40.可视化

In [1]:
```
#常用于数据可视化的Python包
  #Matplotlib
  #Seaborn
  #pandas
  #Bokeh
  #Plotly
  #Vispy
  #Vega
  #gega-lite
```

40.1 Matplotlib可视化

In [2]:
```
import matplotlib.pyplot as plt
%matplotlib inline
```

In [3]:
```
import pandas as pd

women = pd.read_csv('women.csv')
women.head()
```

Out [3]:

	Unnamed: 0	height	weight
0	1	58	115
1	2	59	117
2	3	60	120
3	4	61	123
4	5	62	126

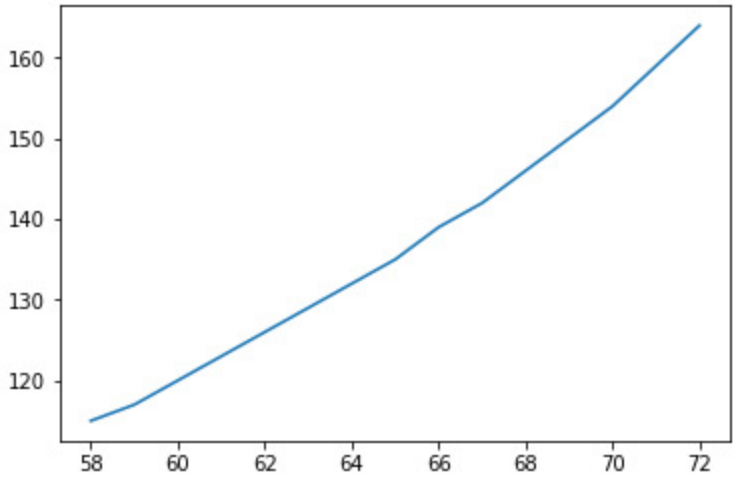
In [4]:
```
women = pd.read_csv('women.csv',index_col =0)
women.head()
```

Out [4]:

	height	weight
1	58	115
2	59	117
3	60	120
4	61	123
5	62	126

In [5]:
```
plt.plot(women["height"], women["weight"])
plt.show()
```

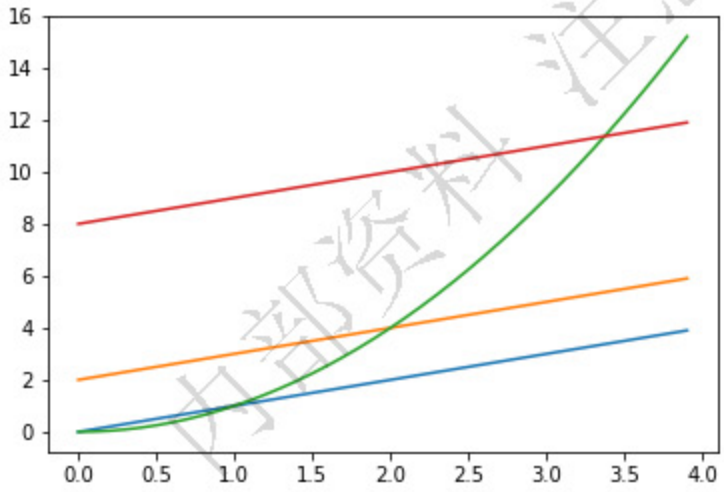
内部资料 注意保密



In [6]:
```
import numpy as np
t=np.arange(0.,4.,0.1)
t
```

Out[6]:
```
array([0. , 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1. , 1.1, 1.2,
       1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2. , 2.1, 2.2, 2.3, 2.4, 2.5,
       2.6, 2.7, 2.8, 2.9, 3. , 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8,
       3.9])
```

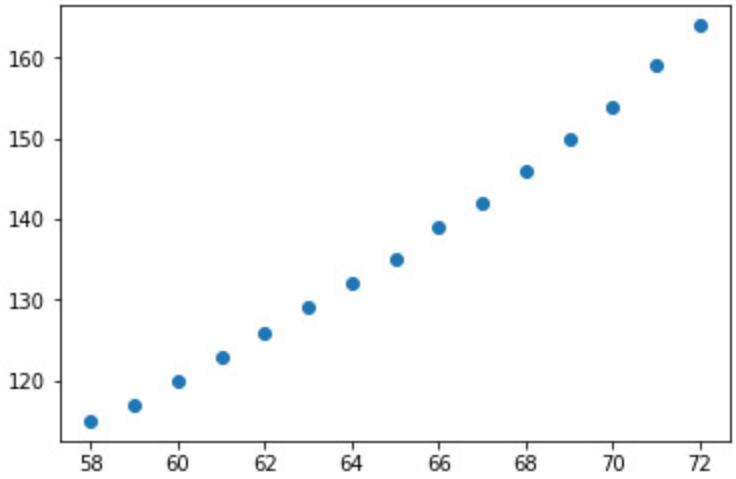
In [8]:
```
plt.plot(t,t,t,t+2,t,t**2,t,t+8)
plt.show()
```



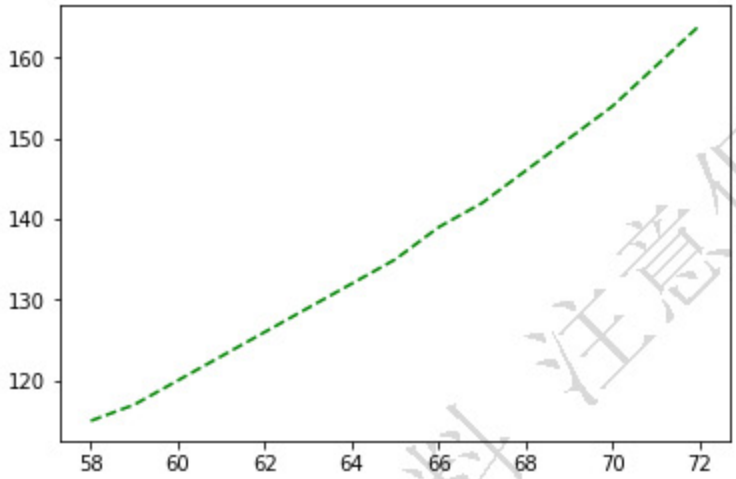
内部资料 注意保密

40.2改变图的属性

In [9]:
```
plt.plot(women["height"], women["weight"],"o")
plt.show()
```

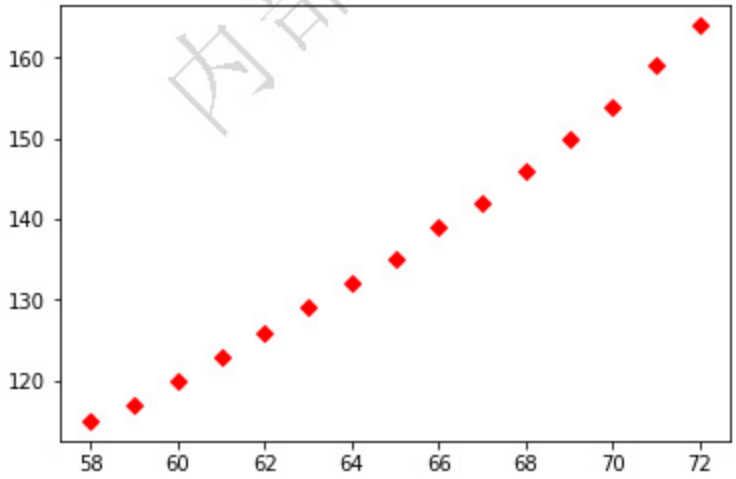


In [10]:
```
plt.plot(women["height"], women["weight"],"g--")
plt.show()
```

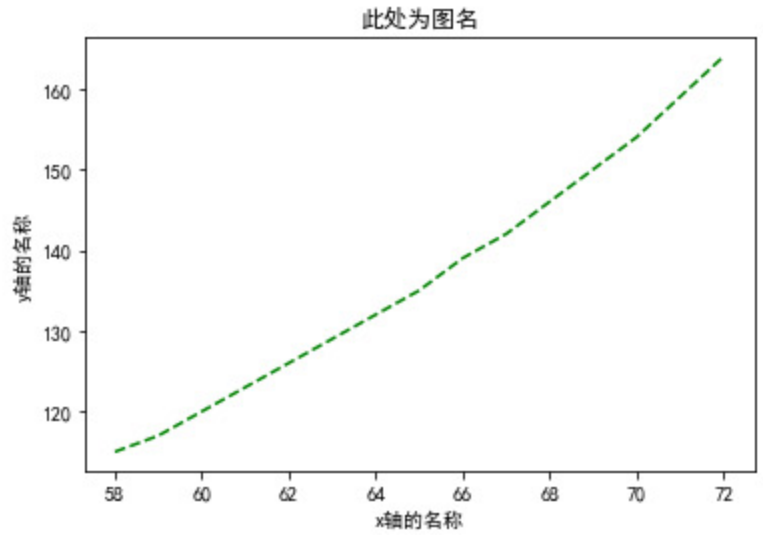


内部资料 注意保密

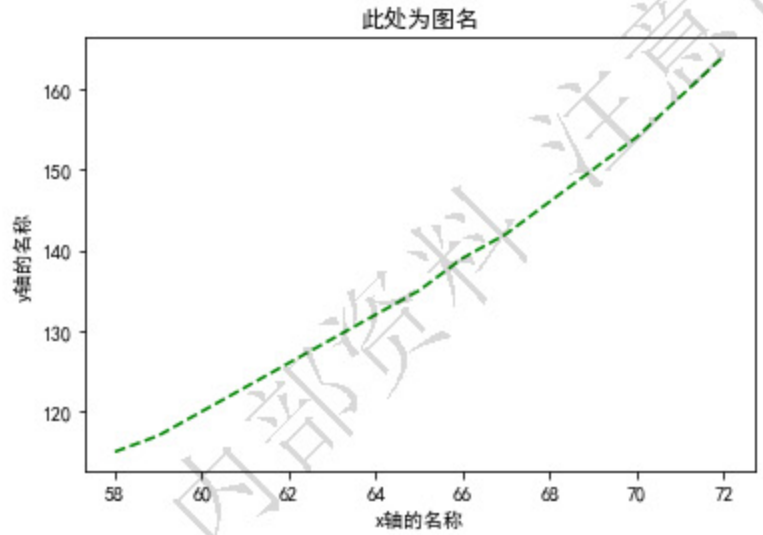
In [11]:
```
plt.plot(women["height"], women["weight"],"rD")
plt.show()
```



In [12]:
```
plt.rcParams['font.family']="SimHei"
plt.plot(women["height"], women["weight"],"g--")
plt.title("此处为图名")
plt.xlabel("x轴的名称")
plt.ylabel("y轴的名称")

plt.show()
```

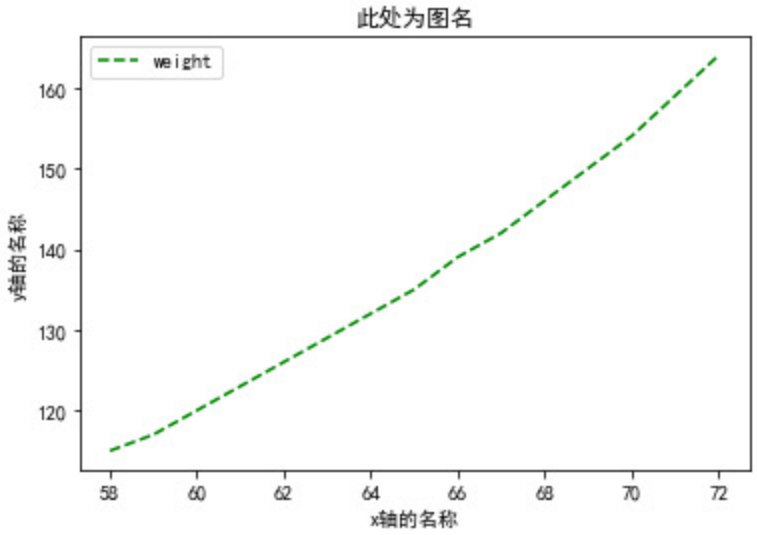


In [13]:
```
plt.rcParams['font.family']="SimHei"
plt.plot(women["height"], women["weight"],"g--")
plt.title("此处为图名")
plt.xlabel("x轴的名称")
plt.ylabel("y轴的名称")

plt.show()
```



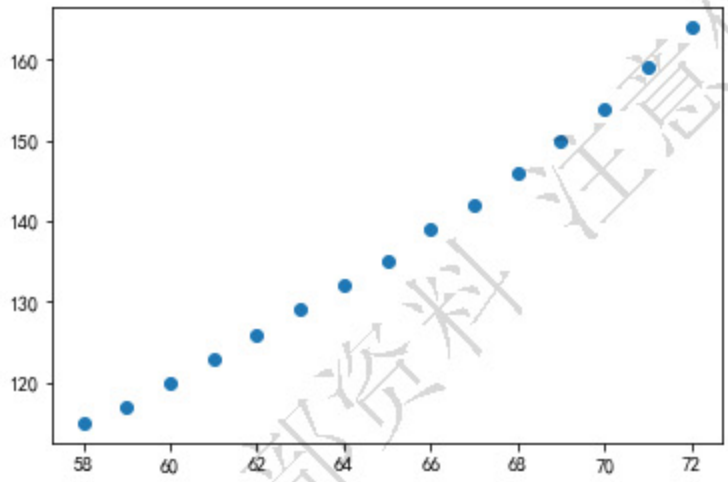
内部资料 注意保密

In [14]:
```
plt.rcParams['font.family']="SimHei"

plt.plot(women["height"], women["weight"],"g--")
plt.title("此处为图名")
plt.xlabel("x轴的名称")
plt.ylabel("y轴的名称")

plt.legend(loc="upper left")
plt.show()
```



40.3 改变图的类型

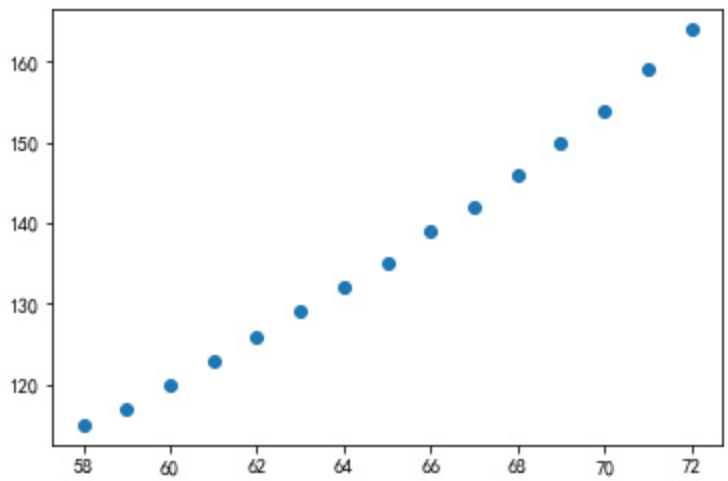
In [15]:
```
plt.scatter(women["height"], women["weight"])
plt.show()
```



内部资料 注意保密

In [16]:
```
%matplotlib inline
plt.scatter(women.height, women.weight)
```

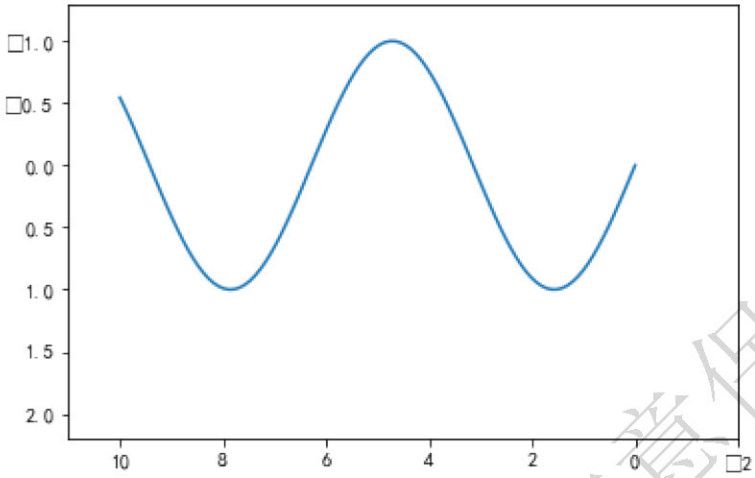
Out[16]:
```
<matplotlib.collections.PathCollection at 0x21b93e51400>
```



40.4 改变图的坐标轴的取值范围

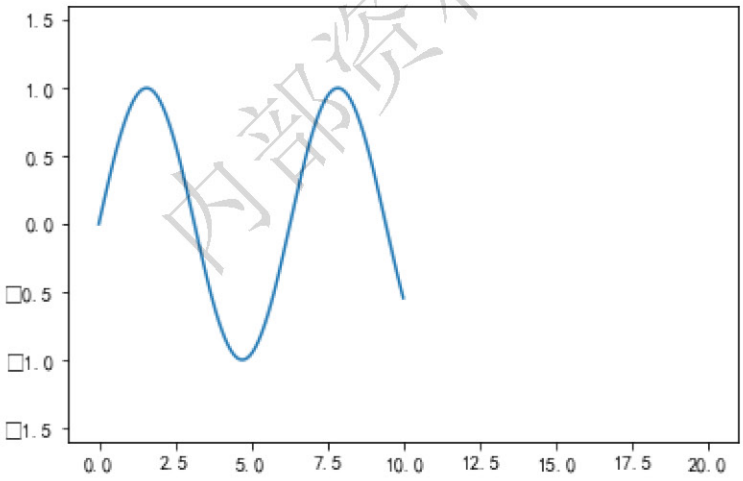
In [17]:
```
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline

x=np.linspace(0,10,100)
plt.plot(x,np.sin(x))
plt.xlim(11,-2)
plt.ylim(2.2,-1.3)
```

Out[17]: (2.2, -1.3)



In [18]:
```
plt.plot(x,np.sin(x))
plt.axis([-1,21,-1.6,1.6])
```

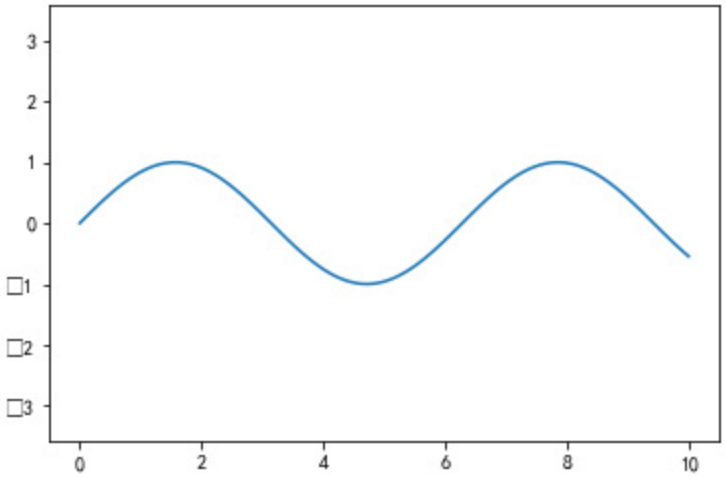
Out[18]: [-1, 21, -1.6, 1.6]



In [19]:
```
plt.plot(x,np.sin(x))
plt.axis([-1,21,-1.6,1.6])
plt.axis("equal")
```

Out[19]: (-0.5, 10.5, -1.0993384025373631, 1.0996461858110391)

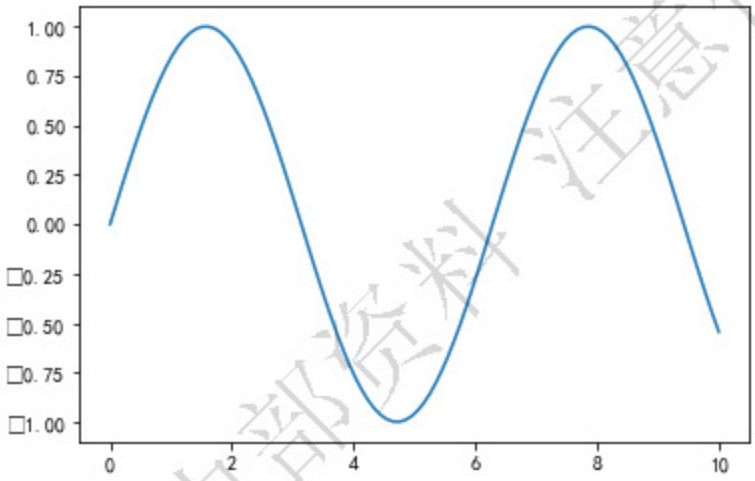
内部资料 注意保密



40.5 去掉边界的空白

In [20]:
```
plt.plot(x,np.sin(x))
plt.axis([-1,21,-1.6,1.6])
plt.axis("tight")
```

Out[20]:
```
(-0.5, 10.5, -1.0993384025373631, 1.0996461858110391)
```

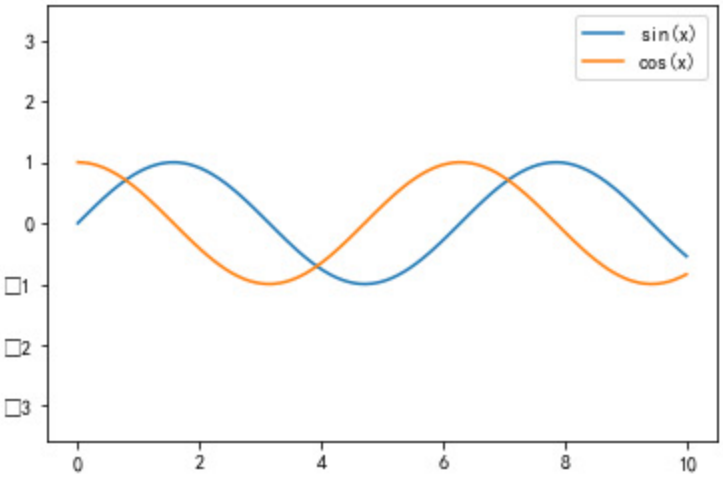


内部资料 注意保密

40.6 在同一个坐标上画两个图

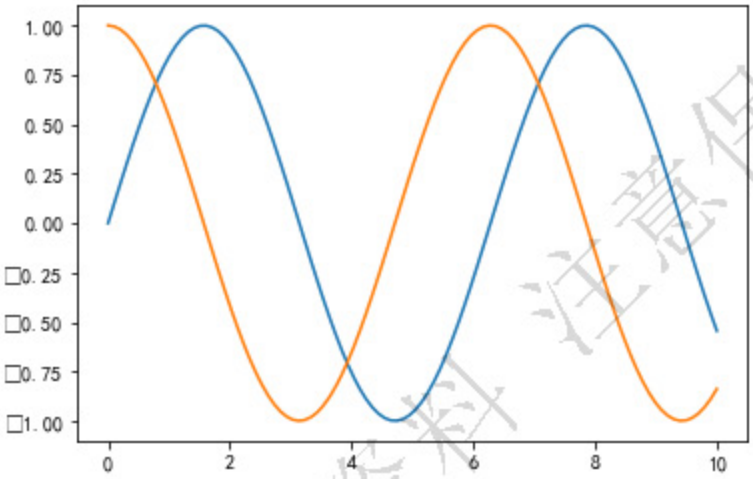
In [21]:
```
plt.plot(x,np.sin(x),label="sin(x)")
plt.plot(x,np.cos(x),label="cos(x)")
plt.axis("equal")
plt.legend()
```

Out[21]:
```
<matplotlib.legend.Legend at 0x21b92a6dbe0>
```



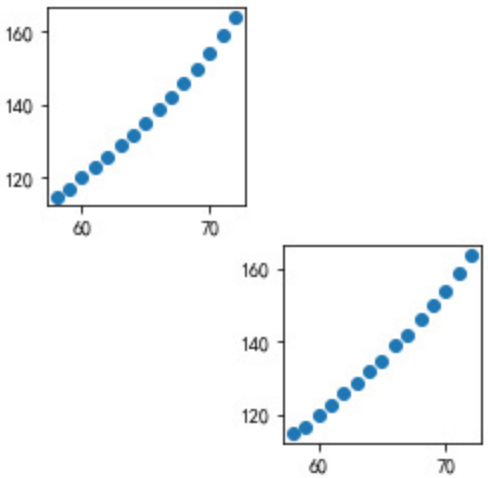
In [22]:
```
plt.plot(x,np.sin(x),label="sin(x)")
plt.plot(x,np.cos(x),label="cos(x)")
```

Out[22]:
```
[<matplotlib.lines.Line2D at 0x21b93d6f630>]
```



40.7 多图显示

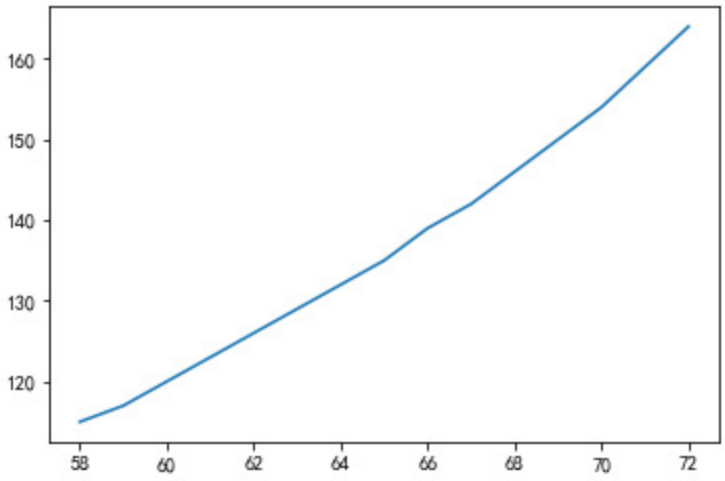
In [23]:
```
plt.subplot(2,3,5)
plt.scatter(women["height"], women["weight"])
plt.subplot(2,3,1)
plt.scatter(women["height"], women["weight"])
plt.show()
```



内部资料 注意保密

40.8 图的保存

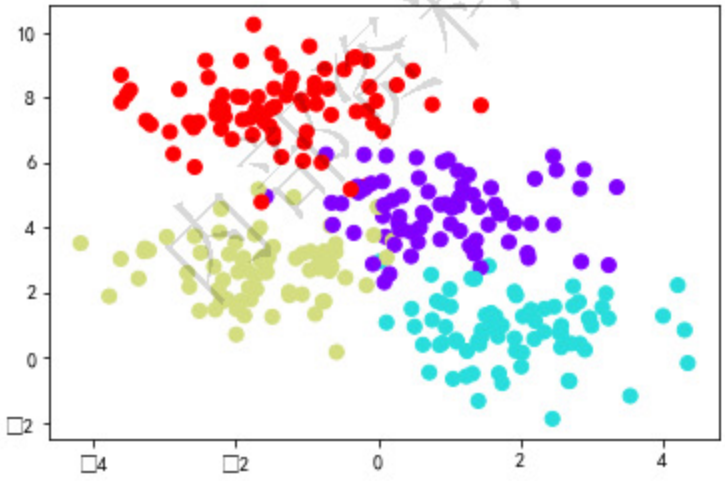
In [24]:
```
women = pd.read_csv('women.csv')
plt.plot(women.height, women.weight)
plt.savefig("sagefig.png")
```



40.9 散点图的画法

In [25]:
```
from sklearn.datasets.samples_generator import make_blobs
X,y=make_blobs(n_samples=300,centers=4,random_state=0, cluster_std=1.0)
plt.scatter(X[:,0],X[:,1],c=y,s=50,cmap="rainbow")
```

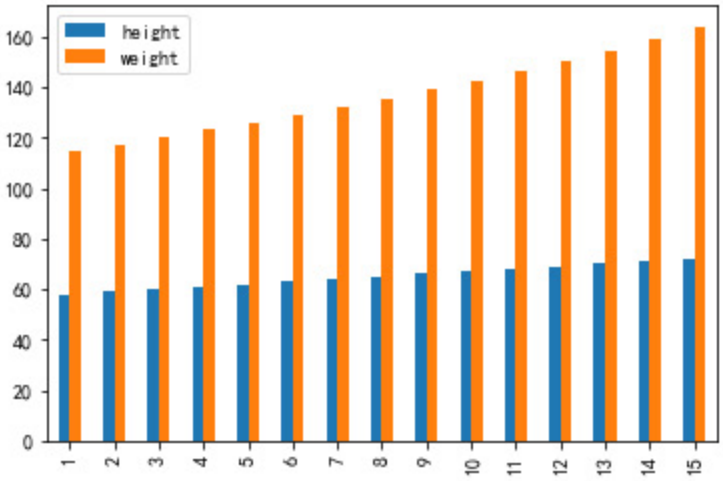
Out[25]:
```
<matplotlib.collections.PathCollection at 0x21b9484ee80>
```



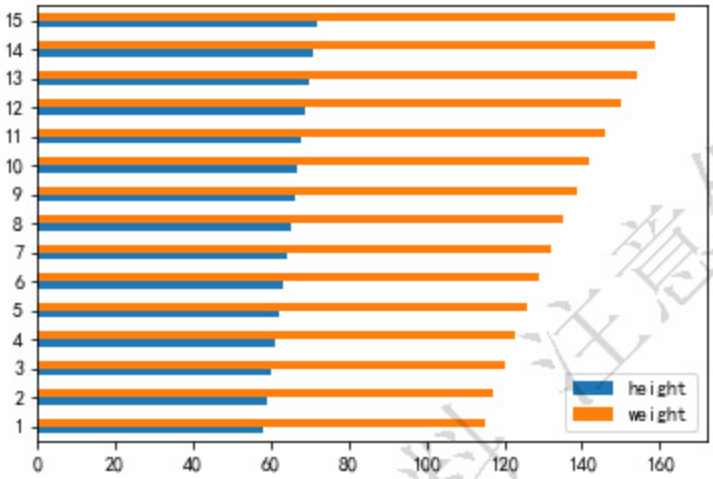
40.10 Pandas可视化

In [26]:
```
import pandas as pd
women = pd.read_csv('women.csv',index_col =0)
women.plot(kind="bar")
plt.show()
```

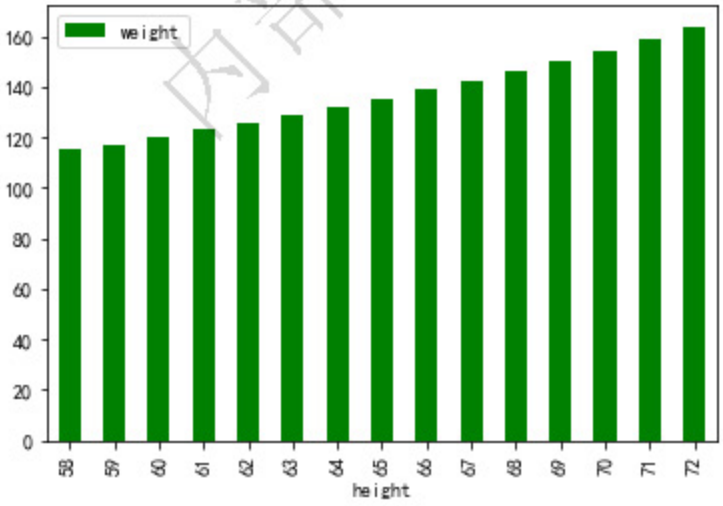
内部资料 注意保密



In [27]:
```
women.plot(kind="barh")
plt.show()
```

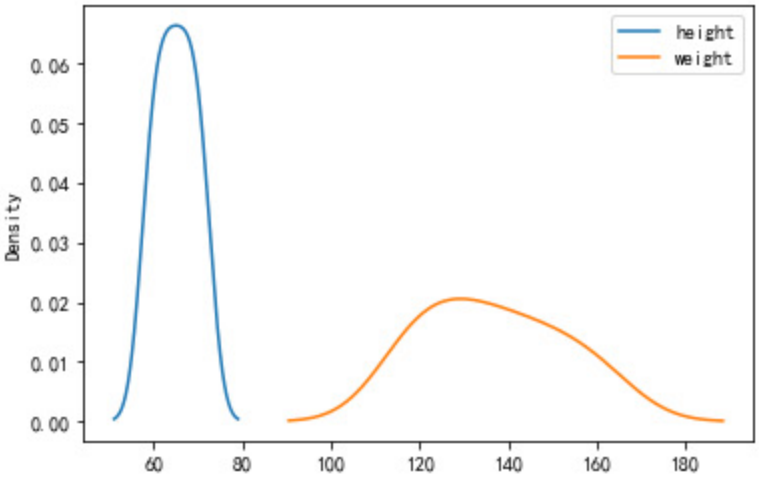


In [28]:
```
women.plot(kind="bar",x="height",y="weight",color="g")
plt.show()
```

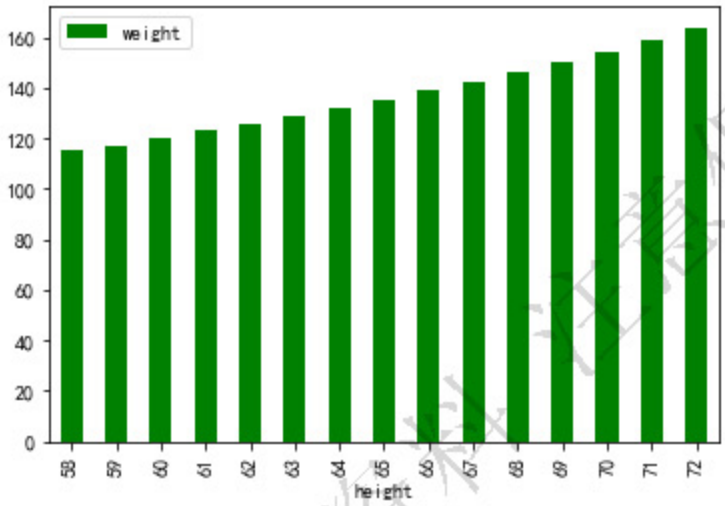


In [29]:
```
women.plot(kind="kde")
plt.show()
```

内部资料 注意保密



In [30]:
```
women.plot(kind="bar",x="height",y="weight",color="g")
plt.legend(loc="best")
plt.show()
```



40.11 Seaborn可视化

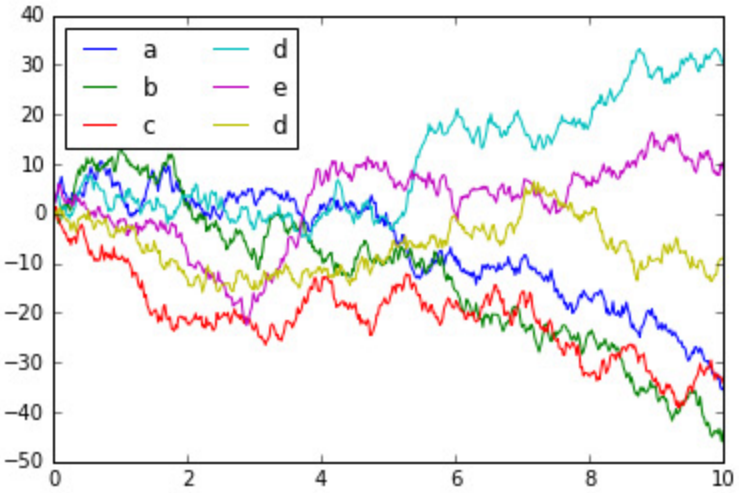
In [31]:
```
import matplotlib.pyplot as plt
plt.style.use("classic")
%matplotlib inline
import numpy as np
import pandas as pd
```

In [32]:
```
rng= np.random.RandomState(0)
x=np.linspace(0,10,500)
y=np.cumsum(rng.randn(500,6),0)
```

In [33]:
```
plt.plot(x,y)
plt.legend("abcded",ncol=2,loc="upper left")
```

Out[33]:
```
<matplotlib.legend.Legend at 0x21b94a676a0>
```

内部资料 注意保密



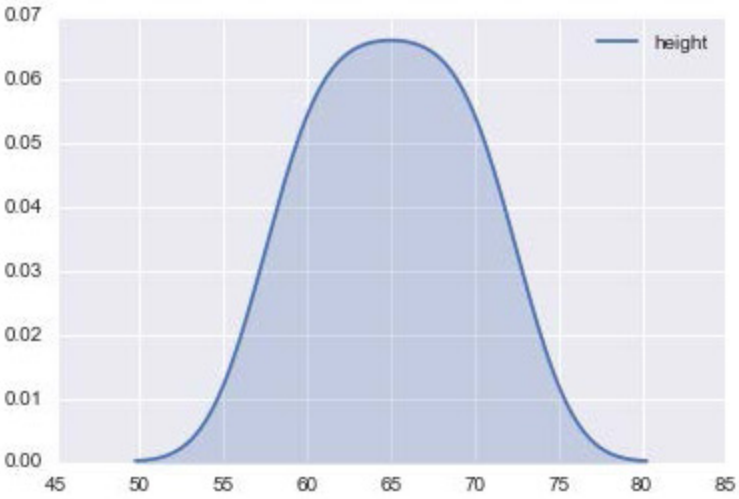
In [34]:
```
import seaborn as sns
sns.set()
plt.plot(x,y)
plt.legend("abcdef",ncol=2,loc="upper left")
```

Out[34]: <matplotlib.legend.Legend at 0x21b94be14a8>

内部资料注意保密

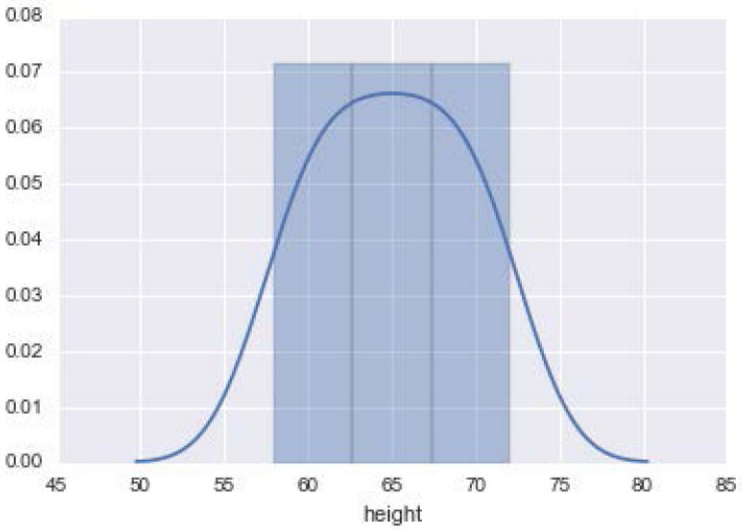
In [35]:
```
sns.kdeplot(women.height, shade=True)
```

Out[35]: <matplotlib.axes._subplots.AxesSubplot at 0x21b94b3fc18>



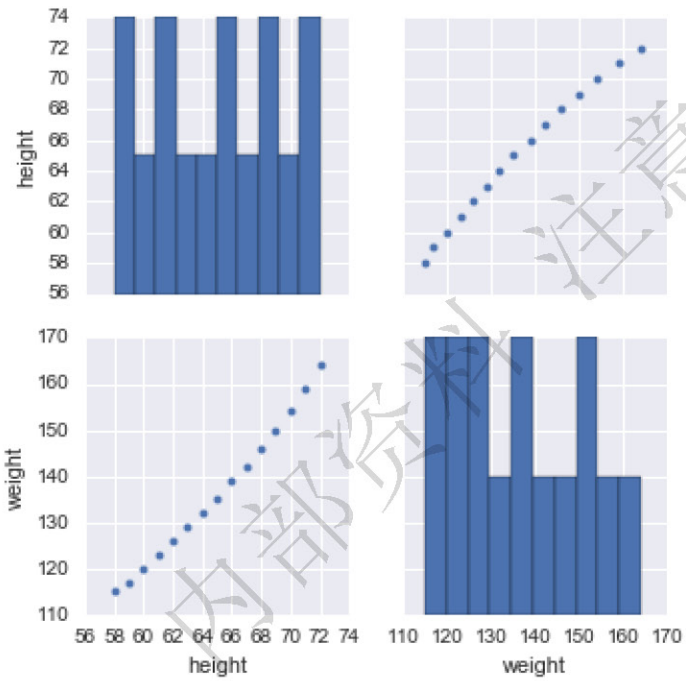
In [36]:
```
sns.distplot(women.height)
```

Out[36]: <matplotlib.axes._subplots.AxesSubplot at 0x21b94a14710>



In [37]:
```
sns.pairplot(women)
```

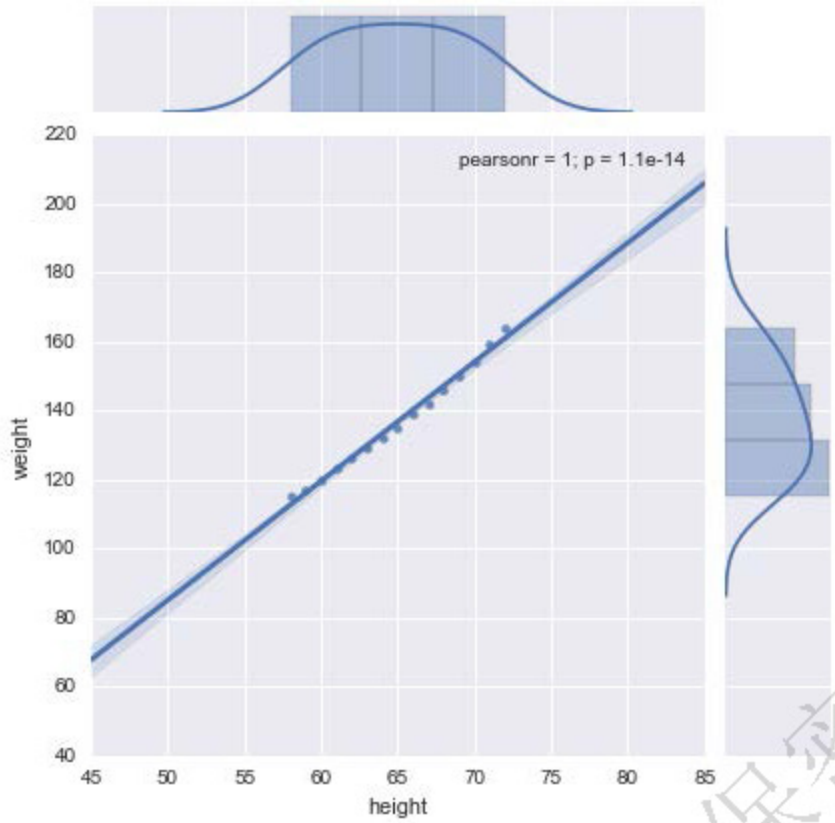
Out[37]: <seaborn.axisgrid.PairGrid at 0x21b948b4a20>



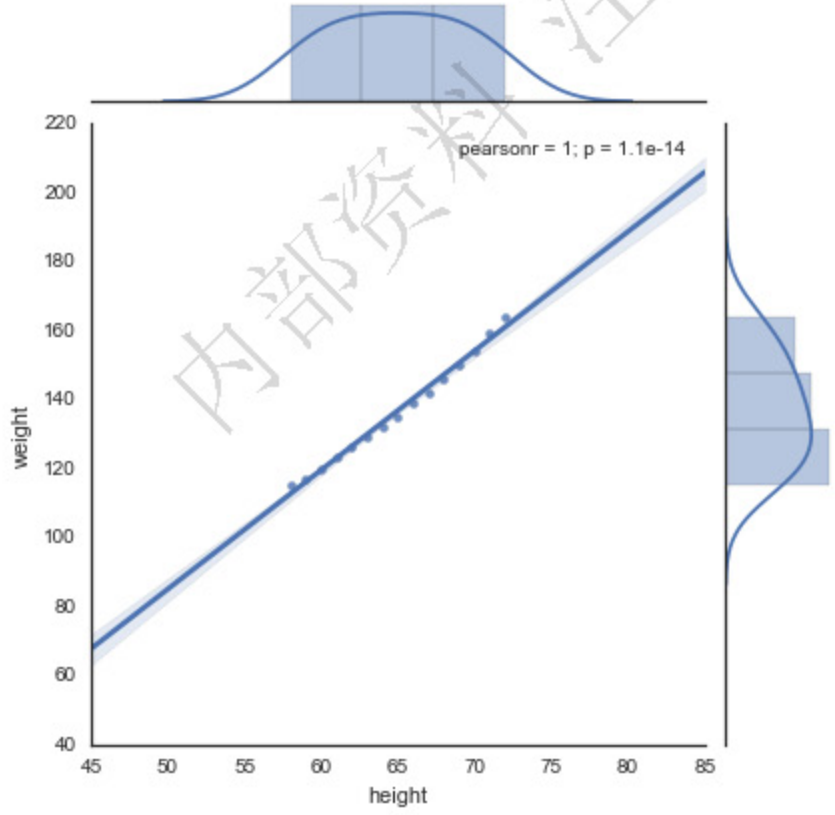
In [38]:
```
sns.jointplot(women.height,women.weight,kind="reg")
```

Out[38]: <seaborn.axisgrid.JointGrid at 0x21b94f5d7f0>

内部资料 注意保密

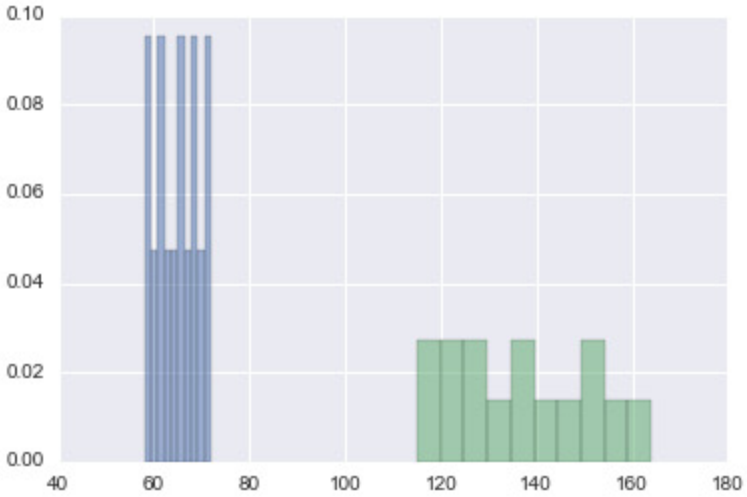


In [39]:
```
with sns.axes_style("white"):
    sns.jointplot(women.height,women.weight,kind="reg")
```



In [40]:
```
for x in ["height","weight"]:
    plt.hist(women[x],normed=True,alpha=0.5)
```

内部资料 注意保密



内部资料 注意保密