# Data Analytics
# Lab # 7
# Data Analysis using python-pandas

Pandas is a popular Python library used for working in tabular data (similar to the data stored in a spreadsheet). Pandas provides helper functions to read data from various file formats like CSV, Excel spreadsheets, HTML tables, JSON, SQL, and more.

Considering an example of day-wise Covid-19 data for Italy in the tabular form as follows,
**date,new_cases,new_deaths,new_tests**
2020-04-21,2256.0,454.0,28095.0
2020-04-22,2729.0,534.0,44248.0
2020-04-23,3370.0,437.0,37083.0
2020-04-24,2646.0,464.0,95273.0
2020-04-25,3021.0,420.0,38676.0
2020-04-26,2357.0,415.0,24113.0
2020-04-27,2324.0,260.0,26678.0
2020-04-28,1739.0,333.0,37554.0
This format of storing data is known as comma-separated values or CSV.
We can now import the pandas module. As a convention, it is imported with the alias pd.

```
: import pandas as pd
```

```
: covid_df = pd.read_csv('italy-covid-daywise.csv')
```

```
: type(covid_df)
```

```
: pandas.core.frame.DataFrame
```

Data from the file is read and stored in a DataFrame object - one of the core data structures in Pandas for storing and working with tabular data. We typically use the _df suffix in the variable names for dataframes.

```
: covid_df
```

:

| | date | new_cases | new_deaths | new_tests |
|---|---|---|---|---|
| **0** | 2019-12-31 | 0 | 0 | NaN |
| **1** | 2020-01-01 | 0 | 0 | NaN |
| **2** | 2020-01-02 | 0 | 0 | NaN |
| **3** | 2020-01-03 | 0 | 0 | NaN |
| **4** | 2020-01-04 | 0 | 0 | NaN |
| **...** | ... | ... | ... | ... |
| **243** | 2020-08-30 | 1444 | 1 | 53541.0 |
| **244** | 2020-08-31 | 1365 | 4 | 42583.0 |
| **245** | 2020-09-01 | 996 | 6 | 54395.0 |
| **246** | 2020-09-02 | 975 | 8 | NaN |
| **247** | 2020-09-03 | 1326 | 6 | NaN |

248 rows × 4 columns

Here's what we can tell by looking at the dataframe:

- The file provides four day-wise counts for COVID-19 in Italy
- The metrics reported are new cases, deaths, and tests
- Data is provided for 248 days: from Dec 12, 2019, to Sep 3, 2020

Keep in mind that these are officially reported numbers. The actual number of cases & deaths may be higher, as not all cases are diagnosed.

We can view some basic information about the data frame using the .info method.

```
covid_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 248 entries, 0 to 247
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   date        248 non-null    object 
 1   new_cases   248 non-null    int64  
 2   new_deaths  248 non-null    int64  
 3   new_tests   135 non-null    float64
dtypes: float64(1), int64(2), object(1)
memory usage: 7.9+ KB
```

It appears that each column contains values of a specific data type. You can view statistical information for numerical columns (mean, standard deviation, minimum/maximum values, and the number of non-empty values) using the .describe method.

```
covid_df.describe()
```

| | new_cases | new_deaths | new_tests |
|---|---|---|---|
| count | 248.000000 | 248.000000 | 135.000000 |
| mean | 1094.818548 | 143.133065 | 31699.674074 |
| std | 1554.508002 | 227.105538 | 11622.209757 |
| min | -148.000000 | -31.000000 | 7841.000000 |
| 25% | 123.000000 | 3.000000 | 25259.000000 |
| 50% | 342.000000 | 17.000000 | 29545.000000 |
| 75% | 1371.750000 | 175.250000 | 37711.000000 |
| max | 6557.000000 | 971.000000 | 95273.000000 |

```
covid_df.columns
```

```
Index(['date', 'new_cases', 'new_deaths', 'new_tests'], dtype='object')
```

```
covid_df.shape
```

```
(248, 4)
```

- pd.read_csv - Read data from a CSV file into a Pandas DataFrame object
- .info() - View basic infomation about rows, columns & data types
- .describe() - View statistical information about numeric columns
- .columns - Get the list of column names
- .shape - Get the number of rows & columns as a tuple

## Tasks:

Find the total number of reported cases and deaths related to Covid-19 in Italy.
Find the overall death rate (ratio of reported deaths to reported cases).
Find the overall number of tests conducted? A total of 935310 tests were conducted before daily test numbers were reported.
Find the positive rate i.e. fraction of tests returned a positive result.