Plotting and Visualization

In [133]:

```
#pip install matplotlib
```

In [134]:

```
import matplotlib.pyplot as plt
```

In [135]:

```
import numpy as np
import pandas as pd
```

In [136]:

```
from numpy.random import randn
import random
np.random.seed(10)
```

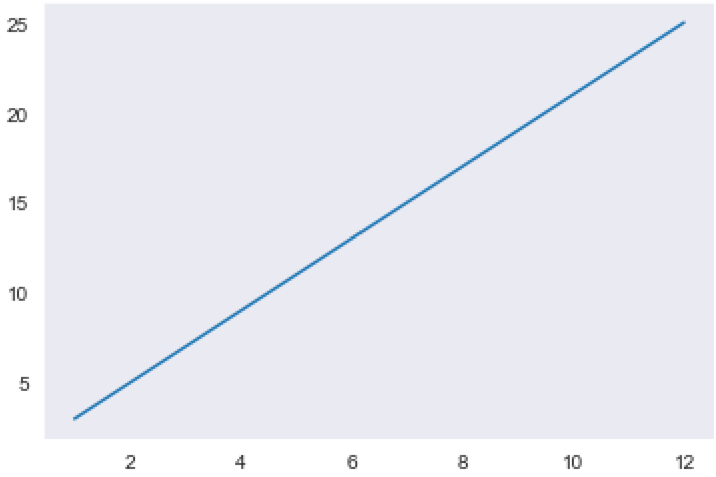
plot

In [137]:

```
x = [1, 2, 5, 12]
y = [3, 5, 11, 25]
```

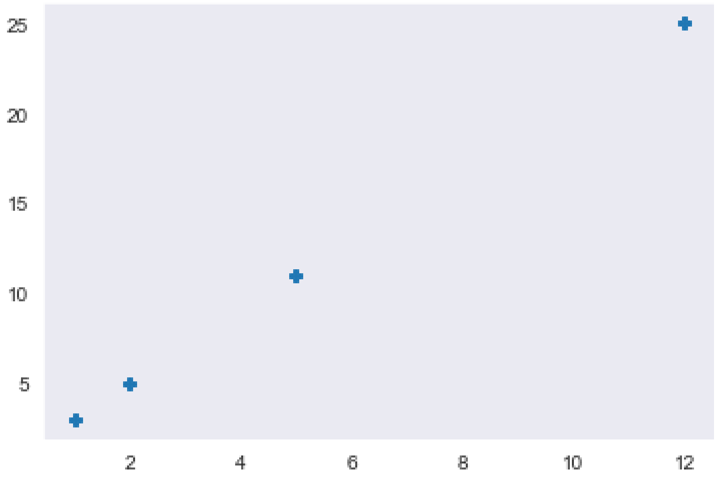
In [138]:

```
plt.plot(x, y)
plt.show()
```



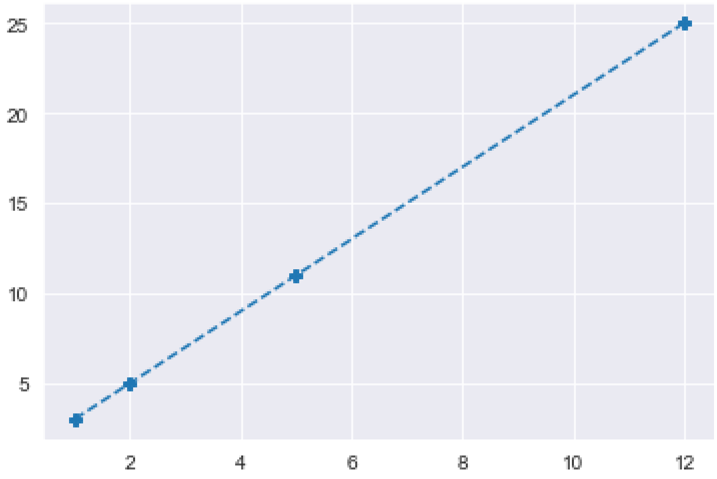
In [139]:

```
plt.plot(x, y,'P')
plt.show()
```



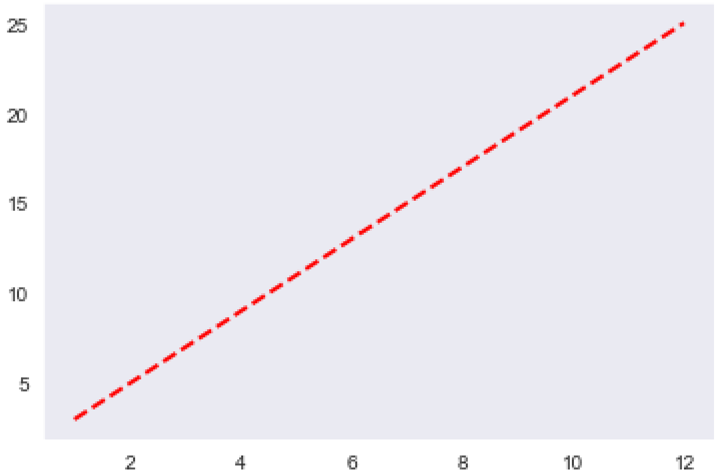
In [140]:

```
plt.plot(x, y,'--P')
plt.grid()
plt.show()
```



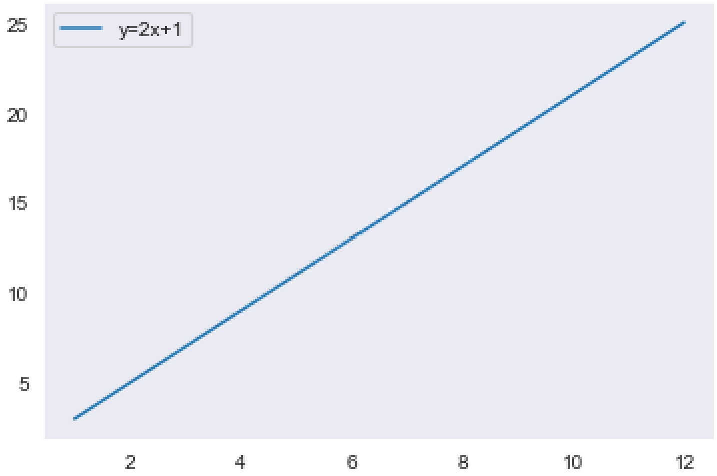
In [141]:

```
plt.plot(x, y, color='red', linestyle='dashed', linewidth = 2)
plt.show()
```



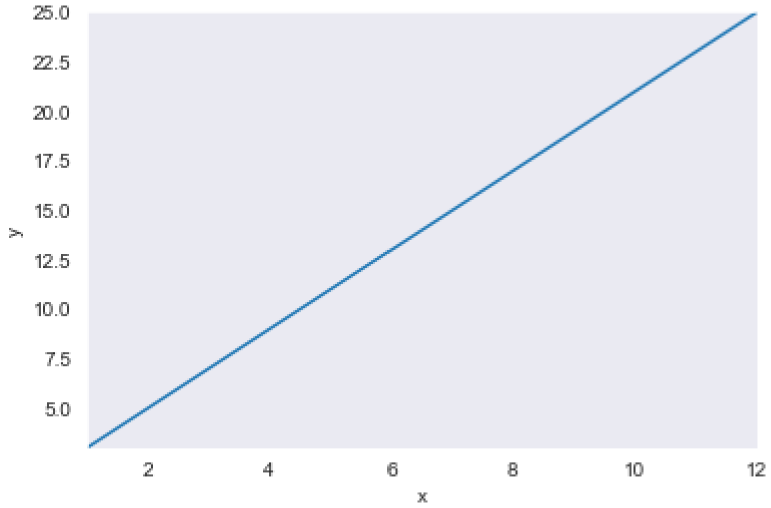
In [142]:

```
plt.plot(x, y, label='y=2x+1')
plt.legend()
plt.show()
```



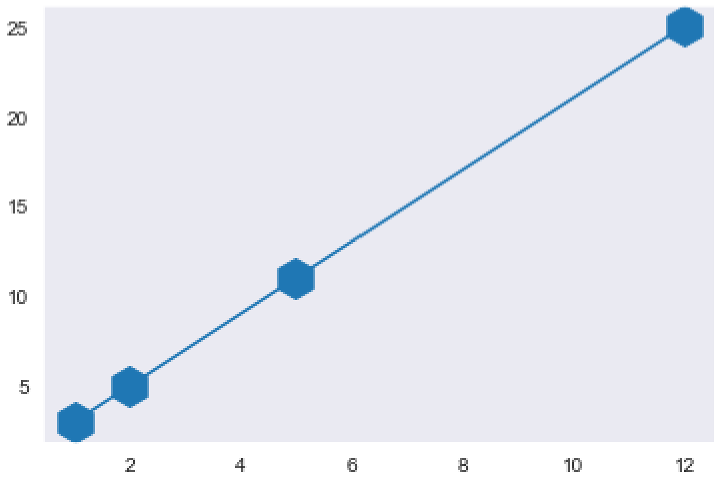
In [143]:

```
plt.plot(x, y)

plt.xlabel('x')
plt.ylabel('y')

plt.xlim(1,12)
plt.ylim(3,25)

plt.show()
```



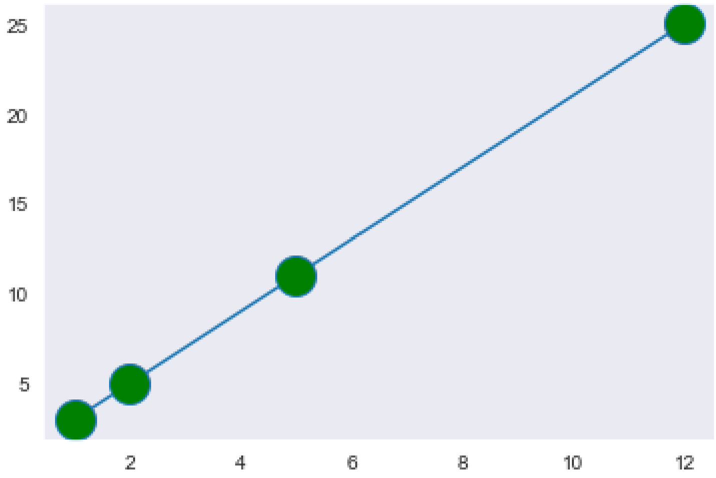
In [144]:

```
plt.plot(x, y , marker='h', markersize=20)      # marker : o , x ,X , p, P , h
plt.show()
```



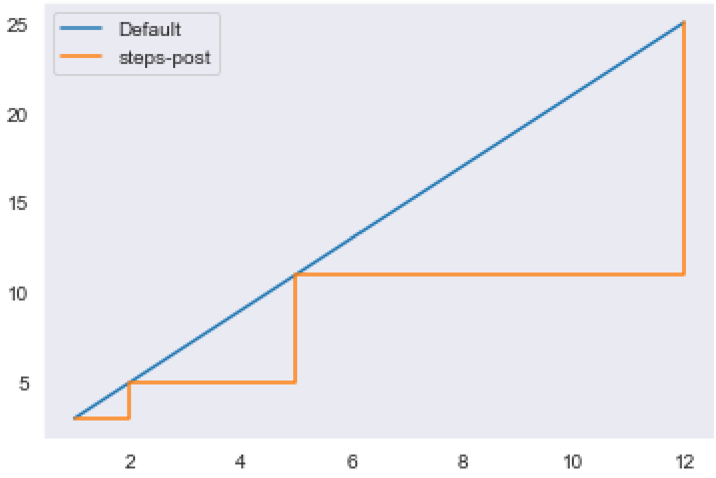
In [145]:

```
plt.plot(x, y, marker='o', markersize=20, markerfacecolor='green')
plt.show()
```



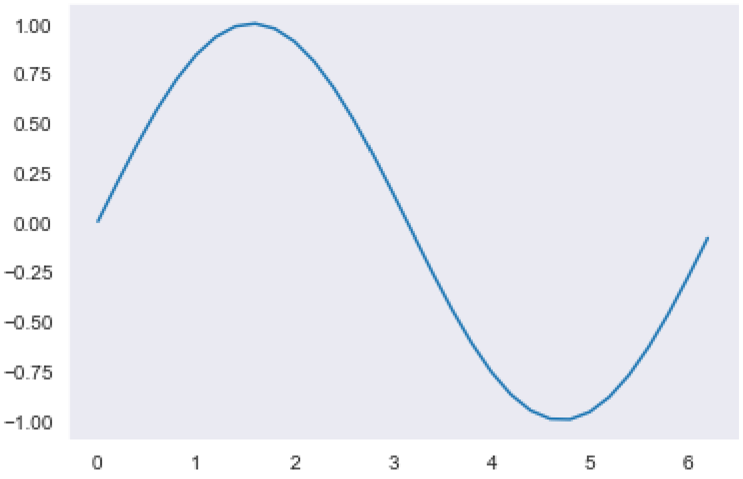
In [146]:

```
plt.plot(x,y, label='Default')
plt.plot(x,y, drawstyle='steps-post', label='steps-post')
plt.legend(loc='best')
plt.show()
```



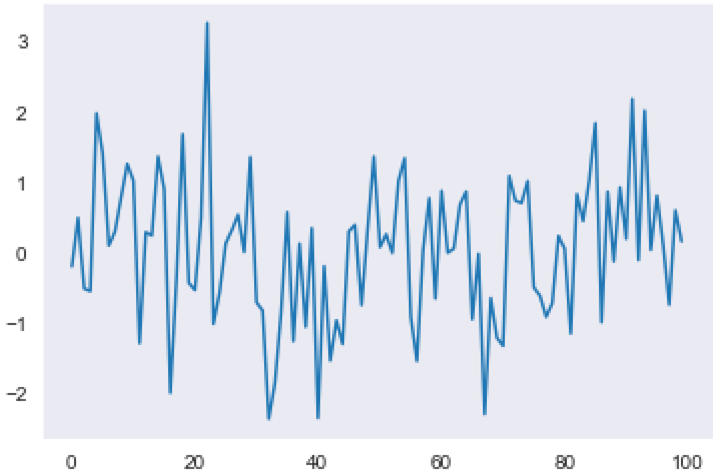
In [147]:

```
x = np.arange(0, 2*(np.pi), 0.2)
y = np.sin(x)
plt.plot(x, y)
plt.show()
```



In [148]:

```
np.random.seed(12345)
plt.plot(np.random.randn(100))
plt.show()
```



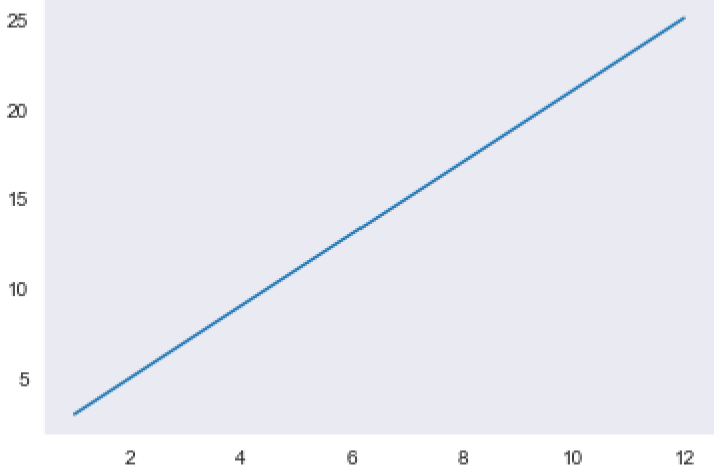
subplot

In [149]:

```
x = [1, 2, 5, 12]
y = [3, 5, 11, 25]
```

In [150]:

```
plt.plot(x, y)
plt.show()
```

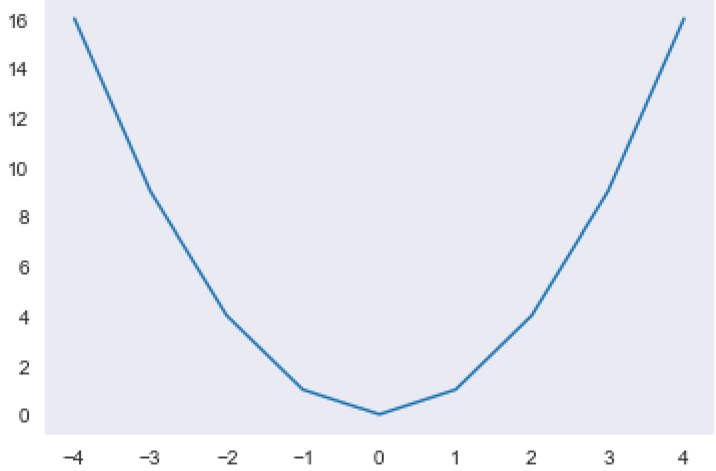


In [151]:

```
z = [-4, -3, -2, -1, 0, 1, 2 , 3, 4]
w = [16, 9, 4, 1, 0, 1, 4, 9, 16]
```

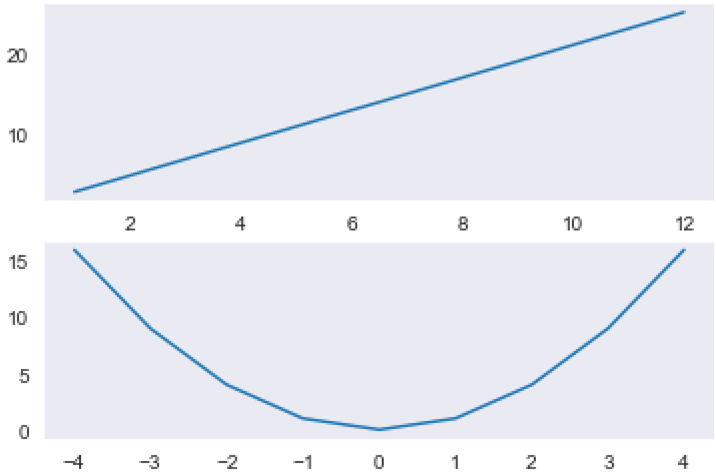
In [152]:

```
plt.plot(z, w)
plt.show()
```



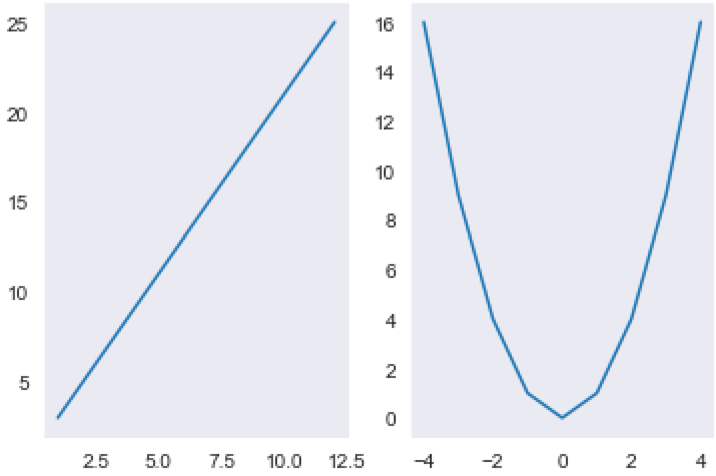
In [153]:

```
plt.subplot(211)
plt.plot(x, y)

plt.subplot(212)
plt.plot(z, w)

plt.show()
```



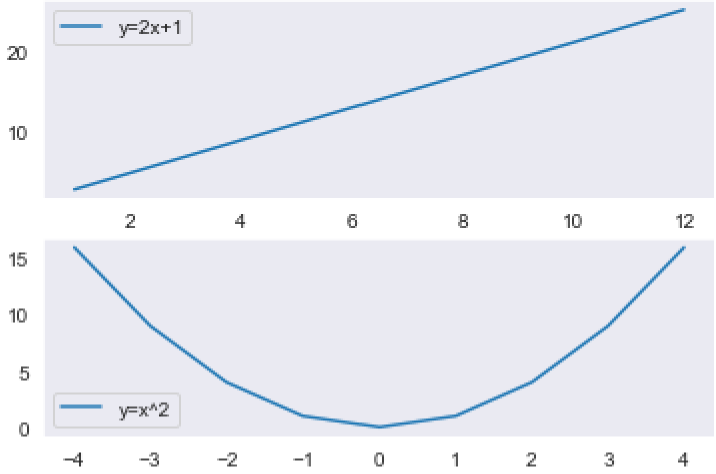
In [154]:

```
plt.subplot(121)
plt.plot(x, y)

plt.subplot(122)
plt.plot(z, w)

plt.show()
```



In [155]:

```
plt.subplot(211)
plt.plot(x, y, label='y=2x+1')
plt.legend()

plt.subplot(212)
plt.plot(z, w, label='y=x^2')
plt.legend()

plt.savefig('test.png')
```

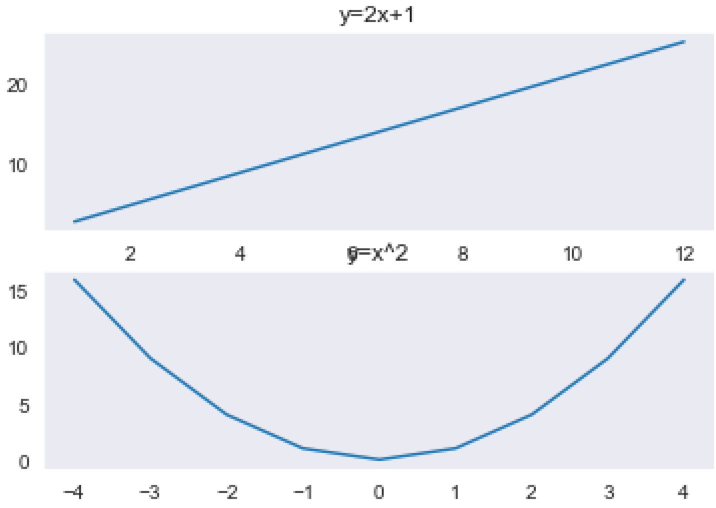


In [156]:

```
# subplots
```

In [157]:

```
fig, (ax1, ax2) = plt.subplots(2, 1)

ax1.plot(x, y)
ax1.set(title='y=2x+1')

ax2.plot(z, w)
ax2.set(title='y=x^2')

plt.show()
```



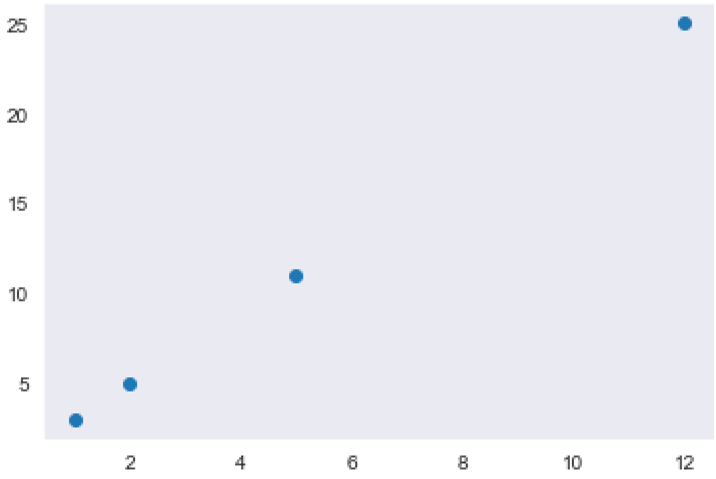
scatter

In [158]:

```
x = [1, 2, 5, 12]
y = [3, 5, 11, 25]
```

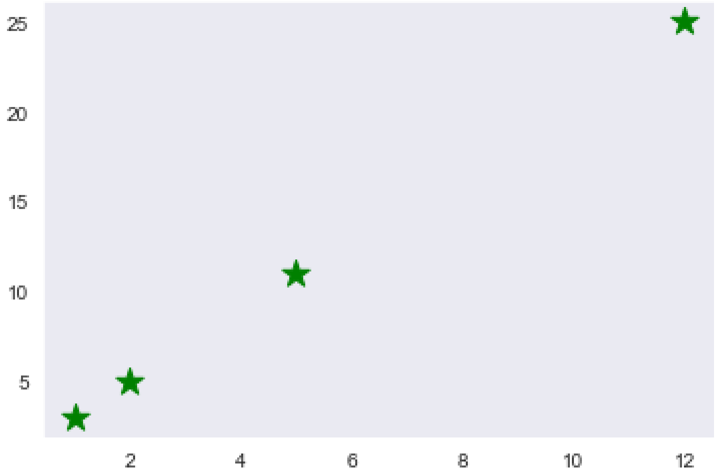
In [159]:

```
plt.scatter(x, y)
plt.show()
```



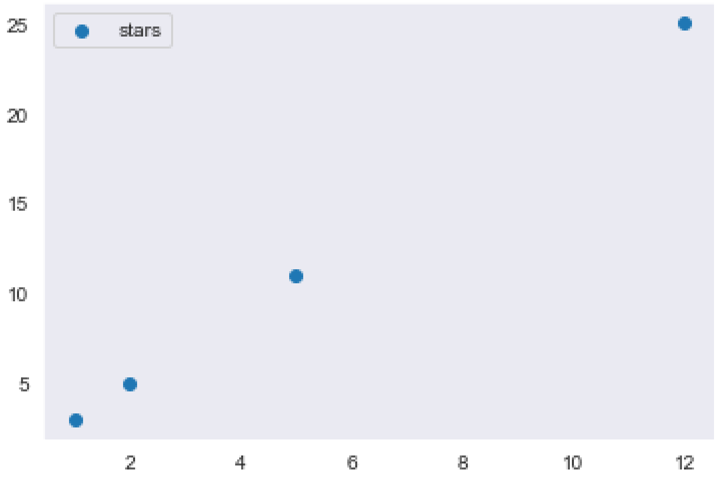
In [160]:

```
plt.scatter(x, y, color= "g", marker= "*", s=200)
plt.show()
```



In [161]:

```
plt.scatter(x, y, label= "stars")
plt.legend()
plt.show()
```

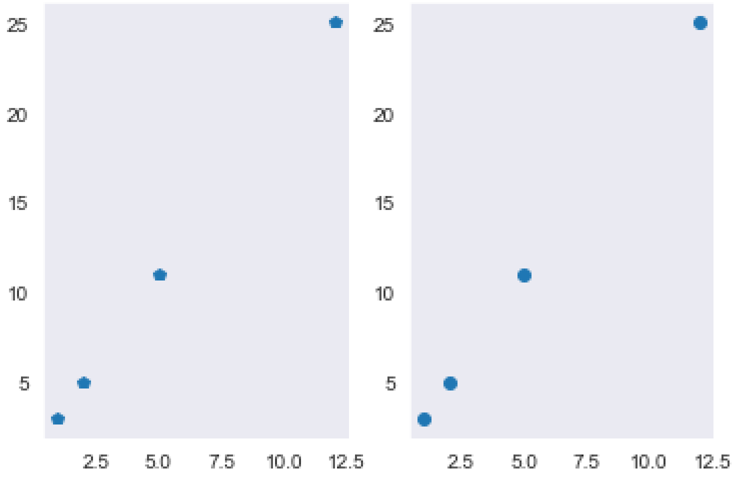


In [162]:

```
x = [1, 2, 5, 12]
y = [3, 5, 11, 25]
```

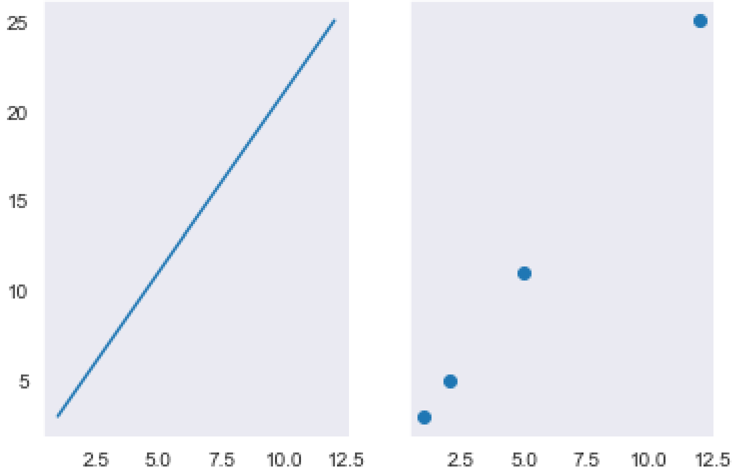
In [163]:

```
f, (ax1, ax2) = plt.subplots(1, 2)
ax1.plot(x, y,'p')
ax2.scatter(x, y)
plt.show()
```



In [164]:

```
f, (ax1, ax2) = plt.subplots(1, 2, sharey=True)
ax1.plot(x, y)
ax2.scatter(x, y)
plt.show()
```



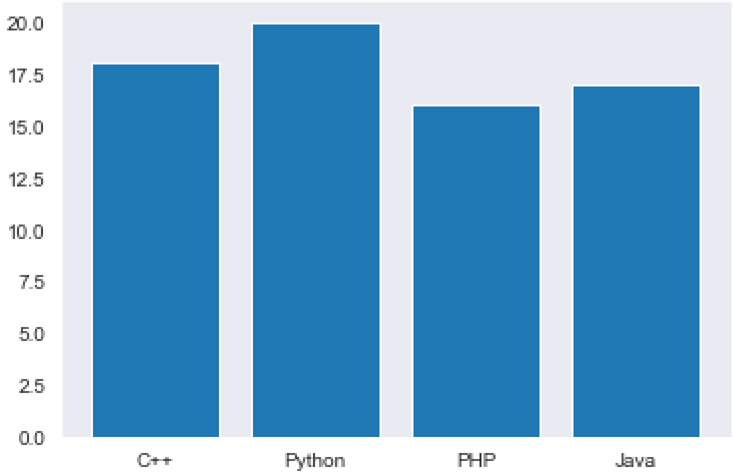
bar

In [165]:

```
dars = ['C++', 'Python', 'PHP','Java']
score = [18, 20 , 16, 17]
```

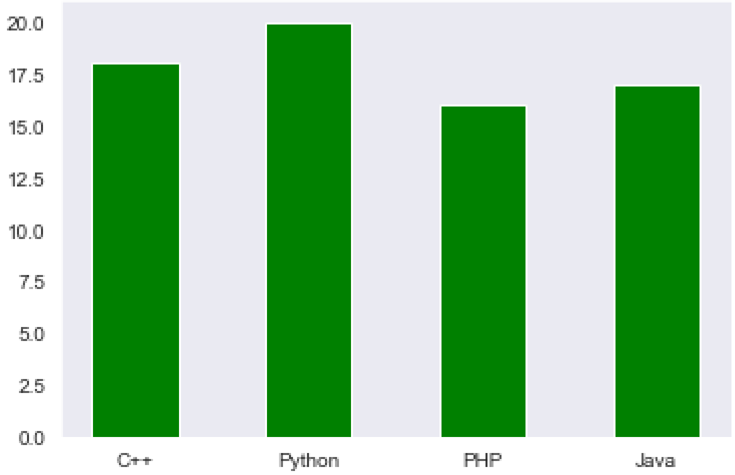
In [166]:

```
plt.bar(dars, score)
plt.show()
```



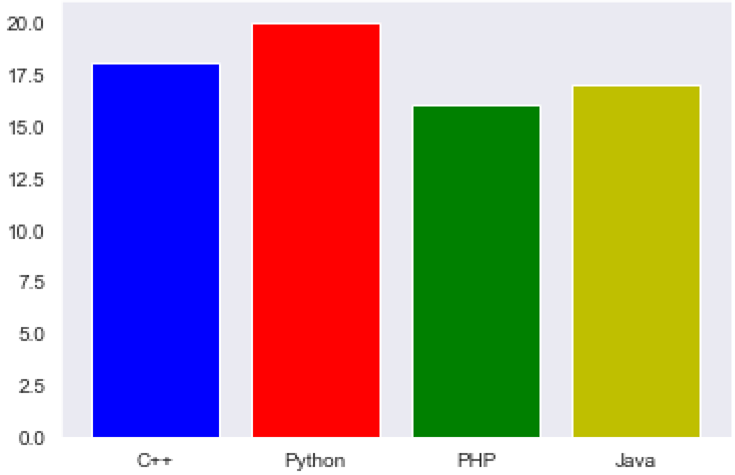
In [167]:

```
plt.bar(dars, score, color='g', width=0.5)
plt.show()
```



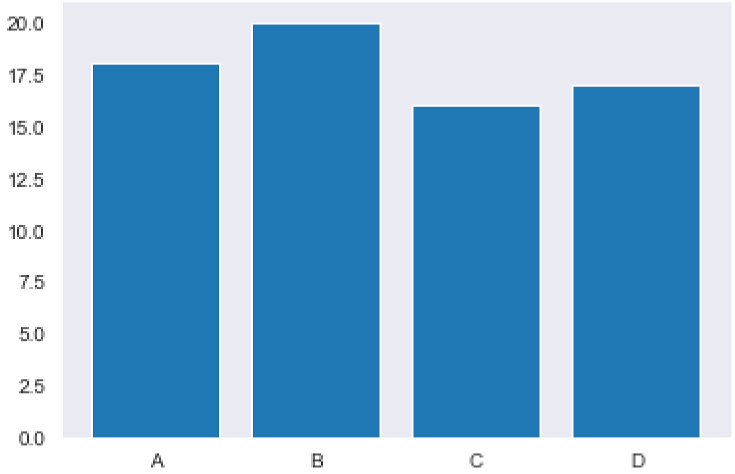
In [168]:

```
plt.bar(dars, score, color = ['b', 'r', 'g', 'y'])
plt.show()
```



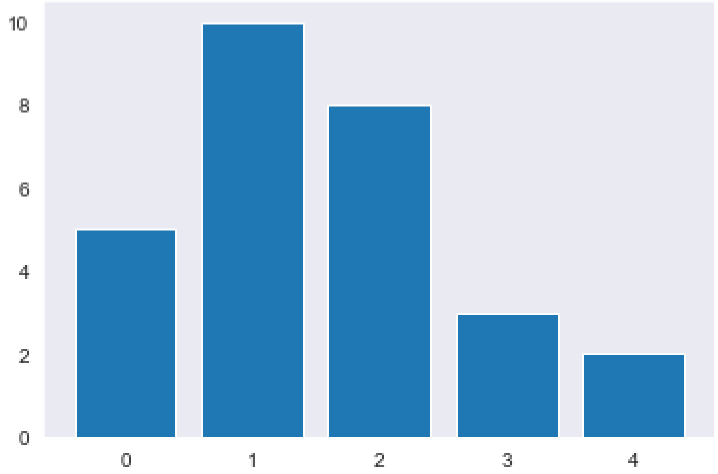
In [169]:

```
plt.bar(dars, score, tick_label = ['A', 'B', 'C', 'D'] )
plt.show()
```



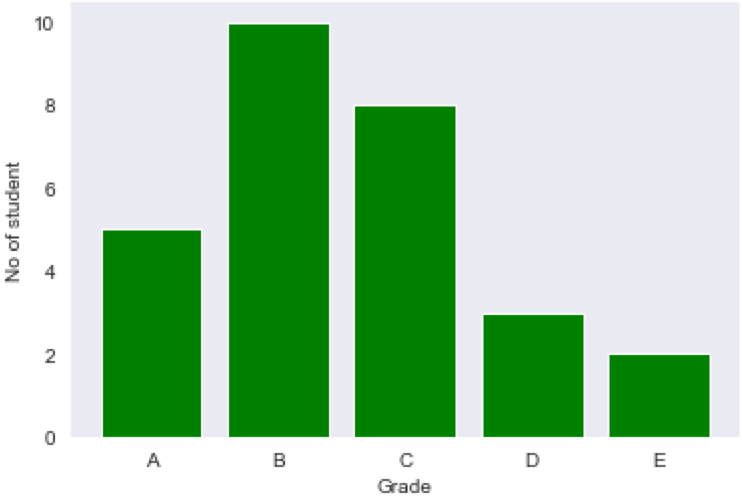
In [170]:

```
x = [0, 1, 2, 3, 4]
y = [5, 10, 8, 3, 2]
plt.bar(x, y)
plt.show()
```



In [171]:

```
plt.xticks(x, ['A', 'B', 'C', 'D', 'E'])
plt.bar(x, y, color='Green')

plt.xlabel('Grade')
plt.ylabel('No of student')
plt.show()
```



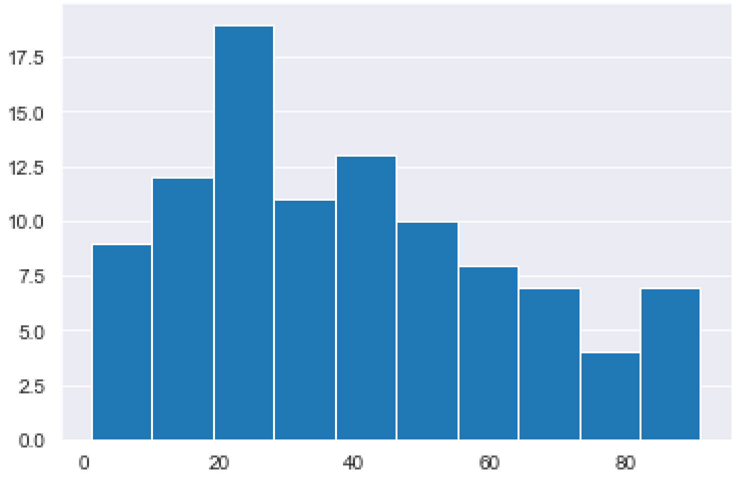
histogram

In [172]:

```
age = [1,1,2,3,3,5,7,8,9,10,10,11,11,13,13,15,16,17,18,18,18,19,20,21,21,23,
       24,24,25,25,25,25,26,26,26,27,27,27,27,27,29,30,30,31,33,34,34,34,35,
       36,36,37,37,38,38,39,40,41,41,42,43,44,45,45,46,47,48,48,49,50,51,52,
       53,54,55,55,56,57,58,60,61,63,64,65,66,68,70,71,72,74,75,77,81,83,84,
       87,89,90,90,91]
```

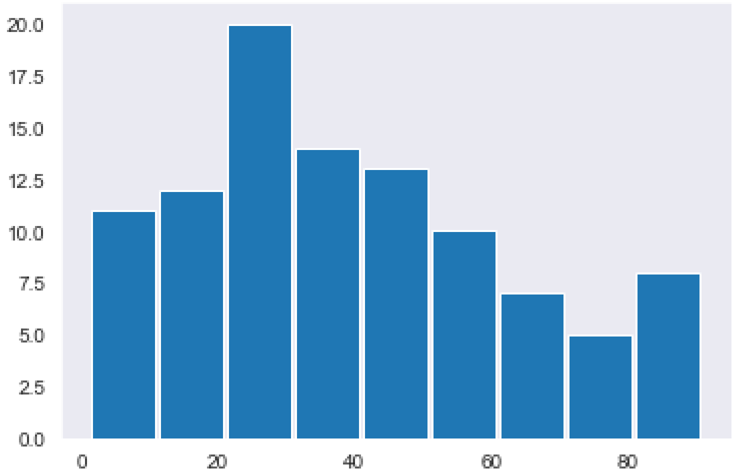
In [173]:

```
plt.hist(age)
plt.grid(axis='y')
plt.show()
```



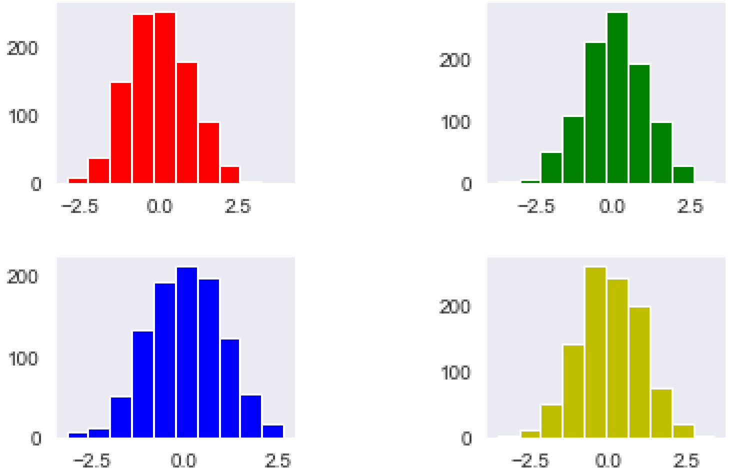
In [174]:

```
plt.hist(age, rwidth=0.95, bins=9)
plt.show()
```



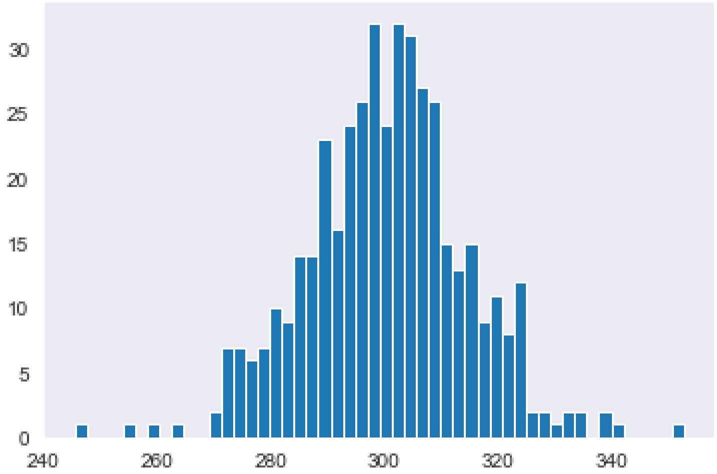
In [175]:

```
fig, ax = plt.subplots(2, 2)

ax[0, 0].hist(np.random.randn(1000), color='r')
ax[0, 1].hist(np.random.randn(1000), color='g')
ax[1, 0].hist(np.random.randn(1000), color='b')
ax[1, 1].hist(np.random.randn(1000), color='y')

plt.subplots_adjust(wspace=0.8, hspace=0.4)
plt.show()
```



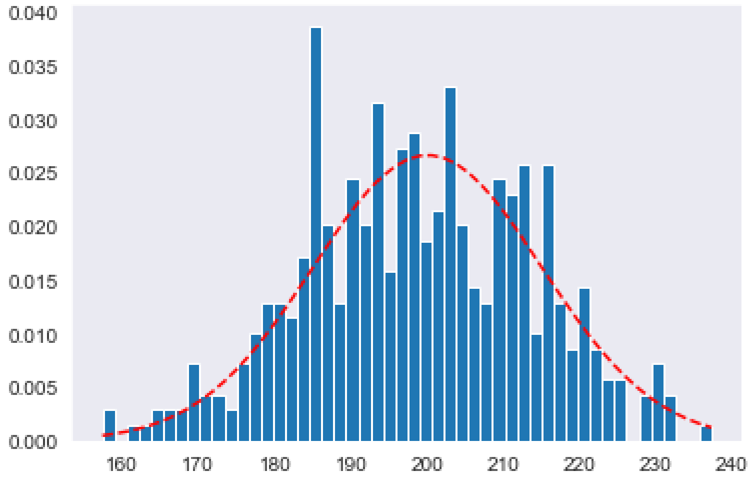
In [176]:

```
mu = 300
sigma = 15
data = mu + sigma * np.random.randn(437)

fig, ax = plt.subplots()
ax.hist(data, bins=50)
plt.show()
```



In [177]:

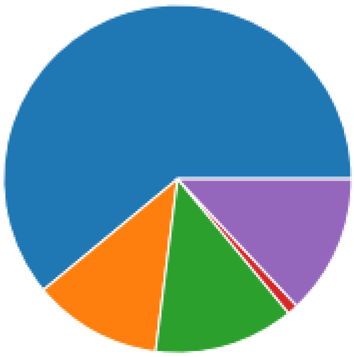
```
mu = 200
sigma = 15
data = mu + sigma * np.random.randn(437)
fig, ax = plt.subplots()

n, bins, patches = ax.hist(data, bins=50, density=True)
y = ((1 / (np.sqrt(2 * np.pi) * sigma)) * np.exp(-0.5 * (1 / sigma * (bins -
ax.plot(bins, y, '--r')
plt.show()
```



pie

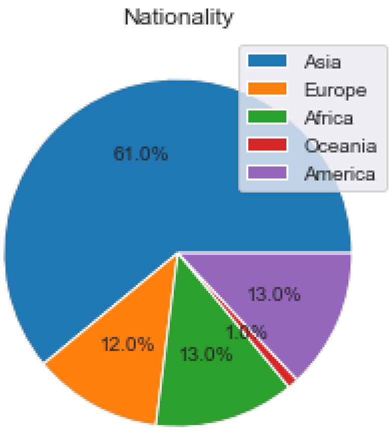
In [178]:

```
data = [61, 12, 13, 1, 13]
plt.pie(data)
plt.show()
```



In [179]:

```
data = [61, 12, 13, 1, 13]
labels = ["Asia", "Europe", "Africa", "Oceania", "America" ]

p = plt.pie(data, autopct='%1.1f%%')

plt.legend(p[0], labels)

plt.title('Nationality')
plt.show()
```

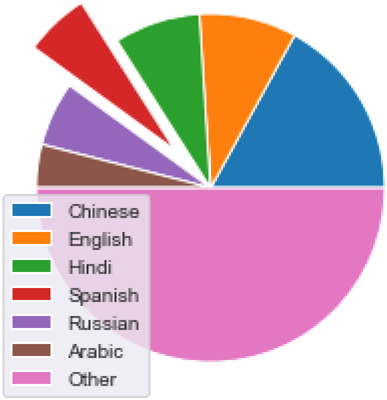


In [180]:

```
#
```

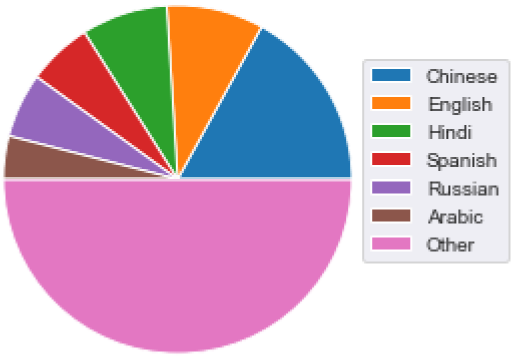
In [181]:

```
labels = ["Chinese", "English", "Hindi", "Spanish", "Russian", "Arabic", "Oth
d = [17, 9, 8, 6, 6, 4, 50]

p = plt.pie(d, explode=[0, 0, 0, 0.3, 0, 0, 0])

plt.legend(p[0], labels)
plt.show()
```



In [182]:

```
labels = ["Chinese", "English", "Hindi", "Spanish", "Russian", "Arabic", "Oth
d = [17, 9, 8, 6, 6, 4, 50]

p = plt.pie(d)

plt.legend(p[0], labels, bbox_to_anchor=(0.9,0.8))
plt.show()
```



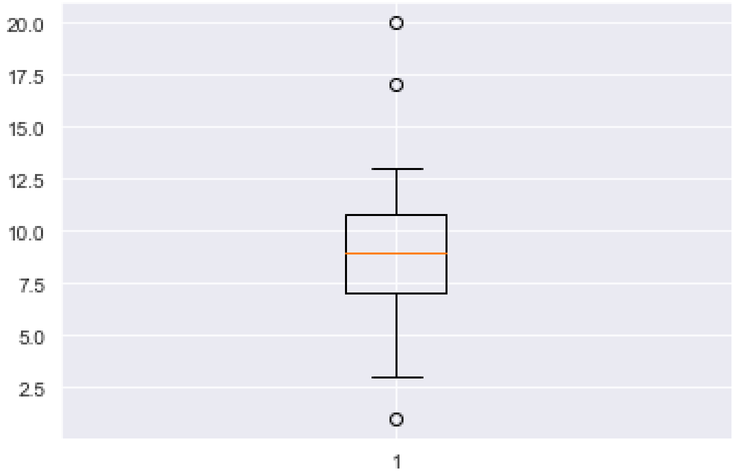
Box Plot

In [183]:

```
score = [1, 3, 6, 7, 7, 7, 8, 9, 9, 10, 10, 10, 13, 13, 17, 20]
```

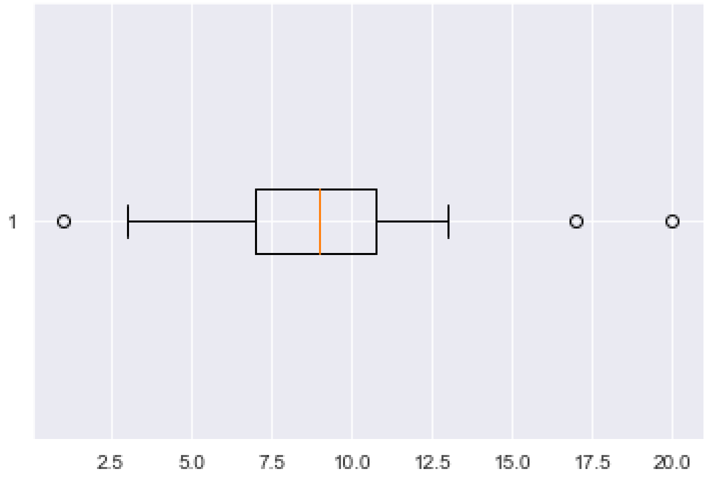
In [184]:

```
plt.boxplot(score)
plt.grid()
plt.show()
```



In [185]:

```
plt.boxplot(score, vert=False)
plt.grid()
plt.show()
```



In [186]:

```
q1 = np.quantile(score, .25)
q1
```

Out[186]:

```
7.0
```

In [187]:

```
q2 = np.quantile(score, .50)
q2
```

Out[187]:

```
9.0
```

In [188]:

```
q3 = np.quantile(score, .75)
q3
```

Out[188]:

```
10.75
```

In [189]:

```
iqr = q3 - q1
iqr
```

Out[189]:

```
3.75
```

In [190]:

```
lv = q1 - 1.5 * iqr
lv
```

Out[190]:

```
1.375
```

In [191]:

```
hv = q3 + 1.5 * iqr
hv
```

Out[191]:

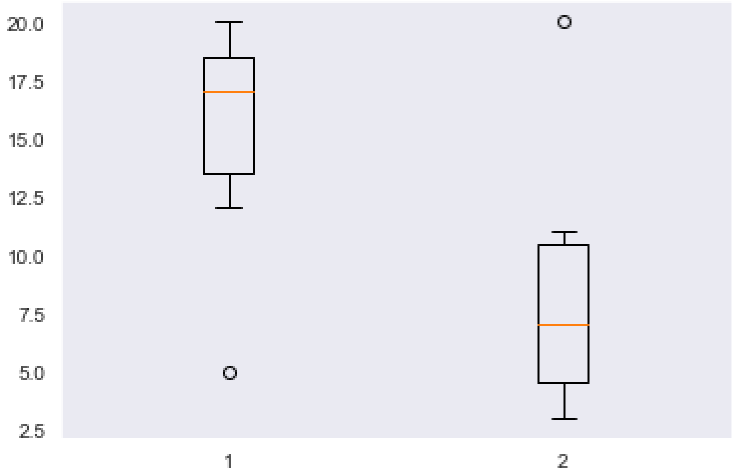
```
16.375
```

In [192]:

```
#
```

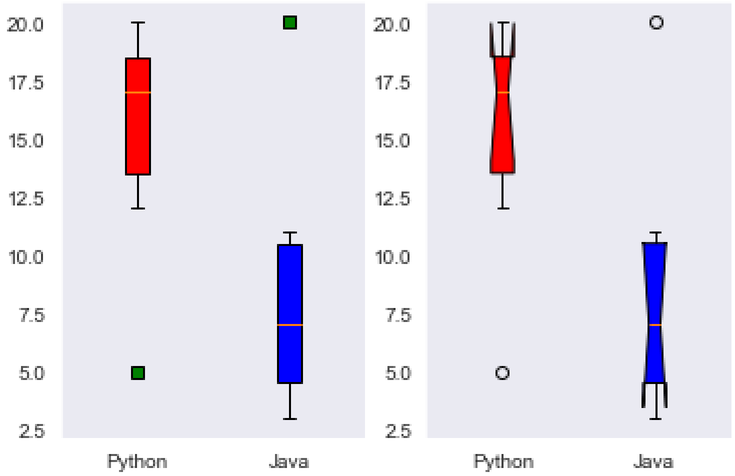
In [193]:

```
score1 = [5, 12, 15, 17, 18, 19, 20]
score2 = [3, 4, 5, 7, 10, 11, 20]
data = np.array(score1),np.array(score2)

plt.boxplot(data)
plt.show()
```



In [194]:

```
fig, (ax1, ax2) = plt.subplots(1, 2)

f = dict(markerfacecolor='g', marker='s')

b1 = ax1.boxplot(data, patch_artist=True, labels=['Python', 'Java'], flierpro
b2 = ax2.boxplot(data, patch_artist=True, labels=['Python', 'Java'], notch=Tr

for b in (b1, b2):
    for p, c in zip(b['boxes'], ['r', 'b']):
        p.set_facecolor(c)

plt.show()
```



Plotting with pandas

In [195]:

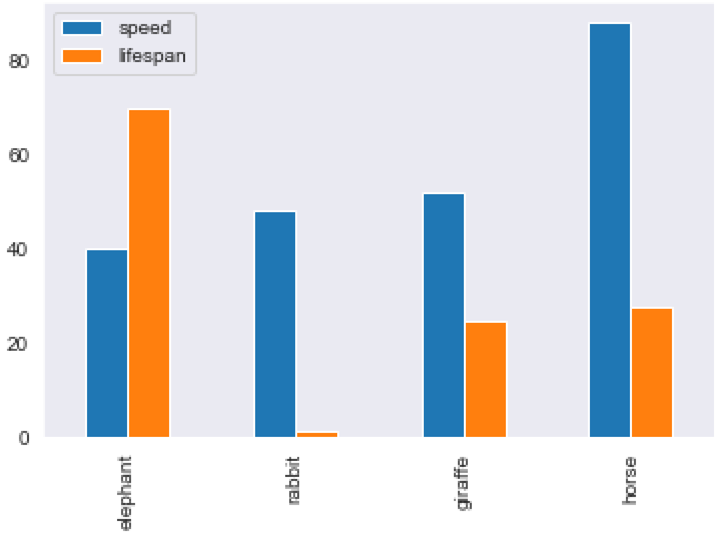
```
d = {'speed': [40, 48, 52, 88], 'lifespan': [70, 1.5, 25, 28]}
df = pd.DataFrame(d, index=['elephant', 'rabbit', 'giraffe', 'horse'])
df
```

Out[195]:

	speed	lifespan
elephant	40	70.0
rabbit	48	1.5
giraffe	52	25.0
horse	88	28.0

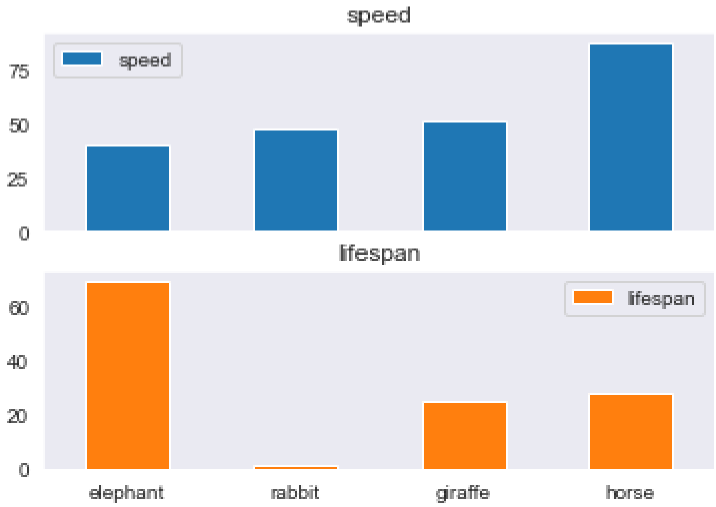
In [196]:

```
df.plot.bar()
plt.show()
```



In [197]:

```
df.plot.bar(subplots=True, rot=0)
plt.show()
```



In [198]:

```
#
```

In [199]:

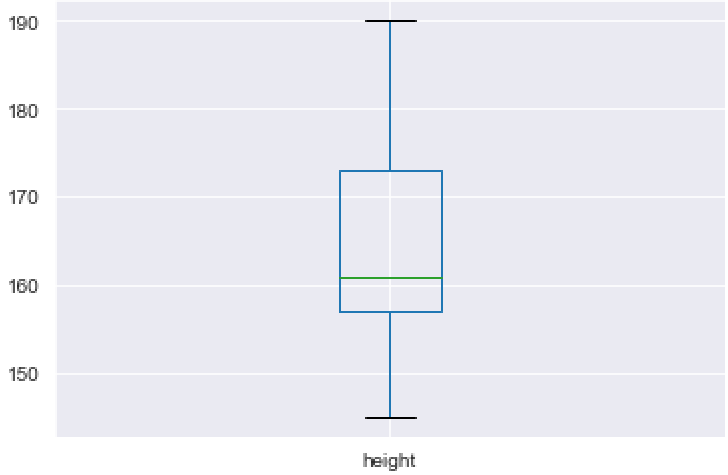
```
data = [
        ['men', 155], ['men', 158], ['men', 160], ['men', 161], ['men', 162],
        ['men', 165], ['men', 169], ['men', 170], ['men', 173], ['men', 175],
        ['men', 180], ['men', 185], ['men', 185], ['men', 190],
        ['women', 145], ['women', 150], ['women', 156], ['women', 157],
        ['women', 158],['women', 150], ['women', 156], ['women', 158],
        ['women', 160], ['women', 161], ['women', 178]
       ]

frame = pd.DataFrame(data, columns = ['gender', 'height'])
frame
```

Out[199]:

	gender	height
0	men	155
1	men	158
2	men	160
3	men	161
4	men	162
5	men	165
6	men	169
7	men	170
8	men	173
9	men	175
10	men	180
11	men	185
12	men	185
13	men	190
14	women	145
15	women	150
16	women	156
17	women	157
18	women	158
19	women	150
20	women	156
21	women	158
22	women	160
23	women	161
24	women	178

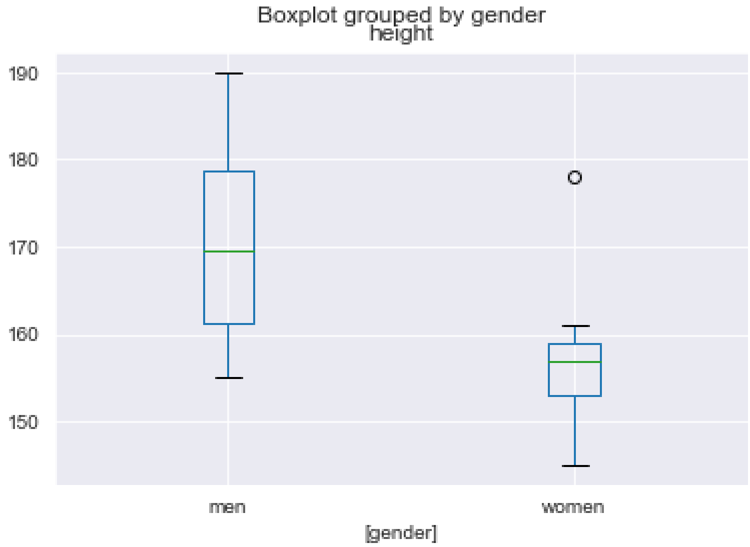
In [200]:

```
frame.boxplot()
plt.show()
```



In [201]:

```
frame.boxplot(by=['gender'])
plt.show()
```



In [202]:

```
#
```

In [203]:

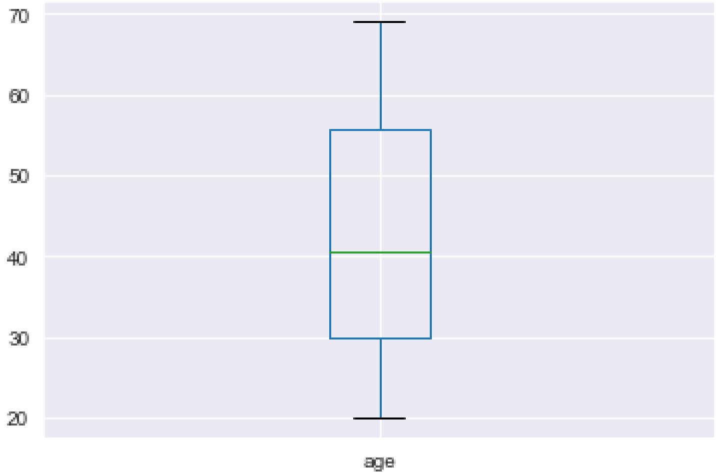
```
g = [random.choice(['male', 'female']) for _ in range(50)]
a = [x for x in np.random.randint(20, 70, 50)]

dfage = pd.DataFrame({'gender': g, 'age':a})
dfage.head()
```

Out[203]:

	gender	age
0	female	49
1	male	33
2	male	54
3	male	33
4	female	30

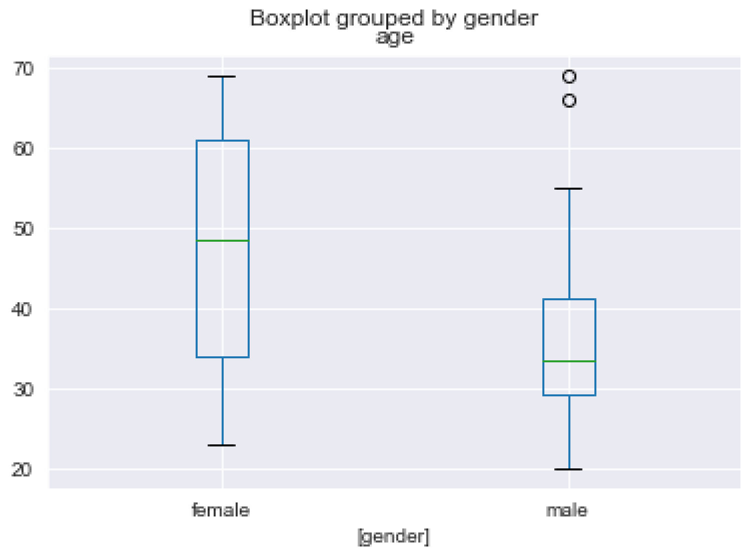
In [204]:

```
dfage.boxplot()
plt.show()
```



In [205]:

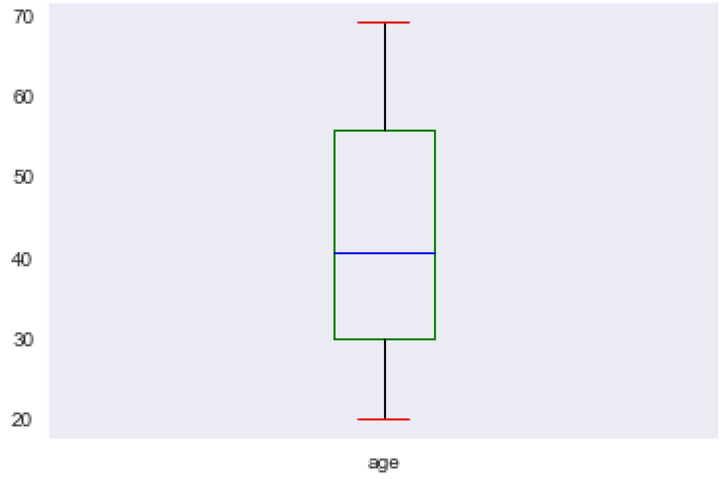
```
dfage.boxplot(by=['gender'])
plt.show()
```



plot.box

In [206]:

```
c = {'boxes': 'Green',
     'whiskers': 'black',
     'medians': 'Blue',
     'caps': 'r'}

dfage.plot.box(color=c)
plt.show()
```



seaborn

In [207]:

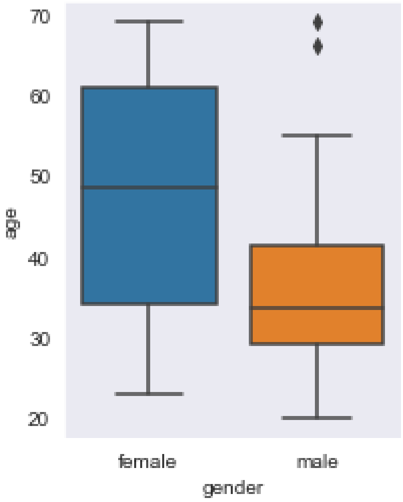
```
#!pip install seaborn
```

In [208]:

```
import seaborn as sns
```

In [209]:

```
plt.figure(figsize=(3, 4))
sns.boxplot(x='gender', y='age', data=dfage)
plt.show()
```



In [210]:

```
# iris
```

In [211]:

```
df = sns.load_dataset('iris')
df
```

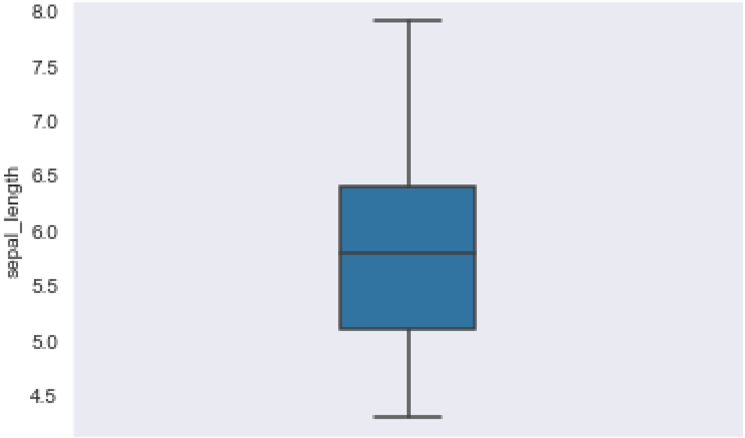
Out[211]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

150 rows × 5 columns

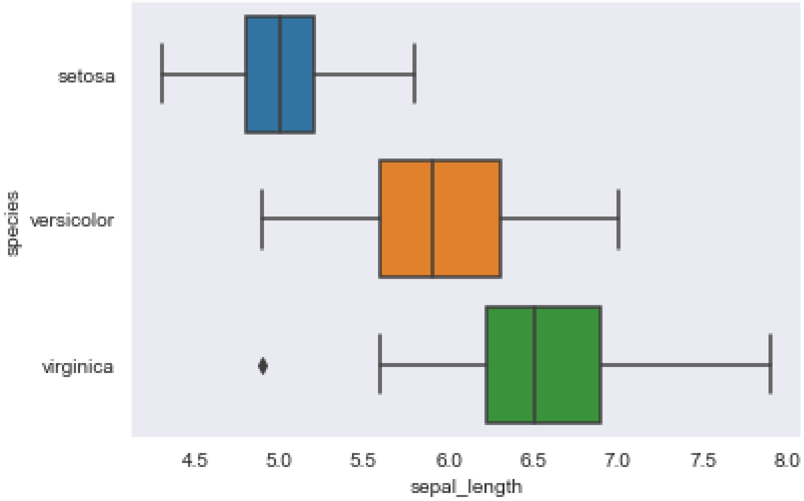
In [212]:

```
sns.boxplot(y=df["sepal_length"], width=0.2);
```

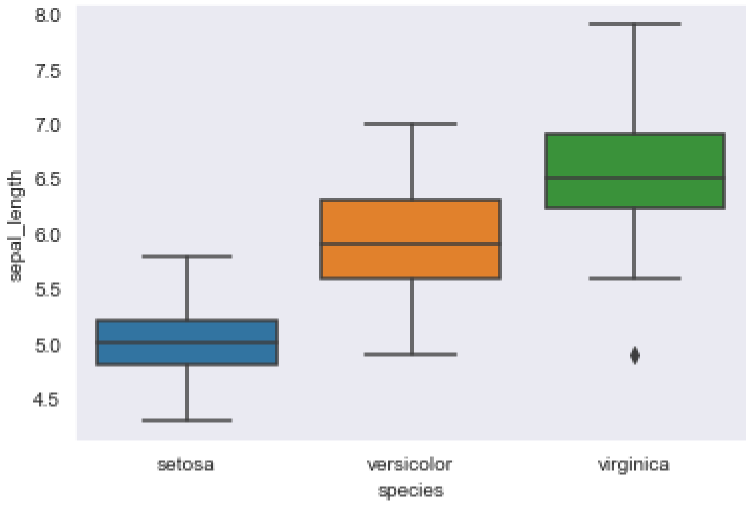


In [213]:

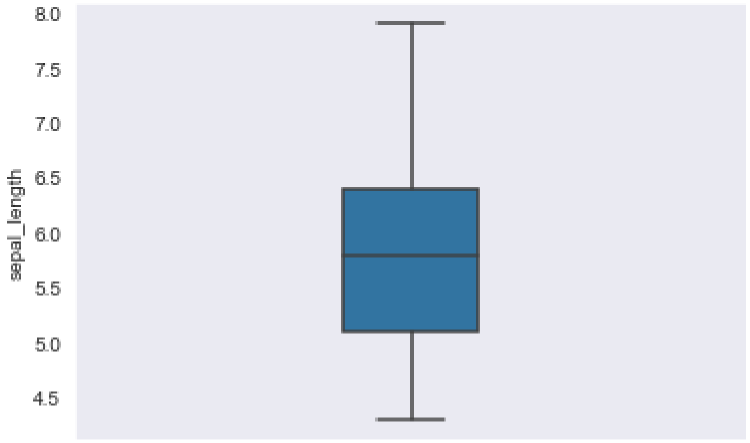
```
sns.boxplot( x=df["sepal_length"], y=df["species"] );
```



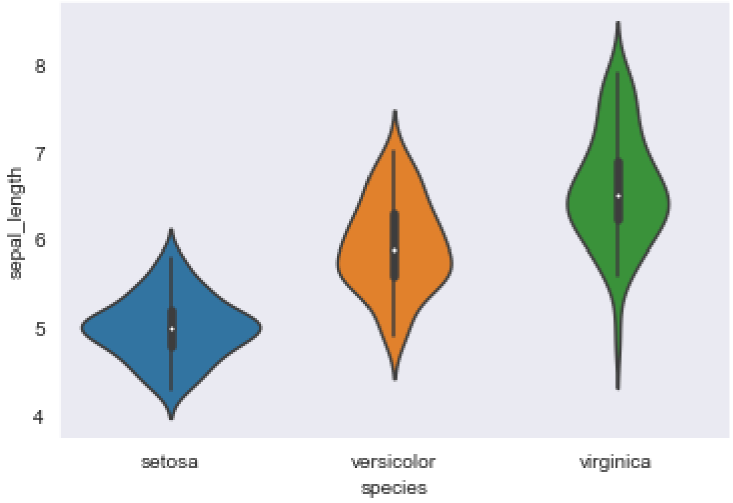
In [214]:
```
sns.boxplot( x=df["species"], y=df["sepal_length"]);  # , palette="Blues" ,de
```



In [215]:
```
sns.set_style("dark")  # dark , whitegrid
sns.boxplot(y=df["sepal_length"], width=0.2);
```



In [216]:
```
sns.violinplot(x=df["species"], y=df["sepal_length"]);
```



دانشگاه شهید مدنی آذربایجان
برنامه نویسی پیشرفته با پایتون
امین گلزاری اسکوئی
١٤٠٠-١٤٠١

Codes and Projects (click here) (https://github.com/Amin-Golzari-Oskouei/Python-Programming-Course-Advanced-2021) slides and videos (click here) (https://drive.google.com/drive/folders/1Dx3v7fD1QBWL-MNP2hd7iIxaRbeALkkA)