Chapter 7 - Exercise 4: 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.

- https://eropi.com/news/tieu-chuan-danh-gia-kim-cuong/ (https://eropi.com/news/tieu-chuan-danh-gia-kim-cuong/)
- https://www.youtube.com/watch?v=hDQ0T6-i1rk&t=183s (https://www.youtube.com/watch?v=hDQ0T6-i1rk&t=183s)

Có phải kim cương càng lớn thì có giá càng cao ?

Cho dữ liệu diamonds có sẵn trong seaborn library. 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
sns.set_style("darkgrid")
```

In [2]:
```
# Câu 1: Đọc dữ liệu diamonds và lưu vào biến diamonds
diamonds = sns.load_dataset('diamonds')
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]:
```
diamonds.shape
```

Out[3]:
```
(53940, 10)
```

In [4]:
```
# tiêu chuẩn từ thấp đến cao (4C: cut, color, clarity, carat)
cut_cats = ['Fair', 'Good', 'Very Good', 'Premium', 'Ideal']
color_cats = ['J', 'I', 'H', 'G', 'F', 'E', 'D']
clarity_cats = ['I1', 'SI2', 'SI1', 'VS2', 'VS1', 'VVS2', 'VVS1', 'IF']
```

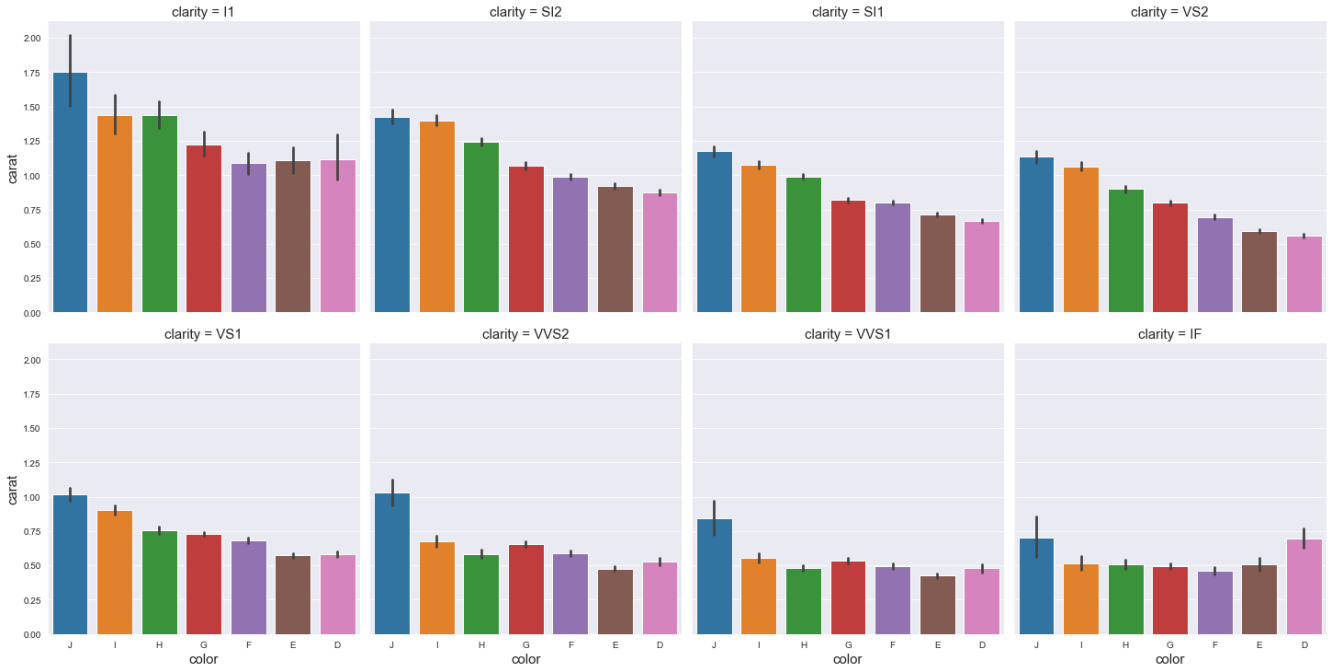
In [5]:
```
# Câu 2: Vẽ biểu đồ bar so sánh giá của kim cương theo color, cut và clarity
# Bạn nhận xét gì qua biểu đồ này

```

Nhấn vào đây để xem kết quả!

In [6]:

```
1 # Câu 3: Bây giờ, hãy thử Phân tích chi tiết hơn
2 # thuộc tính 'carat' theo 'color' và 'clarity' qua biểu đồ catplot - bar plot
3 # Bạn nhận xét gì qua biểu đồ này
4
```

Nhấn vào đây để xem kết quả!

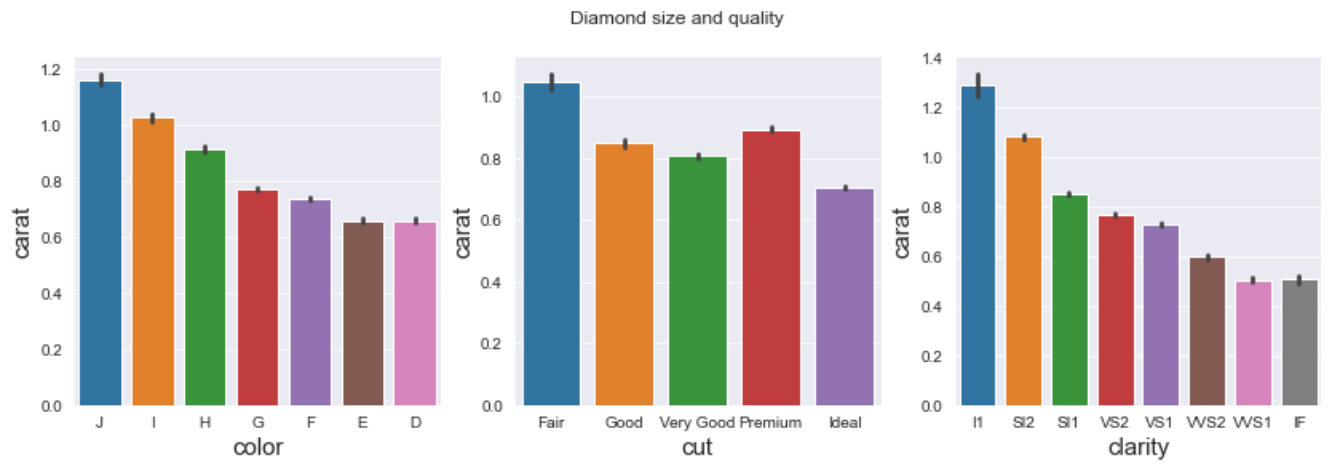


=> Kim cương kích thước nhỏ thì chất lượng thường cao

In [7]:

```
1 # Câu 4: Vẽ biểu đồ bar so sánh 'carat' của kim cương theo color, cut và clarity
2
```

Nhấn vào đây để xem kết quả!



In [8]:

```
# Câu 5: Hãy chia carat ra làm 5 khoảng giá trị,
# tạo cột diamonds['carat_category'] chứa khoảng giá trị tương ứng
# https://pbpython.com/pandas-qcut-cut.html
# Hướng dẫn: sử dụng hàm pd.qcut
# diamonds['carat_category'] = pd.qcut(diamonds.carat, 5)

```

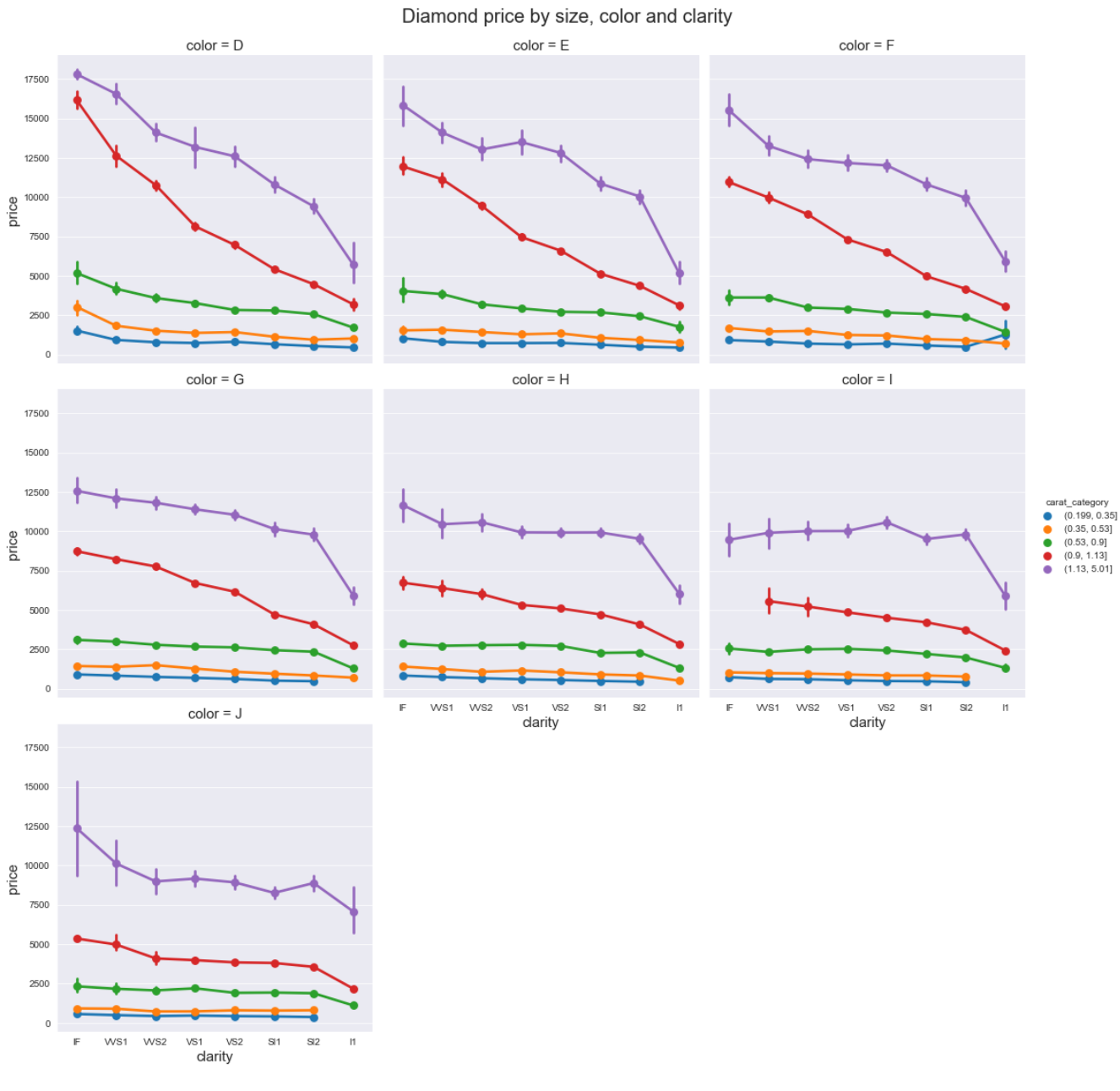
Nhấn vào đây để xem kết quả!

	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 [9]:

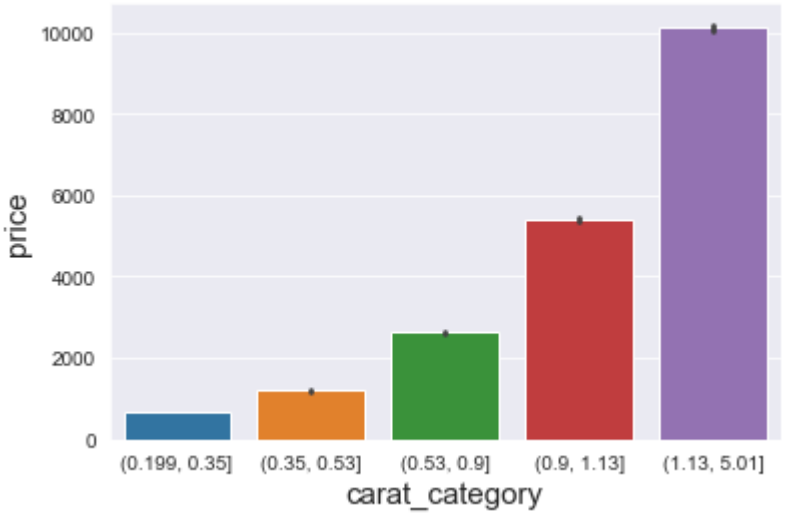
```
# Câu 6: Phân tích chi tiết hơn thuộc tính 'price' theo  'clarity','carat_category'
# 'color' qua biểu đồ catplot - point plot

```

Nhấn vào đây để xem kết quả!



In [10]:
```
# Câu 7: Vẽ barplot

```

Nhấn vào đây để xem kết quả!



In [11]:
```
# Kết luận:carat càng lớn thì giá càng cao!
```

In [ ]:
```

```