In [1]:

```
#import the libraries

import pandas as pd
```

In [8]:

```
# convert dataframe to  csv file format
df = pd.read_csv("https://raw.githubusercontent.com/justmarkham/DAT8/master/data/drinks.csv"
df.to_csv("drinks.csv",index=False)
```

In [9]:

```
# first five records
df.head(5)
```

Out[9]:

	country	beer_servings	spirit_servings	wine_servings	total_litres_of_pure_alcohol	continent
0	Afghanistan	0	0	0	0.0	AS
1	Albania	89	132	54	4.9	EU
2	Algeria	25	0	14	0.7	AF
3	Andorra	245	138	312	12.4	EU
4	Angola	217	57	45	5.9	AF

In [12]:

```
# which continent drinks more beer on average
df.groupby('continent').beer_servings.mean()
```

Out[12]:

```
continent
AF     61.471698
AS     37.045455
EU    193.777778
OC     89.687500
SA    175.083333
Name: beer_servings, dtype: float64
```

In [13]:

```
# FInd the unique record of a column
df.continent.nunique()
```

Out[13]:

```
5
```

In [14]:

```
# find the count of unique records for a perticular column
df.continent.value_counts()
```

Out[14]:

```
AF    53
EU    45
AS    44
OC    16
SA    12
Name: continent, dtype: int64
```

In [21]:
```
# for each continent print the statistics for wine consumation

df.groupby('continent').wine_servings.describe()
```

Out[21]:

	count	mean	std	min	25%	50%	75%	max
continent								
AF	53.0	16.264151	38.846419	0.0	1.0	2.0	13.00	233.0
AS	44.0	9.068182	21.667034	0.0	0.0	1.0	8.00	123.0
EU	45.0	142.222222	97.421738	0.0	59.0	128.0	195.00	370.0
OC	16.0	35.625000	64.555790	0.0	1.0	8.5	23.25	212.0
SA	12.0	62.416667	88.620189	1.0	3.0	12.0	98.50	221.0

In [22]:
```
## print the average alcohol consumation for continent wise

df.groupby('continent').mean()
```

Out[22]:

	beer_servings	spirit_servings	wine_servings	total_litres_of_pure_alcohol
continent				
AF	61.471698	16.339623	16.264151	3.007547
AS	37.045455	60.840909	9.068182	2.170455
EU	193.777778	132.555556	142.222222	8.617778
OC	89.687500	58.437500	35.625000	3.381250
SA	175.083333	114.750000	62.416667	6.308333

In [23]:
```
df.groupby('continent').max()
```

Out[23]:

	country	beer_servings	spirit_servings	wine_servings	total_litres_of_pure_alcohol
continent					
AF	Zimbabwe	376	152	233	9.1
AS	Yemen	247	326	123	11.5
EU	United Kingdom	361	373	370	14.4
OC	Vanuatu	306	254	212	10.4
SA	Venezuela	333	302	221	8.3

In [24]:
```
df.groupby('continent').min()
```

Out[24]:

	country	beer_servings	spirit_servings	wine_servings	total_litres_of_pure_alcohol
continent					
AF	Algeria	0	0	0	0.0
AS	Afghanistan	0	0	0	0.0
EU	Albania	0	0	0	0.0
OC	Australia	0	0	0	0.0
SA	Argentina	93	25	1	3.8

In [26]:

```
df.nlargest(10,'beer_servings')
```

Out[26]:

	country	beer_servings	spirit_servings	wine_servings	total_litres_of_pure_alcohol	continent
117	Namibia	376	3	1	6.8	AF
45	Czech Republic	361	170	134	11.8	EU
62	Gabon	347	98	59	8.9	AF
65	Germany	346	117	175	11.3	EU
98	Lithuania	343	244	56	12.9	EU
135	Poland	343	215	56	10.9	EU
188	Venezuela	333	100	3	7.7	SA
81	Ireland	313	118	165	11.4	EU
129	Palau	306	63	23	6.9	OC
140	Romania	297	122	167	10.4	EU

In [27]:

```
df.nlargest(5,'wine_servings')
```

Out[27]:

	country	beer_servings	spirit_servings	wine_servings	total_litres_of_pure_alcohol	continent
61	France	127	151	370	11.8	EU
136	Portugal	194	67	339	11.0	EU
3	Andorra	245	138	312	12.4	EU
166	Switzerland	185	100	280	10.2	EU
48	Denmark	224	81	278	10.4	EU

Euro_2012_stats_TEAM

In [ ]:

In [34]:
```
euro = pd.read_csv("https://raw.githubusercontent.com/guipsamora/pandas_exercises/master/02_F
euro
```

Out[34]:

	Team	Goals	Shots on target	Shots off target	Shooting Accuracy	% Goals-to-shots	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored	...	Saves made	Save t sho rat
0	Croatia	4	13	12	51.9%	16.0%	32	0	0	0	...	13	81.3
1	Czech Republic	4	13	18	41.9%	12.9%	39	0	0	0	...	9	60.1
2	Denmark	4	10	10	50.0%	20.0%	27	1	0	0	...	10	66.7
3	England	5	11	18	50.0%	17.2%	40	0	0	0	...	22	88.1
4	France	3	22	24	37.9%	6.5%	65	1	0	0	...	6	54.6
5	Germany	10	32	32	47.8%	15.6%	80	2	1	0	...	10	62.6
6	Greece	5	8	18	30.7%	19.2%	32	1	1	1	...	13	65.1
7	Italy	6	34	45	43.0%	7.5%	110	2	0	0	...	20	74.1
8	Netherlands	2	12	36	25.0%	4.1%	60	2	0	0	...	12	70.6
9	Poland	2	15	23	39.4%	5.2%	48	0	0	0	...	6	66.7
10	Portugal	6	22	42	34.3%	9.3%	82	6	0	0	...	10	71.5
11	Republic of Ireland	1	7	12	36.8%	5.2%	28	0	0	0	...	17	65.4
12	Russia	5	9	31	22.5%	12.5%	59	2	0	0	...	10	77.0
13	Spain	12	42	33	55.9%	16.0%	100	0	1	0	...	15	93.8
14	Sweden	5	17	19	47.2%	13.8%	39	3	0	0	...	8	61.6
15	Ukraine	2	7	26	21.2%	6.0%	38	0	0	0	...	13	76.5

16 rows × 35 columns

In [35]:
```
euro.head(1)
```

Out[35]:

	Team	Goals	Shots on target	Shots off target	Shooting Accuracy	% Goals-to-shots	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored	...	Saves made	Saves-to-shots ratio	F
0	Croatia	4	13	12	51.9%	16.0%	32	0	0	0	...	13	81.3%	

1 rows × 35 columns

In [36]:
```
euro.shape
```

Out[36]:
```
(16, 35)
```

In [37]:
```
euro.groupby('Goals')
```

Out[37]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x000002B528C3E190>
```

Select only one column

```
In [55]: euro['Goals']
         #euro.loc[:,'Goals']
         # euro.Goals
```

```
Out[55]: 0      4
         1      4
         2      4
         3      5
         4      3
         5     10
         6      5
         7      6
         8      2
         9      2
         10     6
         11     1
         12     5
         13    12
         14     5
         15     2
         Name: Goals, dtype: int64
```

```
In [ ]: ## How many team are participating in dataset?
```

```
In [57]: euro.Team.nunique()
```

```
Out[57]: 16
```

```
In [59]: euro.shape[0]
```

```
Out[59]: 16
```

```
In [61]: len(euro['Team'])
```

```
Out[61]: 16
```

```
In [62]: euro.shape[1]
```

```
Out[62]: 35
```

```
In [ ]: ## How many columns we have in this dataframe?
```

In [63]:
```
euro.columns
```

Out[63]:
```
Index(['Team', 'Goals', 'Shots on target', 'Shots off target',
       'Shooting Accuracy', '% Goals-to-shots', 'Total shots (inc. Blocked)',
       'Hit Woodwork', 'Penalty goals', 'Penalties not scored', 'Headed goals',
       'Passes', 'Passes completed', 'Passing Accuracy', 'Touches', 'Crosses',
       'Dribbles', 'Corners Taken', 'Tackles', 'Clearances', 'Interceptions',
       'Clearances off line', 'Clean Sheets', 'Blocks', 'Goals conceded',
       'Saves made', 'Saves-to-shots ratio', 'Fouls Won', 'Fouls Conceded',
       'Offsides', 'Yellow Cards', 'Red Cards', 'Subs on', 'Subs off',
       'Players Used'],
      dtype='object')
```

In [64]:
```
euro.loc[:,['Team','Goals','Shots on target']]
```

Out[64]:

	Team	Goals	Shots on target
0	Croatia	4	13
1	Czech Republic	4	13
2	Denmark	4	10
3	England	5	11
4	France	3	22
5	Germany	10	32
6	Greece	5	8
7	Italy	6	34
8	Netherlands	2	12
9	Poland	2	15
10	Portugal	6	22
11	Republic of Ireland	1	7
12	Russia	5	9
13	Spain	12	42
14	Sweden	5	17
15	Ukraine	2	7

In [ ]:
```
## view only the columns we want to add
```

In [66]:

```
euro[['Team','Goals','Shots on target']]
```

Out[66]:

	Team	Goals	Shots on target
0	Croatia	4	13
1	Czech Republic	4	13
2	Denmark	4	10
3	England	5	11
4	France	3	22
5	Germany	10	32
6	Greece	5	8
7	Italy	6	34
8	Netherlands	2	12
9	Poland	2	15
10	Portugal	6	22
11	Republic of Ireland	1	7
12	Russia	5	9
13	Spain	12	42
14	Sweden	5	17
15	Ukraine	2	7

In [67]:

```
euro.tail(-3)
```

Out[67]:

	Team	Goals	Shots on target	Shots off target	Shooting Accuracy	% Goals-to-shots	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored	...	Saves made	Save t sho rat
3	England	5	11	18	50.0%	17.2%	40	0	0	0	...	22	88.1
4	France	3	22	24	37.9%	6.5%	65	1	0	0	...	6	54.6
5	Germany	10	32	32	47.8%	15.6%	80	2	1	0	...	10	62.6
6	Greece	5	8	18	30.7%	19.2%	32	1	1	1	...	13	65.1
7	Italy	6	34	45	43.0%	7.5%	110	2	0	0	...	20	74.1
8	Netherlands	2	12	36	25.0%	4.1%	60	2	0	0	...	12	70.6
9	Poland	2	15	23	39.4%	5.2%	48	0	0	0	...	6	66.7
10	Portugal	6	22	42	34.3%	9.3%	82	6	0	0	...	10	71.5
11	Republic of Ireland	1	7	12	36.8%	5.2%	28	0	0	0	...	17	65.4
12	Russia	5	9	31	22.5%	12.5%	59	2	0	0	...	10	77.0
13	Spain	12	42	33	55.9%	16.0%	100	0	1	0	...	15	93.8
14	Sweden	5	17	19	47.2%	13.8%	39	3	0	0	...	8	61.6
15	Ukraine	2	7	26	21.2%	6.0%	38	0	0	0	...	13	76.5

13 rows × 35 columns

```
In [ ]:
```