

## LAB – 2: GETTING STARTED WITH PANDAS

### PANDAS:

Pandas is a Python software library for data analysis and manipulation widely used in data science and machine learning applications enabling cleansing, merging, and reshaping of data structures for big datasets, handling various file formats including Excel, JSON, Parquet, SQL database tables or queries, and CSV files. **Installation: `pip install pandas`**

Here is a brief description and syntax for the Pandas methods:

1. **Series()**: This method creates a new Series with the specified index and data.  
Syntax: `pd.Series(data, index=['a', 'b', 'c'])`
2. **DataFrame()**: This method creates a new DataFrame with the specified data, where each row is a pandas Series.  
Syntax: `pd.DataFrame(data)`
3. **read\_csv()**: This method reads a CSV file into a pandas DataFrame.  
Syntax: `pd.read_csv('path/to/file.csv')`
4. **loc()**: This method is used to select rows and columns of the DataFrame based on labels or index values.  
Syntax: `df.loc[label1, 'column1']`
5. **head()**: This method returns the first N rows of the DataFrame by default, or the number specified as an argument.  
Syntax: `df.head(n=5)`
6. **tail()**: This method returns the last N rows of the DataFrame by default, or the number specified as an argument.  
Syntax: `df.tail(n=5)`
7. **sample()**: This method returns a random sample of the DataFrame of the specified size or type.  
Syntax: `df.sample(n=5)`
8. **info()**: This method provides a summary of the DataFrame, including the index type, column types, and memory usage.  
Syntax: `df.info()`

Lab program with all the above mentioned methods is given below:

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

```
In [2]: print(pd.__version__)
```

1.5.3

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

```
In [3]: series_dataset = [23, 98, 32, 43, 77]
series_df = pd.Series(series_dataset)
print(series_df)
```

```
0    23
1    98
2    32
3    43
4    77
dtype: int64
```

Custom Labels

```
In [4]: series_dataset = [23, 98, 32, 43, 77]
series_custom_label_df = pd.Series(series_dataset, index=["roll", "a", "b", "c", "d"])
print(series_custom_label_df)
```

```
roll    23
a        98
b        32
c        43
d        77
dtype: int64
```

From Json like key/value object pairs

```
In [5]: color_dataset = {"red": "FF0000", "green": "008000", "blue": "0000FF"}
color_df = pd.Series(color_dataset)
print(color_df)
```

```
red      FF0000
green    008000
blue     0000FF
dtype: object
```

A Pandas DataFrame is a 2 dimensional data structure, like a 2 dimensional array, or a table with rows and columns.

```
In [6]: country_dataset = {
    'country': ["USA", "UK", "Nepal"],
    'population': [331.9, 67.33, 30.03]
}
country_dataset
```

```
Out[6]: {'country': ['USA', 'UK', 'Nepal'], 'population': [331.9, 67.33, 30.03]}
```

```
In [7]: country_df = pd.DataFrame(country_dataset)
print(country_df)
```

```
country  population
0      USA      331.90
```

```
1      UK      67.33
2      Nepal    30.03
```

```
In [8]: print(country_df.shape)
```

```
(3, 2)
```

```
In [9]: print(country_df.loc[2])
```

```
country      Nepal
population    30.03
Name: 2, dtype: object
```

```
In [10]: print(country_df.loc[[1,2]])
```

```
country  population
1      UK      67.33
2      Nepal    30.03
```

```
In [11]: country_custom_label_df = pd.DataFrame(country_dataset, index = ["C1", "C2", "C3"])
print(country_custom_label_df)
```

```
country  population
C1      USA    331.90
C2      UK      67.33
C3      Nepal   30.03
```

```
In [12]: print(country_custom_label_df.loc["C2"])
```

```
country      UK
population    67.33
Name: C2, dtype: object
```

```
In [13]: cancer_df = pd.read_csv('https://raw.githubusercontent.com/becauseiliketolike/Software/master/cancer.csv')
```

```
In [14]: cancer_df.shape
```

```
Out[14]: (569, 33)
```

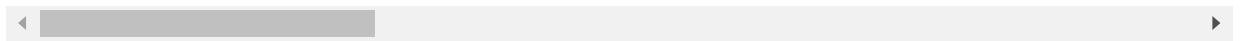
```
In [15]: cancer_df
```

```
Out[15]:
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothn
--	----	-----------	-------------	--------------	----------------	-----------	---------

0	842302	M	17.99	10.38	122.80	1001.0	
1	842517	M	20.57	17.77	132.90	1326.0	
2	84300903	M	19.69	21.25	130.00	1203.0	
3	84348301	M	11.42	20.38	77.58	386.1	
4	84358402	M	20.29	14.34	135.10	1297.0	
...	...	...	...	...	...	...	
564	926424	M	21.56	22.39	142.00	1479.0	
565	926682	M	20.13	28.25	131.20	1261.0	
566	926954	M	16.60	28.08	108.30	858.1	
567	927241	M	20.60	29.33	140.10	1265.0	
568	92751	B	7.76	24.54	47.92	181.0	

569 rows × 33 columns

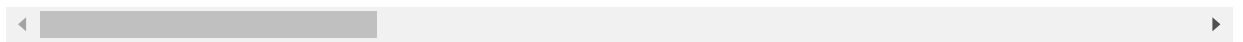


In [16]: `cancer_df.head()`

Out[16]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness
0	842302	M	17.99	10.38	122.80	1001.0	(
1	842517	M	20.57	17.77	132.90	1326.0	(
2	84300903	M	19.69	21.25	130.00	1203.0	(
3	84348301	M	11.42	20.38	77.58	386.1	(
4	84358402	M	20.29	14.34	135.10	1297.0	(

5 rows × 33 columns

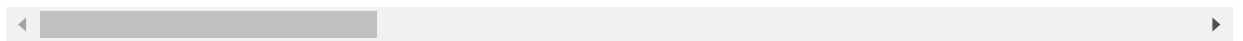


In [17]: `cancer_df.loc[[20,21,22,50]]`

Out[17]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness
20	8510653	B	13.080	15.71	85.63	520.0	(
21	8510824	B	9.504	12.44	60.34	273.9	(
22	8511133	M	15.340	14.26	102.50	704.4	(
50	857343	B	11.760	21.60	74.72	427.9	(

4 rows × 33 columns



In [18]: `cancer_df.tail(10)`

Out[18]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness
559	925291	B	11.51	23.93	74.52	403.5	(
560	925292	B	14.05	27.15	91.38	600.4	(
561	925311	B	11.20	29.37	70.67	386.0	(
562	925622	M	15.22	30.62	103.40	716.9	(
563	926125	M	20.92	25.09	143.00	1347.0	(
564	926424	M	21.56	22.39	142.00	1479.0	(
565	926682	M	20.13	28.25	131.20	1261.0	(
566	926954	M	16.60	28.08	108.30	858.1	(
567	927241	M	20.60	29.33	140.10	1265.0	(
568	92751	B	7.76	24.54	47.92	181.0	(

10 rows × 33 columns

```
In [19]: cancer_df.sample(3)
```

```
Out[19]:
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness
127	866203	M	19.00	18.91	123.40	1138.0	(
94	862028	M	15.06	19.83	100.30	705.6	(
54	857438	M	15.10	22.02	97.26	712.8	(

3 rows × 33 columns

```
In [20]: cancer_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 33 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   id                                     569 non-null    int64
1   diagnosis                             569 non-null    object
2   radius_mean                           569 non-null    float64
3   texture_mean                           569 non-null    float64
4   perimeter_mean                         569 non-null    float64
5   area_mean                             569 non-null    float64
6   smoothness_mean                       569 non-null    float64
7   compactness_mean                      569 non-null    float64
8   concavity_mean                        569 non-null    float64
9   concave points_mean                   569 non-null    float64
10  symmetry_mean                         569 non-null    float64
11  fractal_dimension_mean                 569 non-null    float64
12  radius_se                             569 non-null    float64
13  texture_se                             569 non-null    float64
14  perimeter_se                           569 non-null    float64
15  area_se                               569 non-null    float64
16  smoothness_se                         569 non-null    float64
17  compactness_se                        569 non-null    float64
18  concavity_se                          569 non-null    float64
19  concave points_se                     569 non-null    float64
20  symmetry_se                           569 non-null    float64
21  fractal_dimension_se                   569 non-null    float64
22  radius_worst                          569 non-null    float64
23  texture_worst                         569 non-null    float64
24  perimeter_worst                       569 non-null    float64
25  area_worst                            569 non-null    float64
26  smoothness_worst                      569 non-null    float64
27  compactness_worst                     569 non-null    float64
28  concavity_worst                       569 non-null    float64
29  concave points_worst                   569 non-null    float64
30  symmetry_worst                        569 non-null    float64
31  fractal_dimension_worst                569 non-null    float64
32  Unnamed: 32                           0 non-null      float64
dtypes: float64(31), int64(1), object(1)
```