

第八次作业内容

0 / 3

编程题

1.# 请将"_"处替换成能实现注释中功能的代码。

示例:

#设置列表lst中位置3的值为1

lst_

根据注释中说明可知, 此处"_"应替换成[3]=1,即有

lst[3]=1

请先将下面的代码复制到答题框, 注意添加缩进

导入相关的包

import pandas as pd

import numpy as np

from numpy import nan as NA

def fact():

answer = []

df1 = pd.DataFrame({'key': ['b', 'b', 'a', 'c', 'a', 'a', 'b'], 'data1': range(7)})

df2 = pd.DataFrame({'key': ['a', 'b', 'd'], 'data2': range(3)})

指定key这一列取交集

dataframe1 = _

answer.append(dataframe1)

取df1和df2的交集

dataframe2 = _

answer.append(dataframe2)

取左连接, df1左连接df2

dataframe3 = _

answer.append(dataframe3)

df3 = pd.DataFrame({"lkey": ["b", "b", "a", "c", "a", "a", "b"], "data1": range(7)})

df4 = pd.DataFrame({"rkey": ["a", "b", "d"], "data2": range(3)})

```
# 取df3, df4的交集
dataframe4 = _

answer.append(dataframe4)

s1 = pd.Series([0, 1], index=["a", "b"])
s2 = pd.Series([2, 3, 4], index=["c", "d", "e"])
s3 = pd.Series([5, 6], index=["f", "g"])

# 将多个Series拼接成一个DataFrame,即一个Series就是DataFrame的一系列数据
dataframe5 = _

answer.append(dataframe5)

df5 = pd.DataFrame({"a": [1, NA, 5, NA], "b": [NA, 2, NA, 6], "c": range(2, 18, 4)})
df6 = pd.DataFrame({"a": [5, 4, NA, 3, 7], "b": [NA, 3, 4, 6, 8]})

# 用df6的数据为df5中的数据打补丁
dataframe6 = _

answer.append(dataframe6)

data = pd.DataFrame(np.arange(6).reshape(2, 3), index=pd.Index(["上海", "北京"],
name="省份"),
columns=pd.Index([2011, 2012, 2013], name="年份"))
# 将data的列索引转换到行索引
result1 = _

answer.append(result1)

# 将result1的行索引转化为列索引
result2 = _

answer.append(result2)

# 将result1的行索引转化为列索引, 指定要转化为层次化索引的名称为"省份"
result3 = _

answer.append(result3)

data1 = pd.DataFrame({"k1": ["one"] * 3 + ["two"] * 4, "k2": [1, 1, 2, 3, 3, 4, 4]})
# 使用DataFrame的内置函数去除重复数据, 默认保留第一次出现的值
result4 = _

answer.append(result4)

return answer
```

你的答案:

```

data = pd.DataFrame(np.arange(6).reshape(2, 3), index=pd.Index(["上海",
"北京"], name="省份"),
                    columns=pd.Index([2011, 2012, 2013], name="年份"))
# 将data的列索引转换到行索引
result1 = data.stack()

answer.append(result1)

# 将result1的行索引转化为列索引
result2 = result1.unstack()

answer.append(result2)

# 将result1的行索引转化为列索引，指定要转化为层次化索引的名称为"省份"

```

2.# 请将 "_" 处替换成能实现注释中功能的代码。

示例:

#设置列表lst中位置3的值为1

lst_

根据注释中说明可知，此处 "_" 应替换成 [3]=1, 即有

lst[3]=1

请先将下面的代码复制到答题框（注意添加缩进）：

从pandas库导入Series, DataFrame

from pandas import Series, DataFrame

def fact():

answer = []

创建字典data

```

data = {'state': ['Ohio', 'Ohio', 'Ohio', 'Nevada', 'Nevada'],
'year': [2000, 2001, 2002, 2001, 2002],
'pop': [1.5, 1.7, 3.6, 2.4, 2.9]}

```

创建列表col, index1

col = ['year', 'state', 'pop', 'debt']

index1 = ['one', 'two', 'three', 'four', 'five']

利用data创建DataFrame对象frame1, 并指定该对象列为col, 索引为index1

```

frame1 = _
answer.append(frame1.copy())

```

排序

根据索引，对frame1进行降序排序，并指定轴为1

```

frame2 = frame1._
answer.append(frame2.copy())

```

根据值，对frame1的year列进行排序(升序) 并打印

```

frame3 = frame1._
answer.append(frame3.copy())

```

处理缺失数据

对于frame1，只要有某行有NaN就全部删除

```
frame4 = frame1._  
answer.append(frame4.copy())  
  
# 对于frame1, 某行全部为NaN才删除  
frame5 = frame1._  
answer.append(frame5.copy())  
  
# 填充缺失数据  
# 对于frame1, 将元素为NaN替换成0  
frame6 = frame1._  
answer.append(frame6.copy())  
  
return answer
```

你的答案:

```
from pandas import Series, DataFrame  
  
def fact():  
    answer = []  
# 创建字典data  
    data = {'state': ['Ohio', 'Ohio', 'Ohio', 'Nevada', 'Nevada'],  
            'year': [2000, 2001, 2002, 2001, 2002],  
            'pop': [1.5, 1.7, 3.6, 2.4, 2.9]}  
  
# 创建列表col, index1  
    col = ['year', 'state', 'pop', 'debt']  
    index1 = ['one', 'two', 'three', 'four', 'five']
```

3.# 请将"_"处替换成能实现注释中功能的代码。

```
# 示例:  
# #设置列表lst中位置3的值为1  
# lst_  
# 根据注释中说明可知, 此处"_"应替换成[3]=1,即有  
# lst[3]=1
```

请先将下面的代码复制到答题框 (注意添加缩进) :

```
# 从pandas库导入Series,DataFrame  
from pandas import Series, DataFrame  
  
def fact():  
    answer = []  
# 创建列表lst  
    lst = [4, 7, -5, 3]  
  
# 使用列表list生成Series对象obj  
    obj =_  
    answer.append(obj.copy())  
  
# 创建数组index  
    index1 = ['d', 'b', 'a', 'c']
```

```
# 创建数据为lst，索引为index1的Series对象obj2
obj2 = _
answer.append(obj2.copy())

# 将obj2中索引值为d对应的值赋值为6
obj2 = _
answer.append(obj2.copy())

# 将obj2中索引值为d对应的值存储到ans1中
ans1 = _
answer.append(ans1)

# 从obj2找出大于0的元素并存储到ans2中
ans2 = _
answer.append(ans2)

# 创建字典sdata
sdata = {'Ohio': 45000, 'Texas': 71000, 'Oregon': 16000, 'Utah': 5000}

# 利用sdata生成Series对象obj3
obj3 = _
answer.append(obj3.copy())

# 创建列表states
states = ['California', 'Ohio', 'Oregon', 'Texas']

# 创建数据为sdata，索引为states的Series对象obj4
obj4 = _
answer.append(obj4.copy())

# 将obj3和obj4进行相加，相同索引部分相加，存储到obj5
obj5 = _
answer.append(obj5.copy())

# 指定obj4的名字为population
obj4 = _
answer.append(obj4.copy())

# 指定obj4的索引的名字为state
obj4 = _
answer.append(obj4.copy())
return answer
```

你的答案：

```
# 使用列表lst生成Series对象obj
    obj = Series(lst)
    answer.append(obj.copy())

# 创建数组index
    index1 = ['d', 'b', 'a', 'c']

# 创建数据为lst, 索引为index1的Series对象obj2
    obj2 = Series(lst, index=index1)
    answer.append(obj2.copy())

# 将obj2中索引值为d对应的值赋值为6
    obj2['d'] = 6
    answer.append(obj2.copy())
```

[返回](#)