Chapter 7 - Exercise 1: Tips

Cho dữ liệu tips có sẵn trong seaborn library. Hãy vẽ những biểu đồ theo yêu cầu:

In [1]:

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
```

In [2]:

```
# Load dữ liệu tips có sẵn trong seaborn library
#total_bill: Total bill (cost of the meal), including tax, in US dollars
#tip: Tip (gratuity) in US dollars
#sex: Sex of person paying for the meal (0=male, 1=female)
#smoker: Smoker in party? (0=No, 1=Yes)
#day: 3=Thur, 4=Fri, 5=Sat, 6=Sun
#time: 0=Day, 1=Night
#size: Size of the party
tips = sns.load_dataset("tips")
tips.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 244 entries, 0 to 243
Data columns (total 7 columns):
total_bill    244 non-null float64
tip           244 non-null float64
sex           244 non-null category
smoker        244 non-null category
day           244 non-null category
time          244 non-null category
size          244 non-null int64
dtypes: category(4), float64(2), int64(1)
memory usage: 7.3 KB
```

In [3]:

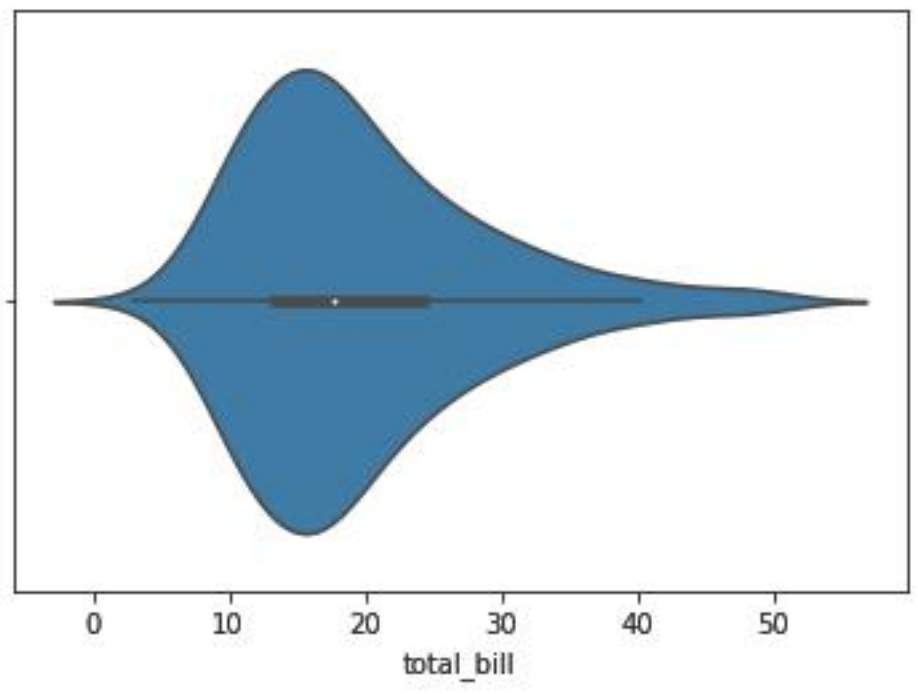
```
tips.tail(10)
```

Out[3]:

	total_bill	tip	sex	smoker	day	time	size
234	15.53	3.00	Male	Yes	Sat	Dinner	2
235	10.07	1.25	Male	No	Sat	Dinner	2
236	12.60	1.00	Male	Yes	Sat	Dinner	2
237	32.83	1.17	Male	Yes	Sat	Dinner	2
238	35.83	4.67	Female	No	Sat	Dinner	3
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

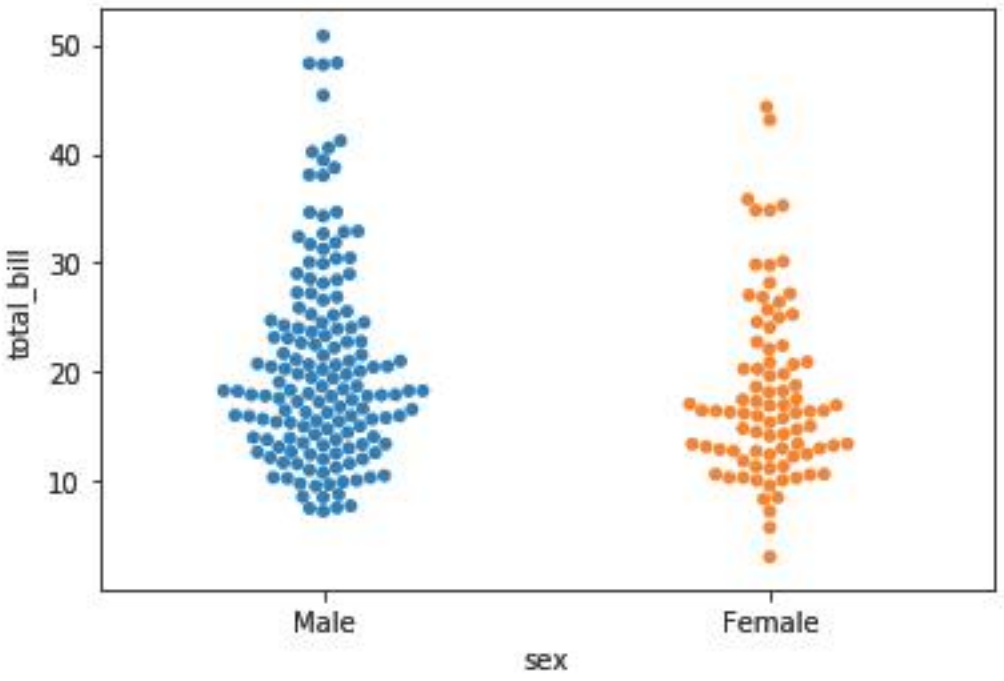
In [4]:

```
# Câu 1: Vẽ violinplot cho cho cột total_bill
# Bạn nhận xét gì về biểu đồ vừa tạo
sns.violinplot(x = "total_bill", data=tips)
plt.show()
```



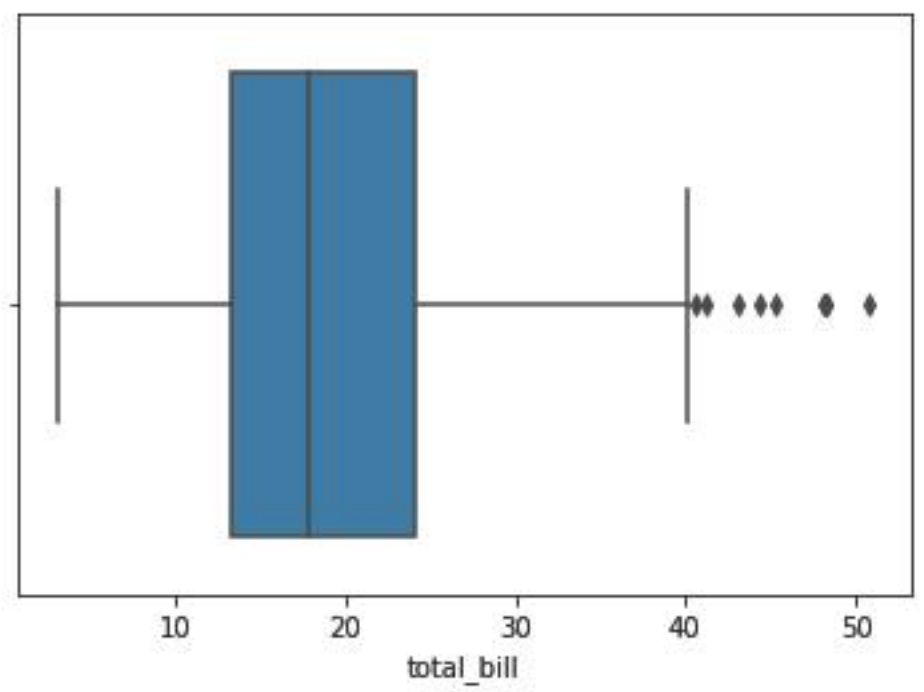
In [5]:

```
# Câu 2: Vẽ swarmplot cho cột total_bill theo sex
# Bạn nhận xét gì về biểu đồ vừa tạo
sns.swarmplot(x="sex", y="total_bill", data=tips)
plt.show()
```



In [6]:

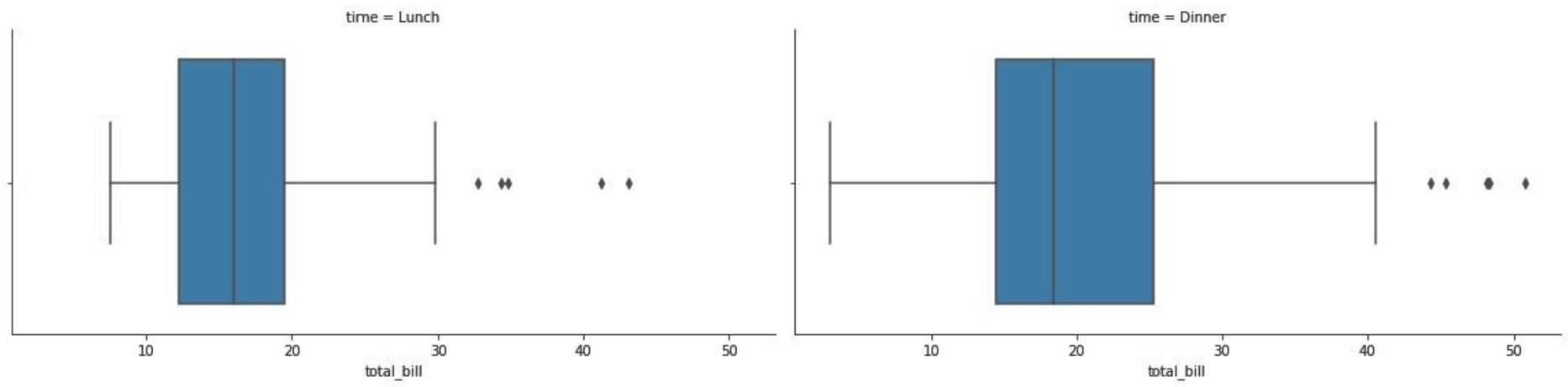
```
# Câu 3: Vẽ boxplot cho cột total_bill
# Bạn nhận xét gì về biểu đồ vừa tạo
sns.boxplot(x="total_bill", data=tips)
plt.show()
```



In [7]:

```
# Câu 4: Tạo FacetGrid của total_bill theo 'time' và chỉ định thứ tự của các cột bằng
col_order
# Bạn nhận xét gì về biểu đồ vừa tạo
plt.figure(figsize=(8,6))
fg = sns.FacetGrid(data=tips, col="time", col_order=['Lunch', 'Dinner'], height=4,
aspect=2)

fg.map(sns.boxplot, 'total_bill', order=None)

plt.show()
plt.clf()
```

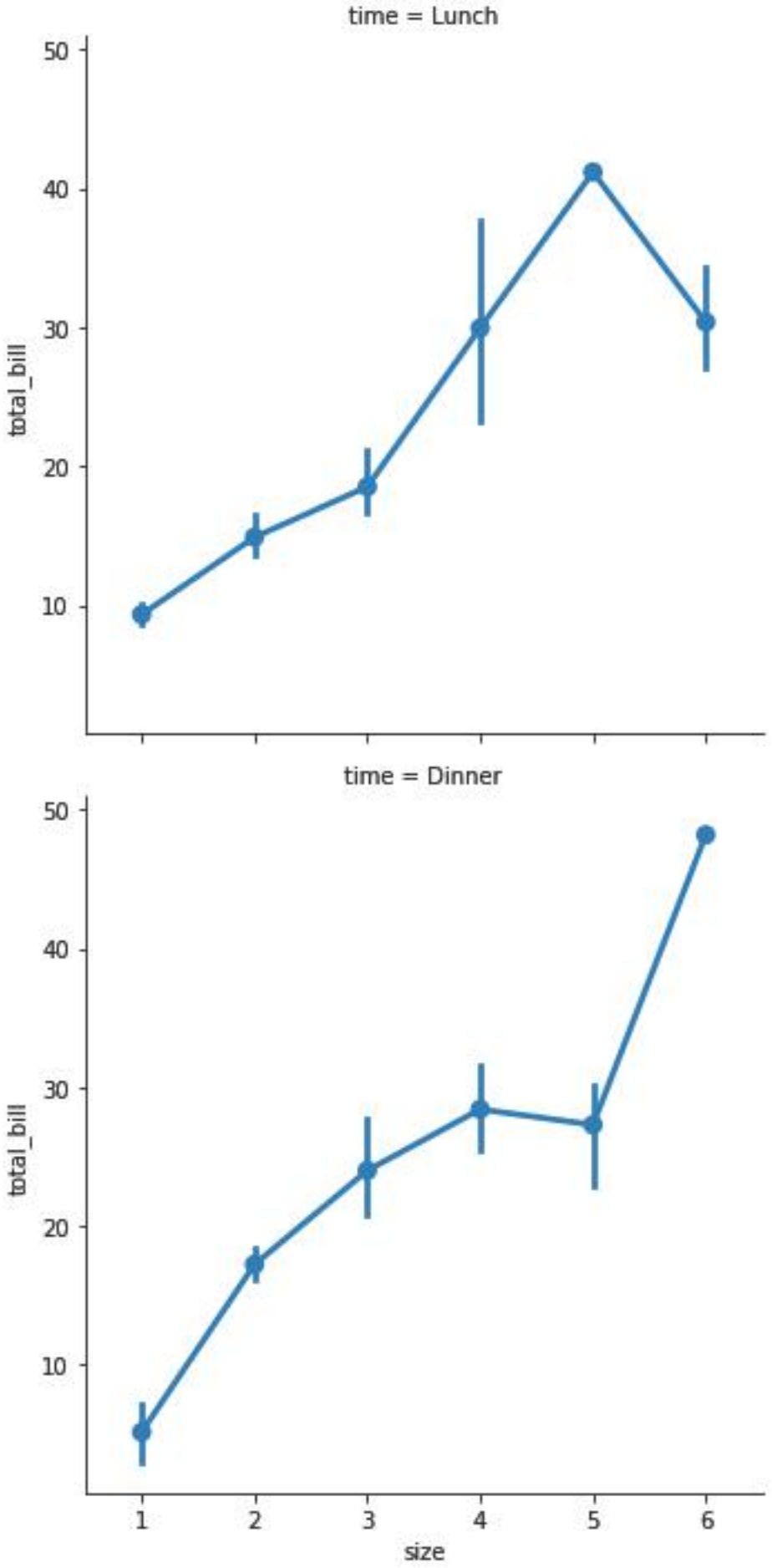
```
<Figure size 576x432 with 0 Axes>
```




```
<Figure size 432x288 with 0 Axes>
```

In [8]:

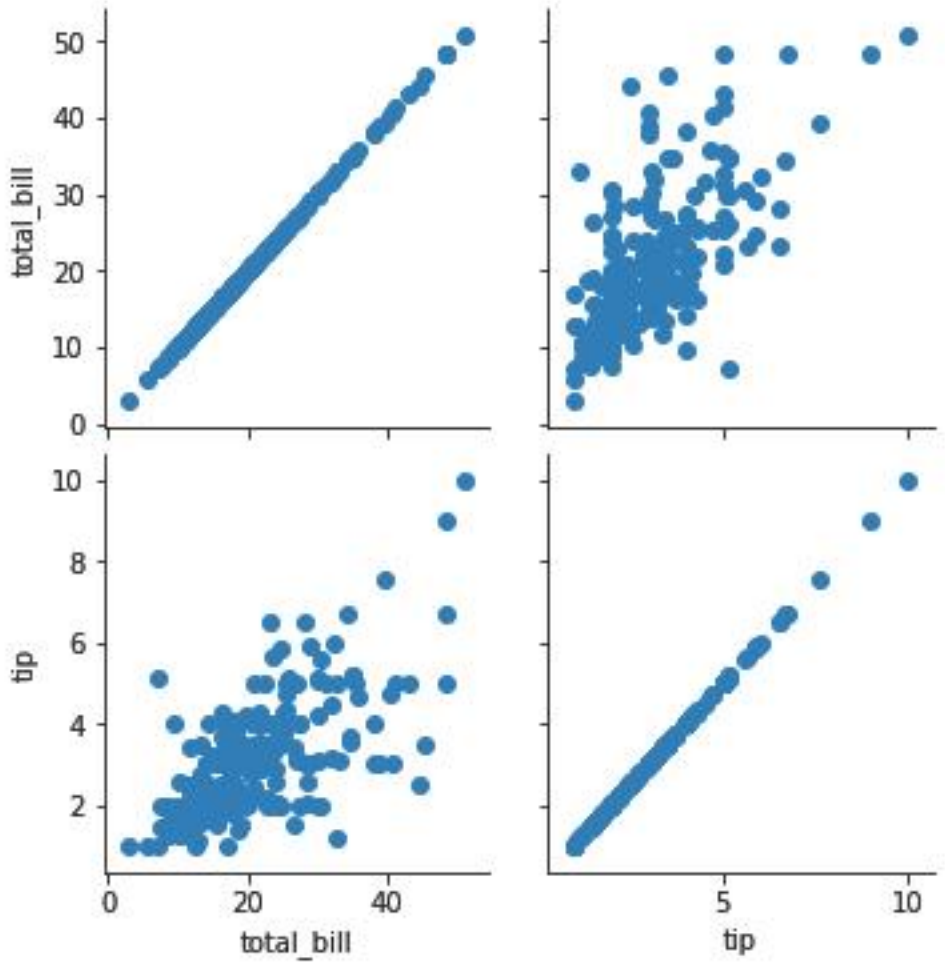
```
# Câu 5: Tạo catplot chứa point plot của giá trị 'total_bill' thay đổi theo size và
tách dòng theo ‘time’ .
# Bạn nhận xét gì về biểu đồ vừa tạo

sns.catplot(data=tips, x='size', y='total_bill', kind='point', row='time')

plt.show()
plt.clf()
```


time = Lunch
time = Dinner
total_bill
size
50
40
30
20
10
1
2
3
4
5
6

```
<Figure size 432x288 with 0 Axes>
```

In [9]:

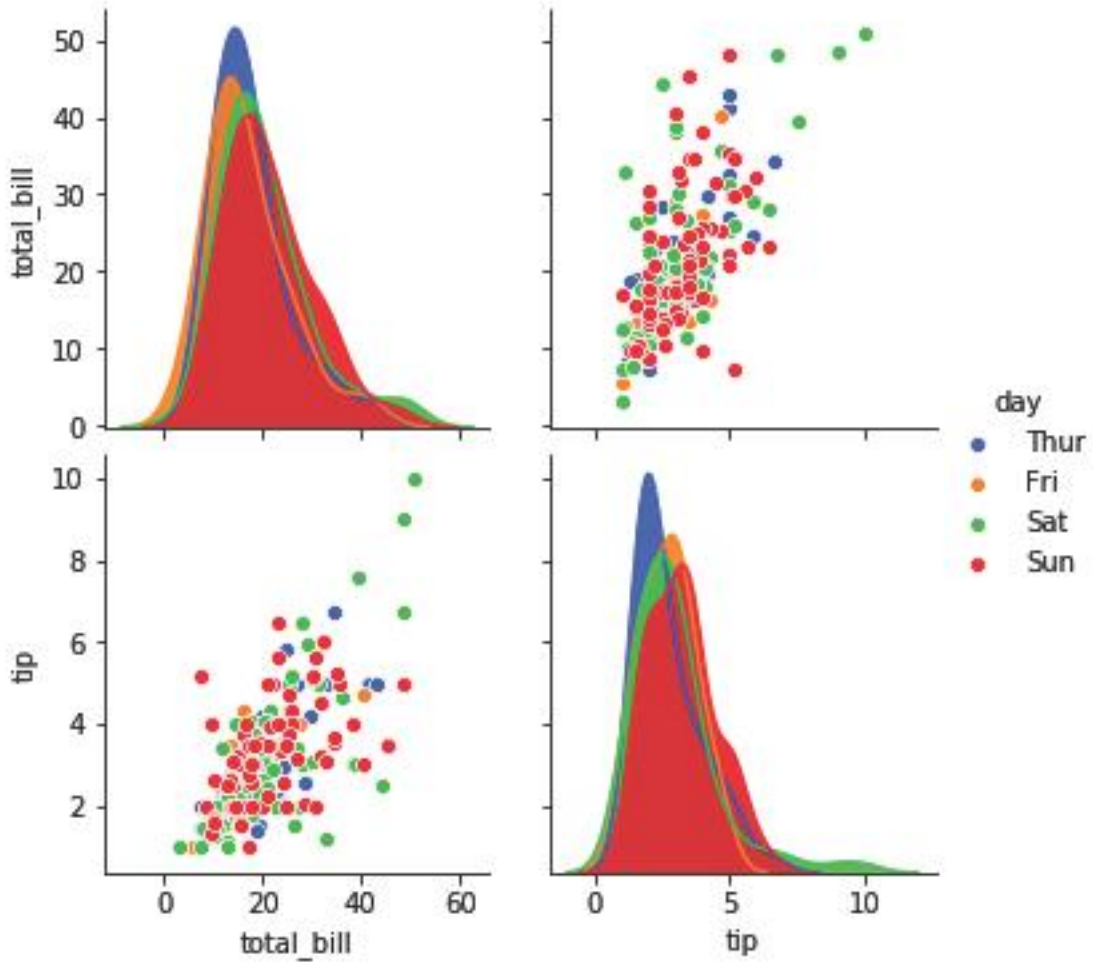
```
# Câu 6: Tạo PairGrid với scatter plot thể hiện liên quan giữa "total_bill" và "tip"
# Bạn nhận xét gì về biểu đồ vừa tạo
pg = sns.PairGrid(data=tips, vars=["total_bill", "tip"])
pg2 = pg.map(plt.scatter)
plt.show()
plt.clf()
```




```
<Figure size 432x288 with 0 Axes>
```

In [10]:

```
# Câu 7: Tạo Pairplot với scatter plot thể hiện liên quan giữa "total_bill" và "tip",
sử dụng palette color = 'day'
# Bạn nhận xét gì về biểu đồ vừa tạo
sns.pairplot(data=tips, vars=["total_bill", "tip"], kind='scatter', hue='day', palette=
'bright', diag_kws={'alpha':.9})
plt.show()
plt.clf()
```



<Figure size 432x288 with 0 Axes>

Chapter 7 - Exercise 2: Titanic

Cho dữ liệu titanic có sẵn trong seaborn library. Hãy vẽ những biểu đồ theo yêu cầu, và cho biết nhận xét sau biểu đồ vừa vẽ:

In [1]:

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
```

In [2]:

```
# Load dữ liệu titanic có sẵn trong seaborn library
titanic = sns.load_dataset("titanic")
titanic.info()
titanic.head()
```

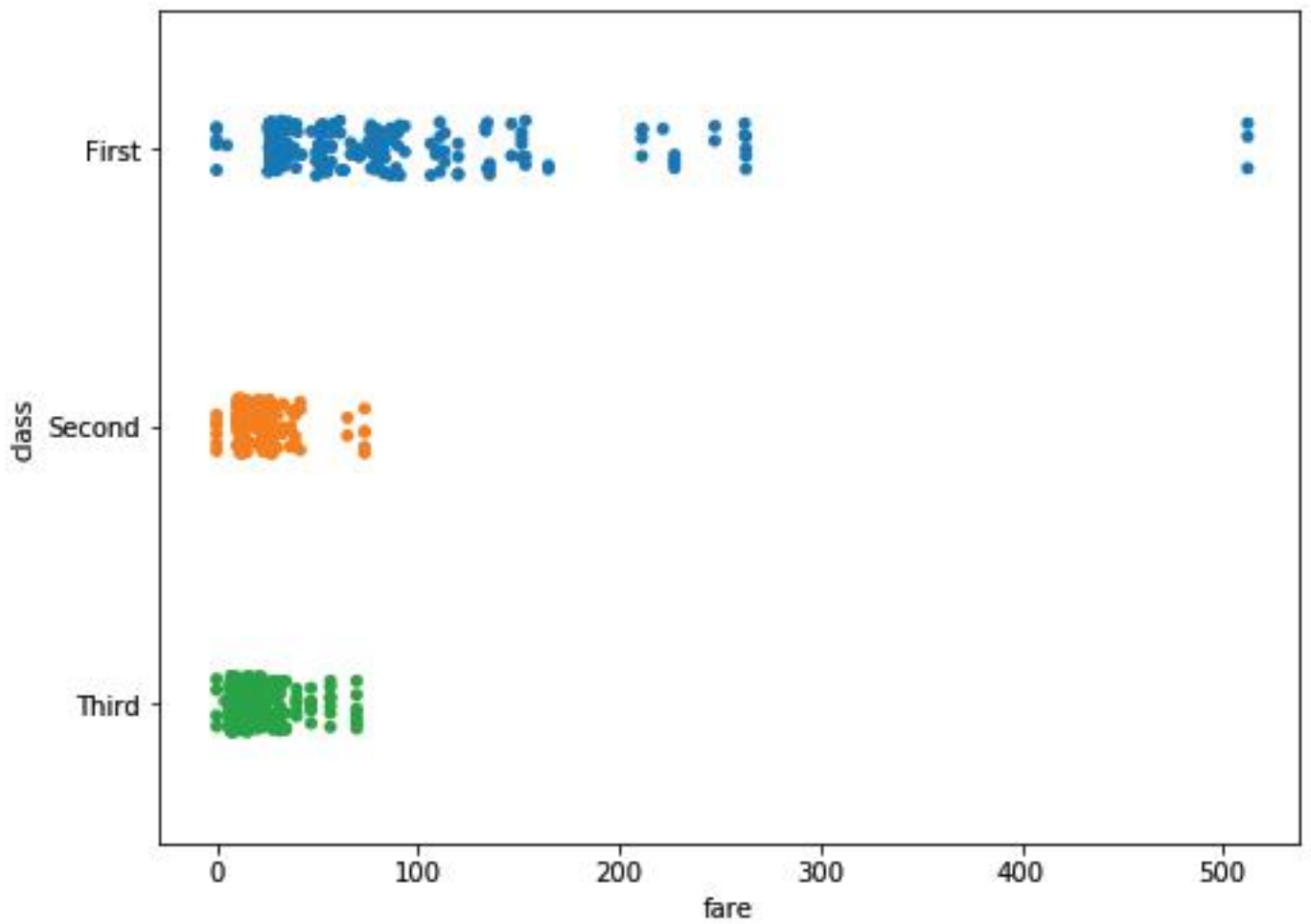
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
survived       891 non-null int64
pclass         891 non-null int64
sex            891 non-null object
age            714 non-null float64
sibsp          891 non-null int64
parch          891 non-null int64
fare           891 non-null float64
embarked       889 non-null object
class          891 non-null category
who            891 non-null object
adult_male     891 non-null bool
deck           203 non-null category
embark_town    889 non-null object
alive          891 non-null object
alone          891 non-null bool
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.6+ KB
```

Out[2]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_ma
0	0	3	male	22.0	1	0	7.2500	S	Third	man	Tr
1	1	1	female	38.0	1	0	71.2833	C	First	woman	Fal
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	Fal
3	1	1	female	35.0	1	0	53.1000	S	First	woman	Fal
4	0	3	male	35.0	0	0	8.0500	S	Third	man	Tr

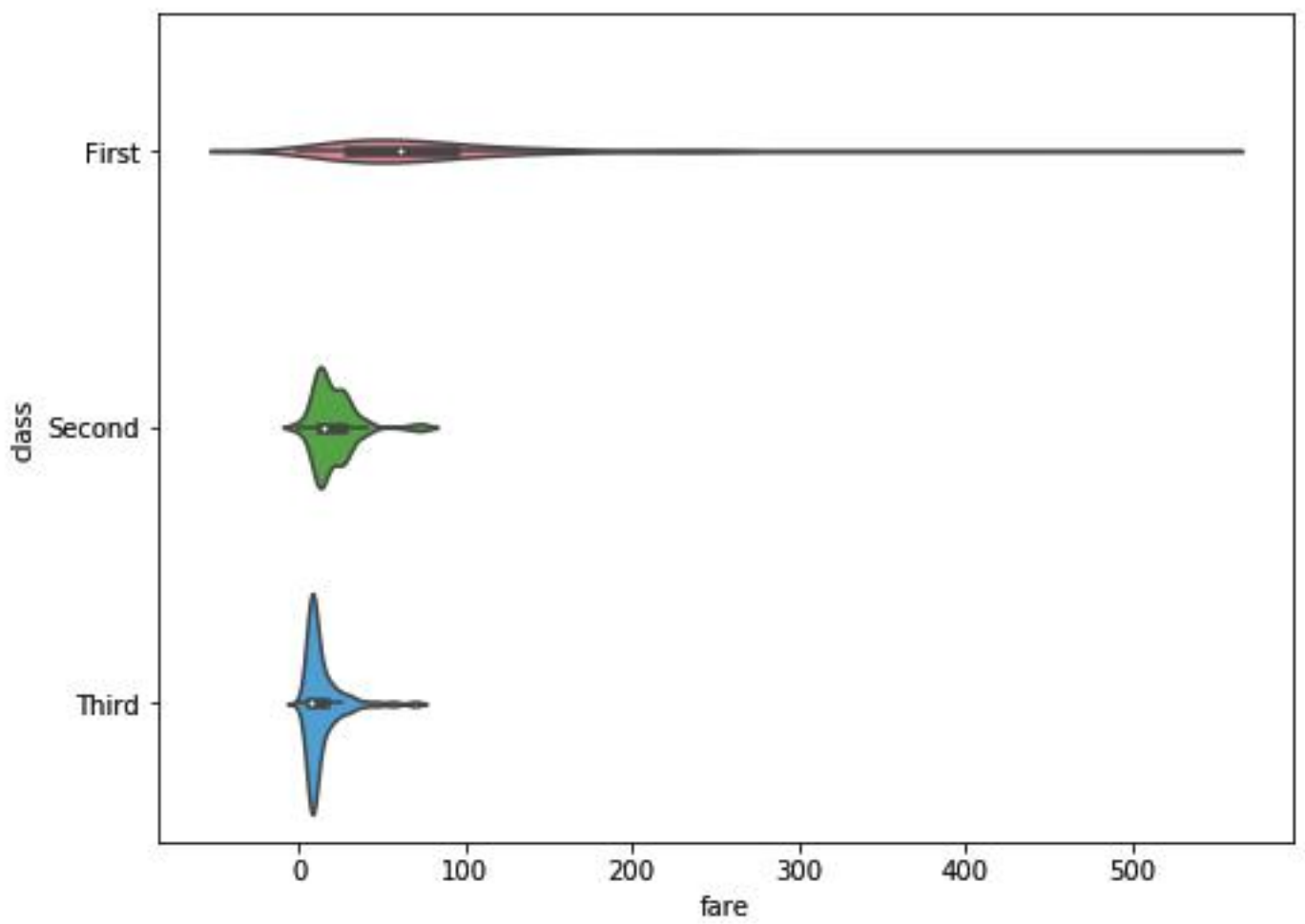
In [3]:

```
# Câu 1: Vẽ stripplot thể hiện sự phân bố của fare theo class
# Bạn nhận xét gì về biểu đồ vừa tạo
plt.figure(figsize=(8,6))
sns.stripplot(data=titanic, x='fare', y='class', jitter=True)
plt.show()
```



In [4]:

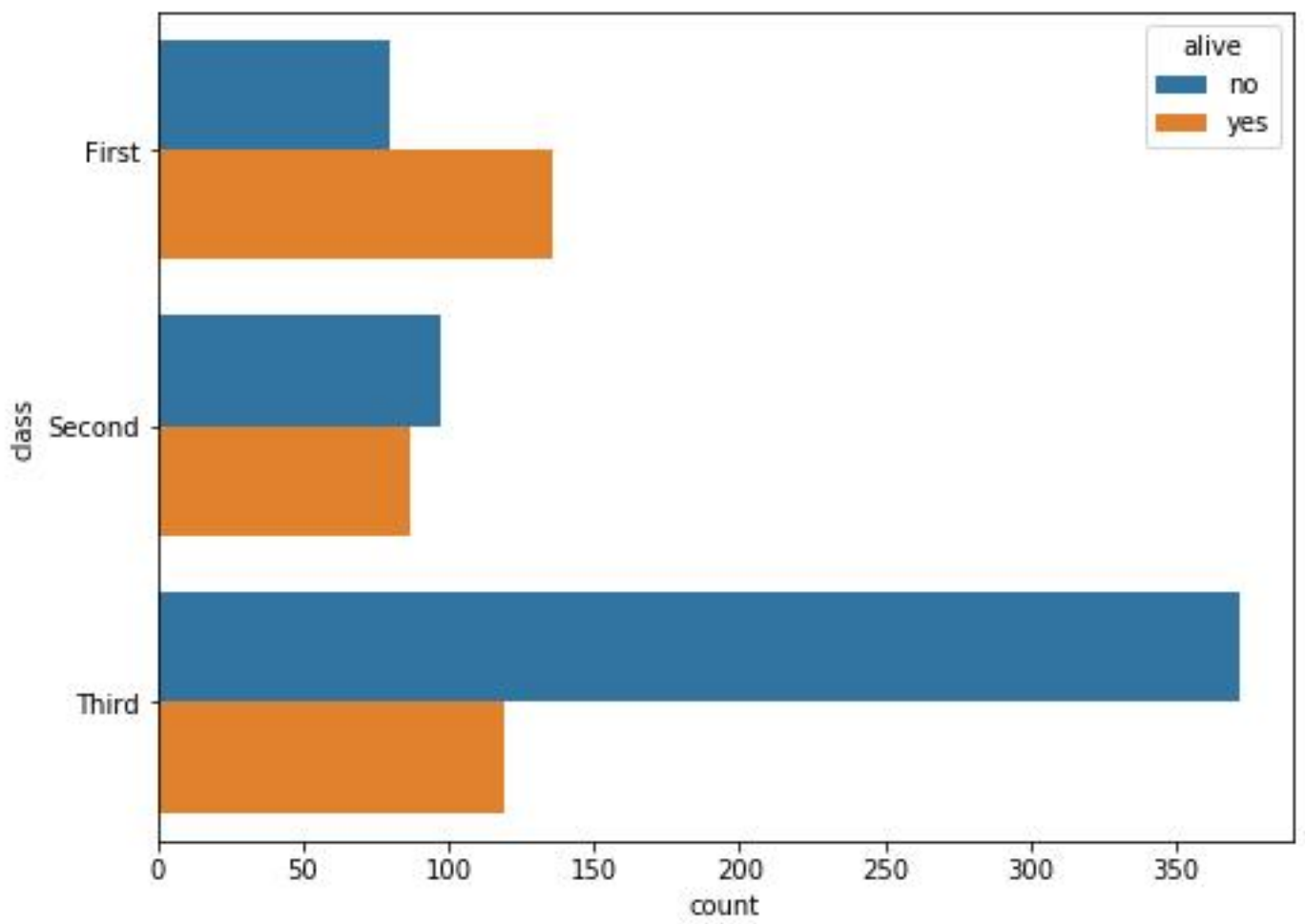
```
# Câu 2: Vẽ violinplot thể hiện sự phân bố của fare theo class
# Bạn nhận xét gì về biểu đồ vừa tạo
plt.figure(figsize=(8,6))
sns.violinplot(data=titanic, x='fare', y='class', palette='husl')
plt.show()
plt.clf()
```



<Figure size 432x288 with 0 Axes>

In [5]:

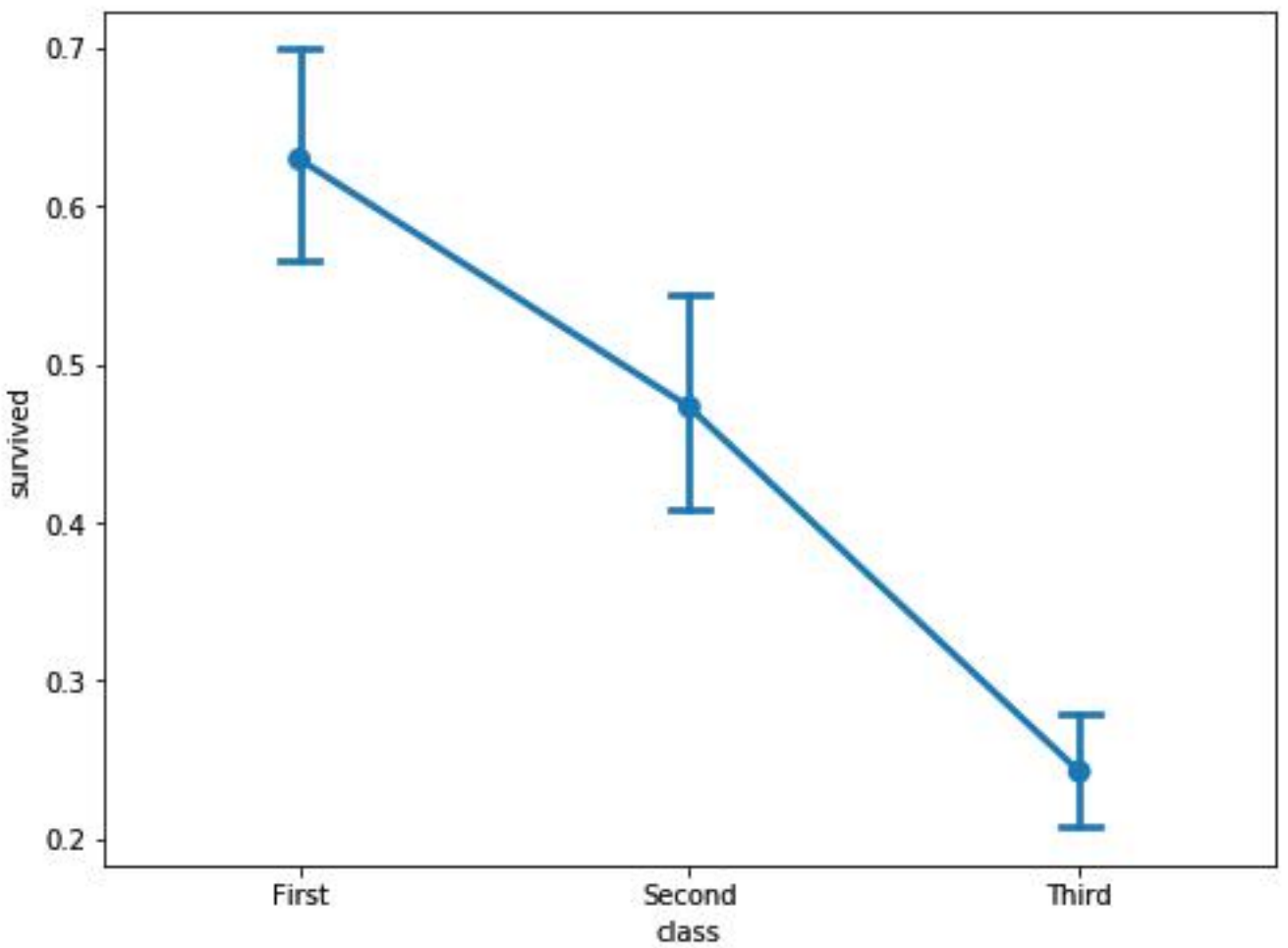
```
# Câu 3: Vẽ countplot đếm số lượng alive/not alive theo từng class
# Bạn nhận xét gì về biểu đồ vừa tạo
plt.figure(figsize=(8,6))
sns.countplot(data=titanic, y="class", hue="alive")
plt.show()
plt.clf()
```




```
<Figure size 432x288 with 0 Axes>
```

In [6]:

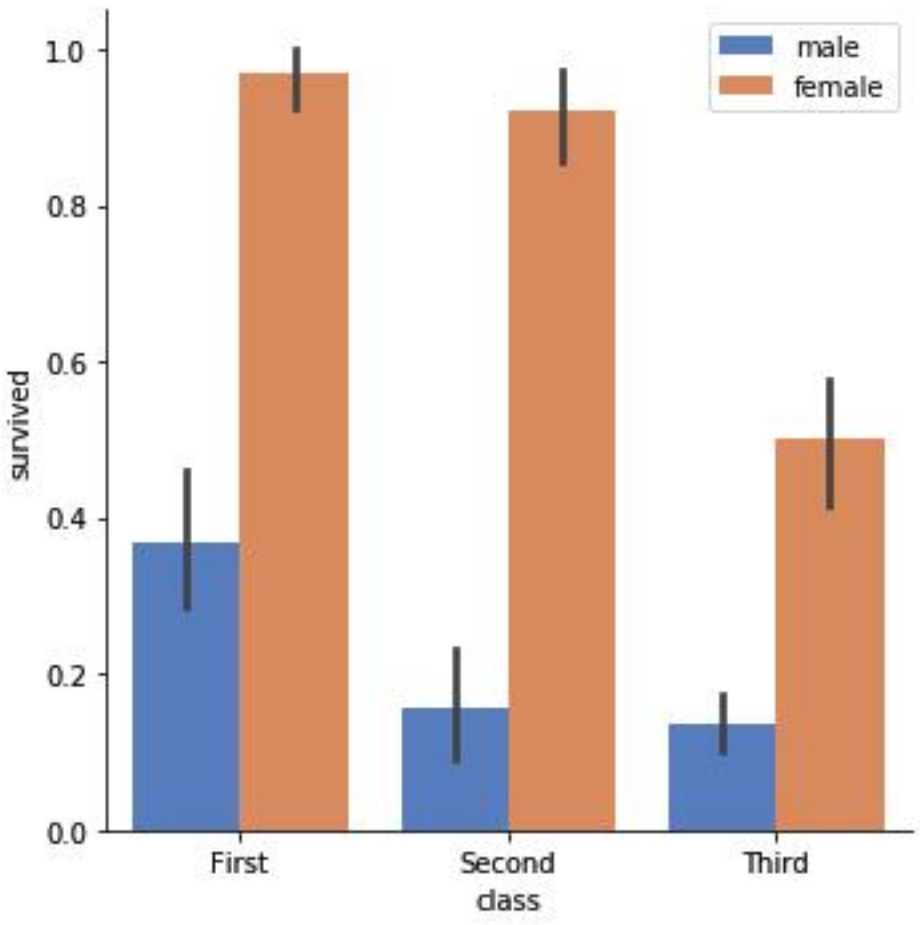
```
# Câu 4: Vẽ pointplot thể hiện khả năng sống sót 'survived' theo class
# Bạn nhận xét gì về biểu đồ vừa tạo
plt.figure(figsize=(8,6))
sns.pointplot(data=titanic, y='survived', x='class', capsize=.1)
plt.show()
plt.clf()
```




```
<Figure size 432x288 with 0 Axes>
```

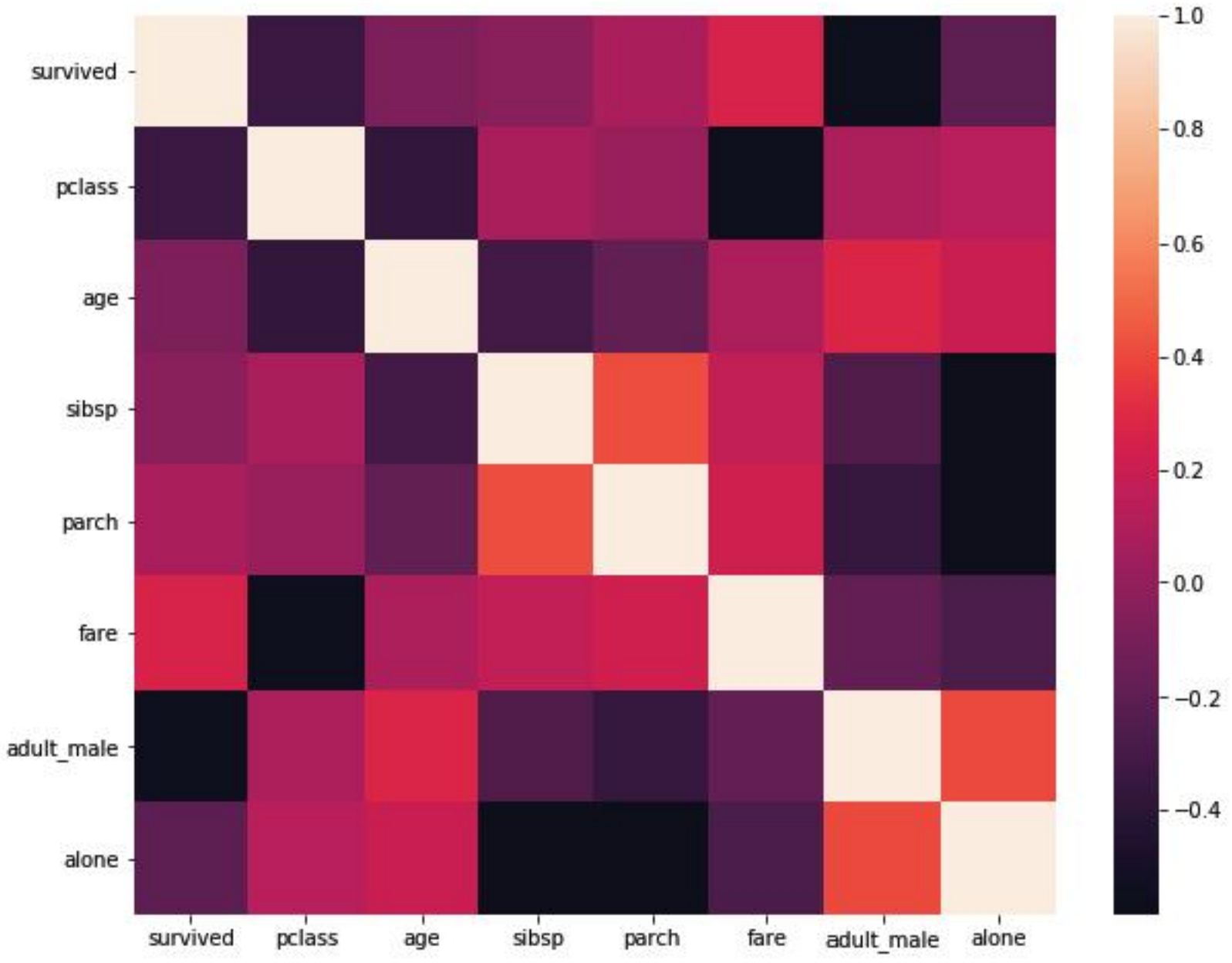
In [7]:

```
# Câu 5: Vẽ catplot dạng bar thể hiện survived của từng sex, phân loại theo class
# Bạn nhận xét gì về biểu đồ vừa tạo
g = sns.catplot("class", "survived", "sex", data=titanic, kind="bar", palette="muted",
legend=False)
plt.legend()
plt.show()
```



In [8]:

```
# Câu 6: Vẽ correlation matrix (heatmap) của titanic
# Bạn nhận xét gì về biểu đồ vừa tạo
plt.figure(figsize=(10,8))
sns.heatmap(titanic.corr())
plt.show()
```



Chapter 7 - Exercise 3: Visualization with Seaborn - Diamond

Nghịch lý Simpson hay hiệu ứng Yule–Simpson, là một nghịch lý trong xác suất và thống kê, trong đó một xu hướng xuất hiện trong dữ liệu sẽ bị đảo ngược khi được phân tích dưới góc nhìn khác.

Cho file dữ liệu diamonds.csv. Hãy thực hiện các yêu cầu sau, để phát hiện nghịch lý Simpson khi phân tích giá kim cương bằng các công cụ trực quan hóa dữ liệu:

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
sns.set_style("darkgrid")
```

In [2]:

```
# Câu 1: Đọc dữ liệu diamonds.csv, đưa vào biến diamonds
diamonds = pd.read_csv(r'data/diamonds.csv')
diamonds.head()
```

Out[2]:

	carat	cut	color	clarity	depth	table	price	x	y	z
0	0.23	Ideal	E	SI2	61.5	55.0	326	3.95	3.98	2.43
1	0.21	Premium	E	SI1	59.8	61.0	326	3.89	3.84	2.31
2	0.23	Good	E	VS1	56.9	65.0	327	4.05	4.07	2.31
3	0.29	Premium	I	VS2	62.4	58.0	334	4.20	4.23	2.63
4	0.31	Good	J	SI2	63.3	58.0	335	4.34	4.35	2.75

In [3]:

```
# Câu 2: Vẽ biểu đồ bar so sánh giá của kim cương theo color, cut và clarity
# Bạn nhận xét gì qua biểu đồ này

fig, (ax1, ax2, ax3) = plt.subplots(1, 3, figsize=(16,4))
sns.barplot(x='color', y='price', data=diamonds, ax=ax1)
sns.barplot(x='cut', y='price', data=diamonds, ax=ax2)
sns.barplot(x='clarity', y='price', data=diamonds, ax=ax3)
fig.suptitle('Price Decreasing with Increasing Quality?',fontsize=15)
```

Out[3]:

```
Text(0.5, 0.98, 'Price Decreasing with Increasing Quality?')
```

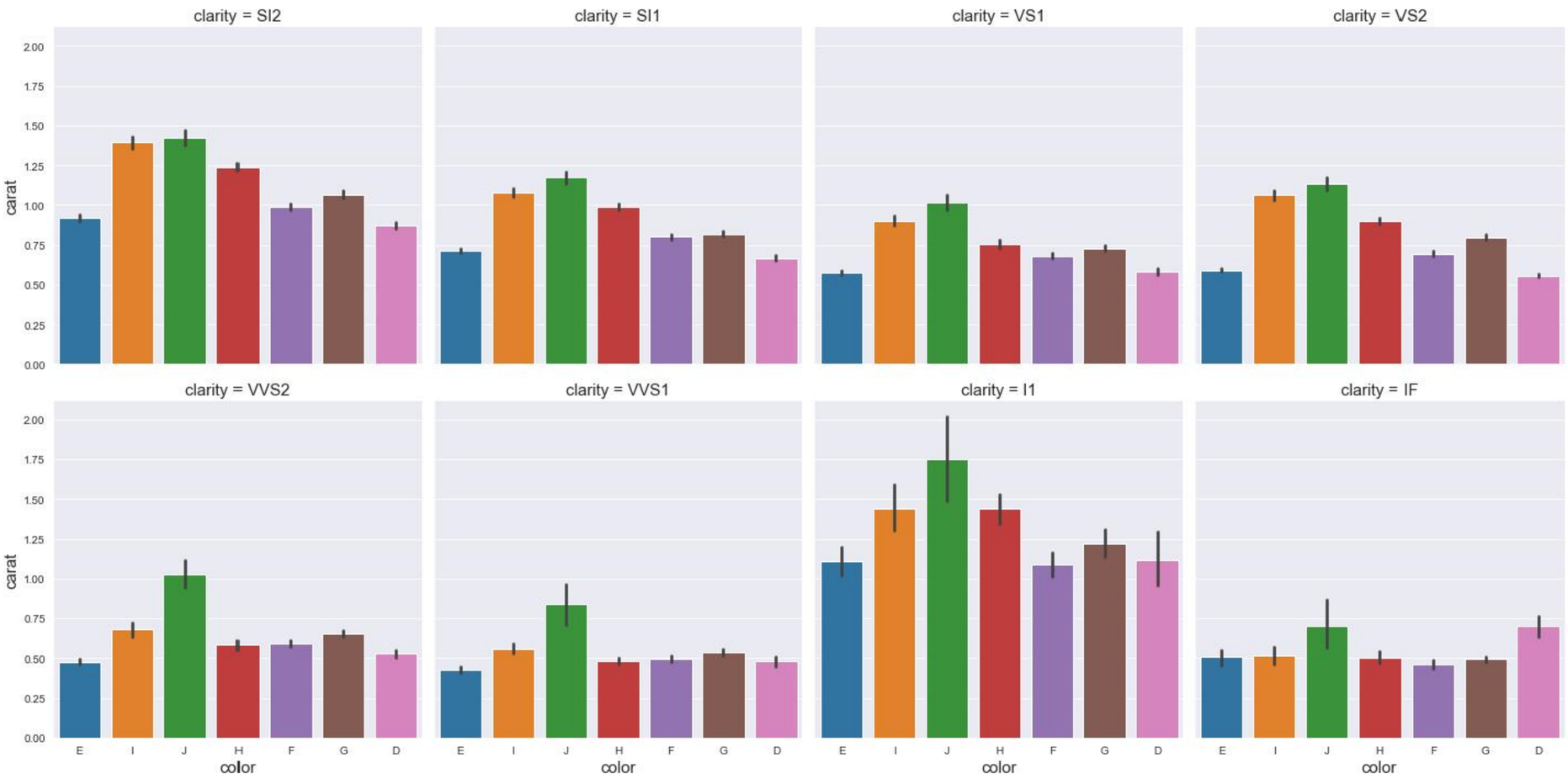
Có gì đó không ổn ???

In [4]:

```
# Câu 3: Bây giờ, hãy thử Phân tích chi tiết hơn thuộc tính 'carat' theo 'color' và
'clarity' qua biểu đồ catplot - bar plot
# Bạn nhận xét gì qua biểu đồ này
plt.rcParams["axes.labelsize"] = 15
sns.catplot(x='color', y='carat', col='clarity', col_wrap=4, data=diamonds, kind='bar')
```

Out[4]:

```
<seaborn.axisgrid.FacetGrid at 0x529859da08>
```



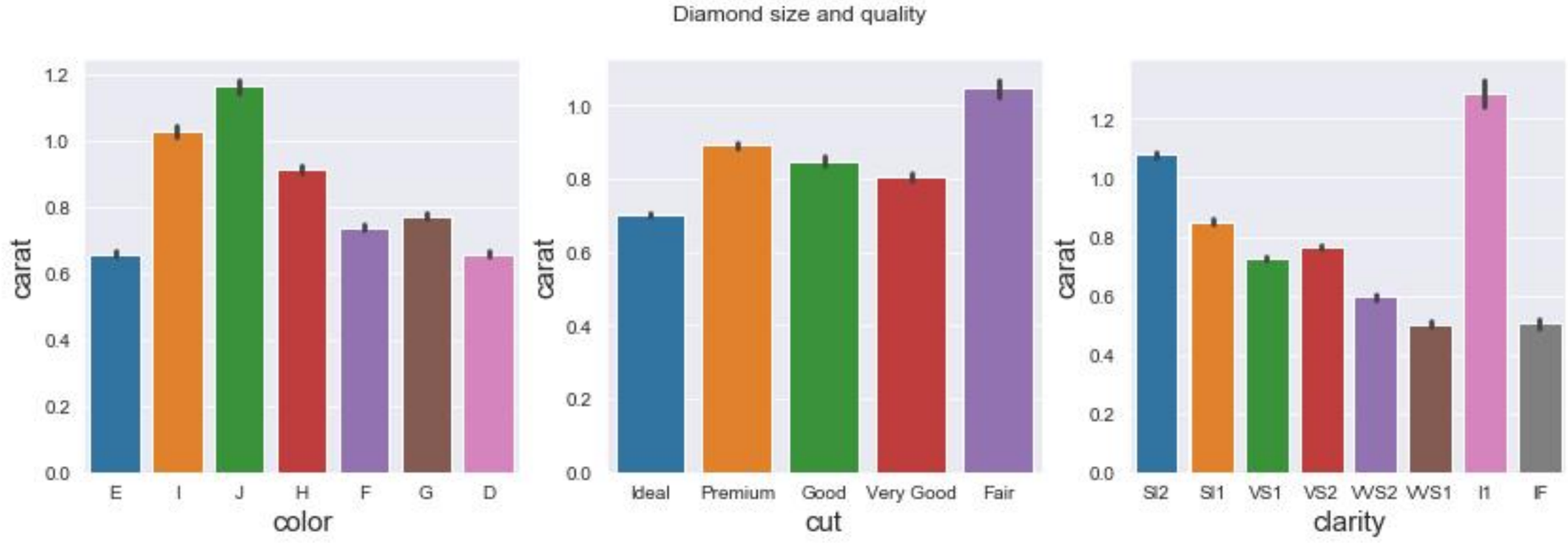
=> Kim cương kích thước nhỏ thì chất lượng thường cao

In [5]:

```
# Câu 4: Vẽ biểu đồ bar so sánh 'carat' của kim cương theo color, cut và clarity
fig, (ax1, ax2, ax3) = plt.subplots(1, 3, figsize=(14,4))
sns.barplot(x='color', y='carat', data=diamonds, ax=ax1)
sns.barplot(x='cut', y='carat', data=diamonds, ax=ax2)
sns.barplot(x='clarity', y='carat', data=diamonds, ax=ax3)
fig.suptitle('Diamond size and quality')
```

Out[5]:

```
Text(0.5, 0.98, 'Diamond size and quality')
```



In [6]:

```
# Câu 5: Hãy chia carat ra làm 5 khoảng giá trị, tạo cột diamonds['carat_category']
chứa khoảng giá trị tương ứng
# Hướng dẫn: sử dụng hàm pd.qcut
diamonds['carat_category'] = pd.qcut(diamonds.carat, 5)
diamonds.head()
```

Out[6]:

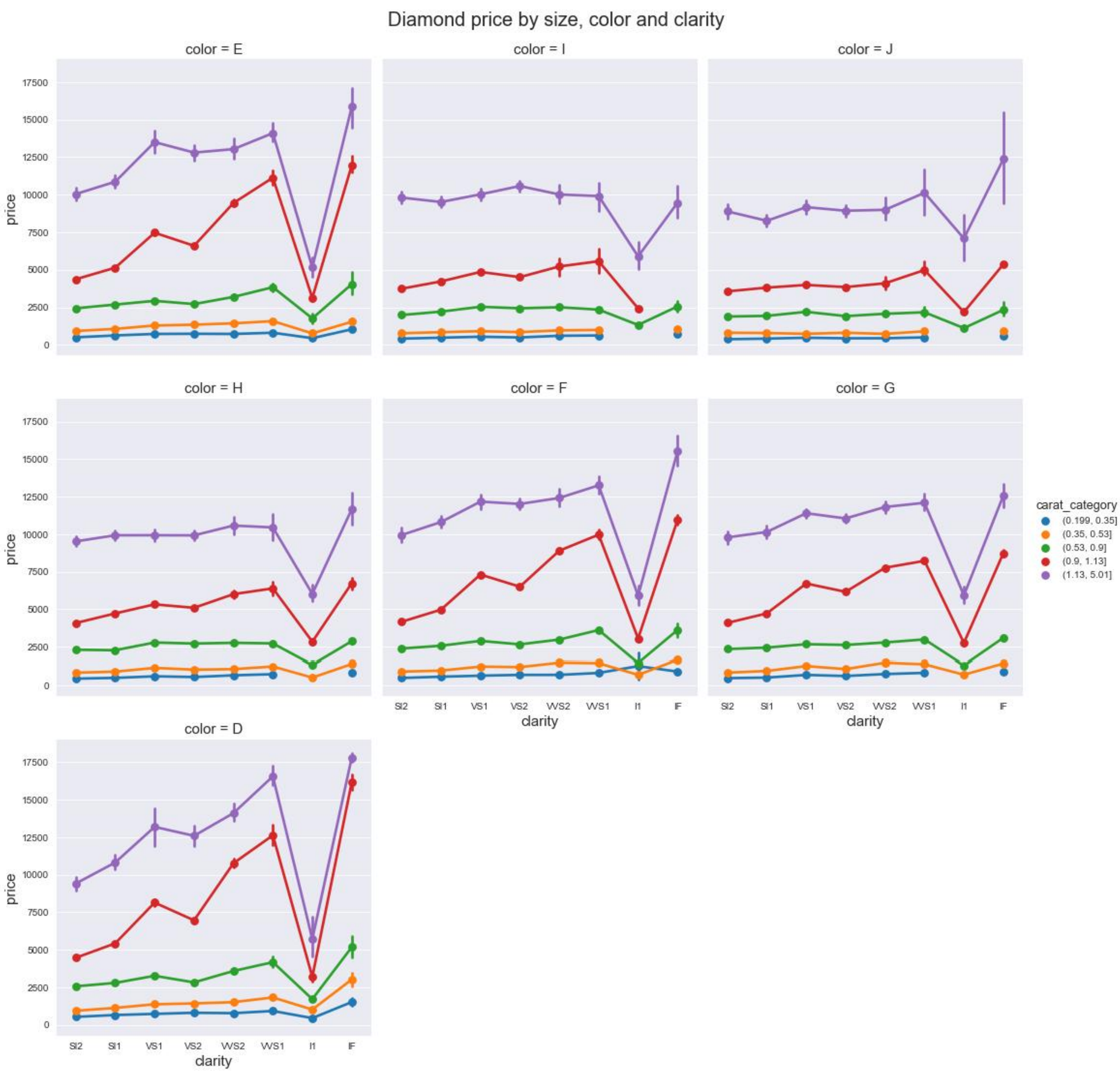
	carat	cut	color	clarity	depth	table	price	x	y	z	carat_category
0	0.23	Ideal	E	SI2	61.5	55.0	326	3.95	3.98	2.43	(0.199, 0.35]
1	0.21	Premium	E	SI1	59.8	61.0	326	3.89	3.84	2.31	(0.199, 0.35]
2	0.23	Good	E	VS1	56.9	65.0	327	4.05	4.07	2.31	(0.199, 0.35]
3	0.29	Premium	I	VS2	62.4	58.0	334	4.20	4.23	2.63	(0.199, 0.35]
4	0.31	Good	J	SI2	63.3	58.0	335	4.34	4.35	2.75	(0.199, 0.35]

In [7]:

```
# Câu 6: Phân tích chi tiết hơn thuộc tính 'price' theo  'clarity','carat_category',
'color' qua biểu đồ catplot - point plot

from matplotlib.cm import Greys
greys = Greys(np.arange(50,250,40))

g = sns.catplot(x='clarity', y='price', data=diamonds,
                    hue='carat_category', col='color',
                    col_wrap=3, kind='point') #, palette=greys)
g.fig.suptitle('Diamond price by size, color and clarity',
                y=1.02, size=20)
```

Out[7]:

```
Text(0.5, 1.02, 'Diamond price by size, color and clarity')
```



In [8]:

```
# Câu 7: Kết luận
```