|  |  |
| --- | --- |
| 实验名称： | 实验2：PHP基本使用 |
| 班级： | 信息安全224 |
| 学号： | 2022132003 |

1. **实验过程：**

（实验的具体过程，可以分成两个部分，第一个部分写思路，第二个部分写具体的实现代码。）

1. 实验思路：
2. **创建HTML表单**：首先，需要创建一个HTML表单，让用户能够选择CSV文件进行上传。
3. **处理文件上传**：在PHP端，使用$\_FILES全局数组来接收上传的文件。使用move\_uploaded\_file()函数将临时文件移动到服务器上的目标位置。

**(3)解析CSV文件**：使用PHP的fgetcsv()函数来逐行读取CSV文件，并将每行数据解析为一个数组。

**(4)展示CSV数据**：将解析后的CSV数据以HTML表格的形式展示给用户。

**(5)实现排序功能**：允许用户选择按照某列进行升序或降序排序。可以使用PHP的usort()函数，配合自定义的比较函数来实现排序。

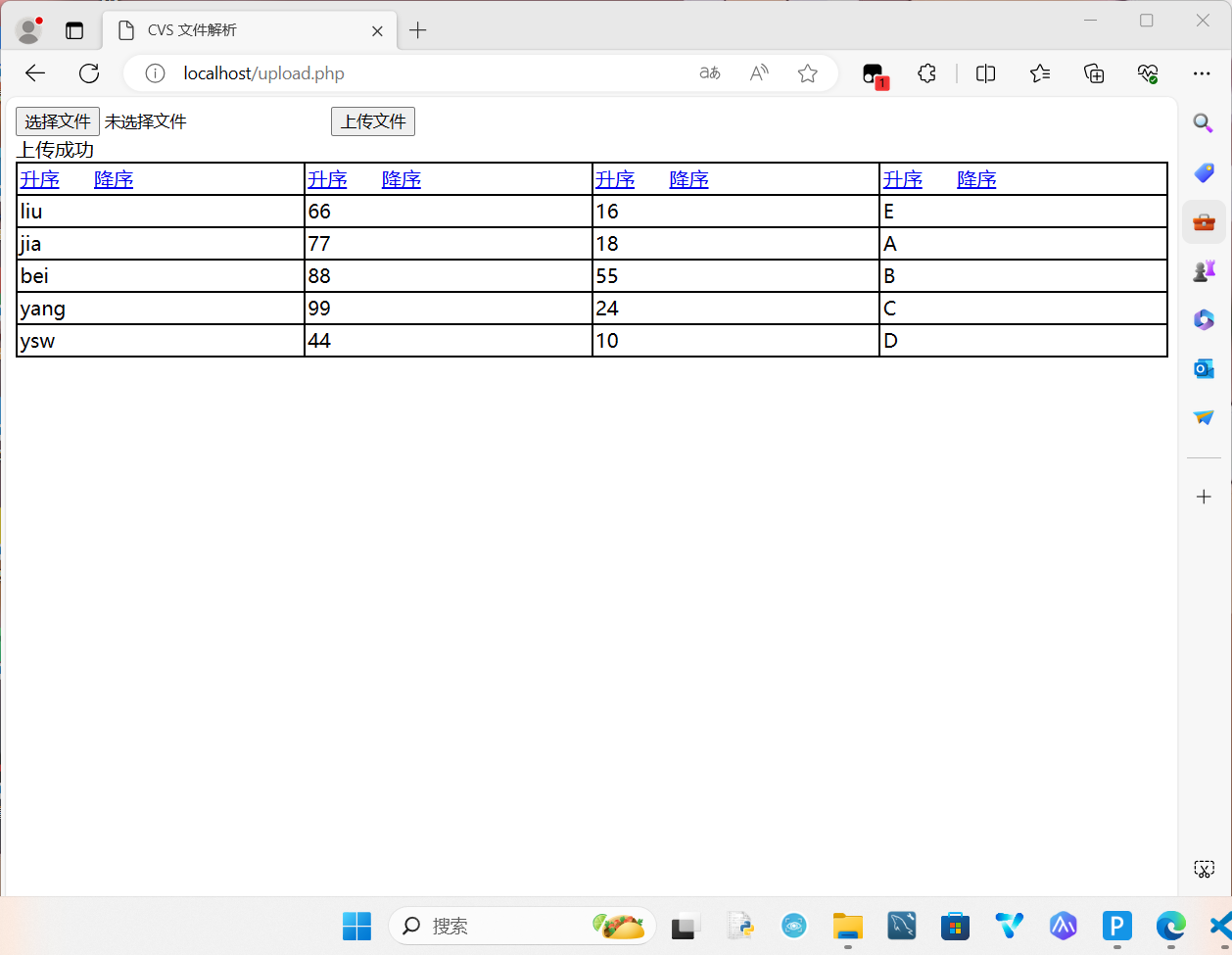
**(6)渲染排序后的数据**：将排序后的数据再次以HTML表格的形式展示给用户。

1. 代码实现：

重要代码片段说明：

<?php  
            $fileName="list.csv";  
            $hasFile=is\_file($fileName);  
            $hasNewFile=false;  
            if (isset($\_FILES['csv'])){  
                if($\_FILES['csv']['error']==0){  
                    if(substr($\_FILES['csv']['name'],-3)=="csv"){  
                        $hasNewFile=true;  
                        move\_uploaded\_file($\_FILES['csv']['tmp\_name'],$fileName);  
                        $hasFile=true;  
                    echo "上传成功";  
                    }  
                }  
            }  
            //如果有文件，解析文件  
            $csvArray=null;  
            $firstLine=true;  
            if($hasFile){  
                $file=fopen($fileName,'r');  
                $row=fgets($file);  
                while ($row!=false){  
                    if($firstLine){  
                        $BOM=chr(239).chr(187).chr(191);  
                        $row=str\_replace($BOM,'',$row);  
                        $firstLine=false;  
                    }  
                    if($row!=false){  
                        $csvArray[]=explode(',',$row);  
                    }  
                    $row=fgets($file);  
                }  
                fclose($file);  
            }  
            //从二维数组中提取某一列的值  
            function getColArray($inArray,$upload){  
                $sortArray=null;  
                foreach($inArray as $key=>$val){  
                    $sortArray[$key]=$val[$upload];  
                }  
                return $sortArray;  
            }  
            //检查是否有get参数s和c,如果有，则根据指定的列进行升序或降序排序。  
            $sortArray=null;  
            if(isset($\_GET['s'])){  
                $sortArray=getColArray($csvArray,$\_GET['c']);  
                if($\_GET['s']){  
                    asort($sortArray,SORT\_REGULAR);  
                }else{  
                    arsort($sortArray,SORT\_REGULAR);  
                }  
            }  
            //利用 php 输出表头的超链接  
            if($csvArray){  
                echo "<tr>";  
                for($i=0;$i<count($csvArray[0]);$i++){  
?>  
                    <td>  
                        <a href="upload.php?s=1&c=<?=$i?>">升序</a>  
                        &nbsp;&nbsp;&nbsp;&nbsp;  
                        <a href="upload.php?s=0&c=<?=$i?>">降序</a>  
                    </td>  
<?php

1. **实验结果**

****

1. **总结与思考**

首先，我意识到在Web开发中，用户交互和数据处理是两个至关重要的环节。其次，我深刻体会到了代码逻辑和算法的重要性。此外，我也意识到错误处理和用户体验的重要性。通过实践，我不仅掌握了PHP文件处理和数据处理的相关知识，还学会了如何将这些知识应用到实际的Web开发中。这将为我未来的学习和工作提供宝贵的经验和基础。同时，我也意识到了自己在编程和Web开发方面的不足和需要改进的地方，比如代码的优化、安全性的加强以及用户体验的提升等。