080>），确保能正常进入主页面。

3. 重复步骤5.1的所有测试操作，确保局域网内可正常跳转页面、预览和下载文件，无权限或加载异常。

步骤6：优化与异常处理（适配后续拓展，降低维护成本）

6.1 优化建议（便于后续拓展）

1. 统一样式管理：将三个HTML文件的<style>标签内容提取到一个单独的CSS文件（如style.css），放入「ExcelPreview」主文件夹，三个页面均引用该CSS文件，后续修改样式仅需修改一个文件，无需逐个修改页面。
2. 备份代码模板：将子页面1（system-account.html）保存为模板文件（如template.html），后续新增子页面时，直接复制模板，修改标题、子文件夹路径即可，大幅提升拓展效率。
3. 规范文件命名：后续新增子页面时，文件名采用“英文标识.html”（如salary-table.html），对应子文件夹采用中文名称（如“工资表”），主页面跳转地址与子页面文件名严格对应，避免混乱。
4. 增加异常提示：可在主页面添加“页面加载失败，请刷新重试”提示，在子页面添加“文件夹不存在，请检查配置”提示，提升用户体验。

6.2 常见异常及解决方法（重点适配主/子页面结构）

- 异常1：访问<http://localhost:8080> 无法进入主页面，提示“404未找到”。

解决：1. 检查「D:\ExcelPreview」主文件夹内是否存在「index.html」文件，文件名是否正确（区分大小写）；2. 检查IIS网站的物理路径是否为「D:\ExcelPreview」；3. 重启IIS网站（IIS管理界面，选中网站，右侧点击「重启」）。

- 异常2：主页面点击选项，无法跳转至子页面，提示“404未找到”。

解决：1. 检查子页面文件名是否正确（system-account.html、business-trip.html），与主页面跳转地址一致；2. 检查子页面是否放在「D:\ExcelPreview」主文件夹内，未放入子文件夹；3. 清除浏览器缓存，重新测试。

- 异常3：子页面无法加载文件列表，提示“文件列表加载失败”。

解决：1. 检查IIS目录浏览是否已启用（步骤3.4）；2. 检查子文件夹名称是否正确（与子页面JS代码中的folderPath一致）；3. 检查子文件夹内是否有Excel文件，文件格式是否为.xlsx/.xls。

- 异常4：子页面预览/下载文件失败，提示“加载失败”。

解决：1. 检查Excel文件是否放入对应子文件夹，未放错路径；2. 检查Excel文件是否损坏，重新复制完整文件；3. 检查服务器防火墙是否放行8080端口，临时关闭防火墙测试。

三、后期拓展指南（重点，贴合你的拓展需求）

后续新增“表格类型”（如工资表、考勤表）时，可按照以下步骤快速拓展，无需修改现有页面和配置：

1. 新增子文件夹：在「D:\ExcelPreview」主文件夹内，新建一个子文件夹，命名为新增表格类型（如“工资表”，纯中文，与后续子页面标题一致）。

2. 新增子页面：复制子页面1（system-account.html）模板，修改以下3处内容，保存为新的子页面文件（如salary-table.html，英文标识）：

- 页面标题：将<title>标签内容改为“工资表 - Excel预览”。

- 页面头部标题：将<h1>标签内容改为“工资表 - Excel在线预览”。

- 子文件夹路径：将JS代码中const folderPath = '系统账号权限表' 改为 const folderPath = '工资表'（与新增子文件夹名称一致）。

3. 主页面新增选项：打开index.html，复制现有「table-item」模块，修改标题、提示文字和跳转地址，示例：

```
<div class="table-item" onclick="window.location.href='salary-table.html'">  
    <h2>工资表</h2>  
    <p>点击进入，预览/下载「工资表」文件夹内的Excel文件</p>  
</div>
```

4. 放置文件并测试：将新增表格类型的Excel文件放入新增子文件夹，访问主页面，测试跳转、预览、下载功能，无需修改IIS配置。

四、总结

本方案已完成「1个主页面+2个子页面」的完整部署，主页面作为入口提供选项跳转，两个子页面对应独立文件夹，分别存储并预览对应Excel文件，支持点击预览、右键下载，无需手动上传，结构模块化、便于后续拓展。

核心关键点：IIS网站配置（主目录、MIME类型、目录浏览）、主/子页面HTML文件的正确存放（均在主文件夹）、子页面与子文件夹路径的对应、SheetJS解析逻辑的复用。按照本流程操作，可快速完成部署，后续新增子页面/文件夹仅需简单复制修改，无需重复配置IIS和核心代码。

（注：文档部分内容可能由 AI 生成）