In [3]:
```
import pandas as pd
coffee=pd.read_csv('https://raw.githubusercontent.com/KeithGalli/complete-pandas-tutorial/refs/heads/master/warmup-da
coffee
```

Out[3]:

	Day	Coffee Type	Units Sold
0	Monday	Espresso	25
1	Monday	Latte	15
2	Tuesday	Espresso	30
3	Tuesday	Latte	20
4	Wednesday	Espresso	35
5	Wednesday	Latte	25
6	Thursday	Espresso	40
7	Thursday	Latte	30
8	Friday	Espresso	45
9	Friday	Latte	35
10	Saturday	Espresso	45
11	Saturday	Latte	35
12	Sunday	Espresso	45
13	Sunday	Latte	35

In [4]:
```
# ***Does not WORK***
# coffee.loc(['Day','Units Sold'])
# coffee.loc(5:8)
# coffee[5:8,['Day', 'Units Sold']]
# coffee[:]['Day','Units Sold']
# coffee.iloc[:, ['Day' , 'Units Sold']]
# coffee[[1,2,4],:]
# ***WORKs***

# coffee[1:8][:]  # به براکت دوم دقت شود که داخل براکت دیگر نیست و عملا روی براکت اول عملبات میکند

# coffee.iloc[:, [1,2]]
# coffee.iloc[:,:]
# coffee[:][['Day' , 'Units Sold']]
# coffee[5:8][['Day', 'Units Sold']] # دو براکت یکی داخلش براکت داره
# coffee.loc[:,['Day','Units Sold']]


# *******در کل از iloc و loc استفاده کنید بهتر است*******
```

In [5]:
```
coffee.iloc[5:, [0,2]].head()
```

Out[5]:

	Day	Units Sold
5	Wednesday	25
6	Thursday	40
7	Thursday	30
8	Friday	45
9	Friday	35

In [6]:
```
# ********NOT WORKS*********
# coffee.Units Sold  #این روش برای دسترسی به ستون "Units Sold" است اما کار نمیکند



# *******WORKS********
# coffee.Day
# coffee.Day # دسترسی به ستون "Day"

# coffee.sort_values(by=['Day','Units Sold'] , ascending=False) # مرتب سازی ستون اول نسبت به Day بعد از آن Units Sold
coffee.sort_values(['Day','Units Sold'] , ascending=False)
```

Out[6]:

	Day	Coffee Type	Units Sold
4	Wednesday	Espresso	35
5	Wednesday	Latte	25
2	Tuesday	Espresso	30
3	Tuesday	Latte	20
6	Thursday	Espresso	40
7	Thursday	Latte	30
12	Sunday	Espresso	45
13	Sunday	Latte	35
10	Saturday	Espresso	45
11	Saturday	Latte	35
0	Monday	Espresso	25
1	Monday	Latte	15
8	Friday	Espresso	45
9	Friday	Latte	35

In [7]:

```
for index, row in coffee.iterrows():
    print(index)
    print(row)
    print('\n\n')
```

```
0
Day              Monday
Coffee Type    Espresso
Units Sold           25
Name: 0, dtype: object
```

```
1
Day            Monday
Coffee Type     Latte
Units Sold         15
Name: 1, dtype: object
```

```
2
Day             Tuesday
Coffee Type    Espresso
Units Sold           30
Name: 2, dtype: object
```

```
3
Day            Tuesday
Coffee Type      Latte
Units Sold          20
Name: 3, dtype: object
```

```
4
Day            Wednesday
Coffee Type     Espresso
Units Sold            35
Name: 4, dtype: object
```

```
5
Day            Wednesday
```

```
Coffee Type        Latte
Units Sold            25
Name: 5, dtype: object
```

```
6
Day            Thursday
Coffee Type    Espresso
Units Sold           40
Name: 6, dtype: object
```

```
7
Day            Thursday
Coffee Type       Latte
Units Sold           30
Name: 7, dtype: object
```

```
8
Day              Friday
Coffee Type    Espresso
Units Sold           45
Name: 8, dtype: object
```

```
9
Day            Friday
Coffee Type     Latte
Units Sold         35
Name: 9, dtype: object
```

```
10
Day            Saturday
Coffee Type    Espresso
Units Sold           45
```

```
Name: 10, dtype: object



11
Day             Saturday
Coffee Type        Latte
Units Sold            35
Name: 11, dtype: object



12
Day               Sunday
Coffee Type     Espresso
Units Sold            45
Name: 12, dtype: object



13
Day             Sunday
Coffee Type      Latte
Units Sold          35
Name: 13, dtype: object
```

In [8]:

```
for index, row in coffee.iterrows():
    print(index)
    print(row)
    print('\n\n')
```

```
0
Day                 Monday
Coffee Type       Espresso
Units Sold              25
Name: 0, dtype: object
```

```
1
Day               Monday
Coffee Type        Latte
Units Sold            15
Name: 1, dtype: object
```

```
2
Day                Tuesday
Coffee Type       Espresso
Units Sold              30
Name: 2, dtype: object
```

```
3
Day               Tuesday
Coffee Type         Latte
Units Sold             20
Name: 3, dtype: object
```

```
4
Day               Wednesday
Coffee Type        Espresso
Units Sold               35
Name: 4, dtype: object
```

```
5
Day               Wednesday
```

```
Coffee Type        Latte
Units Sold            25
Name: 5, dtype: object
```

```
6
Day            Thursday
Coffee Type    Espresso
Units Sold           40
Name: 6, dtype: object
```

```
7
Day            Thursday
Coffee Type       Latte
Units Sold           30
Name: 7, dtype: object
```

```
8
Day              Friday
Coffee Type    Espresso
Units Sold           45
Name: 8, dtype: object
```

```
9
Day            Friday
Coffee Type     Latte
Units Sold         35
Name: 9, dtype: object
```

```
10
Day            Saturday
Coffee Type    Espresso
Units Sold           45
```

```
Name: 10, dtype: object



11
Day            Saturday
Coffee Type       Latte
Units Sold           35
Name: 11, dtype: object



12
Day              Sunday
Coffee Type    Espresso
Units Sold           45
Name: 12, dtype: object



13
Day            Sunday
Coffee Type     Latte
Units Sold         35
Name: 13, dtype: object
```

In [9]:

```
coffee.columns
```

Out[9]:

```
Index(['Day', 'Coffee Type', 'Units Sold'], dtype='object')
```

In [10]:

```
bios=pd.read_excel("C:/Users/LENOVO/Downloads/olympics-data.xlsx")
bios.head()
```

Out[10]:

	athlete_id	name	born_date	born_city	born_region	born_country	NOC	height_cm	weight_kg	died_date
0	1	Jean-François Blanchy	1886-12-12	Bordeaux	Gironde	FRA	France	NaN	NaN	1960-10-02
1	2	Arnaud Boetsch	1969-04-01	Meulan	Yvelines	FRA	France	183.0	76.0	NaN
2	3	Jean Borotra	1898-08-13	Biarritz	Pyrénées-Atlantiques	FRA	France	183.0	76.0	1994-07-17
3	4	Jacques Brugnon	1895-05-11	Paris VIIIe	Paris	FRA	France	168.0	64.0	1978-03-20
4	5	Albert Canet	1878-04-17	Wandsworth	England	GBR	France	NaN	NaN	1930-07-25

In [11]:

```
# a=bios.iloc[:,['height_cm']]>210
# a
b=bios['height_cm']>215
bios.loc[b,['name','height_cm','born_country']]
# bios.loc[a,['year']]
# bios.columns

# این یک DataFrame برمی‌گرداند:
#bios.loc[:,['height_cm']]        # shape: (n, 1)

# این یک Series برمی‌گرداند:
# bios['height_cm']                # shape: (n,)
# همیشه برای شرایط فیلتر کردن از Series استفاده کن، نه DataFrame!
```

Out[11]:

	name	height_cm	born_country
5089	Viktor Pankrashkin	220.0	RUS
5583	Paulinho Villas Boas	217.0	BRA
5673	Gunther Behnke	221.0	GER
5716	Uwe Blab	218.0	GER
5781	Tommy Burleson	223.0	USA
5796	Andy Campbell	218.0	AUS
6223	Lars Hansen	216.0	DEN
6270	Hu Zhangbao	216.0	NaN
6409	Sergey Kovalenko	216.0	NaN
6420	Jānis Krūmiņš	218.0	LAT
6504	Luc Longley	220.0	AUS
6722	Shaquille O'Neal	216.0	USA
6937	David Robinson	216.0	USA
6978	Arvydas Sabonis	223.0	LTU
7074	Paulo da Silva	217.0	BRA
7188	Vladimir Tkachenko	220.0	RUS
7281	Stojko Vranković	217.0	CRO
7376	Eurelijus Žukauskas	218.0	LTU
52608	Aleksey Kazakov	217.0	RUS
82100	Frédéric Weis	218.0	FRA
89070	Yao Ming	226.0	CHN
89075	Roberto Dueñas	221.0	ESP

	name	height_cm	born_country
107408	Peter John Ramos	219.0	PUR
112312	Stanko Barać	217.0	BIH
112332	Andreas Glyniadakis	216.0	GRE
112337	Hamed Haddadi	218.0	IRI
118663	Timofey Mozgov	216.0	NaN
118676	Dmitry Musersky	219.0	UKR
120266	Zhang Zhaoxu	221.0	CHN
121694	Salah Mejri	216.0	TUN
123850	Tyson Chandler	216.0	USA
130460	Li Muhao	218.0	CHN
130461	Zhou Qi	217.0	CHN
138671	Ondřej Balvín	216.0	CZE
139365	Moustapha Fall	218.0	FRA

In [12]:

```
# فیلتر کردن با دو شرط
bios.loc[(bios['height_cm']>215) & (bios['born_country']=='USA'),['name','height_cm','born_country']]

# فیلتر کردن براساس اسم دقیق
bios.loc[(bios['name'].str.contains('hamed|pouria',case=False)) & (bios['born_country'].str.lower()=='iri') & (bios[
```

Out[12]:

	athlete_id	name	born_date	born_city	born_region	born_country	NOC	height_cm	weight_kg	died_date
112337	113593	Hamed Haddadi	1985-05-19	Ahvaz	Khuzestan	IRI	Islamic Republic of Iran	218.0	110.0	NaN
112338	113594	Hamed Afagh	1983-02-01	Mashhad	Khorasan Razavi	IRI	Islamic Republic of Iran	188.0	82.0	NaN
112340	113596	Hamed Sohrabnejad	1983-05-07	Sanandaj	Kurdistan	IRI	Islamic Republic of Iran	202.0	95.0	NaN

In [ ]:

```
# Series.isin(list_of_values)  #چک میکنه که هر مقدار در ستون، آیا در لیست مشخص شده وجود داره یا نه
list_of_countries = ['USA', 'FRA']
# bios.loc[bios['born_country'].isin(list_of_countries)]

# endswith() → پایان رشته      # startswith()-> آغاز رشته
# bios.loc[bios['name'].str.lower().str.startswith('p') & bios['name'].str.lower().str.endswith('p')]

# match() → بررسی الگوی رشته با regex      #case(bool)= حساس به بزرگ و کوچک بودن حروف   #na(bool)= بررسی مقادیر خالی
bios.loc[bios['name'].str.lower().str.match(r'^h.*p$', case=False ) & bios['born_city'].str.match(r'^a' ,case=False
```

Out[ ]:

	athlete_id	name	born_date	born_city	born_region	born_country	NOC	height_cm	weight_kg	died_date
31752	31991	Heidi Astrup	1972-05-31	NaN	NaN	NaN	Denmark	175.0	65.0	NaN
59595	60021	Harry Challstorp	1884-02-07	NaN	NaN	NaN	Sweden	183.0	NaN	1961-10-17
128369	130893	Helah Kiprop	1985-04-07	NaN	NaN	NaN	Kenya	160.0	50.0	NaN

In [63]:

```
# *****NO WORK*****

# bios.query('height_cm > 210 & born_country == "IRI" or "USA"')

# *****WORKs*****
# bios.query('height_cm > 210 & (born_country == "IRI" | born_country == "USA")')
```

```
countries=['IRI' , 'USA']

# bios.query('height_cm > 210 & (born_country == @countries)')
# bios.query('height_cm > 210 & (born_country in @countries)')
# bios.query('height_cm > 210 & born_country ==["IRI", "USA"]')
bios.query('height_cm > 210 & born_country in["IRI", "USA"]')
```

Out[63]:

	athlete_id	name	born_date	born_city	born_region	born_country	NOC	height_cm	weight_kg	died_date
5681	5704	Walt Bellamy	1937-07-24	New Bern	North Carolina	USA	United States	211.0	98.0	2013-11-02
5781	5804	Tommy Burleson	1952-02-24	Crossnore	North Carolina	USA	United States	223.0	102.0	NaN
5894	5919	Mel Counts	1941-10-16	Coos Bay	Oregon	USA	United States	213.0	102.0	NaN
6106	6133	Marc Freiberger	1928-11-27	Amarillo	Texas	USA	United States	211.0	98.0	2005-06-29
6281	6309	Darrall Imhoff	1938-10-11	San Gabriel	California	USA	United States	211.0	98.0	2017-06-30
6379	6410	Joe Kleine	1962-01-04	Colorado Springs	Colorado	USA	United States	211.0	122.0	NaN
6393	6424	Jon Koncak	1963-05-17	Cedar Rapids	Iowa	USA	United States	213.0	113.0	NaN
6442	6473	Christian Laettner	1969-08-17	Angola	New York	USA	United States	211.0	107.0	NaN
6615	6648	Tom McMillen	1952-05-26	Mansfield	Pennsylvania	USA	United States	212.0	95.0	NaN
6722	6755	Shaquille O'Neal	1972-03-06	Newark	New Jersey	USA	United States	216.0	137.0	NaN
6937	6972	David Robinson	1965-08-06	Key West	Florida	USA	United States	216.0	107.0	NaN
7012	7048	Daniel Santiago	1976-06-24	Tarpon Springs	Florida	USA	Puerto Rico	213.0	111.0	NaN
112319	113575	Chris Kaman	1982-04-28	Wyoming	Michigan	USA	Germany	213.0	122.0	NaN
112337	113593	Hamed Haddadi	1985-05-19	Ahvaz	Khuzestan	IRI	Islamic Republic of Iran	218.0	110.0	NaN

	athlete_id	name	born_date	born_city	born_region	born_country	NOC	height_cm	weight_kg	died_date
112347	113603	Mousa Nabipour	1983-08-14	Behshahr	Mazandaran	IRI	Islamic Republic of Iran	212.0	105.0	NaN
112366	113622	Dwight Howard	1985-12-08	Atlanta	Georgia	USA	United States	211.0	120.0	NaN
118573	120309	Al-Farouq Aminu	1990-09-21	Atlanta	Georgia	USA	Nigeria	211.0	98.0	NaN
123850	126093	Tyson Chandler	1982-10-02	Hanford	California	USA	United States	216.0	107.0	NaN
132188	135037	Domantas Sabonis	1996-05-03	Portland	Oregon	USA	Lithuania	211.0	109.0	NaN
133332	136271	DeAndre Jordan	1988-07-21	Houston	Texas	USA	United States	211.0	120.0	NaN
142044	145640	Jahlil Okafor	1995-12-15	Fort Smith	Arkansas	USA	Nigeria	211.0	NaN	NaN
142930	146560	Mike Tobey	1994-10-10	Monroe	New York	USA	Slovenia	213.0	NaN	NaN
143712	147367	JaVale McGee	1988-01-19	Flint	Michigan	USA	United States	213.0	NaN	NaN

```
In [64]: coffee
```

Out[64]:

	Day	Coffee Type	Units Sold
0	Monday	Espresso	25
1	Monday	Latte	15
2	Tuesday	Espresso	30
3	Tuesday	Latte	20
4	Wednesday	Espresso	35
5	Wednesday	Latte	25
6	Thursday	Espresso	40
7	Thursday	Latte	30
8	Friday	Espresso	45
9	Friday	Latte	35
10	Saturday	Espresso	45
11	Saturday	Latte	35
12	Sunday	Espresso	45
13	Sunday	Latte	35

```
In [ ]:
```