Ex2 - Filtering and Sorting Data

This time we are going to pull data directly from the internet.

Step 1. Import the necessary libraries

In [18]:

```python
import pandas as pd
```

Step 2. Import the dataset from this address.

In [32]:

```python
df = pd.read_csv("https://raw.githubusercontent.com/guipsamora/pandas_exercises/master/02_Filtering_%26_Sorting/Euro12/Euro_2012_stats_TEAM.csv")
df
```

Out[32]:

	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	Fouls Won	Fouls Conceded	Offsides	Yellow Cards	Red Cards	Subs on	Subs off	Players Used
0	Croatia	4	13	12	51.9%	16.0%	32	0	0	0	...	13	81.3%	41	62	2	9	0	9	9	16
1	Czech Republic	4	13	18	41.9%	12.9%	39	0	0	0	...	9	60.1%	53	73	8	7	0	11	11	19
2	Denmark	4	10	10	50.0%	20.0%	27	1	0	0	...	10	66.7%	25	38	8	4	0	7	7	15
3	England	5	11	18	50.0%	17.2%	40	0	0	0	...	22	88.1%	43	45	6	5	0	11	11	16
4	France	3	22	24	37.9%	6.5%	65	1	0	0	...	6	54.6%	36	51	5	6	0	11	11	19
5	Germany	10	32	32	47.8%	15.6%	80	2	1	0	...	10	62.6%	63	49	12	4	0	15	15	17
6	Greece	5	8	18	30.7%	19.2%	32	1	1	1	...	13	65.1%	67	48	12	9	1	12	12	20
7	Italy	6	34	45	43.0%	7.5%	110	2	0	0	...	20	74.1%	101	89	16	16	0	18	18	19
8	Netherlands	2	12	36	25.0%	4.1%	60	2	0	0	...	12	70.6%	35	30	3	5	0	7	7	15
9	Poland	2	15	23	39.4%	5.2%	48	0	0	0	...	6	66.7%	48	56	3	7	1	7	7	17
10	Portugal	6	22	42	34.3%	9.3%	82	6	0	0	...	10	71.5%	73	90	10	12	0	14	14	16
11	Republic of Ireland	1	7	12	36.8%	5.2%	28	0	0	0	...	17	65.4%	43	51	11	6	1	10	10	17
12	Russia	5	9	31	22.5%	12.5%	59	2	0	0	...	10	77.0%	34	43	4	6	0	7	7	16
13	Spain	12	42	33	55.9%	16.0%	100	0	1	0	...	15	93.8%	102	83	19	11	0	17	17	18
14	Sweden	5	17	19	47.2%	13.8%	39	3	0	0	...	8	61.6%	35	51	7	7	0	9	9	18
15	Ukraine	2	7	26	21.2%	6.0%	38	0	0	0	...	13	76.5%	48	31	4	5	0	9	9	18

16 rows × 35 columns

Step 3. Assign it to a variable called euro12.

In [61]:

```python
df
```

Out[61]:

	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	Fouls Won	Fouls Conceded	Offsides	Yellow Cards	Red Cards	Subs on	Subs off	Players Used
0	Croatia	4	13	12	51.9%	16.0%	32	0	0	0	...	13	81.3%	41	62	2	9	0	9	9	16
1	Czech Republic	4	13	18	41.9%	12.9%	39	0	0	0	...	9	60.1%	53	73	8	7	0	11	11	19
2	Denmark	4	10	10	50.0%	20.0%	27	1	0	0	...	10	66.7%	25	38	8	4	0	7	7	15
3	England	5	11	18	50.0%	17.2%	40	0	0	0	...	22	88.1%	43	45	6	5	0	11	11	16
4	France	3	22	24	37.9%	6.5%	65	1	0	0	...	6	54.6%	36	51	5	6	0	11	11	19
5	Germany	10	32	32	47.8%	15.6%	80	2	1	0	...	10	62.6%	63	49	12	4	0	15	15	17
6	Greece	5	8	18	30.7%	19.2%	32	1	1	1	...	13	65.1%	67	48	12	9	1	12	12	20
7	Italy	6	34	45	43.0%	7.5%	110	2	0	0	...	20	74.1%	101	89	16	16	0	18	18	19
8	Netherlands	2	12	36	25.0%	4.1%	60	2	0	0	...	12	70.6%	35	30	3	5	0	7	7	15
9	Poland	2	15	23	39.4%	5.2%	48	0	0	0	...	6	66.7%	48	56	3	7	1	7	7	17
10	Portugal	6	22	42	34.3%	9.3%	82	6	0	0	...	10	71.5%	73	90	10	12	0	14	14	16
11	Republic of Ireland	1	7	12	36.8%	5.2%	28	0	0	0	...	17	65.4%	43	51	11	6	1	10	10	17
12	Russia	5	9	31	22.5%	12.5%	59	2	0	0	...	10	77.0%	34	43	4	6	0	7	7	16
13	Spain	12	42	33	55.9%	16.0%	100	0	1	0	...	15	93.8%	102	83	19	11	0	17	17	18
14	Sweden	5	17	19	47.2%	13.8%	39	3	0	0	...	8	61.6%	35	51	7	7	0	9	9	18
15	Ukraine	2	7	26	21.2%	6.0%	38	0	0	0	...	13	76.5%	48	31	4	5	0	9	9	18

16 rows × 35 columns

In [ ]:

Step 4. Select only the Goal column.

In [44]:

```python
df["Goals"]
```

Out[44]:

```
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
```

Step 5. How many team participated in the Euro2012?

In [43]:

```python
df.count()
```

Out[43]:

```
Team                          16
Goals                         16
Shots on target               16
Shots off target              16
Shooting Accuracy             16
% Goals-to-shots              16
Total shots (inc. Blocked)    16
Hit Woodwork                  16
Penalty goals                 16
Penalties not scored          16
Headed goals                  16
Passes                        16
Passes completed              16
Passing Accuracy              16
Touches                       16
Crosses                       16
Dribbles                      16
Corners Taken                 16
Tackles                       16
Clearances                    16
Interceptions                 16
Clearances off line           15
Clean Sheets                  16
Blocks                        16
Goals conceded                16
Saves made                    16
Saves-to-shots ratio          16
Fouls Won                     16
Fouls Conceded                16
Offsides                      16
Yellow Cards                  16
Red Cards                     16
Subs on                       16
Subs off                      16
Players Used                  16
dtype: int64
```

In [47]:

```python
print(len(df.Team))
```

```
16
```

Step 6. What is the number of columns in the dataset?

In [46]:

```python
print(len(df.columns))
```

```
35
```

Step 7. View only the columns Team, Yellow Cards and Red Cards and assign them to a dataframe called discipline

In [51]:

```python
discipline = df[["Team","Yellow Cards","Red Cards"]]
discipline
```

Out[51]:

	Team	Yellow Cards	Red Cards
0	Croatia	9	0
1	Czech Republic	7	0
2	Denmark	4	0
3	England	5	0
4	France	6	0
5	Germany	4	0
6	Greece	9	1
7	Italy	16	0
8	Netherlands	5	0
9	Poland	7	1
10	Portugal	12	0
11	Republic of Ireland	6	1
12	Russia	6	0
13	Spain	11	0
14	Sweden	7	0
15	Ukraine	5	0

In [53]:

```python
discipline
```

Out[53]:

	Team	Yellow Cards	Red Cards
0	Croatia	9	0
1	Czech Republic	7	0
2	Denmark	4	0
3	England	5	0
4	France	6	0
5	Germany	4	0
6	Greece	9	1
7	Italy	16	0
8	Netherlands	5	0
9	Poland	7	1
10	Portugal	12	0
11	Republic of Ireland	6	1
12	Russia	6	0
13	Spain	11	0
14	Sweden	7	0
15	Ukraine	5	0

Step 8. Sort the teams by Red Cards, then to Yellow Cards

In [59]:

```python
discipline.sort_values(by=['Red Cards', 'Yellow Cards'], inplace = True)
discipline
```

```
C:\Users\Mua\anaconda3\lib\site-packages\pandas\util\_decorators.py:311: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  return func(*args, **kwargs)
```

Out[59]:

	Team	Yellow Cards	Red Cards
2	Denmark	4	0
5	Germany	4	0
3	England	5	0
8	Netherlands	5	0
15	Ukraine	5	0
4	France	6	0
12	Russia	6	0
1	Czech Republic	7	0
14	Sweden	7	0
0	Croatia	9	0
13	Spain	11	0
10	Portugal	12	0
7	Italy	16	0
11	Republic of Ireland	6	1
9	Poland	7	1
6	Greece	9	1

Step 9. Calculate the mean Yellow Cards given per Team

In [62]:

```python
df["Yellow Cards"].mean()
```

Out[62]:

```
7.4375
```

Step 10. Filter teams that scored more than 6 goals

In [64]:

```python
df["Goals"]>6
```

Out[64]:

```
0     False
1     False
2     False
3     False
4     False
5      True
6     False
7     False
8     False
9     False
10    False
11    False
12    False
13     True
14    False
15    False
Name: Goals, dtype: bool
```

Step 11. Select the teams that start with G

In [65]:

```python
df[df.Team.str.startswith("G")]
```

Out[65]:

	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	Fouls Won	Fouls Conceded	Offsides	Yellow Cards	Red Cards	Subs on	Subs off	Players Used
5	Germany	10	32	32	47.8%	15.6%	80	2	1	0	...	10	62.6%	63	49	12	4	0	15	15	17
6	Greece	5	8	18	30.7%	19.2%	32	1	1	1	...	13	65.1%	67	48	12	9	1	12	12	20

2 rows × 35 columns

Step 12. Select the first 7 columns

In [67]:

```python
df.head(7)
```

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	Saves-to-shots ratio	Fouls Won	Fouls Conceded	Offsides	Yellow Cards	Red Cards	Subs on	Subs off	Players Used
0	Croatia	4	13	12	51.9%	16.0%	32	0	0	0	...	13	81.3%	41	62	2	9	0	9	9	16
1	Czech Republic	4	13	18	41.9%	12.9%	39	0	0	0	...	9	60.1%	53	73	8	7	0	11	11	19
2	Denmark	4	10	10	50.0%	20.0%	27	1	0	0	...	10	66.7%	25	38	8	4	0	7	7	15
3	England	5	11	18	50.0%	17.2%	40	0	0	0	...	22	88.1%	43	45	6	5	0	11	11	16
4	France	3	22	24	37.9%	6.5%	65	1	0	0	...	6	54.6%	36	51	5	6	0	11	11	19
5	Germany	10	32	32	47.8%	15.6%	80	2	1	0	...	10	62.6%	63	49	12	4	0	15	15	17
6	Greece	5	8	18	30.7%	19.2%	32	1	1	1	...	13	65.1%	67	48	12	9	1	12	12	20

7 rows × 35 columns

Step 13. Select all columns except the last 3.

In [72]:

```python
df.iloc[:, :-3]
```

Out[72]:

	Team	Goals	Shots on target	Shots off target	Shooting Accuracy	% Goals-to-shots	Total shots (inc. Blocked)	Hit Woodwork	Penalty goals	Penalties not scored	...	Clean Sheets	Blocks	Goals conceded	Saves made	Saves-to-shots ratio	Fouls Won	Fouls Conceded	Offsides	Yellow Cards	Red Cards
0	Croatia	4	13	12	51.9%	16.0%	32	0	0	0	...	0	10	3	13	81.3%	41	62	2	9	0
1	Czech Republic	4	13	18	41.9%	12.9%	39	0	0	0	...	1	10	6	9	60.1%	53	73	8	7	0
2	Denmark	4	10	10	50.0%	20.0%	27	1	0	0	...	1	10	5	10	66.7%	25	38	8	4	0
3	England	5	11	18	50.0%	17.2%	40	0	0	0	...	2	29	3	22	88.1%	43	45	6	5	0
4	France	3	22	24	37.9%	6.5%	65	1	0	0	...	1	7	5	6	54.6%	36	51	5	6	0
5	Germany	10	32	32	47.8%	15.6%	80	2	1	0	...	1	11	6	10	62.6%	63	49	12	4	0
6	Greece	5	8	18	30.7%	19.2%	32	1	1	1	...	1	23	7	13	65.1%	67	48	12	9	1
7	Italy	6	34	45	43.0%	7.5%	110	2	0	0	...	2	18	7	20	74.1%	101	89	16	16	0
8	Netherlands	2	12	36	25.0%	4.1%	60	2	0	0	...	0	9	5	12	70.6%	35	30	3	5	0
9	Poland	2	15	23	39.4%	5.2%	48	0	0	0	...	0	8	3	6	66.7%	48	56	3	7	1
10	Portugal	6	22	42	34.3%	9.3%	82	6	0	0	...	2	11	4	10	71.5%	73	90	10	12	0
11	Republic of Ireland	1	7	12	36.8%	5.2%	28	0	0	0	...	0	23	9	17	65.4%	43	51	11	6	1
12	Russia	5	9	31	22.5%	12.5%	59	2	0	0	...	0	8	3	10	77.0%	34	43	4	6	0
13	Spain	12	42	33	55.9%	16.0%	100	0	1	0	...	5	8	1	15	93.8%	102	83	19	11	0
14	Sweden	5	17	19	47.2%	13.8%	39	3	0	0	...	1	12	5	8	61.6%	35	51	7	7	0
15	Ukraine	2	7	26	21.2%	6.0%	38	0	0	0	...	0	4	4	13	76.5%	48	31	4	5	0

16 rows × 32 columns

Step 14. Present only the Shooting Accuracy from England, Italy and Russia

In [74]:

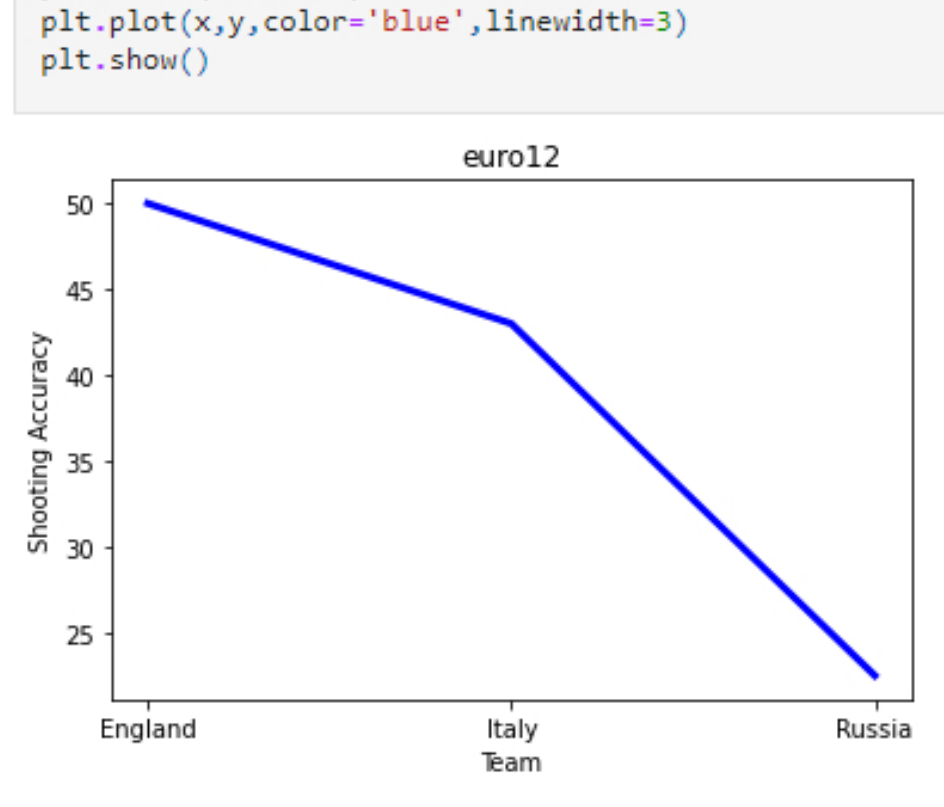
```python
df.loc[df.Team.isin(['England', 'Italy', 'Russia']), ['Team','Shooting Accuracy']]
```

Out[74]:

	Team	Shooting Accuracy
3	England	50.0%
7	Italy	43.0%
12	Russia	22.5%

In [98]:

```python
import matplotlib.pyplot as plt
%matplotlib inline
df = pd.read_csv("C:/Users/Mua/Downloads/euro12.csv").dropna()
x=("England","Italy","Russia")
y=(50.0,43.0,22.5)
plt.xlabel('Team')
plt.ylabel('Shooting Accuracy')
plt.title('euro12')
plt.plot(x,y,color='blue',linewidth=3)
plt.show()
```



In [ ]: