|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **课题** | 项目二 Ｅxcel电子表格操作 | | | **备课时间** | |  |
| **课型** | 理实一体化 | **课时** | 16课时 | **授课班级** | |  |
| **教学目标** | 知识目标：  （1）认识Python常用的第三方扩展库。  （2）认识openpyxl库的作用。  能力目标：  （1）能根据需要安装必要的Python第三方扩展库。  （2）能简单应用openpyxl库。  素质目标：  （1）能根据需要安装必要的Python第三方扩展库。  （2）能简单应用openpyxl库。 | | | | | |
| **教材**  **分析** | **重难点** | 能根据需要安装必要的Python第三方扩展库。  能简单应用openpyxl库。 | | | | |
| **教具** | PPT、课本 | | | | | |
| **教法** | 案例分析、合作学习、电脑模拟等方法 | | | | | |
| **学法** | 学生在教师的指导下课前复习课本的内容，课后及时巩固上课所学的知识，完成上课布置的作业。 | | | | | |
| **教 学 过 程** | | | | | | |
| **教学环节** | **教 学 内 容** | | | | **教师活动、学生活动** | |
| **导入新课** | 教师引入：openpyxl是一个Python库，用于读取/写入.xlsx、.xlsm、.xltx、.xltm文件，但不支持.xls文件。  在Python程序中要引用openpyxl模块，首先必须安装openpyxl模块库，方法为从Windows进入COM命令窗口，运行“pip install openpyxl”命令，安装openpyxl模块库。  你对Python库了解多少呢？ | | | | 教师通过设问引入本课内容，学生分小组讨论。 | |
| **教学过程** | **任务一 增加表格中的订单数量** 任务要求： （1）现有文件table1.xlsx，table1.xlsx中Sheet1工作表是一份订单表，其中订单为数量。  （2）请读取表中的数据，并把订单数量增加2。  （3）结果保存在文件table2.xlsx中。  **实现步骤：**  （1）在D盘创建文件夹mypython，并把table1.xlsx复制到D:\mypython\xls目录下，如图所示。    （2）启动Visual Studio Code，执行“文件/打开文件夹”命令。  （3）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，如图所示。    （4）执行“运行/启动调试”命令，如图所示。    （5）运行程序时，会看终端输出了print括号中的变量值，如图所示。    （6）运行程序后，查看xls子目录，能看到程序生成的table2.xlsx文件，如图所示。    （7）打开table2.xlsx文件，能看到工作表中订单的数值比table1.xlsx文件中订单的数值都多了2，如图所示。    **任务二 计算表格中的“合计”值** 任务要求： （1）现有文件“销售单.xlsx”，是一份“合计”列为空的销售单。  （2）按公式“合计=数量\*单价”计算，得到“合计”的值。  （3）在数据的最后添加“合计”行，字体为粗体。  （4）结果保存在文件“销售单合计.xlsx”中。  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码编程实现如下功能：导入openpyxl模块，导入Font用于设置字体格式，打开“销售单.xlsx”文件，把Sheet1工作表的“数量乘以单价”写入“合计”单元格，并设置货币格式，如图所示。    （2）在mypython.py文件继续输入程序代码实现功能：在原数据最后添加“合计”一行，如图所示。    （3）运行程序后，查看xls目录的“销售单合计.xlsx”文件内容是否符合要求。  **任务三 创建新的景点表** 任务要求： （1）编写程序实现“景点表”的创建功能。  （2）“景点表”包括“景点名称”“所在省市”表头。  （3）表名为“景点表”。  （4）A2:A7内容是一列景点名称，B2:B7内容是景点所在省市的名称。  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：导入openpyxl模块，定义data变量的值拟存入表格中，自定数组变量data\_excel，并把data变量的值赋值给数组变量data\_excel，如图所示。    （2）输入程序代码，创建工作簿文件实例赋值给wb变量，把数组变量data\_excel的值写入单元格中，如图所示。    **任务四 合并单元格** 任务要求： （1）编写程序实现“派工单”的创建功能，并按效果图实现单元格合并。  （2）“派工单”包括“派工单”表头、项目名称、工作内容、派工人、负责人等。  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：从openpyxl模块导入Workbook，从openpyxl.styles模块导入Alignment，表格第1行A1:F1单元格合并，写入“派工单”，代码实现文本内容水平居中，垂直居中。    （2）在mypython.py文件继续输入程序代码，实现功能：第2行第1列输入“项目名称：”，合并B2:F2单元格；合并A3:F6单元格，写入“工作内容，设置文本内容水平方向左对齐、垂直方向靠顶部对齐的格式。    （3）在mypython.py文件继续输入程序代码，实现功能：在第7行第1列输入“派工人：”，合并B7:C7单元格；在第7行第4列输入“负责人：”，合并E7:F7单元格，保存文件名为“派工单.xlsx”。    **任务五 更改单元格样式** 任务要求： （1）打开“派工单.xlsx”，按要求修改单元格样式。  （2）第1行的高度为40磅。  （3）表对单元格填充一种合适的背景色。  （4）第1行的单元格设置边框线，其中底部边框线的style='thick'，其他边框线的style='thin'，形成加粗下边框线的效果。  （5）“项目名称：”“派工人：”“负责人：”等单元格填充一种合适的背景色。  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：从openpyxl模块导入load\_workbook，从openpyxl.styles模块导入Alignment、PatternFill、 colors、Border、Side等功能模块。    （2）在mypython.py文件继续输入程序代码，实现功能：打开“派工单.xlsx”文件，打开第1个工作表，设置第1行的高度为40。    （3）在mypython.py文件输入程序代码，实现功能：定义单元格颜色的填充方式fills = PatternFill("solid", fgColor="FFC0CB")，参数solid表示纯色方式填充，参数fgColor设置填充的前景色，ws.cell(1, 1).fill = fills实现对所指定第1行第1列单元格的填充。编写代码用fills填充第1行第1列、第2行第1列、第7行第1列、第7行第4列。    （4）在mypython.py文件继续输入程序代码，实现功能：Border()函数设置单元格左边框、右边框、上边框、下边框的线型和颜色，设置A1、B1、C1、D1、E1、F1等单元格的边框，最后保存文件为“派工单2.xlsx”。    **任务六 插入文件夹中的多张图片**  **任务要求：**  （1）创建新的工作簿，把imgs子目录中的所有图片插入工作表。  （2）表头依次是“序号”“图片”“文件名称”。  （3）每张图片一行，图片左列是序号数字，图片右列是图片文件名。  （4）设置合适的行高和列宽。  **实现步骤：**  （1）在D:\mypython文件夹的imgs子目录中存放若干张图片。    （2）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：从openpyxl模块导入Workbook，从openpyxl.drawing.image模块导入Image，导入os模块。    （3）在mypython.py文件继续输入程序代码，实现功能：打开“课程表.xlsx”文件，打开第1个工作表，设置A列的宽度为20，设置第1行的高度为40。      （5）在mypython.py文件继续输入程序代码，实现功能：遍历file\_name\_list的所有文件，序号写入A列，文件名写入C列，图片插入到B列。    **任务七 读取工作表中的图片**  **任务要求：**  （1）现“景点列表.xlsx”文件的工作表中，有多张景点图片和标题。  （2）编程获取工作表中的图片，把图片保存在指定文件夹imgs中。  （3）文件名以标题命名，扩展名为png，如景点1.png、景点2.png、景点3.png、景点4.png。  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：从PIL模块导入Image函数，从openpyxl模块导入load\_workbook函数，打开“景点列表.xlsx”文件，获取第1个工作表，如图所示。    （2）在mypython.py文件继续输入程序代码，实现功能：把获取到的图片文件用标题命名后保存到电脑指定的imgs目录中，如图所示。    （3）运行程序之后，在D:\mypython文件夹的imgs子目录中可查看到获取的图片已按要求保存，如图所示。    **任务八 删除与插入单元格**  **任务要求：**  （1）现有文件“一周生产计划表.xlsx”。  （2）原生产计划表在周末要上班，现准备以后不在周末双休，删除周末的上班计划，为了不影响生产，把原周末的生产量平均分配到每天的加班中，所以，需要添加“加班”计划。  （4）完成后输出文件“一周生产计划表(加班).xlsx”。  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：从openpyxl模块导入load\_workbook函数，从openpyxl.styles模块导入Alignment函数，打开“一周生产计划表.xlsx”文件，获取第1个工作表，如图所示。    （2）在mypython.py文件继续输入程序代码，实现功能：插入第6行，删除第7行两次，用for语句在第6行的第2列至第6列单元格填入“1680”表示加班生产数量，在第6行第1列填入“加班”，并设水平居中且垂直居中，最后保存文件名为“一周生产计划表(加班).xlsx”。    （3）运行程序之后，在程序mypython.py文件的目录下，打开“一周生产计划表(加班).xlsx”。    **任务九 生成柱形图表**  **任务要求：**  （1）现有文件“产品月利润表.xlsx”，数据在工作表Sheet1中A1:D区域。  （2）以A1:D区域的数据创建柱形图，在F1单元格插入图表。  （3）设置图表标题“利润图表”。  （4）设置y轴方向标题“数量”、x轴方向标题“项目”。  （5）显示分类图例名称和项目名称。  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：从openpyxl.chart 模块导入BarChart和Reference函数，从openpyxl模块导入load\_workbook函数，打开“产品月利润表.xlsx”文件，打开第1个工作表，如图所示。    （2）在mypython.py文件继续输入程序代码，实现功能：用values记录Reference()设置图表数据源区域，用BarChart()创建一个柱形图实例，用series记录Reference()设置图例数据源区域，用add\_data()生成图表，用set\_categories()设置图表图例，如图所示。    （3）在D:\mypython文件夹的imgs子目录里存放若干张图片，如图所示。    **任务十 分割工作簿文件**  **任务要求：**  （1）现有文件记录多个公司产品记录的“多公司汇总表.xlsx”。  （2）请把每个公司的记录存为一张表，文件名按公司名称命名。  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：导入openpyxl模块，打开“多公司汇总表.xlsx”文件的第1个工作表，获取表格中的公司名，用于在后续的操作步骤中，以公司名为区分进行数据分割，如图所示。    （2）在mypython.py文件继续输入程序代码，实现功能：获取数据的表头标题，获取表中的所有记录，为在后续的操作步骤中以公司名为区分进行数据分割作为数据准备，如图所示。    （3）在mypython.py文件继续输入程序代码，实现功能：获，如图所示。    （4）运行程序之后，在xls的目录下会看到分出来的多个公司的工作簿文件，如A公司、B公司、C公司等，如图所示。    **任务十一 合并工作簿文件**  **任务要求：**  （1）现xls目录中存放A公司.xlsx、B公司.xlsx、C公司.xlsx等工作簿文件。  （2）查看确保每个工作簿文件的第1个工作表有数据且结构相同。  （3）编程实现把多个工作表内容合并在同一个工作表中。  （4）结果以“新的总表.xlsx”文件名保存在xls目录中  **实现步骤：**  （1）打开D:\mypython文件夹作为项目目录，执行“文件/新建文件”命令，新建mypython.py文件，输入程序代码，实现功能：导入openpyxl模块，导入os模块，获取xls子目录中所有的文件，如图所示。    （2）在mypython.py文件继续输入程序代码，实现功能：遍历所有文件名，逐个打开工作簿文件，并打开Shee1工作表，读取所有的行记录存入row\_lists数据中，如图所示。    （3）在mypython.py文件继续输入程序代码，实现功能：获取数据的表头标题，获取表中的所有记录，为在后续的操作步骤中以公司名为区分进行数据分割作为数据准备，如图所示。 | | | |  | |
| **作业布置** | 完成思考与练习题。 | | | | | |
| **本课总结** | 本项目学习了Python的openpyxl模块库应用案例，讲述了电子表格工作簿文件的创建、打开、数值的运算、合并单元格、更改单元格边框线样式、文本水平居中、文本垂直居中、字体加粗、读取图片、插入图片、删除单元格、插入单元格、生成图表、一个工作簿文件按内容条件分割成多个文件、多个文件合并成一个文件等操作技能。 | | | | | |
| **教学反思** |  | | | | | |