

高三上信息技术信息系统+pandas 综合题练习 1

1. 小蒋同学为自家的樱桃番茄大棚搭建了一个环境温湿度监测系统，该系统能通过传感器获取樱桃番茄大棚的温度、湿度及光线值，出现异常时，通过执行器发出警报。用户可通过浏览器查看实时数据及历史数据，系统结构如图 a 所示。请回答下列问题：

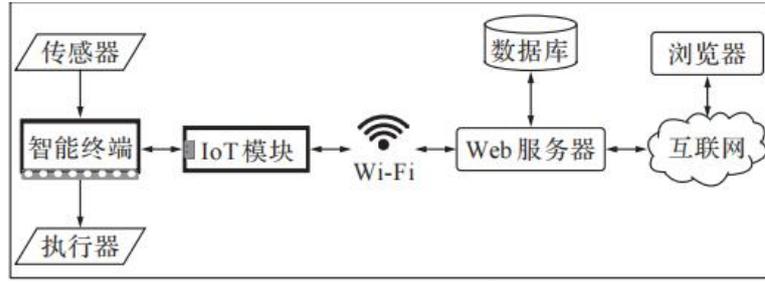


图 a

(1) 该系统中，智能终端与服务器之间的数据传输 C (单选，填字母：A. 只能由智能终端到服务器端 / B. 只能由服务器端到智能终端 / C. 既可以由智能终端到服务器端，也可以由服务器端到智能终端)。

(2) 小蒋基于 Flask Web 框架编写服务器端的程序，部分 Python 程序如下。编写完成后，若要通过浏览器获取视图函数 `index()` 返回的页面，则应访问的 URL 是 <http://192.168.12.136:8080/>

#导入 Flask 框架模块及其他相关模块，代码略

```

app=Flask(__name__)
@app.route("/")
def index():
    '''从数据库读取温度、湿度及光线数据，并返回页面，代码略'''
    #服务器其他功能，代码略
if __name__=="__main__":
    app.run(host="192.168.12.136",port=8080)
    
```

服务端、智能终端代码中
可以分析得到服务器IP地址
一般不能查看智能终端IP地址

(3) 樱桃番茄生长的环境温度 t 应保持在 $18^{\circ}\text{C} \sim 30^{\circ}\text{C}$ 之间 (包含 18°C 和 30°C)，若符合此标准则将系统状态变量 `flag` 设为 0；若低于此标准则将系统状态变量 `flag` 设为 1；若高于此标准则将系统状态变量 `flag` 设为 2。下列 Python 程序段符合要求的有 BD (多选，填字母)。(注：全部选对的得 2 分，选对但不全的得 1 分，不选或有选错的得 0 分)

<p>A.</p> <pre> if t>18 or t<30: flag=0 elif t<=18: flag=1 else: flag=2 </pre>	<p>B.</p> <pre> if not t<18 and t<=30: flag=0 elif t<18: flag=1 else: flag=2 </pre>	<p>C.</p> <pre> if t<18: flag=1 if t>30: flag=2 else: 包含t<18 flag=0 </pre>	<p>D.</p> <pre> flag=0 if t<18: flag=1 if t>30: flag=2 </pre>
---	--	---	---

(4) 将该年 1~10 月的温度数据导出，部分数据如图 b 所示，分析 1~10 月中每天 21 时刻的月平均温度值并绘制线形图如图 c 所示，其中时刻值为数值型。实现上述功能的部分 Python 程序如下，请在划线处填入合适的代码。

(4) 将该年 1~10 月的温度数据导出，部分数据如图 b 所示，分析 1~10 月中每天 21 时刻的月平均温度值并绘制线形图如图 c 所示，其中时刻值为数值型。实现上述功能的部分 Python 程序如下，请在划线处填入合适的代码。

1	日期	时刻	温度
2	2023-01-01	5	17
3	2023-01-01	8	18
4	2023-01-01	11	28
5	2023-01-01	14	30
6	2023-01-01	17	20
7	2023-01-01	21	13
8	2023-01-01	23	14
201	2023-10-31	7	17
202	2023-10-31	10	27
203	2023-10-31	11	29
204	2023-10-31	14	30
205	2023-10-31	17	19
206	2023-10-31	21	14
207	2023-10-31	23	12

图 b

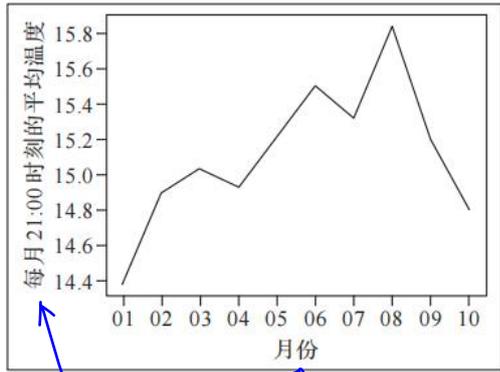


图 c

```

#其他代码略
df=pd.read_excel("data.xlsx")
df["月份"]=0 年-月-日 → 月 为groupby做准备
for i in df.index:
    df.at[i,"月份"]=df.at[i,"日期"][5:7]
#筛选出每天 21 时刻的数据
dft=df[① df.时刻==21 或 df["时刻"]==21]
g=dft.groupby("月份").mean()
plt.plot(② g.index,g.温度)
plt.ylabel("每月 21:00 时刻的平均温度")
plt.xlabel("月份")
plt.show()
    
```



2. 为了简化借书的繁琐流程，提升图书馆服务水平，学校构建智慧图书系统并采购了智慧图书电子借阅机，其外形如图所示，方便师生 24 小时自助借还图书。其中借书流程为：把校园卡放置在“读卡区”，点击“借书”按钮后，把书籍放置在“图书感应区”，系统读取电子标签信息后，即可完成借书。同时图书馆服务器也将该借书流水等信息存入数据库中。

使用智能终端及 Web 服务器模拟智慧图书系统，实现图书查询子功能：把书籍放置在图书感应设备上，感应设备读取到图书电子标签的唯一编号后将该编号数据通过 Wi-Fi 传输到 Web 服务器，并通过该唯一编号在 books.db 数据库的 brrow 表中查询到该书籍的状态及借阅日期（假设仅存储了 2023 年的数据）。系统结构示意图如图 a 所示，查询页面如图 b 所示。请回答下列问题：

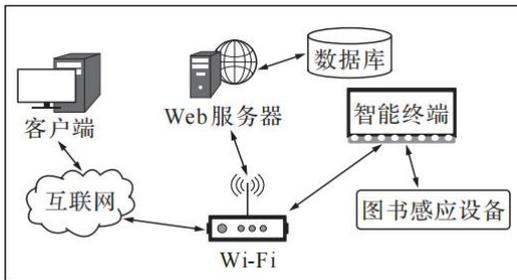


图 a



图 b

(1) 下列关于该智慧图书系统的说法，正确的是 **BD** (多选，填字母)。(注：全部选对的得 2 分，选对但不全的得 1 分，不选或有选错的得 0 分)

- A. 该系统需要在客户端安装配套 App
- B. 智能终端连接 IoT 模块进行数据传输
- C. 数据库是服务器的外接数据存储设备 **x**
- D. 设计如图 b 所示用户界面属于详细设计阶段

(2) 该系统实现查询功能的部分服务器端程序如下，请在划线处填入合适的代码。

数据库是软件
数据库文件是数据，
可以存在服务器上，
也可以存在其他地方

编号	书名	状态	借阅日期
2021030405	红楼梦	外借	2023年3月7日

概要设计中提及的界面：
界面友好

图 c

```

@app.route("/search")
def get_bh():
    days=[31,28,31,30,31,30,31,31,30,31,30,31]
    id=request.args.get("bbh")
    conn=sqlite3.connect("books.db")
    #数据库中的数据组织如图 c 所示
    '''在数据库中查询编号为 id 的图书信息存入变量 rslt 中, 代码略'''
    #查询“年”在字符串中的索引位置
    t1=rslt[0][3].find("年")
    t2=rslt[0][3].find("月")
    m0=int(rslt[0][3][t1+1:t2])
    d0=int(rslt[0][3][t2+1:-1])
    '''获取查询当天日期, 年份为 2023 年, 月份、日期存入整数变量 m1 和 d1 中, 代码略'''
    sum=d1-d0
    for m in range(m0,m1):
        sum+=days[m-1]
    if sum>30: #以 30 天为借阅期限
        rslt[0][2]+=" (逾期)"
    else:
        rslt[0][2]+=" (正常)"
    return render_template("result.html", data=rslt[0])
    
```

flask要读取数据库
第1步: 建立连接
数据库的名称, 带扩展名
select * form 表名称 where id=="xxxx"
rslt
月份
日
#借阅当天不计数 此处的值可能是正, 也可能是负
叠加每月天数 后一定是正

编号	书名	状态	借阅日期
2021030405	红楼梦	外借	2023年3月7日



图 d

(3) 图 b 所示网页的模板文件名是 result.html。
 (4) 使用客户端打开浏览器, 在地址栏中输入 http://192.168.3.10:8081/intro 后, 出现错误如图 d 所示, 请尝试分析原因 该服务器上没有该路由; 路由地址输入错误 说明在最后一页
 3. 小王收集某超市 2023 年全年超市流水数据, 每月的流水数据分别保存在以 6 位日期字符串命名的 Excel 文件中, 部分文件如图 a 所示, 每个文件记录了当月的销售流水数据, 示例如图 b 所示。

图 a

- 202301.xlsx
- 202302.xlsx
- 202303.xlsx
- 202304.xlsx
- 202305.xlsx
- 202306.xlsx
- 202307.xlsx
- 202308.xlsx
- 202309.xlsx

图 b

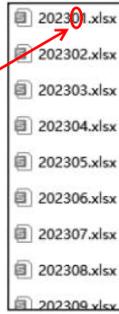
销售时间	商品名称	销售单价	销售数量	销售金额
2023/1/1	...	1.7	1	
2023/1/2	...	4.4	1	
2023/1/3	...	86.9	1	
2023/1/4	...	7.2	1	
2023/1/5	...	5.6	1	
2023/1/6	...	4.3	1	
2023/1/7	...	1.5	2	
2023/1/8	...	5	1	
2023/1/28	...	4.3	1	
2023/1/29	...	4.3	1	
2023/1/30	...	5.2	1	

为统计分析超市全年各商品销售额情况, 编写 Python 程序。请回答下列问题:
 (1) 定义 s_month 函数, 功能为: 读取某月的 Excel 文件并返回当月各商品销售金额总和。该函数代码如下, 划线处应填入的代码为 AD (多选, 填字母)。(注: 全部选对的得 2 分, 选对但不全的得 1 分, 不选或有选错的得 0 分)
 A. df.销售单价*df.销售数量 ✓
 B. df[["销售单价", "销售数量"]].sum()
 C. df[df["销售单价"]*df["销售数量"]] ✓
 D. df["销售单价"]*df["销售数量"] ✓
 import pandas as pd
 def s_month(dayfile):
 df=pd.read_excel(dayfile)
 df["销售金额"]=
 dfg_sum=df.groupby("商品名称", as_index=False).销售金额.sum()
 #返回当月各商品的销售总额
 return dfg_sum

(2) 统计 2023 年销售总额前 3 的商品 (若销售总额相同则一起输出) 并绘制柱形图, 部分 Python 程序如下,

请在划线处填入合适的代码。

```
import matplotlib.pyplot as plt
def mstr(t):
    if t<10:
        return "0"+str(t)
    else:
        return str(t)
df=s_month("202301.xlsx")
```



销售时间	商品名称	销售单价	销售数量	销售金额
2023/1/1	福临门东北	1.7	1	
2023/1/2	福临门东北	4.4	1	
2023/1/3	福临门东北	86.9	1	
2023/1/4	福临门东北	7.2	1	
2023/1/5	福临门东北	5.6	1	
2023/1/6	福临门东北	4.3	1	
2023/1/7	福临门东北	1.5	2	
2023/1/8	福临门东北	5	1	
2023/1/28	福临门东北	4.3	1	
2023/1/29	福临门东北	4.3	1	
2023/1/30	福临门东北	5.2	1	

```
for i in range(1,12):
    ① mthstr=mstr(i+1) 将2~12月的记录加入到1月记录中, i是1, 所以要i+1
    monfile="2023"+mthstr+".xlsx"
    df1=s_month(monfile)
    df=pd.concat([df,df1]) #合并两个 DataFrame 对象数据 1~12月份的数据全合并在一起
data_goods={}
for goods in df.values:
    if goods[0] in data_goods:
        data_goods[goods[0]]+=goods[1]
    else: data_goods[goods[0]]=goods[1]
    ②
'''将字典 data_goods 转换成二维列表 list_goods, 形如[["牛奶棒棒糖",7.8],...] , 代码略'''
d_pm=[1]*len(list_goods)
for i in range(len(list_goods)):
    for j in range(len(list_goods)):
        if list_goods[i][1]<list_goods[j][1]:
            d_pm[i]+=1
            ③ d_pm[下标]=排名 (按降序排名)
x=[];y=[]
for i in range(len(d_pm)):
    if ③ d_pm[i]<=3 : 找排名前三的
        x.append(list_goods[i][0])
        y.append(list_goods[i][1])
```

plt.bar(x,y)
#设置绘图参数, 显示:

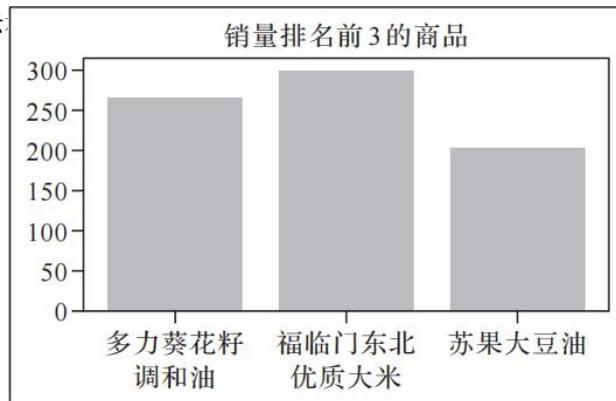


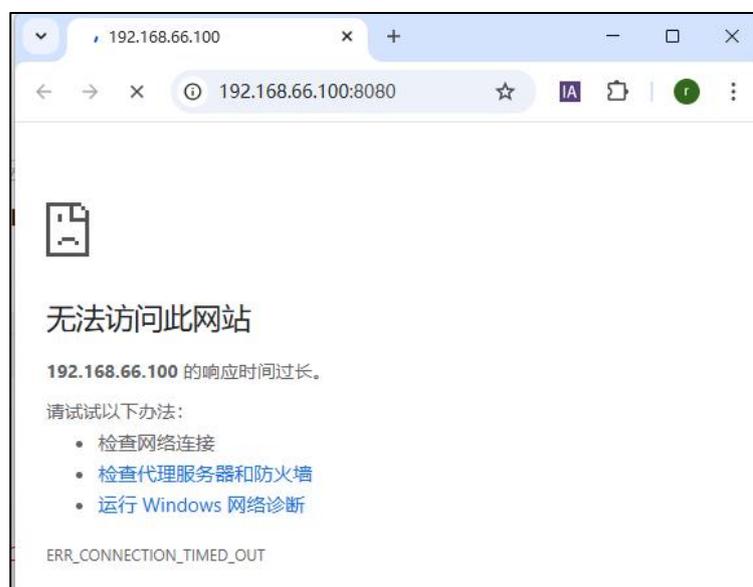
图 c

(3) 由图 c 可知, 2023 年销售总额第一名的商品是 B (单选, 填字母: A. 多力葵花籽调和油 / B. 福临门东北优质大米 / C. 苏果大豆油)。

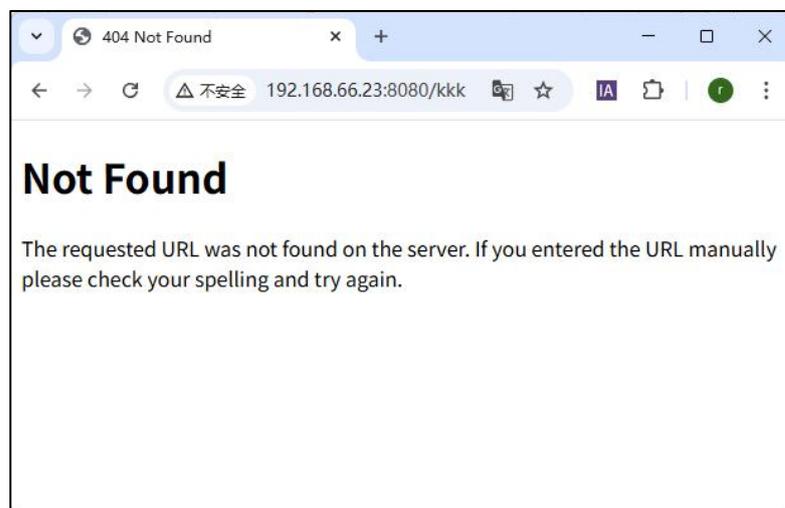
1. 输入正确 ip 地址、端口、路由：可以访问



2 输入不存在的网址或者服务器 IP 地址，或者网址、IP 地址正确但端口错误，显示：



3. 如 IP 地址、端口正确，网址中的路由不存在，显示



4. 不存在 flask 中路由已定义，但是对应的视图函数未定义的情况（python 代码无法会无法运行）