2274802010480_TranKhanhLinh

2.PANDAS

```python
#LAB3
import pandas as pd
import numpy as np
data = {
    "ID":[101,102,103,104,105],
    "Name":["Tuan Kiet","Thu Hang",None,"Phuc Nguyen","Viet
    "Age":[26,18,20,None,19],
    "Salary":[50000,35000,6500,10000,None]
}
#show
df = pd.DataFrame(data)
print(df)
```

[17] Python

```
    ID         Name   Age   Salary
0  101    Tuan Kiet  26.0  50000.0
1  102     Thu Hang  18.0  35000.0
2  103         None  20.0   6500.0
3  104  Phuc Nguyen   NaN  10000.0
4  105      Viet Vu  19.0      NaN
```

```python
#Loại bỏ dòng chứa dữ liệu thiếu
df.dropna(inplace=True)
print(df)
```

[22] Python

```
    ID         Name    Age   Salary
0  101    Tuan Kiet  26.00  50000.0
1  102     Thu Hang  18.00  35000.0
2  103      Unknown  20.00   6500.0
3  104  Phuc Nguyen  20.75  10000.0
4  105      Viet Vu  19.00  10000.0
```

```python
#Điền giá trị thiếu trong Name = "Unknown"
df["Name"]=df["Name"].fillna("Unknown",inplace=True)
print(df)
```

[20] Python

```
    ID         Name    Age   Salary
0  101    Tuan Kiet  26.00  50000.0
1  102     Thu Hang  18.00  35000.0
2  103      Unknown  20.00   6500.0
3  104  Phuc Nguyen  20.75  10000.0
4  105      Viet Vu  19.00      NaN
```

```python
#Điền giá trị thiếu trong Salary = phương pháp loại suy Int
df["Salary"]=df["Salary"].interpolate(method="linear",inpla
print(df)
```

Python

```
   ID         Name    Age   Salary
0  101    Tuan Kiet  26.00  50000.0
1  102     Thu Hang  18.00  35000.0
2  103      Unknown  20.00   6500.0
3  104  Phuc Nguyen  20.75  10000.0
4  105      Viet Vu  19.00  10000.0
```

```python
#cho data
df1 = pd.DataFrame({
    "id":[1,2,3],
    "score_A":[70,90,85],
})
df2 = pd.DataFrame({
    "id":[3,4,5],
    "score_A":[62,91,75],
})
```

```python
#Thực hiện Merge trên cột id (inner join, left join, outer

df3 = pd.merge(df1,df2,on="id",how="inner").reset_index()
print(df3)
```

Python

```
   index  id  score_A_x  score_A_y
0      0   3         85         62
```

```python
#Nối DataFrame theo chiều dọc
df4 = pd.concat([df1,df2],axis=0).reset_index()
print(df4)
```

Python

```
   index  id  score_A
0      0   1       70
1      1   2       90
2      2   3       85
3      0   3       62
4      1   4       91
5      2   5       75
```

```python
#Gộp DF1 và DF2 để điền giá trị thiếu
df1.set_index("id").combine_first(df2.set_index("id")).rese
```

Python

	id	score_A
0	1	70
1	2	90
2	3	85
3	4	91
4	5	75

```python
#Cho dữ liệu sau:
data = pd.DataFrame({
    "id" : range(1,100001),
    "value" : np.random.randint(0,100,100000)
})
```

Python

```python
    #Dùng .astype để tối ưu hóa bộ nhớ
    data["id"] = data["id"].astype("int32")
    data["value"] = data["value"].astype("int16")
    data.info()
    #Tìm 5 giá trị phổ biến trong cột value
    data["value"].value_counts().head(5)
    #Sử dụng query để lọc dữ liệu nhanh hơn df[df["value"] > 90
    data.query("value > 90")
```

Python

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100000 entries, 0 to 99999
Data columns (total 2 columns):
 #   Column  Non-Null Count   Dtype
---  ------  --------------   -----
 0   id      100000 non-null  int32
 1   value   100000 non-null  int16
dtypes: int16(1), int32(1)
memory usage: 586.1 KB
```

	id	value
0	1	95
6	7	91
46	47	95
93	94	92
106	107	98

3.Matplotlib

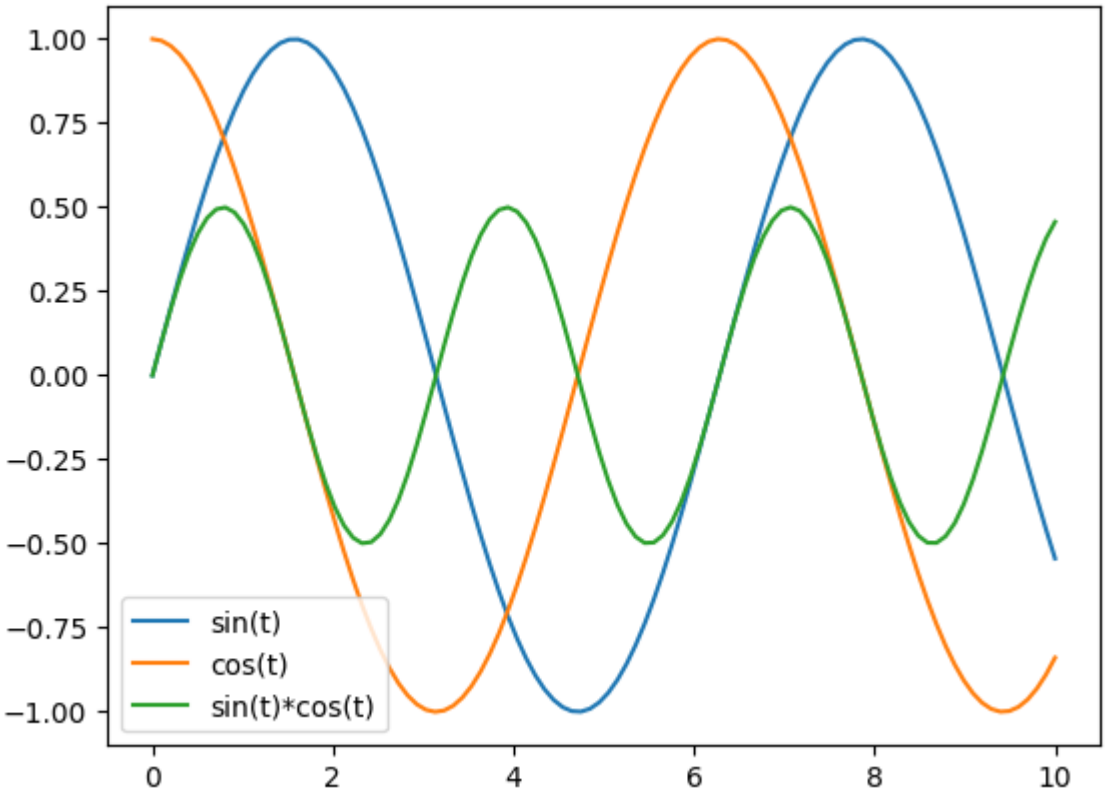
```
# Biểu đồ nhiều đường

#Cho dữ liệu sau

import matplotlib.pyplot as plt
import numpy as np

t = np.linspace(0,10,100)
y1 = np.sin(t)
y2 = np.cos(t)
y3 = np.sin(t) * np.cos(t)
```

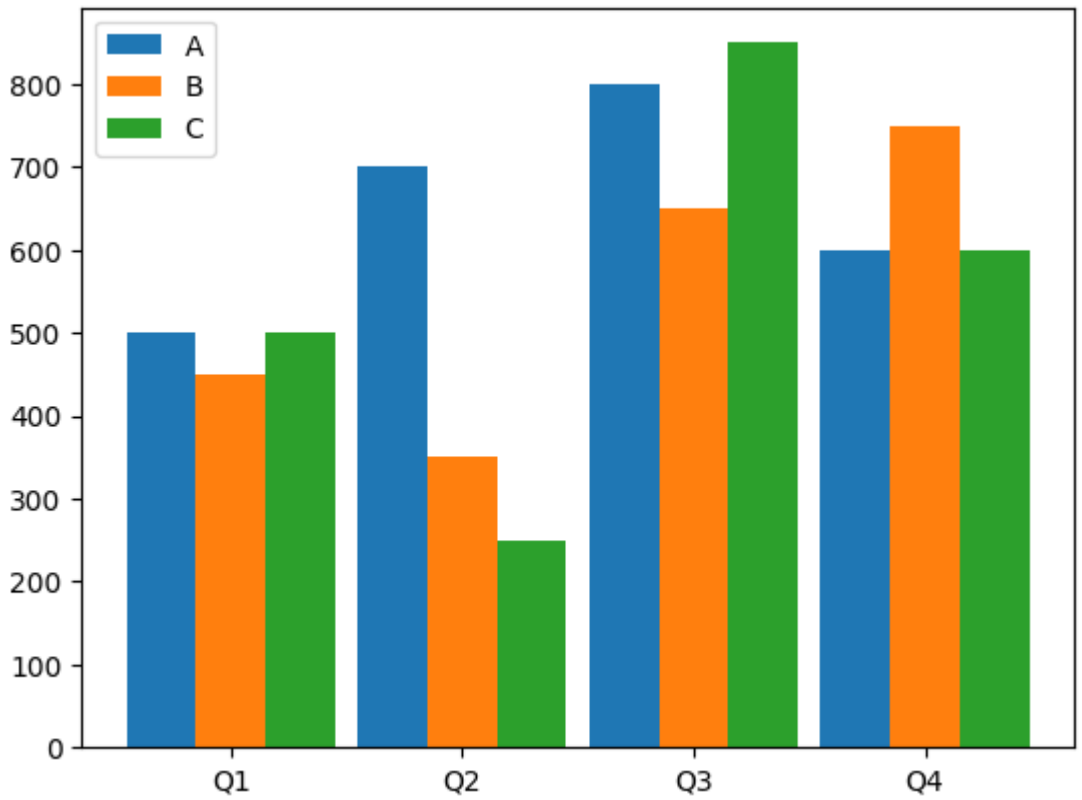
```
#tạo biểu đồ các đường theo thời gian
plt.plot(t,y1,label="sin(t)")
plt.plot(t,y2,label="cos(t)")
plt.plot(t,y3,label="sin(t)*cos(t)")
plt.legend()
plt.show()
```




```
#tạo biểu thanh nhóm
# cho dữ liệu sau
labels = ["Q1","Q2","Q3","Q4"]
A = [500,700,800,600]
B = [450,350,650,750]
C = [500,250,850,600]
```

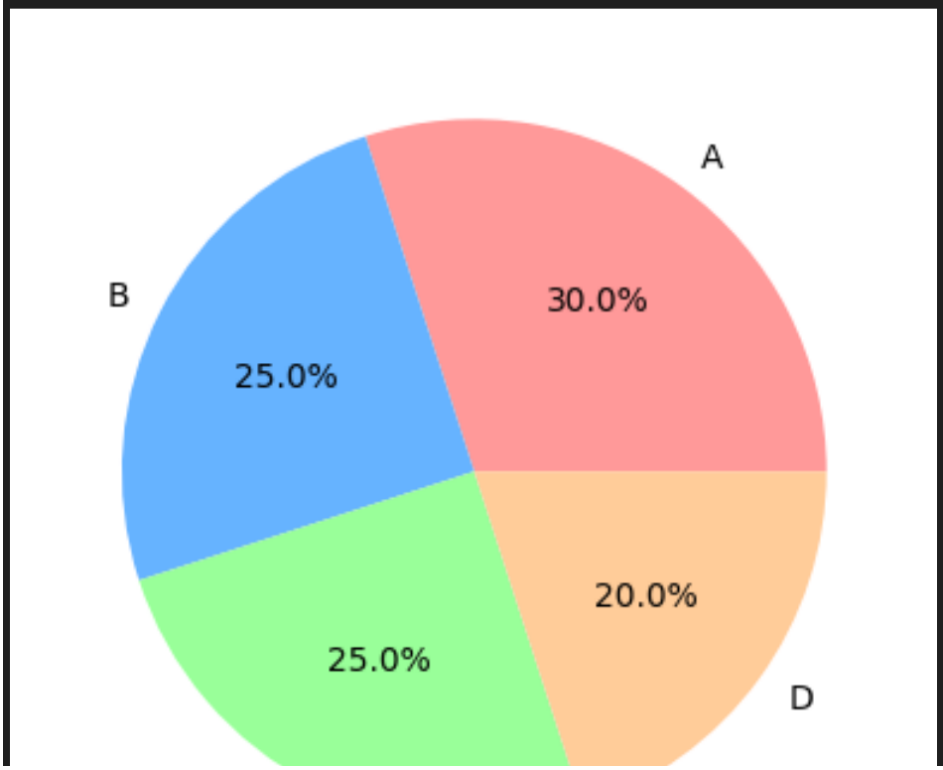
Pytho

```
# vẽ biểu đồ thanh nhóm thể hiện doanh thu của A B C trong
x = np.arange(len(labels))
width = 0.3
plt.bar(x - width, A, width, label="A")
plt.bar(x, B, width, label="B")
plt.bar(x + width, C, width, label="C")
plt.xticks(x, labels)
plt.legend()
plt.show()
```


A
B
C
800
700
600
500
400
300
200
100
0
Q1
Q2
Q3
Q4

```
#Biểu đồ tròn
Cty = ["A","B","C","D"]
thiphan = [30,25,25,20]
color = ["#ff9999","#66b3ff","#99ff99","#ffcc99"]
```

```
#Tạo biểu đồ tròn hiển thị tỉ lệ thị phần của 4 công ty
plt.pie(thiphan, labels=Cty, colors=color, autopct="%1.1f%%")
plt.show()
```



```
# Biểu đồ phân tán
x = np.random.randn(100)
y = np.random.randn(100)
sizes = np.random.randint(100)*300
colors = np.random.rand(100)
```

```
# tạo biểu đồ phân tán của hai biến ngẫu nhiên x và y
plt.scatter(x,y,s=sizes/100,c=colors,alpha=0.5)
plt.show()
```