PANDAS

~~Python is~~

- Pandas is an open source library in Python. It provides excel file which create dataframe
- Dataframe is the key data structure in Pandas. It allows us to store and manipulate tabular data as a 2-D data structure.
- It represent keys & values form

Data frame

- Data frame is a main object in Pandas.
- It is used to represent data with rows & columns.
- Tabular or Excel spreadsheet like data.

Different ways to create a dataframe.

1. Through reading csv file.
2. " " excel "
3. " python Dictionary
4. " " tuple list
5. " " dictionary list

Create Dataframe using python dictionary

```
import pandas as pd
emp = {'id':[101,102,103,104,105],
       'Name':['priya','riya','neha','anjali','shivangi'],
       'Age' :[25,22,24,20,27],
       'Salary':[25000,30000,150000,10000,30000]}
df = pd.Dataframe(emp)
```

df

Output

	id	Name	Age	Salary
0	101	priya	25	25000
1	102	riya	22	30000
2	103	neha	24	150000
3	104	Shivangi	20	10000
4	105	anjali	27	30000

type(df)

pandas.core.frame.DataFrame.

pandas → Library, core → Module, frame → class, DataFrame → method

Create Dataframe using python ~~dictionary~~ tuple list

Using Dataframe() method

```
emp-data = [(101, 'priya', 23, 30000),
            (102, 'riya', 22, 30000),
            (103, 'amit', 25, 120000),
            (104, 'shivi', 24, 20,500)]
df = pd.DataFrame(emp-data, columns = ['id', 'name',
                                       'age', 'salary'])
```

- Each and every element of the tuple list is a actually a row in your dataframe.
- You should mention all column name as a list as a second argument.

Create Dataframe using a list of dictionaries

```
emp_data_list = [{'id': 101, 'name': 'priya', 'age': 24,
                  'salary': 200000},
                 {'id': 102, 'name': 'riya', 'age': 22,
                  'salary': 120000},
                 {'id': 103, 'name': 'amit', 'age': 25,
                  'salary': 70000},
                 {'id': 104, 'name': 'niju', 'age': 24,
                  'salary': 300000}]
df = pd.DataFrame(emp_data_list)
df
```

Create Dataframe using CSV (Comma Separated Value)

We use read_csv()

```
import pandas as pd
df = pd.read(r'path\filename.csv')
print(df)
```

read_csv() method accepts the path of the csv file, if that exist in the home folder or current working directory

os.getcwd() → to find current directory.

1)
```
import os
print(os.getcwd())
```

O/P

C:\Users\USER\Desktop\Priya\Untitled Folder 1

2)
```
df1 = pd.read_csv("stock_data.csv")
df1
```

O/P

File

Pandas Series

↳ A Pandas Series is like a column in a table
It is a one-dimensional array holding data of any type.

↳ It is used in Time series data (e.g.- stock temprature, humidity, stock data)

```
import pandas as pd
a = [1,7,2]
myvar = pd.Series(a)
print(myvar)
```

Output

index no	time series
0	1
1	7
2	2

dtype : int64

Labels

If nothing else is specified, the value are labeled with their index number. First value has index 0, second value has index 1 etc.

The label can be used to access a specified varies.

```
import pandas as pd
a = [1,7,2]
myvar = pd.Series(a)
print(myvar[0])
```

O/P
1

Create Labels

With the index argument, you can name your own labels.

1)
```
a = [1, 7, 2]
myvar = pd.Series(a, index = ["x", "y", "z"])
print(myvar)
```

Output

```
x  1
y  7
z  2
dtype: int64
```

2) or
```
a = [1, 7, 2]
myvar = pd.Series(a, index = ["x", "y", "z"])
print(myvar["y"])
```

O/P

```
7
```

Key / Value Objects as Series

You can also use a key/value object, like a dictionary, when creating a series

e.g.) Create a Simple Pandas Series from a dictionary

```
calories = {"day1": 420, "day2": 380, "day3": 390}
myvar = pd.Series(calories)
print(myvar)
```

O/P

```
day1  420
day2  380
day3  390
dtype int64
```

e.g2) calories = {"day1": 420, "day2": 380, "day3": 390}

myvar = pd.Series(calories, index = ["day1", "day2"])

print(myvar)

O/P

day1 420
day2 380
dtype : int64