Data Visualization - Trực quan hóa dữ liệu là phương pháp truyền đạt dữ liệu, thông tin dưới dạng đồ thị, biểu đồ trực quan sinh động.

Trực quan hóa dữ liệu thường được sử dụng sau khi có kết quả phân tích từ dữ liệu tức là đã có thông tin để trình bày cho người dùng. Ngoài ra Trực quan hóa dữ liệu còn được sử dụng trước khi dữ liệu được đưa vào giai đoạn phân tích để có thể hiểu về các biến dữ liệu, mối liên hệ giữa chúng từ đó có những quyết định phân tích hiệu quả hơn.

Thực hành Trực quan hóa dữ liệu.

Khai báo các thư viện:

- pandas: Làm việc với cấu trúc dữ liệu, hỗ trợ đọc ghi file dữ liệu với nhiều định dạng khác nhau như csv, xls, sql, html...
- matplotlib: Thư viện hỗ trợ vẽ các đồ thị trong Python
- Numpy cung cấp các đối tượng và phương thức để làm việc với mảng nhiều chiều và các phép toán đại số tuyến tính.

```
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
import numpy as np
```

Đọc file dữ liệu "Work_data.csv" khảo sát mức lương của nhân viên có kinh nghiệm từ 0-10 năm của một số ngành vào Dataframes df của python. Dataframes được cấu tạo như một mảng hai chiều gồm các cột và các dòng.

```
df = pd.read_csv("Work_data.csv")
```

```
df.shape   #Số lượng mẫu và Số thuộc tính quan sát
```

```
    (79, 3)
```

```
df.head(10) # Xem 10 dòng dữ liệu đầu tiên
```

	SoNamKinhNghiem	Luong	NganhNghe
0	7	26.0	KeToan
1	4	13.8	KeToan
2	8	21.5	KeToan
3	9	24.0	KeToan
4	1	7.8	KeToan
5	2	10.0	KeToan
6	4	13.5	KeToan
7	5	15.8	KeToan

```
print(df)  # Xem toàn bộ dữ liệu
```

```
    SoNamKinhNghiem  Luong NganhNghe
0                 7   26.0    KeToan
1                 4   13.8    KeToan
2                 8   21.5    KeToan
3                 9   24.0    KeToan
4                 1    7.8    KeToan
..              ...    ...       ...
74                8   22.5      Sale
75                5   16.7      Sale
76                9   25.4      Sale
77                6   18.9      Sale
78                8   22.7      Sale

[79 rows x 3 columns]
```

Xem các thông tin thống kê đối với các dữ liệu định lượng:

- count: tổng số các bản ghi trong dữ liệu
- min, max: giá trị lớn nhất nhỏ nhất của dữ liệu
- mean: giá trị trung bình
- std: độ lệch chuẩn, giá trị STD lớn hay nhỏ phản ánh tập dữ liệu phân bố tập trung quanh điểm trung tâm hay rời xa nó (ở đây là mean)

```
df.describe() #
```

	SoNamKinhNghiem	Luong
count	79.000000	79.000000
mean	5.367089	17.355696
std	3.105732	6.313391
25%	3.000000	12.600000
max	10.000000	27.600000

Đổi tên các thuộc tính cần dùng về dạng đơn giản giúp dễ dàng truy cập sau này

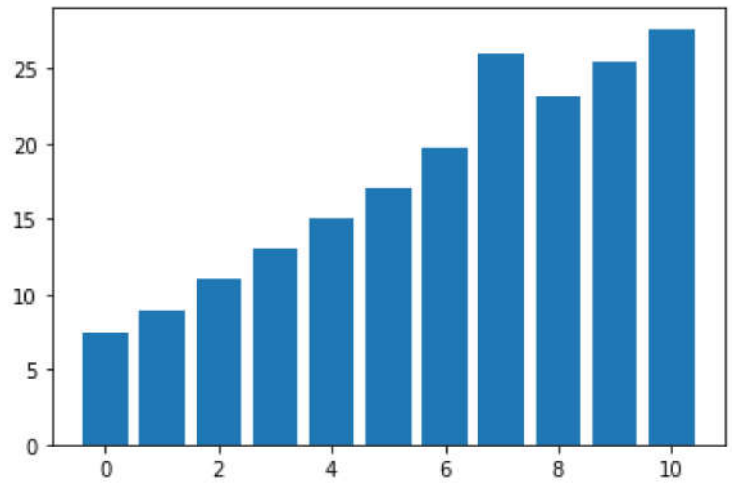
```
nam = df['SoNamKinhNghiem']
luong = df['Luong']
nghe = df['NganhNghe']
```

Vẽ đồ thị, biểu đồ

Biểu đồ cột/thanh dọc Bar chart

- Mối quan hệ giữa số năm kinh nghiệm và lương

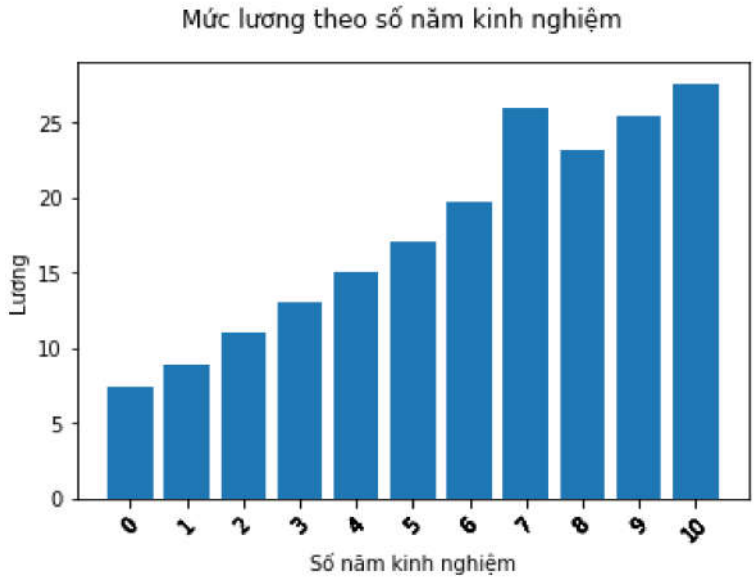
```
plt.bar(nam, luong)
plt.show()
```



Thêm nhãn cho biểu đồ

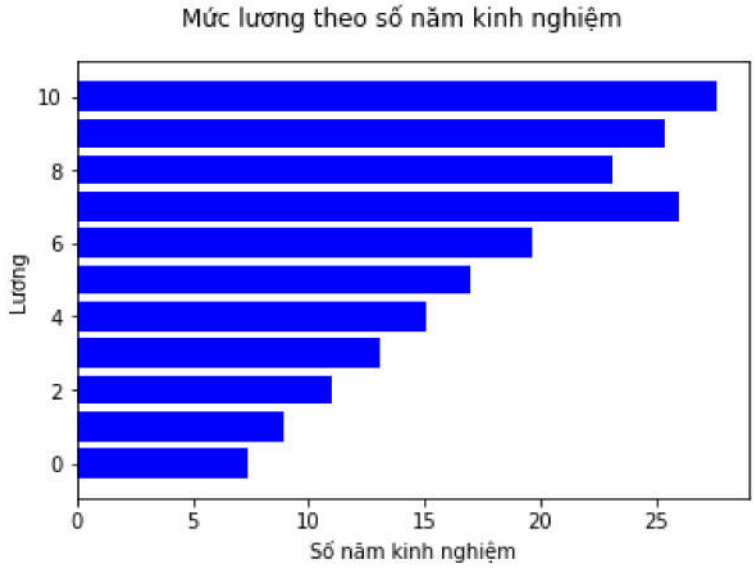
```
plt.xlabel('Số năm kinh nghiệm')
plt.ylabel('Lương')
plt.suptitle('Mức lương theo số năm kinh nghiệm')

# Hiển thị nhãn của các năm
plt.xticks(rotation=45)
```

```
plt.xticks(nam)

plt.bar(nam, luong)
plt.show()
```



Biểu đồ cột ngang

```
plt.xlabel('Số năm kinh nghiệm')
plt.ylabel('Lương')
plt.suptitle('Mức lương theo số năm kinh nghiệm')
plt.barh(nam, luong, color = 'blue')
plt.show()
```



Vẽ nhiều đồ thị trong cùng một ảnh

- Ví dụ vẽ ba đồ thị về mối quan hệ giữa lương và số năm kinh nghiệm của từng ngành nghề

Bước 1: Tách bảng dữ liệu ban đầu thành các bảng con theo các tiêu chí

- Ví dụ tách bảng dữ liệu thành ba bảng con theo ngành nghề

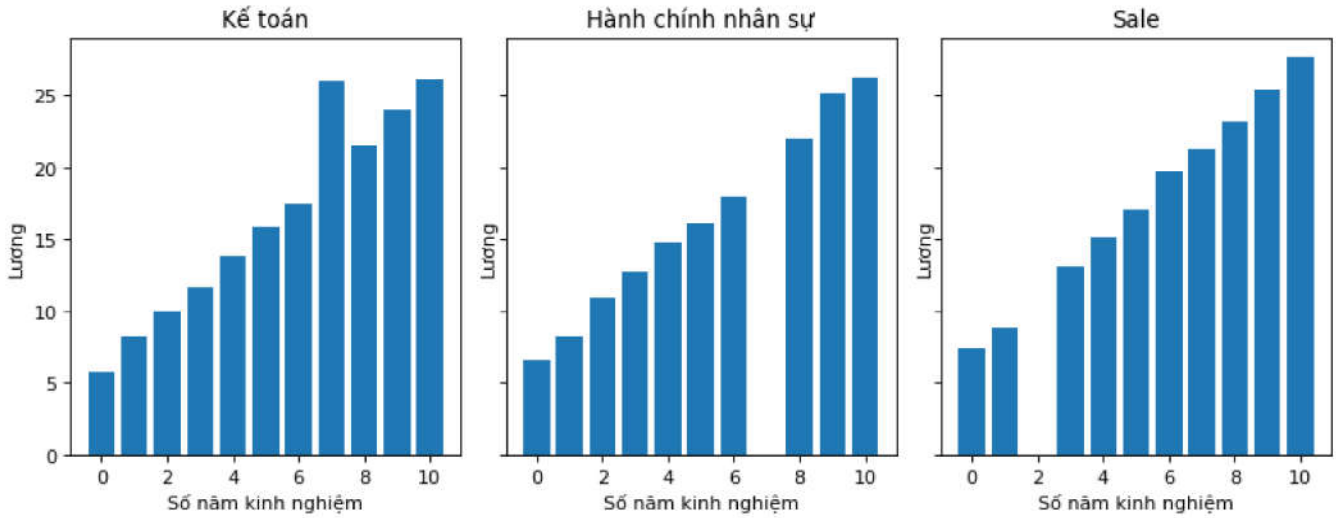
```
df_ketoan = df[df['NganhNghe'] == 'KeToan']
df_hcns = df[df['NganhNghe'] == 'HCNS']
df_sale = df[df['NganhNghe'] == 'Sale']

print ("Số lượng mẫu nhân viên Kế toán: " + str(df_ketoan.shape[0]))
print ("Số lượng mẫu nhân viên HCNS: " + str(df_hcns.shape[0]))
print ("Số lượng mẫu nhân viên SALE: " + str(df_sale.shape[0]))

print("Dữ liệu bảng kế toán")
print(df_ketoan)
```

```
Số lượng mẫu nhân viên Kế toán: 19
Số lượng mẫu nhân viên HCNS: 20
Số lượng mẫu nhân viên SALE: 40
Dữ liệu bảng kế toán
    SoNamKinhNghiem  Luong NganhNghe
0                 7   26.0    KeToan
1                 4   13.8    KeToan
2                 8   21.5    KeToan
3                 9   24.0    KeToan
4                 1    7.8    KeToan
5                 2   10.0    KeToan
6                 4   13.5    KeToan
7                 5   15.8    KeToan
8                 6   17.5    KeToan
9                10   26.1    KeToan
10                1    8.3    KeToan
11                9   23.9    KeToan
12                0    5.8    KeToan
13                7   19.9    KeToan
14                4   13.8    KeToan
15                3   11.6    KeToan
16                3   11.7    KeToan
17               10   25.8    KeToan
18                2    9.6    KeToan
```

Bước 2: Vẽ 3 đồ thị Trong Matplotlib, mỗi plt.plot() trả về một đối tượng Figure (là hình ảnh bên ngoài), trong Figure lại có thể có nhiều các đối tượng Axes là các đồ thị con bên trong.

```
fig, (ax1, ax2, ax3) = plt.subplots(1,3, figsize=(10,4), sharey=True, dpi=80) # Tạo đồ thị gồ

ax1.bar(df_ketoan['SoNamKinhNghiem'],df_ketoan['Luong'])
ax2.bar(df_hcns['SoNamKinhNghiem'],df_hcns['Luong'])
ax3.bar(df_sale['SoNamKinhNghiem'],df_sale['Luong'])
```

```
# Tiêu đề và nhãn của các đồ thị con
ax1.set_title('Kế toán')
ax2.set_title('Hành chính nhân sự')
ax3.set_title('Sale')

ax1.set_xlabel('Số năm kinh nghiệm')
ax2.set_xlabel('Số năm kinh nghiệm')
ax3.set_xlabel('Số năm kinh nghiệm')

ax1.set_ylabel('Lương')
ax2.set_ylabel('Lương')
ax3.set_ylabel('Lương')


plt.tight_layout()
plt.show()
```



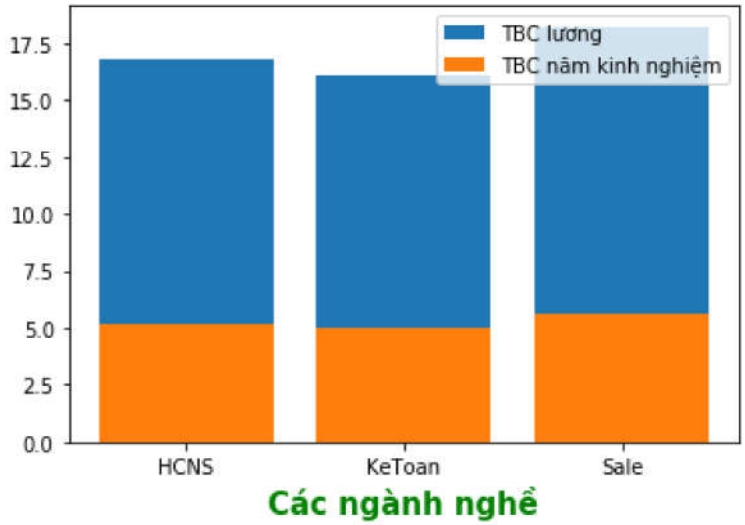
Biểu đồ thanh xếp chồng - Stacked Bar Chart

- Vẽ biểu đồ thanh xếp chồng của trung bình cộng lương và trung bình cộng số năm kinh nghiệm theo các ngành nghề

```
tbc_luong_nghe = df.groupby('NganhNghe')[['Luong']].mean()
tbc_nam_nghe = df.groupby('NganhNghe')[['SoNamKinhNghiem']].mean()
print("TBC lương theo nghề",'\n', tbc_luong_nghe, '\n')
print("TBC năm kinh nghiệm theo nghề",'\n',tbc_nam_nghe)

plt.xlabel('Các ngành nghề', color = 'green', fontweight = 'bold', fontsize = '15')

plt.bar(tbc_luong_nghe.index,tbc_luong_nghe['Luong'], label='TBC lương')
plt.bar(tbc_nam_nghe.index,tbc_nam_nghe['SoNamKinhNghiem'], label='TBC năm kinh nghiệm')
```

```
plt.legend()  # Hiển thị nhãn label
plt.show()
```

```
TBC lương theo nghề
                Luong
NganhNghe
HCNS       16.790000
KeToan     16.126316
Sale       18.222500

TBC năm kinh nghiệm theo nghề
            SoNamKinhNghiem
NganhNghe
HCNS                  5.200
KeToan                5.000
Sale                  5.625
```



Biểu đồ thanh nhóm - Grouped Bar Chart

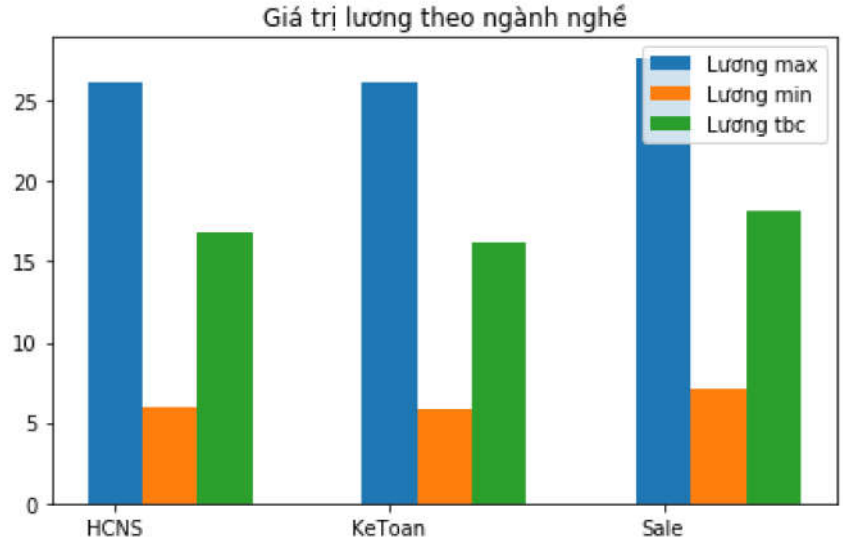
Cho phép các thanh đứng liền nhau

- Vẽ biểu đồ minh họa các giá trị lương lớn nhất, nhỏ nhất, trung bình theo mỗi ngành nghề

```
# Tính lương lớn nhất, nhỏ nhất, trung bình theo mỗi ngành nghề
max_luong_nghe = df.groupby('NganhNghe')[['Luong']].max()
min_luong_nghe = df.groupby('NganhNghe')[['Luong']].min()
tbc_luong_nghe = df.groupby('NganhNghe')[['Luong']].mean()

x = np.arange(len(max_luong_nghe.index))  # Tính vị trí hiển thị của các nhóm. Kết quả trả về
                                          # Nhóm 1 có vị trí trên trục x từ 0-1, nhóm 2 từ vị
width = 0.2  # Kích thước các thanh

fig, ax = plt.subplots()
ax.bar(x , max_luong_nghe['Luong'], width, label='Lương max')
```

```
ax.bar(x + 0.2,  min_luong_nghe['Luong'], width, label='Lương min')
ax.bar(x + 0.4,  tbc_luong_nghe['Luong'], width, label='Lương tbc')

# Add some text for labels, title and custom x-axis tick labels, etc.

ax.set_title('Giá trị lương theo ngành nghề')
ax.set_xticks(x)
ax.set_xticklabels(max_luong_nghe.index)  # Thiết lập chuỗi các nhãn
ax.legend()

fig.tight_layout()

plt.show()
```

```
[0 1 2]
```



Biểu đồ hộp

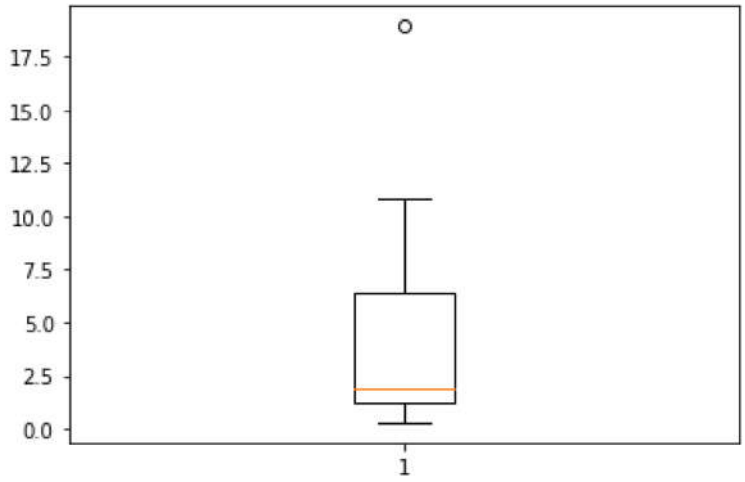
Biểu đồ hộp (Box-plot) hay còn gọi là biểu đồ hộp-và-râu (box-and-whisker plot) là biểu đồ diễn tả 5 vị trí phân bố của dữ liệu, đó là: giá trị nhỏ nhất không ngoại lai (râu dưới) (L), tứ phân vị thứ nhất (Q1), trung vị (median), tứ phân vị thứ 3 (Q3) và giá trị lớn nhất không ngoại lai (râu trên) (U).

Biểu đồ hộp cho biết phân bố của dữ liệu và xác định các giá trị ngoại lai:

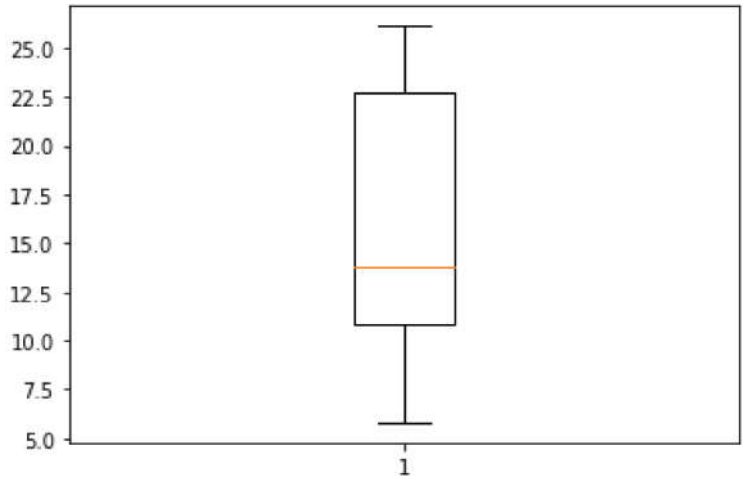
- Nếu đường trung vị chia chiếc hộp thành 2 nửa đều nhau, thì tập dữ liệu này đối xứng (symmetric). Nếu nửa dưới nhỏ hơn nửa trên thì tập dữ liệu bị lệch phải (right-skewed), và ngược lại, nếu nửa dưới lớn hơn thì tập dữ liệu bị lệch trái (left-skewed).
- Các giá trị ngoại lai (nếu có) sẽ xuất hiện bên ngoài phía dưới râu dưới và phía trên râu trên

```
# Vẽ biểu đồ hộp mô tả tỉ lệ CO2 bình quân đầu người của 8 quốc gia đông dân số nhất trên thế
nuoc = np.array(['china','india','us','indonesia','brazil','pakistan','russia','bangladesh'])
co2 = np.array([4.9,1.4,18.9,1.8,1.9,0.9,10.8,0.3])

plt.boxplot(co2)
```

```
plt.show()

#Ta có kết quả như sau:
#1.       Min = 0.3
#2.       Q1 = 1.275
#3.       Trung vị median= 1.85
#4.       Q3= 6.375
#5.       Max = 18.9
#6.       IQR = Q3-Q1= 5.1
#7.       Giá trị thấp của biến L = Q1-1,5xIQR = -6.375
#8.       Giá trị cao của biến U = Q3 + 1,5xIQR = 14.025
#9.       Từ (7) và (8), ta suy ra us = 18.9 là một giá trị ngoại biên.
```




```
# Biểu đồ hộp phân bố lương của nhân viên Kế toán
plt.boxplot(df_ketoan['Luong'])

plt.show()
```

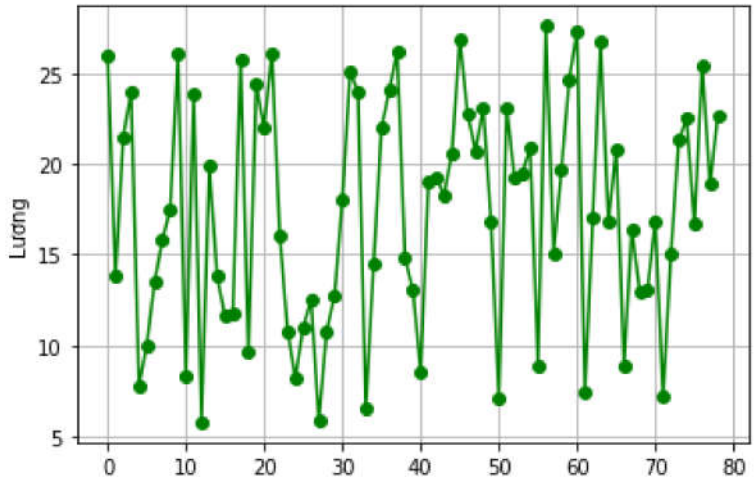


Biểu đồ đường Line Plot

Được sử dụng để mô tả xu hướng biến động tăng hay giảm của dữ liệu

- Vẽ biểu đồ đường của Lương theo 79 dòng dữ liệu
- Một số định dạng:
    - 'go-': các điểm có màu xanh (g) và nối hai điểm là đường thẳng (có thể thay màu và kiểu đường ví dụ 'yo-.')
    - 'go': chỉ hiển thị các điểm của đồ thị
    - 'r*--' các điểm hình ngôi sao màu đỏ, đường nối các điểm dạng –.
    - 'bD-.' các điểm hình kim cương màu xanh dương, đường nối các điểm dạng -.
    - 'g^-' các điểm hình tam giác hướng lên màu xanh lá, đường nối các điểm dạng -.

```
plt.ylabel('Lương')
plt.grid()
plt.plot(luong, 'go-')
plt.show()
```



Vẽ biểu đồ đường mô tả mối quan hệ giữa số năm kinh nghiệm và mức lương trung bình

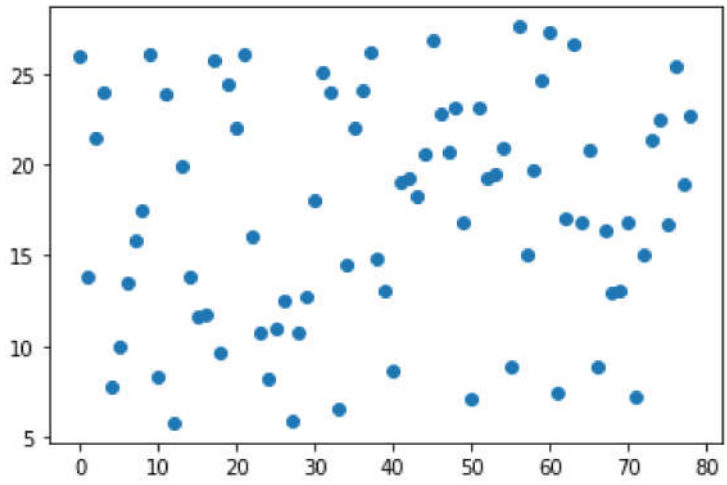
```
tbc_luong_nam = df.groupby('SoNamKinhNghiem')[['Luong']].mean()
plt.xlabel('Số năm kinh nghiệm')
plt.ylabel('Lương trung bình')
plt.grid()
plt.plot(tbc_luong_nam, 'ro-')
plt.show()
```

Biểu đồ phân tán Scatter

Biểu đồ phân tán mô tả sự phân bố của dữ liệu trong không gian hai chiều

- Vẽ biểu đồ mô tả sự phân bố của lương. Do chỉ xét biến lương lên thêm biến df.index là id của các mẫu trong bảng dữ liệu df

```
plt.scatter(df.index,luong)
plt.show()
```

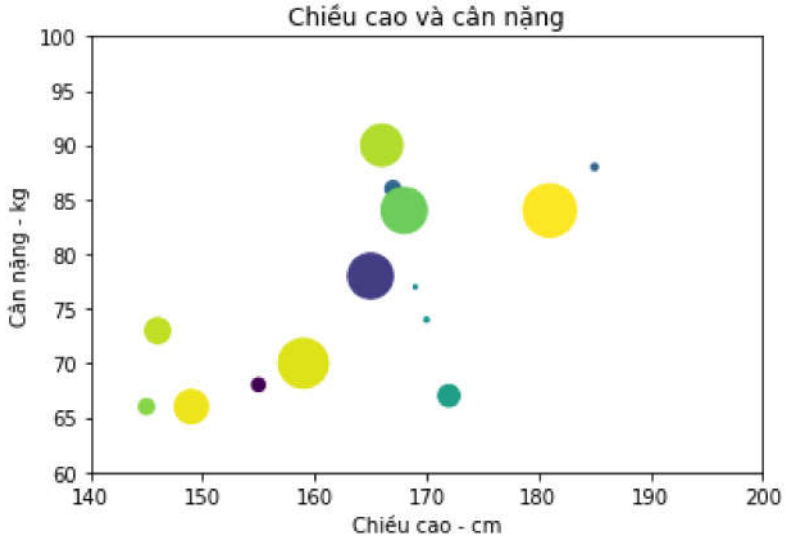


Sự khác biệt giữa plot() và scatter():

plot() không có khả năng thay đổi màu và kích thước điểm trong tập hợp điểm ban đầu nhưng scatter() lại có thể. plot() có thể vẽ các đường nối hai điểm liên tiếp, scatter() thì không. Ví dụ dưới đây vẽ ra các điểm trên đồ thị với dữ liệu về chiều cao và cân nặng, mỗi điểm có màu ngẫu nhiên và có kích thước cũng ngẫu nhiên.

```
height = np.array([167,170,149,165,155,180,166,146,159,185,145,168,172,181,169])
weight = np.array([86,74,66,78,68,79,90,73,70,88,66,84,67,84,77])

colors = np.random.rand(15)  # Sinh ngẫu nhiên 15 giá trị colors
area = (30 * np.random.rand(15))**2

plt.xlim(140,200)   # Giới hạn trục x có giá trị từ 140 đến 200cm
plt.ylim(60,100)    # Giới hạn trục y có giá trị từ 60 đến 100kg
plt.scatter(height, weight, s=area, c=colors)
plt.title("Chiều cao và cân nặng")
plt.xlabel("Chiều cao - cm")
```

```
plt.ylabel("Cân nặng - kg")

plt.show()
```



Bài tập về nhà:

1. Vẽ biểu đồ thanh mỗi cột một màu
2. Vẽ biểu đồ tròn mô tả mối quan hệ giữa mức lương trung bình theo ngành nghề
3. Vẽ biểu đồ hộp so sánh đồng thời lương của nhân viên Kế toán, HCNS và Sale
4. Tìm hiểu thêm các loại biểu đồ khác trong python