In [1]:

```
import pandas as pd
```

In [2]:

```
review = pd.read_csv('Wine_review.csv')
review
```

Out[2]:

	Unnamed: 0	country	description	designation	points	price	province	region_1
0	0	US	This tremendous 100% varietal wine hails from ...	Martha's Vineyard	96	235.0	California	Napa Valley
1	1	Spain	Ripe aromas of fig, blackberry and cassis are ...	Carodorum Selección Especial Reserva	96	110.0	Northern Spain	Toro
2	2	US	Mac Watson honors the memory of a wine once ma...	Special Selected Late Harvest	96	90.0	California	Knights Valley
3	3	US	This spent 20 months in 30% new French oak, an...	Reserve	96	65.0	Oregon	Willamette Valley
4	4	France	This is the top wine from La Bégude, named aft...	La Brûlade	95	66.0	Provence	Bandol
...	...	...	...	...	...	...	...	...
150925	150925	Italy	Many people feel Fiano represents southern Ita...	NaN	91	20.0	Southern Italy	Fiano di Avellino
150926	150926	France	Offers an intriguing nose with ginger, lime an...	Cuvée Prestige	91	27.0	Champagne	Champagne
150927	150927	Italy	This classic example comes from a cru vineyard...	Terre di Dora	91	20.0	Southern Italy	Fiano di Avellino
150928	150928	France	A perfect salmon shade, with scents of peaches...	Grand Brut Rosé	90	52.0	Champagne	Champagne

	Unnamed: 0	country	description	designation	points	price	province	region_1
150929	150929	Italy	More Pinot Grigios should taste like this. A r...	NaN	90	15.0	Northeastern Italy	Alto Adige

150930 rows × 11 columns

1. Who are the most common wine reviewers in the dataset? Create a `Series` whose index is the `taster_twitter_handle` category from the dataset, and whose values count how many reviews each person wrote.

In [5]:

```
#1
taster_twitter_handle = review.country.value_counts().head(15)
taster_twitter_handle
```

Out[5]:

```
US              62397
Italy           23478
France          21098
Spain            8268
Chile            5816
Argentina        5631
Portugal         5322
Australia        4957
New Zealand      3320
Austria          3057
Germany          2452
South Africa     2258
Greece            884
Israel            630
Hungary           231
Name: country, dtype: int64
```

2. What is the best wine I can buy for a given amount of money? Create a Series whose index is wine prices and whose values is the maximum number of points a wine costing that much was given in a review. Sort the values by price, ascending (so that 4.0 dollars is at the top and 3300.0 dollars is at the bottom).

In [9]:

```
#2
best_wine = review.groupby(['price']).points.agg([max])
best_wine
```

Out[9]:

price	max
4.0	86
5.0	90
6.0	90
7.0	90
8.0	91
...	...
1300.0	96
1400.0	100
1900.0	98
2013.0	91
2300.0	99

357 rows × 1 columns

3. What are the minimum and maximum prices for each variety of wine? Create a DataFrame whose index is the variety category from the dataset and whose values are the min and max values thereof.

In [11]:

```
#3
variety = review.groupby(['variety']).price.agg([min, max])
variety
```

Out[11]:

variety	min	max
Agiorgitiko	8.0	65.0
Aglianico	6.0	130.0
Aidani	27.0	27.0
Airen	8.0	10.0
Albana	8.0	66.0
...	...	...
Zierfandler-Rotgipfler	20.0	25.0
Zinfandel	4.0	100.0
Zlahtina	13.0	17.0
Zweigelt	9.0	70.0
Žilavka	13.0	15.0

632 rows × 2 columns

4. What are the most expensive wine varieties? Create a variable sorted_varieties containing a copy of the dataframe from the previous question where varieties are sorted in descending order based on minimum price, then on maximum price (to break ties).

In [12]:

```
#4
sorted_varieties = variety.sort_values(by = ['max', 'min'], ascending=False)
sorted_varieties
```

Out[12]:

variety	min	max
Bordeaux-style Red Blend	7.0	2300.0
Chardonnay	4.0	2013.0
Grüner Veltliner	9.0	1100.0
Bordeaux-style White Blend	8.0	1000.0
Port	11.0	980.0
...	...	...
Rabigato	NaN	NaN
Sacy	NaN	NaN
Sauvignon Blanc-Sauvignon Gris	NaN	NaN
Terret Blanc	NaN	NaN
Zelen	NaN	NaN

632 rows × 2 columns

6. What combination of countries and varieties are most common? Create a Series whose index is a MultiIndexof {country, variety} pairs. For example, a pinot noir produced in the US should map to {"US", "Pinot Noir"}. Sort the values in the Series in descending order based on wine count.

In [19]:

```
#COMBINING ALL THE METHODS INTO ONE DATAFRAME AND FUNC.
#6
var_count = review.groupby(['country', 'variety']).points.agg([len]).sort_values(by=(['len'
var_count
```

Out[19]:

		len
country	**variety**	
US	**Pinot Noir**	10340
	Cabernet Sauvignon	9178
	Chardonnay	8127
France	**Bordeaux-style Red Blend**	4960
US	**Syrah**	4274
...	**...**	...
France	**Pied de Perdrix**	1
Tunisia	**White Blend**	1
	Rosé	1
Switzerland	**White Blend**	1
US	**Carignan-Grenache**	1

1475 rows × 1 columns

In [13]:

```
# SPLITTING QUESTION 6 INTO TWO SETS A,B
#6A
var_count = review.groupby(['country', 'variety']).points.agg([len])
var_count
```

Out[13]:

		len
country	**variety**	
Albania	**Kallmet**	2
Argentina	**Barbera**	2
	Bonarda	151
	Bordeaux-style Red Blend	106
	Bordeaux-style White Blend	3
...	**...**	...
Uruguay	**Rosé**	3
	Tannat	33
	Tannat-Merlot	3
	Tannat-Syrah	2
	Viognier	7

1475 rows × 1 columns

In [14]:

```
#6B
var_count2 = var_count.sort_values(by=(['len']), ascending=False)
var_count2
```

Out[14]:

		len
country	**variety**	
US	**Pinot Noir**	10340
	Cabernet Sauvignon	9178
	Chardonnay	8127
France	**Bordeaux-style Red Blend**	4960
US	**Syrah**	4274
...	**...**	...
France	**Pied de Perdrix**	1
Tunisia	**White Blend**	1
	Rosé	1
Switzerland	**White Blend**	1
US	**Carignan-Grenache**	1

1475 rows × 1 columns