

# 第四部分 Pandas数据分析基础（餐饮小费数据分析）

## 4-1 Pandas数据结构

```
# 创建series
import numpy as np
import pandas as pd
# 从数组创建
ser = pd.Series(np.array([18,25,30,21,19]))
print(ser)
```

```
0    18
1    25
2    30
3    21
4    19
dtype: int32
```

```
print(type(ser))
```

```
<class 'pandas.core.series.Series'>
```

```
print(ser[1])
print(ser[1:3])
```

```
25
1    25
2    30
dtype: int32
```

```
# 从列表创建
ser2 = pd.Series(["M","F","F","M","F"])
print(ser2)
```

```
0    M
1    F
2    F
3    M
4    F
dtype: object
```

```
# 创建DataFrame
import pandas as pd
df = pd.DataFrame({"Age":ser,"Sex":ser2})
df
```

	Age	Sex
0	18	M
1	25	F
2	30	F
3	21	M
4	19	F

```
dict = {"Name":["张三","李四","王五","赵六","钱七"],
        "Age":[18,25,30,21,19],
        "Sex":["男","女","女","男","女"]}
df = pd.DataFrame(dict)
df
```

	Name	Age	Sex
0	张三	18	男
1	李四	25	女
2	王五	30	女
3	赵六	21	男
4	钱七	19	女

## 项目步骤：读取餐饮小费数据创建DataFrame

```
# 从文本文件创建
import pandas as pd
# 两种方法读取csv文件
#tips = pd.read_table("data/tips.csv", delimiter=",")
tips = pd.read_csv("data/tips.csv")
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

```
# DataFrame的每一列就是一个Series
print(tips["tip"])
```

```
0      1.01
1      1.66
2      3.50
3      3.31
4      3.61
...
239    5.92
240    2.00
241    2.00
242    1.75
243    3.00
Name: tip, Length: 244, dtype: float64
```

## 4-2 DataFrame的数据查看

```
# 创建DataFrame
import pandas as pd
df = pd.DataFrame({"Name": ["张三", "李四", "王五", "赵六", "钱七"],
                  "Age": [18, 25, 30, 21, 19],
                  "Sex": ["男", "女", "女", "男", "女"]})
df
```

	Name	Age	Sex
0	张三	18	男
1	李四	25	女
2	王五	30	女
3	赵六	21	男
4	钱七	19	女

```
# 数据框的常用属性
# 索引
print(df.index)
# 所有值，相当于二维数组
print(df.values)
# 列名
print(df.columns)
# 数据类型
print(df.dtypes)
```

```
RangeIndex(start=0, stop=5, step=1)
[['张三' 18 '男']
 ['李四' 25 '女']
 ['王五' 30 '女']
 ['赵六' 21 '男']
 ['钱七' 19 '女']]
Index(['Name', 'Age', 'Sex'], dtype='object')
Name    object
Age      int64
Sex      object
dtype: object
```

```
## 查看DataFrame的元素个数
print(df.size)
## 查看DataFrame的维度数
print(df.ndim)
## 查看DataFrame的形状
print(df.shape)
## 查看DataFrame的行数
print(len(df))
```

```
15
2
(5, 3)
5
```

```
# 设置行索引
df.index=["zhang","li","wang","zhao","qian"]
df
```

	Name	Age	Sex
zhang	张三	18	男
li	李四	25	女
wang	王五	30	女
zhao	赵六	21	男
qian	钱七	19	女

```
# 取某一列，结果就是Series
# 字典方式
df["Name"]
```

```
zhang    张三
li        李四
wang      王五
zhao      赵六
qian      钱七
Name: Name, dtype: object
```

```
# 属性方式
df.Age
```

```
zhang    18
li        25
wang      30
zhao      21
qian      19
Name: Age, dtype: int64
```

```
# 访问某一列的几行
df["Name"][1:4]
```

```
li        李四
wang      王五
zhao      赵六
Name: Name, dtype: object
```

```
# 访问多列数据
df[["Name", "Age"]]
```

	Name	Age
zhang	张三	18
li	李四	25
wang	王五	30
zhao	赵六	21
qian	钱七	19

- loc方法是针对DataFrame索引名称的切片方法，如果传入的不是索引名称，那么切片操作将无法执行。利用loc方法，能够实现所有单层索引切片操作。loc方法使用方法如下。  
DataFrame.loc[行索引名称或条件, 列索引名称]
- iloc和loc区别是iloc接收的必须是行索引和列索引的位置。iloc方法的使用方法如下。  
DataFrame.iloc[行索引位置, 列索引位置]

```
# loc方式索引和切片
df.loc["li": "zhao", "Name"]
```

```
li      李四
wang    王五
zhao    赵六
Name: Name, dtype: object
```

```
# loc方式索引和切片
df.loc["li", "Name"]
```

```
'李四'
```

```
df.loc["li": "zhao", "Age": "Sex"]
```

	Age	Sex
li	25	女
wang	30	女
zhao	21	男

```
# iloc方式索引和切片
df.iloc[1,0]
```

```
'李四'
```

```
df.iloc[1:3,1:3]
```

	Age	Sex
li	25	女
wang	30	女

```
# 条件切片
df.loc[df.Age<20, "Name"]
```

```
zhang    张三
qian     钱七
Name: Name, dtype: object
```

```
df.loc[df.Age<20,:]
```

	Name	Age	Sex
zhang	张三	18	男
qian	钱七	19	女

## 项目步骤：小费数据的查看和选取

```
# 小费数据集的数据选取
import pandas as pd
tips = pd.read_excel("data/tips.xlsx")
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

```
# 选取小费列
print(tips["tip"].head())
# 属性方式选取
print(tips.tip.head())
```

```
0    1.01
1    1.66
2    3.50
3    3.31
4    3.61
Name: tip, dtype: float64
0    1.01
1    1.66
2    3.50
3    3.31
4    3.61
Name: tip, dtype: float64
```



```
# 取出前两列的前5行数据
print(tips.loc[:5,["total_bill","tip"]])
print(tips.iloc[:5,:2])
```

```
   total_bill  tip
0      16.99  1.01
1      10.34  1.66
2      21.01  3.50
3      23.68  3.31
4      24.59  3.61
5      25.29  4.71
   total_bill  tip
0      16.99  1.01
1      10.34  1.66
2      21.01  3.50
3      23.68  3.31
4      24.59  3.61
```

```
# 取出所有性别为male的行(知识点: 条件选取切片)
tips.loc[tips["sex"]=="Male",:]
```

	total_bill	tip	sex	smoker	day	time	size
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
5	25.29	4.71	Male	No	Sun	Dinner	4
6	8.77	2.00	Male	No	Sun	Dinner	2
...	...	...	...	...	...	...	...
236	12.60	1.00	Male	Yes	Sat	Dinner	2
237	32.83	1.17	Male	Yes	Sat	Dinner	2
239	29.03	5.92	Male	No	Sat	Dinner	3
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2

157 rows × 7 columns

## 4-3 DataFrame的增删改

```
# 生成数据
import pandas as pd
dict = {"Name":["张三","李四","王五","赵六","钱七"],
        "Age":[18,25,30,21,19],
        "Sex":["男","女","女","男","女"]}
df = pd.DataFrame(dict)
df
```

	Name	Age	Sex
0	张三	18	男
1	李四	25	女
2	王五	30	女
3	赵六	21	男
4	钱七	19	女

```
# 王五的性别改为“男”
df.loc[df["Name"]=="王五","Sex"] = "男"
df
```

	Name	Age	Sex
0	张三	18	男
1	李四	25	女
2	王五	30	男
3	赵六	21	男
4	钱七	19	女

```
# 行索引为2（王五）的年龄改为20
df.loc[2,"Age"] = 20
df
```

	Name	Age	Sex
0	张三	18	男
1	李四	25	女
2	王五	20	男
3	赵六	21	男
4	钱七	19	女

```
# 增加一列“City”，设为“深圳”
df["City"] = "深圳"
df
```

	Name	Age	Sex	City
0	张三	18	男	深圳
1	李四	25	女	深圳
2	王五	20	男	深圳
3	赵六	21	男	深圳
4	钱七	19	女	深圳

```
# 删除“钱七”这一行
df.drop(4, inplace=True)
df
```

	Name	Age	Sex	City
0	张三	18	男	深圳
1	李四	25	女	深圳
2	王五	20	男	深圳
3	赵六	21	男	深圳

```
# 删除City列
df.drop("City", axis=1, inplace=True)
df
```

	Name	Age	Sex
0	张三	18	男
1	李四	25	女
2	王五	20	男
3	赵六	21	男

## 项目步骤：餐饮小费数据的修改

```
# 小费数据集的数据修改
import pandas as pd
tips = pd.read_excel("data/tips.xlsx")
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

```
# 将第三行的性别改为“Female”
tips.loc[2,"sex"] = "Female"
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Female	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

```
# 增加一列tip_per, 表示小费在帐单中的占比
import numpy as np
tips["tip_per"] = np.round(tips["tip"] / tips["total_bill"] * 100, 1)
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size	tip_per
0	16.99	1.01	Female	No	Sun	Dinner	2	5.9
1	10.34	1.66	Male	No	Sun	Dinner	3	16.1
2	21.01	3.50	Female	No	Sun	Dinner	3	16.7
3	23.68	3.31	Male	No	Sun	Dinner	2	14.0
4	24.59	3.61	Female	No	Sun	Dinner	4	14.7

## 4-4 DataFrame的统计计算

```
#生成数据
import pandas as pd
dict = {"Name":["张三","李四","王五","赵六","钱七"],
        "Age":[18,25,30,21,19],
        "Sex":["男","女","女","男","女"],
        "chn":[90,85,62,58,75],
        "math":[96,66,73,60,90]}
df = pd.DataFrame(dict)
df
```

	Name	Age	Sex	chn	math
0	张三	18	男	90	96
1	李四	25	女	85	66
2	王五	30	女	62	73
3	赵六	21	男	58	60
4	钱七	19	女	75	90

```
# 描述数据
df.describe()
```

	Age	chn	math
count	5.000000	5.000000	5.000000
mean	22.600000	74.000000	77.000000
std	4.929503	13.946326	15.459625
min	18.000000	58.000000	60.000000
25%	19.000000	62.000000	66.000000
50%	21.000000	75.000000	73.000000
75%	25.000000	85.000000	90.000000
max	30.000000	90.000000	96.000000

```
# 按性别统计人数
df["Sex"].value_counts()
```

```
女    3
男    2
Name: Sex, dtype: int64
```

```
# 平均年龄
print(df["Age"].mean())
```

```
22.6
```

```
# 语文和数学平均分
print(df["chn"].mean())
print(df["math"].mean())
```

```
74.0
77.0
```

```
# 添加一列平均分Aver
df["Aver"] = (df["chn"] + df["math"])/2
df
```

	Name	Age	Sex	chn	math	Aver
0	张三	18	男	90	96	93.0
1	李四	25	女	85	66	75.5
2	王五	30	女	62	73	67.5
3	赵六	21	男	58	60	59.0
4	钱七	19	女	75	90	82.5

## 项目步骤：小费数据的计算

```
# 小费数据集的数据选取
import pandas as pd
tips = pd.read_excel("data/tips.xlsx")
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

```
# 数据描述
tips.describe()
```

	total_bill	tip	size
count	244.000000	244.000000	244.000000
mean	19.785943	2.998279	2.569672
std	8.902412	1.383638	0.951100
min	3.070000	1.000000	1.000000
25%	13.347500	2.000000	2.000000
50%	17.795000	2.900000	2.000000
75%	24.127500	3.562500	3.000000
max	50.810000	10.000000	6.000000

```
# 按性别统计数量
tips["sex"].value_counts()
```

```
Male      157
Female     87
Name: sex, dtype: int64
```

```
# 计算账单和小费的均值和中位数
import numpy as np
print(tips["total_bill"].mean(), np.mean(tips["tip"]))
print(tips["total_bill"].median(), np.median(tips["tip"]))
```

```
19.785942622950824 2.9982786885245902
17.795 2.9
```

```
# 计算男性和女性就餐者给出小费的均值（知识点：pandas的计算功能）
tip_male = tips.loc[tips["sex"]=="Male","tip"].mean()
tip_female = tips.loc[tips["sex"]=="Female","tip"].mean()
print("男性和女性就餐者给出的小费均值分别是：{:.2f}和{:.2f}".format(tip_male,tip_female))
```

```
男性和女性就餐者给出的小费均值分别是：3.09和2.83
```

## 4-5 DataFrame的分组聚合



```
#生成数据
import pandas as pd
dict = {"Name":["张三","李四","王五","赵六","钱七"],
        "Age":[18,25,30,21,19],
        "Sex":["男","女","女","男","女"],
        "chn":[90,85,62,58,75],
        "math":[96,66,73,60,90]}
df = pd.DataFrame(dict)
df
```

	Name	Age	Sex	chn	math
0	张三	18	男	90	96
1	李四	25	女	85	66
2	王五	30	女	62	73
3	赵六	21	男	58	60
4	钱七	19	女	75	90

```
# 按性别分组计算语文和数学平均分
print(df.groupby("Sex")["chn"].mean())
print(df.groupby("Sex")["math"].mean())
```

```
Sex
女    74
男    74
Name: chn, dtype: int64
Sex
女    76.333333
男    78.000000
Name: math, dtype: float64
```

```
# apply函数
df[["chn","math"]].apply(np.mean)
```

```
chn    74.0
math   77.0
dtype: float64
```

```
# 将分数变为10分制
df[["chn", "math"]].apply(lambda x: np.round(x / 10))
```

	chn	math
0	9.0	10.0
1	8.0	7.0
2	6.0	7.0
3	6.0	6.0
4	8.0	9.0

```
# agg函数
df[["chn", "math"]].agg([np.mean, np.sum])
```

	chn	math
mean	74.0	77.0
sum	370.0	385.0

## 项目步骤：小费数据的分组统计

```
# 小费数据集的数据选取
import pandas as pd
tips = pd.read_excel("data/tips.xlsx")
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

```
# 按性别分组统计就餐人数的平均值
tip_grp_sex = tips.groupby(by="sex")
print(tip_grp_sex["size"].mean())
```

```
sex
Female    2.459770
Male      2.630573
Name: size, dtype: float64
```

```
# 按是否吸烟分组统计账单和小费的平均值
tip_grp_smoker = tips.groupby(by="smoker")
print(tip_grp_smoker["total_bill", "tip"].mean())
```

```
      total_bill      tip
smoker
No          19.188278  2.991854
Yes         20.756344  3.008710
```

```
<ipython-input-55-3e71ca1540ec>:3: FutureWarning: Indexing with multiple keys
(implicitly converted to a tuple of keys) will be deprecated, use a list
instead.
print(tip_grp_smoker["total_bill", "tip"].mean())
```

```
# agg函数的应用
tips[["total_bill", "tip"]].agg([np.sum, np.mean, np.std])
```

	total_bill	tip
sum	4827.770000	731.580000
mean	19.785943	2.998279
std	8.902412	1.383638

```
# agg函数的应用
tips.agg({"total_bill": np.mean, "tip": np.sum})
```

```
total_bill    19.785943
tip           731.580000
dtype: float64
```

## 4-6 DataFrame的交叉表和透视表

```
#生成数据
import pandas as pd
dict = {"Name":["张三","李四","王五","赵六","钱七"],
        "Age":[18,25,30,21,19],
        "Sex":["男","女","女","男","女"],
        "Class":[1,1,2,2,1],
        "chn":[90,85,62,58,75],
        "math":[96,66,73,60,90]}
df = pd.DataFrame(dict)
df
```

	Name	Age	Sex	Class	chn	math
0	张三	18	男	1	90	96
1	李四	25	女	1	85	66
2	王五	30	女	2	62	73
3	赵六	21	男	2	58	60
4	钱七	19	女	1	75	90

```
# 以性别和班级形成交叉表
pd.crosstab(df["Sex"],df["Class"],margins=True)
```

Class	1	2	All
Sex			
女	2	1	3
男	1	1	2
All	3	2	5

```
# 透视表（按性别和班级求语文数学平均分）
import numpy as np
pd.pivot_table(data=df, values=["chn", "math"], index=["Sex", "Class"],
aggfunc=np.mean)
```

		chn	math
Sex	Class		
女	1	80	78
	2	62	73
男	1	90	96
	2	58	60

# 透视表

```
pd.pivot_table(data=df, values=["chn", "math"], index=["Sex"], columns=
["Class"], aggfunc=np.mean)
```

	chn		math	
Class	1	2	1	2
Sex				
女	80	62	78	73
男	90	58	96	60

## 项目步骤：小费数据的交叉表和透视表

# 小费数据集的数据选取

```
import pandas as pd
tips = pd.read_excel("data/tips.xlsx")
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

```
# 性别和是否吸烟者数量的交叉统计（知识点：交叉表crosstab）
tips_sex_smoker = pd.crosstab(tips.sex, tips.smoker, margins=True)
tips_sex_smoker
```

	smoker	No	Yes	All
sex				
Female		54	33	87
Male		97	60	157
All		151	93	244

```
# 统计按性别和是否吸烟分类的账单和小费的均值（知识点：透视表pivot_table）
pivot = pd.pivot_table(data=tips, values=["total_bill", "tip"], index=["sex", "smoker"], aggfunc=np.mean, margins=True)
pivot
```

		tip	total_bill
sex	smoker		
Female	No	2.773519	18.105185
	Yes	2.931515	17.977879
Male	No	3.113402	19.791237
	Yes	3.051167	22.284500
All		2.998279	19.785943

## 4-7 Pandas绘图

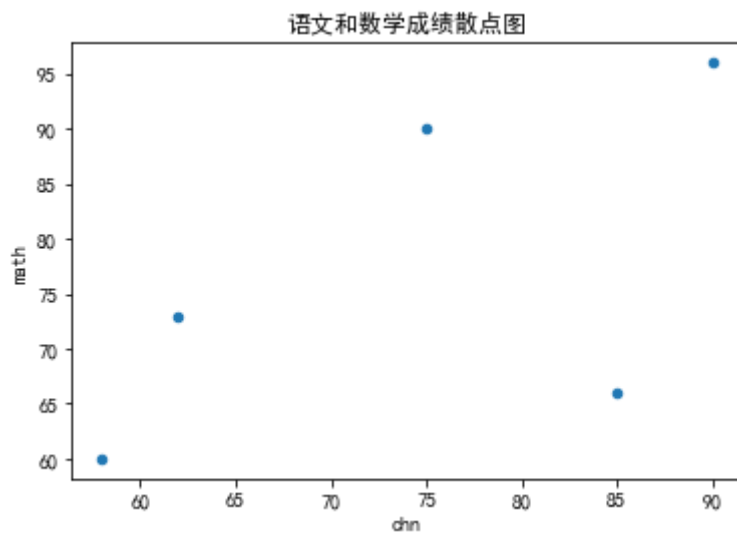
```
#生成数据
import pandas as pd
dict = {"Name":["张三","李四","王五","赵六","钱七"],
        "Age":[18,25,30,21,19],
        "Sex":["男","女","女","男","女"],
        "Class":[1,1,2,2,1],
        "chn":[90,85,62,58,75],
        "math":[96,66,73,60,90]}
df = pd.DataFrame(dict)
df
```

	Name	Age	Sex	Class	chn	math
0	张三	18	男	1	90	96
1	李四	25	女	1	85	66
2	王五	30	女	2	62	73
3	赵六	21	男	2	58	60
4	钱七	19	女	1	75	90

```
import matplotlib.pyplot as plt
## 设置字体为SimHei显示中文
plt.rcParams["font.sans-serif"] = "SimHei"
## 设置正常显示符号
plt.rcParams["axes.unicode_minus"] = False
```

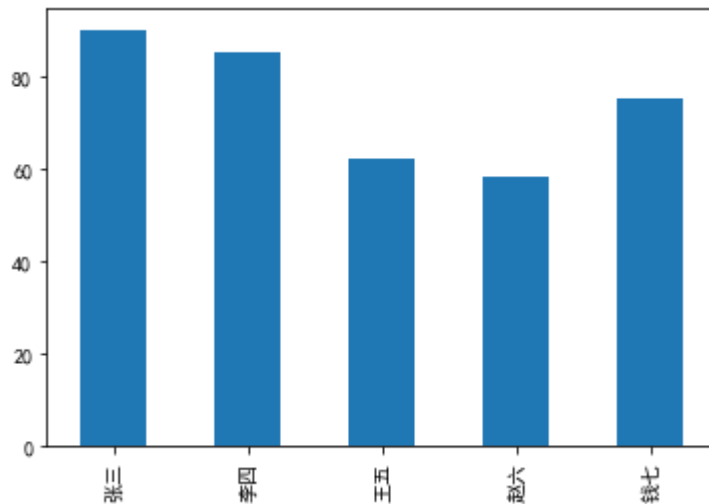
```
# 绘制语文和数学关系的散点图
df.plot("chn", "math", kind="scatter")
plt.title("语文和数学成绩散点图")
```

```
Text(0.5, 1.0, '语文和数学成绩散点图')
```



```
# 语文成绩条形图
df["chn"].plot(kind="bar")
plt.xticks(range(5), df["Name"])
```

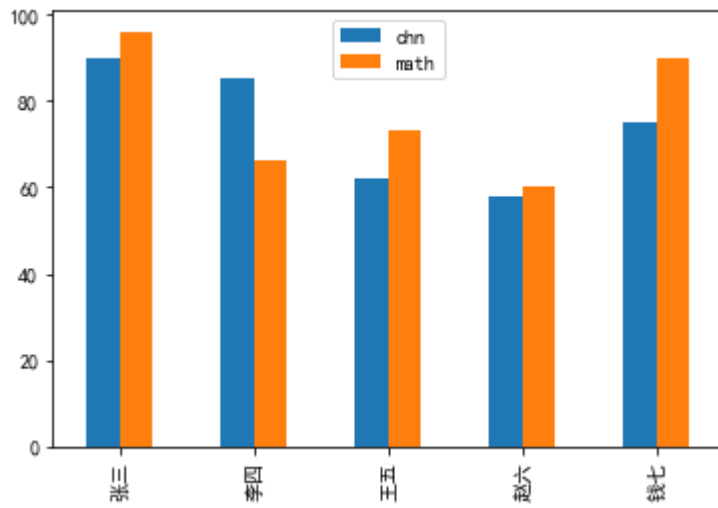
```
([<matplotlib.axis.XTick at 0x2204844d340>,
  <matplotlib.axis.XTick at 0x2204844d310>,
  <matplotlib.axis.XTick at 0x2204846d820>,
  <matplotlib.axis.XTick at 0x2204847da90>,
  <matplotlib.axis.XTick at 0x2204847dfa0>],
 [Text(0, 0, '张三'),
  Text(1, 0, '李四'),
  Text(2, 0, '王五'),
  Text(3, 0, '赵六'),
  Text(4, 0, '钱七')])
```



```
# 语文数学成绩条形图
df[["chn", "math"]].plot(kind="bar")
plt.xticks(range(5), df["Name"])
```

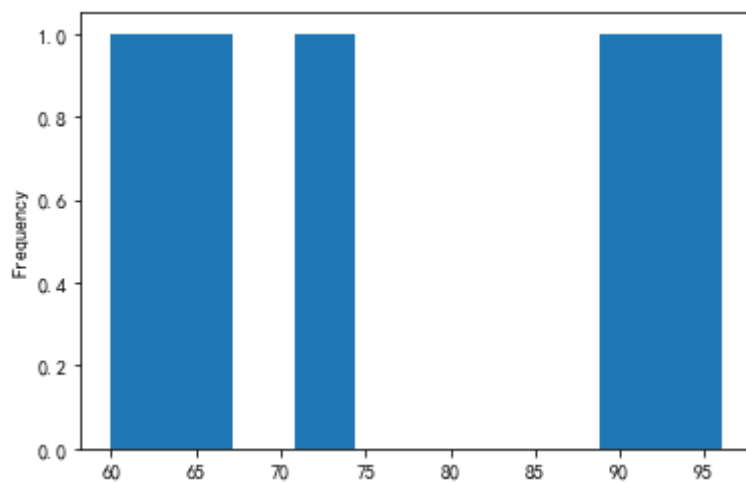
```
([<matplotlib.axis.XTick at 0x2204846d7c0>,
  <matplotlib.axis.XTick at 0x220484b72b0>,
  <matplotlib.axis.XTick at 0x220484e5760>,
  <matplotlib.axis.XTick at 0x220485062b0>,
  <matplotlib.axis.XTick at 0x220485067c0>],
 [Text(0, 0, '张三'),
  Text(1, 0, '李四'),
  Text(2, 0, '王五'),
  Text(3, 0, '赵六'),
  Text(4, 0, '钱七')])
```





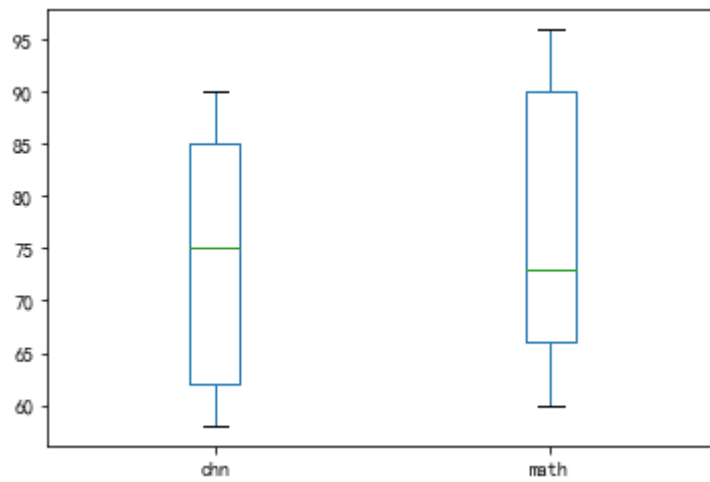
```
# 数学成绩直方图
df["math"].plot(kind="hist")
```

```
<AxesSubplot:ylabel='Frequency'>
```



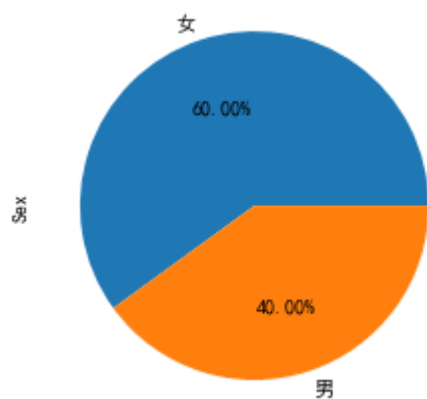
```
# 语文数学成绩的箱型图
df[["chin", "math"]].plot(kind="box")
```

```
<AxesSubplot:>
```



```
# 男女生人数饼图
plt.figure(figsize=(4,4))
df["Sex"].value_counts().plot(kind="pie", autopct="%3.2f%%")
```

```
<AxesSubplot:ylabel='Sex'>
```



## 项目步骤：小费数据绘图分析

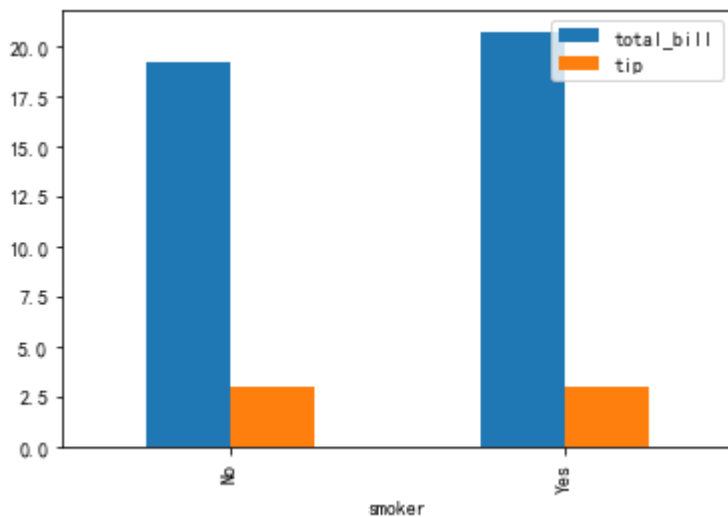
```
# 小费数据集
import pandas as pd
tips = pd.read_excel("data/tips.xlsx")
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

# 以是否吸烟分组

```
smoker_group = tips.groupby(by="smoker")
smoker_group[["total_bill", "tip"]].mean()
smoker_group[["total_bill", "tip"]].mean().plot(kind="bar")
```

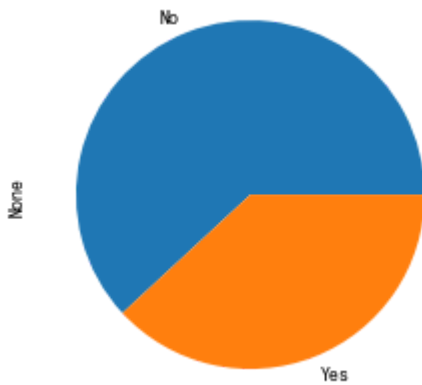
```
<AxesSubplot:xlabel='smoker'>
```



# 是否吸烟的人数对比

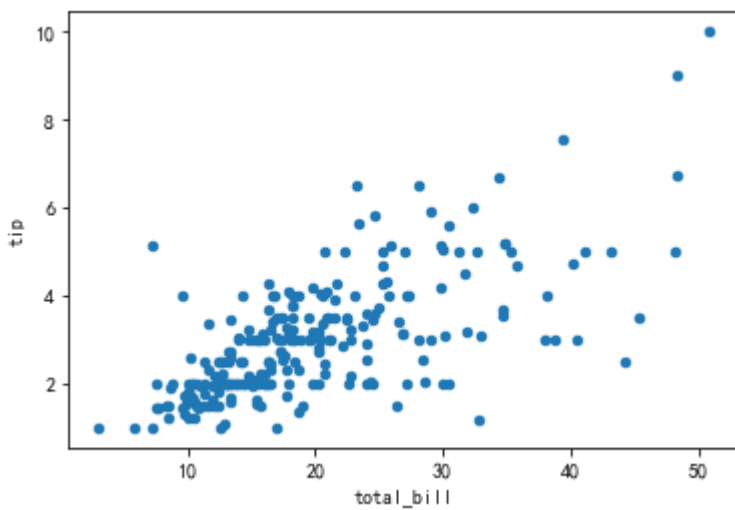
```
smoker_group.size().plot(kind="pie")
```

```
<AxesSubplot:ylabel='None'>
```



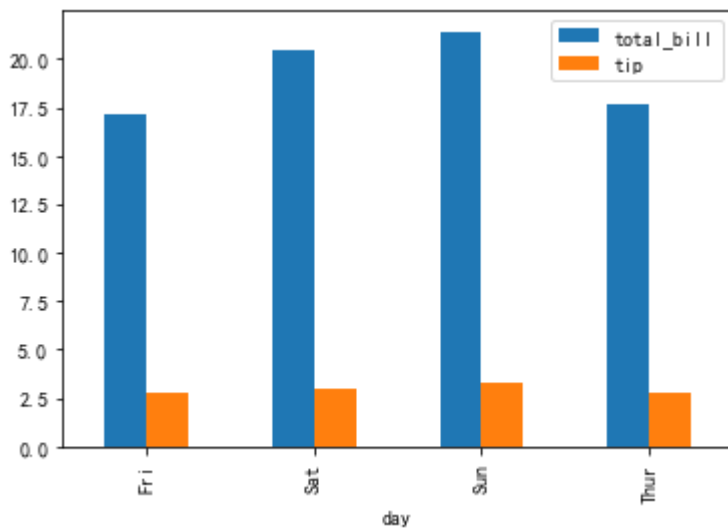
```
# 绘散点图：账单和小费的关系
tips.plot(kind="scatter",x="total_bill", y="tip")
```

```
<AxesSubplot:xlabel='total_bill', ylabel='tip'>
```



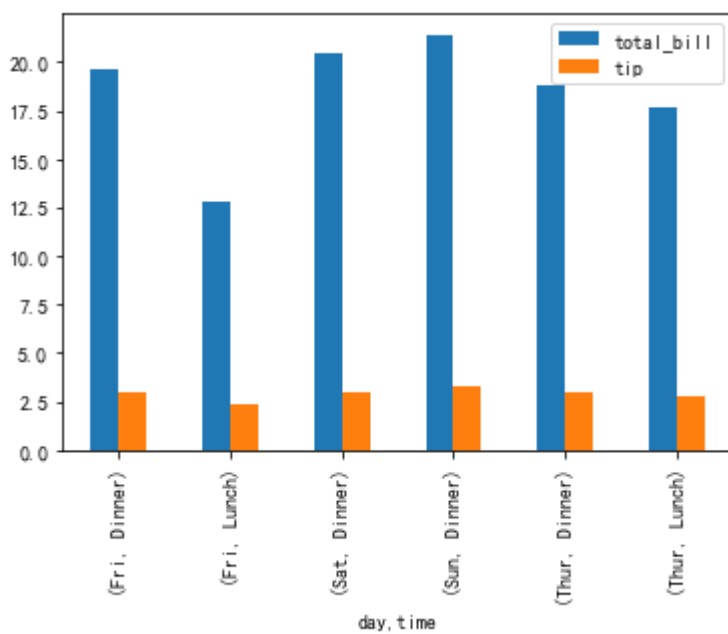
```
# 以tips数据为例，计算按就餐星期值分组的账单均值和小费均值以及就餐桌数。以此来分析周末对就餐和小费的影响。
day_grp = tips.groupby(by="day")
day_grp[["total_bill","tip"]].mean().plot(kind="bar")
```

```
<AxesSubplot:xlabel='day'>
```



```
day_grp = tips.groupby(by=["day","time"])
day_grp[["total_bill","tip"]].mean().plot.bar()
```

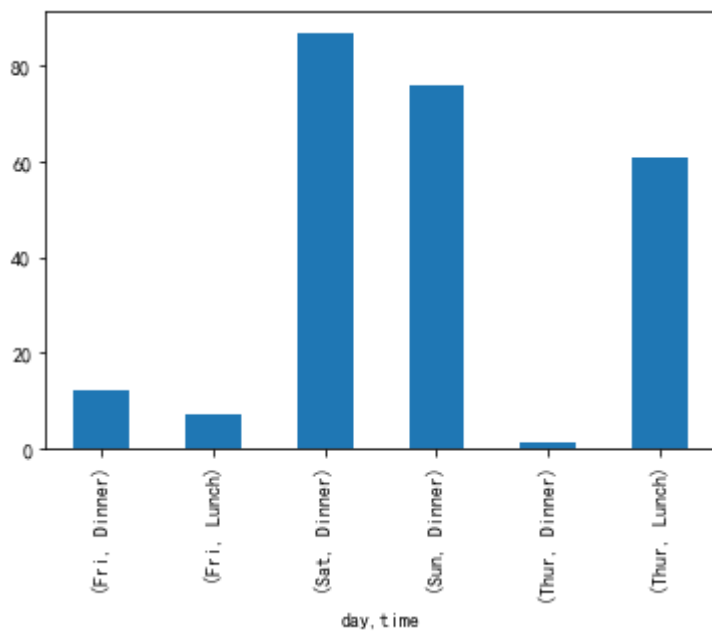
```
<AxesSubplot:xlabel='day,time'>
```



```
# 就餐人数和就餐时间的关系
```

```
day_grp.size().plot.bar()
```

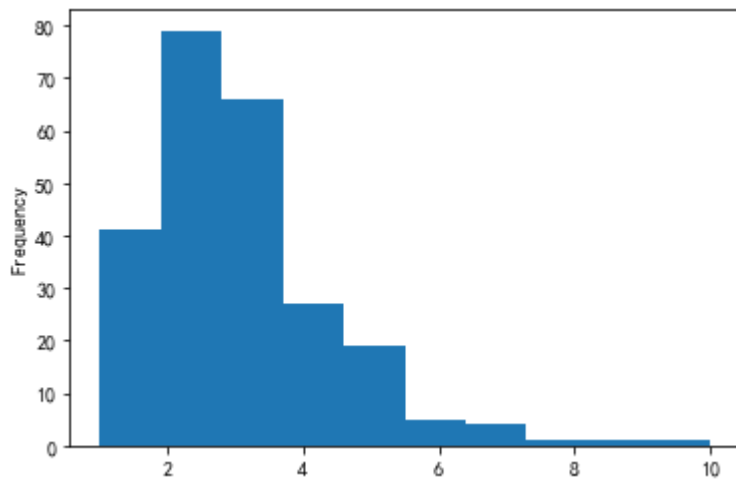
```
<AxesSubplot:xlabel='day,time'>
```



```
# 小费的分布状况
```

```
tips["tip"].plot.hist()
```

```
<AxesSubplot:ylabel='Frequency'>
```



```
tips["tip"].plot.box()
```

<AxesSubplot:>

