

第十一课--Pandas

任务目标

- 1、Pandas数据框
- 2、Pandas数据计数
- 3、数据行列变换
- 4、数据排序
- 5、数据过滤

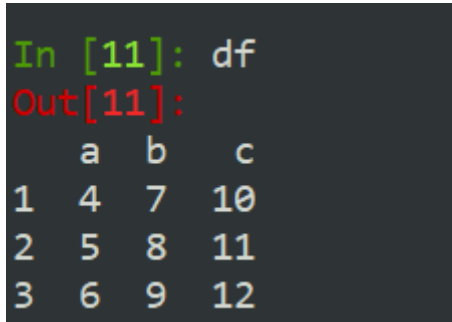
相关知识

- 1、Pandas数据处理

1、Pandas数据框

- 1、按照每一列建立数据框

```
import pandas as pd
df = pd.DataFrame({"a": [4,5,6], "b": [7,8,9], "c": [10,11,12]}, index=[1,2,3], columns=['a', 'b', 'c'])
```



```
In [11]: df
Out[11]:
```

	a	b	c
1	4	7	10
2	5	8	11
3	6	9	12

- 2、按照每一行建立数据框

```
df = pd.DataFrame([[4, 7, 10], [5, 8, 11], [6, 9, 12]], index=[1, 2, 3], columns=['a', 'b', 'c'])
```

- 4、建立多索引数据框

```
df = pd.DataFrame({"a" : [4 ,5, 6], "b" : [7, 8, 9], "c" : [10, 11, 12]}, index = pd.MultiIndex.from_tuples([('d',1), ('d',2), ('e',2)], names=['n', 'v']))
```

		a	b	c
n	v			
d	1	4	7	10
	2	5	8	11
e	2	6	9	12

2、Pandas数据计数

```
df['a'].value_counts() #变量计数
len(df)               #变量的长度
df['a'].nunique()      #唯一的变量
df.describe()         #描述
```

```
Out[17]:
6    1
5    1
4    1
Name: a, dtype: int64
```

3、数据的行列变换

```
df1 = pd.DataFrame([[4,7,7],[5,6,7]],columns=['1','2','3'])
df2 = pd.DataFrame([[1,2,3],[2,3,4]],columns=['1','2','3'])
df = pd.concat([df1,df2])          #按行合并
pd.melt(df)
df.pivot(columns='1',values='2')
df = pd.concat([df1,df2],axis=1) #按列合并
```

4、数据排序

```
In [15]: df
Out[15]:
   1  2  3
0  4  7  7
1  5  6  7
0  1  2  3
1  2  3  4
df.sort_values('1')          #升序
df.sort_values('1',ascending=False) #降序
df.rename(columns={'1':'number'}) #列名修改
df.sort_index()              #根据索引进行排序
Out[21]:
   1  2  3
0  4  7  7
0  1  2  3
1  5  6  7
1  2  3  4
df = df.reset_index()        #增加索引列数据
Out[22]:
   index  1  2  3
0       0  4  7  7
```

```

1      1  5  6  7
2      0  1  2  3
3      1  2  3  4
df.drop(columns=['index'])    #删除index列数据

```

5、数据过滤

```

df = pd.DataFrame(
    [[4, 7, 10],
     [5, 8, 11],
     [6, 9, 12]],
    index=[1, 2, 3],
    columns=['a', 'b', 'c'])

df[df.a > 2]

df = pd.DataFrame([[4, 7, 10], [4, 7, 10], [6, 9, 12], [1, 2, 3], [4, 5, 6], [16, 9,
12], [14, 27, 10], [14, 17, 20], [16, 1, 22]], columns=['a', 'b', 'c'])
df.drop_duplicates() #删除重复值
df.head(6);
df.tail(6);
df.sample(frac=0.5)    #随机采样
df.sample(n=5)         #随机采样5行数据
df.iloc[3:5]
df.nlargest(3, 'a')    #选择'a'列最大的3个值,
df[['a', 'b']]         #选择多列
df.loc[:, 'a': 'b']
df.iloc[:, [1, 2]]     #选择第二列、第三列
df.loc[df['a'] > 10, ['a', 'c']]

```

6、数据统计

```

df.sum()
df.count()
df.median()
quantile([0.25, 0.75])
apply(function)
min()
max()
mean()
var()
std()

```

7、数据分组

```

In [100]: df.groupby('c').agg('mean')
Out[100]:
          a          b
c
3    1.000000    2.000000
6    4.000000    5.000000
10   7.333333   13.666667
12  11.000000    9.000000
20  14.000000   17.000000

```

```
22 16.000000 1.000000
```

```
df.shift(1)
```

```
Out[95]:
```

	a	b	c
0	NaN	NaN	NaN
1	4.0	7.0	10.0
2	4.0	7.0	10.0
3	6.0	9.0	12.0
4	1.0	2.0	3.0
5	4.0	5.0	6.0
6	16.0	9.0	12.0
7	14.0	27.0	10.0
8	14.0	17.0	20.0

```
In [97]: df.shift(-1)
```

```
Out[97]:
```

	a	b	c
0	4.0	7.0	10.0
1	6.0	9.0	12.0
2	1.0	2.0	3.0
3	4.0	5.0	6.0
4	16.0	9.0	12.0
5	14.0	27.0	10.0
6	14.0	17.0	20.0
7	16.0	1.0	22.0
8	NaN	NaN	NaN

8、处理缺失值

```
df1 = pd.DataFrame({'a':[1,2,3], 'b':[4,5,6], 'c':[51,np.nan ,23]})
```

```
df1.dropna() #删除缺失值
```

```
df1.fillna(5) #填充缺失值
```