**The flying Birds**

**Assignment**

**Of**

**Pandas**

**Introduction:-**

**Pandas** is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the [Python](https://www.python.org/) programming language.

**What is Pandas?**

Pandas is a Python library used for working with data sets.

It has functions for analyzing, cleaning, exploring, and manipulating data.

The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.

## Why Use Pandas?

Pandas allows us to analyze big data and make conclusions based on statistical theories.

Pandas can clean messy data sets, and make them readable and relevant.

Relevant data is very important in data science.

Pandas makes it simple to do many of the time consuming, repetitive tasks associated with working with data, including:

* Data cleansing
* Data fill
* Data normalization
* Merges and joins
* Data visualization
* Statistical analysis
* Data inspection
* Loading and saving data
* And much more

In fact, with Pandas, you can do everything that makes world-leading data scientists vote Pandas as the best data analysis and manipulation tool available.

**Installation of Pandas**

If you have [Python](https://www.w3schools.com/python/default.asp) and [PIP](https://www.w3schools.com/python/pandas/python_pip.asp) already installed on a system, then installation of Pandas is very easy.

Install it using this command:

C:\Users\*Your Name*>pip install pandas

If this command fails, then use a python distribution that already has Pandas installed like, Anaconda, and Spyder etc.

**Import Pandas**

Once Pandas is installed, import it in your applications by adding the **import** keyword

**Ex:-**

**C:\Users\Rajat verma\AppData\Local\Microsoft\Windows\INetCache\Content.Word\a1.png**

=>Now Pandas is imported and ready to use.

## Pandas as pd

Pandas is usually imported under the **pd**alias.

**alias:** In Python alias are an alternate name for referring to the same thing.

Create an alias with the **as**keyword while importing:

**Ex:-**

**C:\Users\Rajat verma\AppData\Local\Microsoft\Windows\INetCache\Content.Word\a2.png**

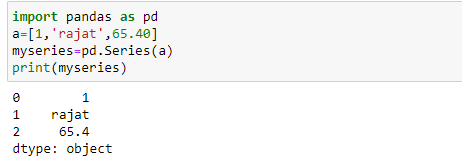
=>Now the Pandas package can be referred to as **pd** instead of **pandas.**

# **Pandas Series**

## What is a Series?

A Pandas Series is like a column in a table.

It is a one-dimensional array holding data of any type.



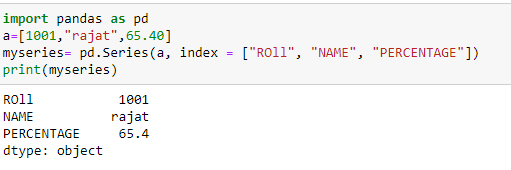
## Labels

If nothing else is specified, the values are labeled with their index number. First value has index 0, second value has index 1 etc. as you see in above Example.

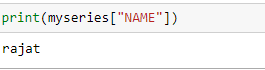
This label can be used to access a specified value.

## Create Labels

With the index argument, you can name your own labels

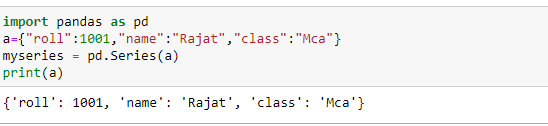


When you have created labels, you can access an item by referring to the label.



## Key/Value Objects as Series

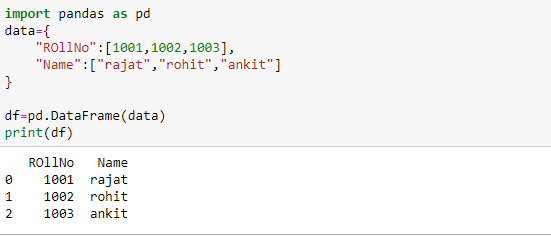
You can also use a key/value object, like a dictionary, when creating a Series.



## DataFrames

Data sets in Pandas are usually multi-dimensional tables, called DataFrames.

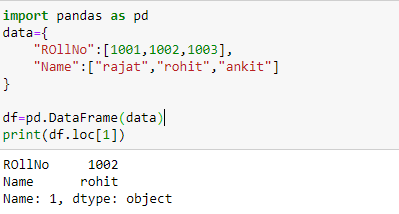
Series is like a column, a DataFrame is the whole table.



## Locate Row

As you can see from the result above, the DataFrame is like a table with rows and columns.

Pandas use the loc attribute to return one or more specified row(s)



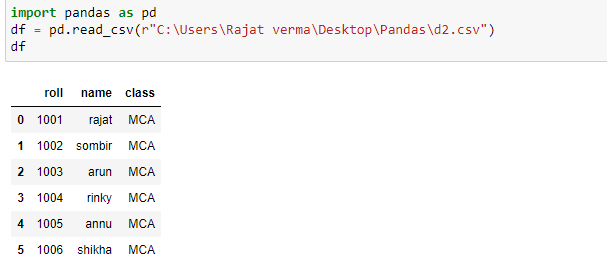
# **Pandas Read and writeCSV**

## Read CSV Files

A simple way to store big data sets is to use CSV files (comma separated files).

CSV files contains plain text and is a well know format that can be read by everyone including Pandas.

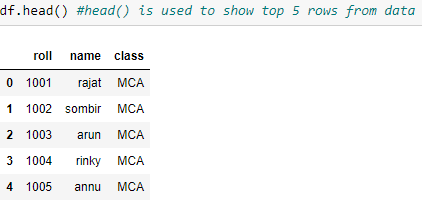
In our examples we will be using a CSV file called 'data.csv'.

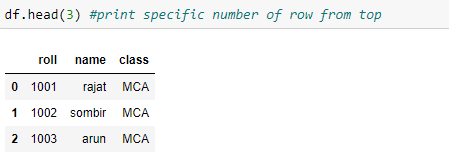


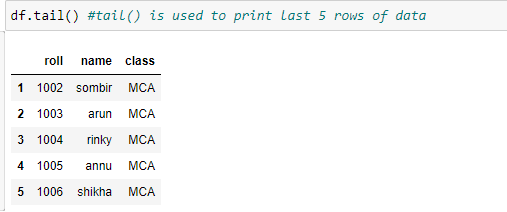
# C:\Users\Rajat verma\Desktop\Pandas\S1.PNG**Pandas - Analyzing DataFrames**

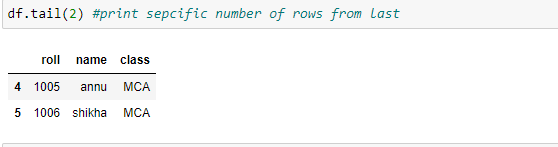
# **Head and tail**

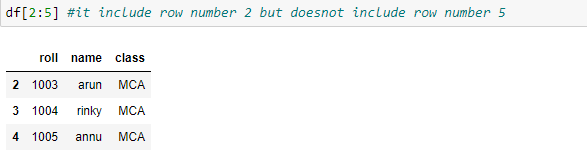
To view a small sample of a Series or DataFrame object, use the [head()](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.head.html#pandas.DataFrame.head) and [tail()](https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.tail.html#pandas.DataFrame.tail) methods. The default number of elements to display is five, but you may pass a custom number.

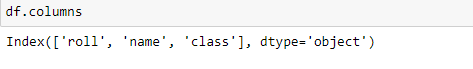


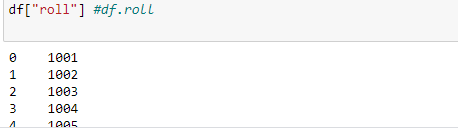


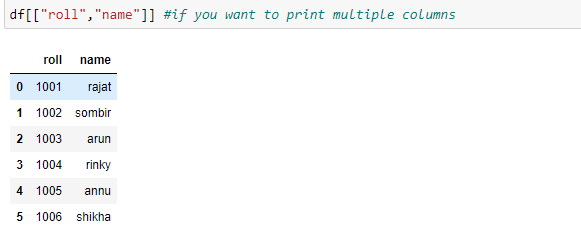












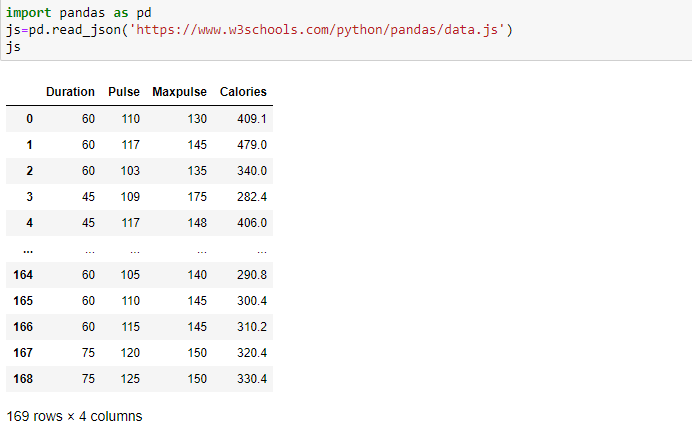
# **Pandas Read JSON**

## Read JSON

Big data sets are often stored, or extracted as JSON.

JSON is plain text, but has the format of an object, and is well known in the world of programming, including Pandas.

In our examples we will be using a JSON file called 'data.json'.



## Dictionary as JSON

JSON = Python Dictionary

JSON objects have the same format as Python dictionaries.

If your JSON code is not in a file, but in a Python Dictionary, you can load it into a DataFrame directly:

