

```
In [1]: import pandas as pd
import numpy as np
```

```
In [3]: data = {'Name': ['ratna', 'smruti', 'Bob', 'Eve', 'Charlie'],
               'Age': [20, 20, 22, 35, 28],
               'Gender': ['Female', 'Female', 'Male', 'Female', 'Male'],
               'Salary': [50000, 55000, 40000, 70000, 48000]}

df = pd.DataFrame(data)
print(df.index) # Accessing the index
```

RangeIndex(start=0, stop=5, step=1)

```
In [5]: # custom index --> set_index() --> allow you to set
# custom index based on column, such as name or age
df_with_index = df.set_index('Name')
print(df_with_index)
```

	Age	Gender	Salary
Name			
ratna	20	Female	50000
smruti	20	Female	55000
Bob	22	Male	40000
Eve	35	Female	70000
Charlie	28	Male	48000

```
In [7]: # reset index -> it will convert current index to regular column
# and create a new default index
df_reset = df.reset_index()
print(df_reset)
```

	index	Name	Age	Gender	Salary
0	0	ratna	20	Female	50000
1	1	smruti	20	Female	55000
2	2	Bob	22	Male	40000
3	3	Eve	35	Female	70000
4	4	Charlie	28	Male	48000

```
In [ ]:
```

```
In [7]: print(df.index)
```

RangeIndex(start=0, stop=5, step=1)

```
In [21]: df[df['Name'] == 'smruti']
```

```
Out[21]:
```

	Name	Age	Gender	Salary
1	smruti	20	Female	55000

```
In [23]: #changing index --> set_index() --> it allows to set one or more columns as the new i
df_with_new_index = df.set_index('Age')
print(df_with_new_index)
```

	Name	Gender	Salary
Age			
20	ratna	Female	50000
20	smruti	Female	55000
22	Bob	Male	40000
35	Eve	Female	70000
28	Charlie	Male	48000

```
In [25]: # access data frame
age_column = df['Age']
print(age_column)
```

```
0    20
1    20
2    22
3    35
4    28
Name: Age, dtype: int64
```

```
In [33]: # access rows
second_row = df.iloc[0]
print(second_row)
```

```
Name    ratna
Age      20
Gender  Female
Salary  50000
Name: 0, dtype: object
```

```
In [37]: # access multiple row and columns
subset = df.loc[0:2,['Name','Age']]
print(subset)
```

```
   Name  Age
0  ratna  20
1  smruti  20
2    Bob  22
```

```
In [41]: # row based on condition-->filter rows
filter_data = df[df['Age']<24]
print(filter_data)
```

```
   Name  Age  Gender  Salary
0  ratna  20  Female  50000
1  smruti  20  Female  55000
2    Bob  22   Male  40000
```

```
In [43]: # access specific cells with at and iat
# at--> its for label-based indexing
# iat-->it for integer position based indexing
salary_at_index_2 = df.at[2,'Salary']
print(salary_at_index_2)
```

```
40000
```

```
In [45]: #---> iat(row position + column position)
salary_iat = df.iat[2, 3]
print(salary_iat)
```

40000

In [47]: df

Out[47]:

	Name	Age	Gender	Salary
0	ratna	20	Female	50000
1	smruti	20	Female	55000
2	Bob	22	Male	40000
3	Eve	35	Female	70000
4	Charlie	28	Male	48000

In [59]: *# row index*
df.loc[0]

Out[59]: Name ratna
Age 20
Gender Female
Salary 50000
Name: 0, dtype: object

In [57]: *# column*
df.loc[0, ['Name', 'Salary']]

Out[57]: Name ratna
Salary 50000
Name: 0, dtype: object

In [55]: *#iloc → position-based indexing*
Select row by position
df.iloc[0]

Out[55]: Name ratna
Age 20
Gender Female
Salary 50000
Name: 0, dtype: object

In [63]: *# specif column*
df.iloc[0, [0, 3]]

Out[63]: Name ratna
Salary 50000
Name: 0, dtype: object

In [75]: *#Select row & column slice*
df.iloc[1:4, 1:3]

Out[75]:

	Age	Gender
1	20	Female
2	22	Male
3	35	Female

In []: