# 广州理工学院实验报告

**学院：计算机科学与工程学院 专业： 数据科学与大数据技术 年级： 2022**

**姓名： 吴恩威 学号： 20220409430251组 实验时间：2023.12.4**

**指导教师签字： 成绩：**

|  |
| --- |
| **实验项目名称：**  实验9 Python文件及数据操作应用   * **实验目的和要求：**   1、熟练掌握Python文件及目录的操作方法及相关模块的使用；  2、掌握Python对csv文件操作及方法；  3、能够完成工程应用下对各类文件及文件目录进行操作。 |
| * **主要仪器设备：**   PC机一台；Windows操作系统；Python运行环境；开发工具（PyCharm)。 |
| * **实验内容和原理：**   1、Python文件操作及方法应用（OS模块、shutil模块、CSV模块）；  2、Python目录操作及方法应用（OS模块、shutil模块）。  3、文件（目录）的实际工程应用。 |
| * **操作方法和实验步骤（绘图）：**   1、“record.txt”文件有3段对话,分别用“=========……”分隔，请对老师下发的“record.txt”文件，进行编程，实现以下功能：  (1)、要根据不同的角色（小甲鱼、小客服），将每段对话内容分别存入对应的文件中；  (2)、小甲鱼的说话内容存入boy\_1，boy\_2......文件中去；  (3)、小客服的说话内容存入girl\_1，girl\_2......文件中去；  (4)、请定义函数完成，完成后，将会生成6个文件，如下图所示：    (5)、统计并输出record.txt文本文件中对话内容最长那一行的对话内容长度以及该行的内容。  操作方法与步骤：  （1）定义open\_read\_file函数，实现文件的打开与读取,读取过程中根据角色不同，仅将角色说的话内容提取出来，调用save\_file函数将一段提出的不同角色内容存入对应文件。  （2）定义save\_file函数，实现将不同角色内容存入对应文件。  （3）定义max\_length函数，实现将每一行角色的说话内容进行按长度排序，然后提取最长的对话内容，再求解长度，输出结果。  （4）主函数调用以上函数，实现题目要求。  过程代码截图如下：      2、使用csv模块读取并输出“iris.csv”文件中的内容，并将“sepal\_length”长度为6以上的存入iris1\_1.csv中。  操作方法与步骤：使用open函数打开文件，使用csv模块的字典读取器读取文件iris.csv,将“sepal\_length”长度为6以上的数据筛选出来，写入iris1\_1.csv即可。注意写入新文件iris1\_1.csv时的编码格式、换行情况的设置等。  过程代码截图如下：    3、使用递归调用实现：遍历指定目录（如F:\test）下所有目录（包含子目录）下. txt文件改为.py文件。  操作方法与步骤：（1）定义递归调用函数seacher\_file文件，函数内先判断传递的目录是否是本地系统真实存在的目录，如果是真实存在的目录，则遍历该目录内的所有对象（子目录、文件），如果是文件，进一步判断是否为txt文件，如果是则修改为py文件，如果不是，则忽略；如果是目录，就递归调用seacher\_file函数，将当前子目录传递给seacher\_file函数，继续遍历该子目录。（2）主函数调用并传递指定目录给seacher\_file函数。  过程代码截图如下： |
| * **实验结果与分析**  1. 实验结果如下：       结果分析：1、open\_read\_file(filename),打开文件并按照对话者分类：使用 open() 函数打开指定文件 filename。(1)创建了两个空列表 jiayu\_word 和 kefu\_word，用于存储不同人物的对话内容。（2）通过遍历文件中的每一行，根据对话开始的标识（例如："小甲鱼" 或 "小客服"），将对应的对话内容存储到不同的列表中。（3）分割对话并保存到文件：当遇到分割线（"==="）时，调用 save\_file() 函数，将之前存储的 "小甲鱼" 和 "小客服" 的对话内容写入到两个不同的文件中。每次遇到分割线，都会增加计数器 count，确保每个文件名都是唯一的。  2、save\_file(data1, data2, count)保存对话内容到文件：（1）创建了两个文件名，分别命名为 "第一题/boy\_<count>.txt" 和 "第一题/girl\_<count>.txt"。（2）使用 open() 函数以写入模式打开这两个文件。（3）将传入的 data1 和 data2 分别写入到对应的文件中，这里假设 data1 是 "小甲鱼" 的对话内容，data2 是 "小客服" 的对话内容。3、max\_length(filename)查找最长的一句话：打开指定的文件 filename。（1）创建了一个空列表 speak\_ans，用于存储文件中所有对话内容。（2）遍历文件中的每一行，将对话内容添加到 speak\_ans 列表中。（3）使用 sorted() 函数并传入一个 key 参数，按对话内容长度对列表进行排序。（4）打印排序后列表中的最后一项，即最长的一句话，以及其长度。  2、实验结果如下：    结果分析：（1）使用 with 语句打开名为 "iris.csv" 的文件，指定使用 'gbk' 编码读取文件，并创建一个文件对象 in\_file。（2）使用 csv.DictReader 创建一个 CSV 读取器 reader，用于按行读取 CSV 文件。这个读取器会把每一行数据转换为一个字典，以文件的第一行作为列名。（3）对于读取器 reader 中的每一行（表示为字典 row），它会检查键为 'sepal\_length' 的值是否大于等于 6。在此处使用了 eval() 函数来将 'sepal\_length' 对应的字符串转换为 Python 对象，这样会将其评估为一个数值（可能是浮点数或整数），然后进行大小比较。（3）使用 with 语句创建一个名为 "iris1\_1.csv" 的文件，以写入模式打开，并指定使用 'gbk' 编码写入文件。newline='' 参数确保在写入时不自动插入额外的换行符。（4）使用 csv.DictWriter 创建一个 CSV 写入器 writer，使用与读取器相同的列名 reader.fieldnames。（5）将 CSV 文件的头部（即列名）写入 "iris1\_1.csv" 文件中。（6）将筛选出的符合条件的行（即 "sepal\_length" 大于等于 6 的行）写入到 "iris1\_1.csv" 文件中。  3、实验结果如下：    结果分析：1、检查路径是否是目录：使用 os.path.isdir() 来检查传入的路径 filename 是否是一个目录，如果不是目录则直接返回，不进行任何操作。2、遍历目录内容：使用 os.listdir() 遍历指定目录中的所有对象（文件和子目录）。对于每一个找到的对象，使用 os.path.join() 创建完整的路径，以便后续处理。3、递归调用：如果当前对象是一个目录（使用 os.path.isdir() 进行判断），则递归调用 seacher\_file() 函数，传入当前目录路径 cur，继续深入遍历该子目录。4、改名操作：如果当前对象是一个文件（使用 os.path.isfile() 进行判断），且文件名以 '.txt' 结尾，则执行文件名修改的操作。使用 os.path.splitext() 拆分文件名和扩展名，将 '.txt' 替换为 '.py'。使用 os.rename() 将文件从当前路径 cur 改名为新的路径 new\_path。 |

**批阅评语：**