In [1]:

```
import numpy as np
import pandas as pd
```

In [2]:

```
s = pd.Series([1, 3, 5, np.nan, 6, 8])
s
```

Out[2]:

```
0    1.0
1    3.0
2    5.0
3    NaN
4    6.0
5    8.0
dtype: float64
```

In [3]:

```
print(s)
```

```
0    1.0
1    3.0
2    5.0
3    NaN
4    6.0
5    8.0
dtype: float64
```

In [7]:

```
dates = pd.date_range("20210801", periods=6)
dates
```

Out[7]:

```
DatetimeIndex(['2021-08-01', '2021-08-02', '2021-08-03', '2021-08-04',
               '2021-08-05', '2021-08-06'],
              dtype='datetime64[ns]', freq='D')
```

In [8]:

```
df = pd.DataFrame(np.random.randn(6, 4), index=dates, columns=list("ABCD"))
```

In [9]:

```
df
```

Out[9]:

	A	B	C	D
2021-08-01	1.172683	-0.320713	0.469766	-0.092718
2021-08-02	1.491463	-0.315457	-0.348181	0.342860
2021-08-03	-1.043681	0.901628	0.815549	-1.447308
2021-08-04	-1.453992	1.185176	-0.942859	-0.969936
2021-08-05	-1.863945	-0.298399	-0.735511	0.595694
2021-08-06	-0.811115	2.193013	0.185606	-0.984710

In [10]:

```
df["A"]
```

Out[10]:

```
2021-08-01    1.172683
2021-08-02    1.491463
2021-08-03   -1.043681
2021-08-04   -1.453992
2021-08-05   -1.863945
2021-08-06   -0.811115
Freq: D, Name: A, dtype: float64
```

In [11]:

```
df[0:3]
```

Out[11]:

	A	B	C	D
2021-08-01	1.172683	-0.320713	0.469766	-0.092718
2021-08-02	1.491463	-0.315457	-0.348181	0.342860
2021-08-03	-1.043681	0.901628	0.815549	-1.447308

In [12]:

```
#accessing a particular index
df.loc[dates[0]]
```

Out[12]:

```
A    1.172683
B   -0.320713
C    0.469766
D   -0.092718
Name: 2021-08-01 00:00:00, dtype: float64
```

In [13]:

```
#Selecting on a multi-axis by label:
df.loc[:, ["A", "B"]]
```

Out[13]:

	A	B
2021-08-01	1.172683	-0.320713
2021-08-02	1.491463	-0.315457
2021-08-03	-1.043681	0.901628
2021-08-04	-1.453992	1.185176
2021-08-05	-1.863945	-0.298399
2021-08-06	-0.811115	2.193013

In [14]:

```
#Select via the position of the passed integers:
df.iloc[3]
```

Out[14]:

```
A   -1.453992
B    1.185176
C   -0.942859
D   -0.969936
Name: 2021-08-04 00:00:00, dtype: float64
```

In [15]:

```
#For slicing rows explicitly:
df.iloc[1:3, :]
```

Out[15]:

	A	B	C	D
2021-08-02	1.491463	-0.315457	-0.348181	0.342860
2021-08-03	-1.043681	0.901628	0.815549	-1.447308

In [16]:

```
#For slicing columns explicitly:
df.iloc[:, 1:3]
```

Out[16]:

	B	C
2021-08-01	-0.320713	0.469766
2021-08-02	-0.315457	-0.348181
2021-08-03	0.901628	0.815549
2021-08-04	1.185176	-0.942859
2021-08-05	-0.298399	-0.735511
2021-08-06	2.193013	0.185606

In [17]:

```
#Using a single column’s values to select data.
df[df["A"] > 0]
```

Out[17]:

	A	B	C	D
2021-08-01	1.172683	-0.320713	0.469766	-0.092718
2021-08-02	1.491463	-0.315457	-0.348181	0.342860

In [18]:

```
df2 = df.copy()
df2["E"] = ["one", "one", "two", "three", "four", "three"]
df2
```

Out[18]:

	A	B	C	D	E
2021-08-01	1.172683	-0.320713	0.469766	-0.092718	one
2021-08-02	1.491463	-0.315457	-0.348181	0.342860	one
2021-08-03	-1.043681	0.901628	0.815549	-1.447308	two
2021-08-04	-1.453992	1.185176	-0.942859	-0.969936	three
2021-08-05	-1.863945	-0.298399	-0.735511	0.595694	four
2021-08-06	-0.811115	2.193013	0.185606	-0.984710	three

In [19]:

```
df2[df2["E"].isin(["two", "four"])]
```

Out[19]:

	A	B	C	D	E
2021-08-03	-1.043681	0.901628	0.815549	-1.447308	two
2021-08-05	-1.863945	-0.298399	-0.735511	0.595694	four