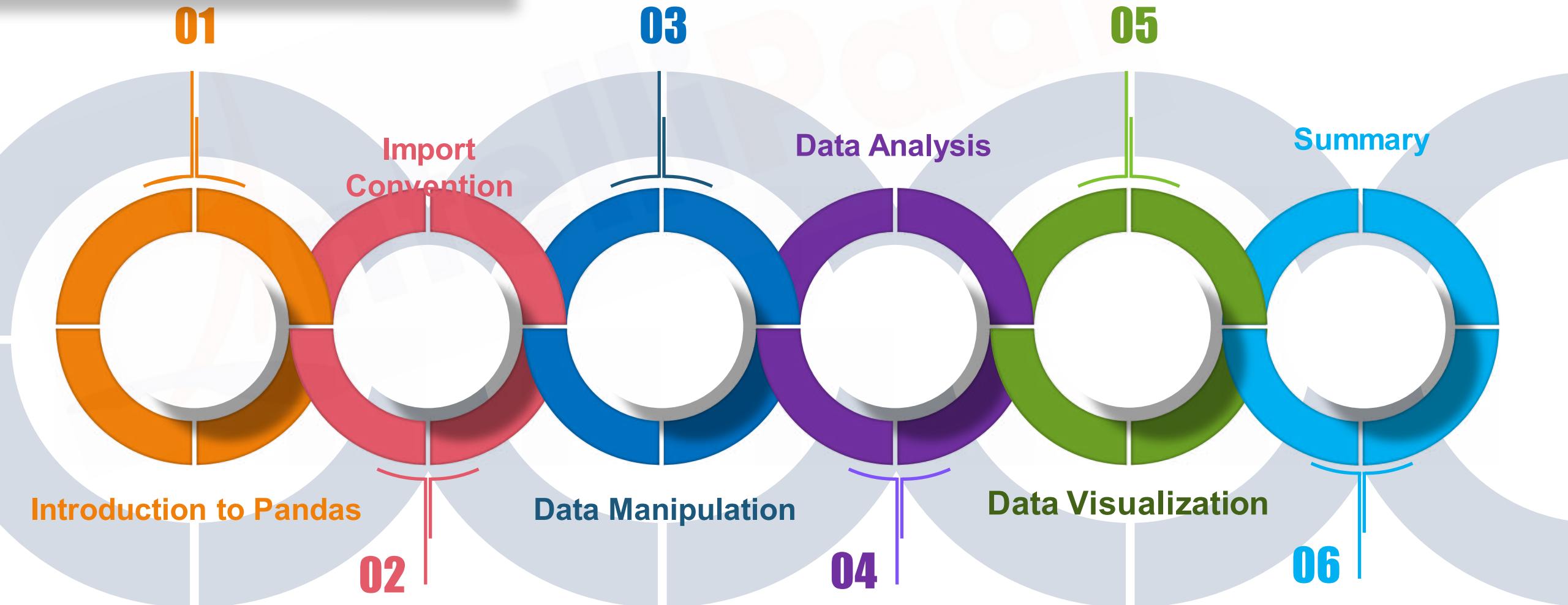
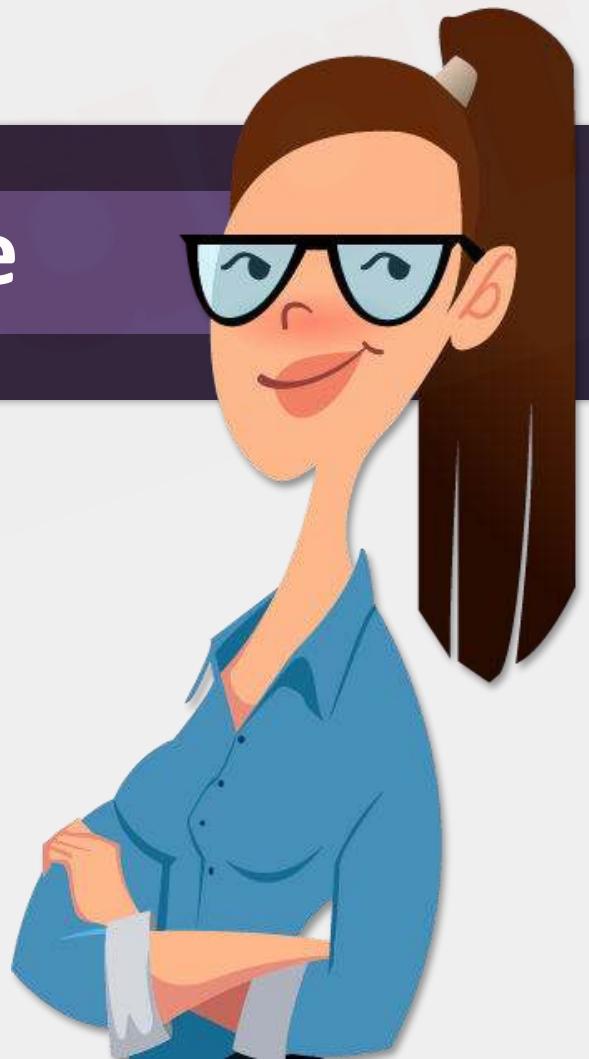


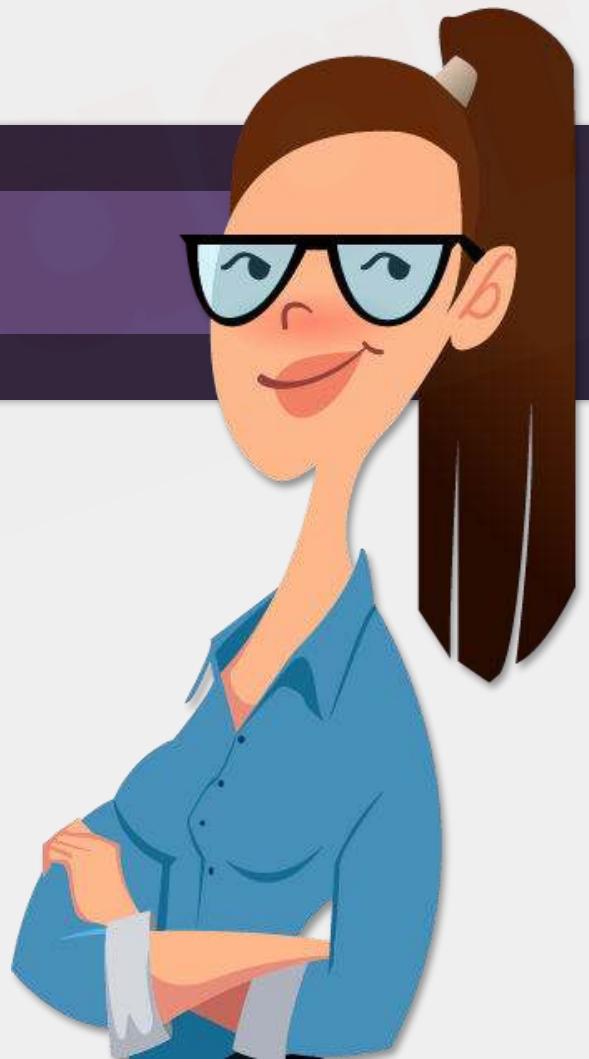
# Agenda for Today's Session



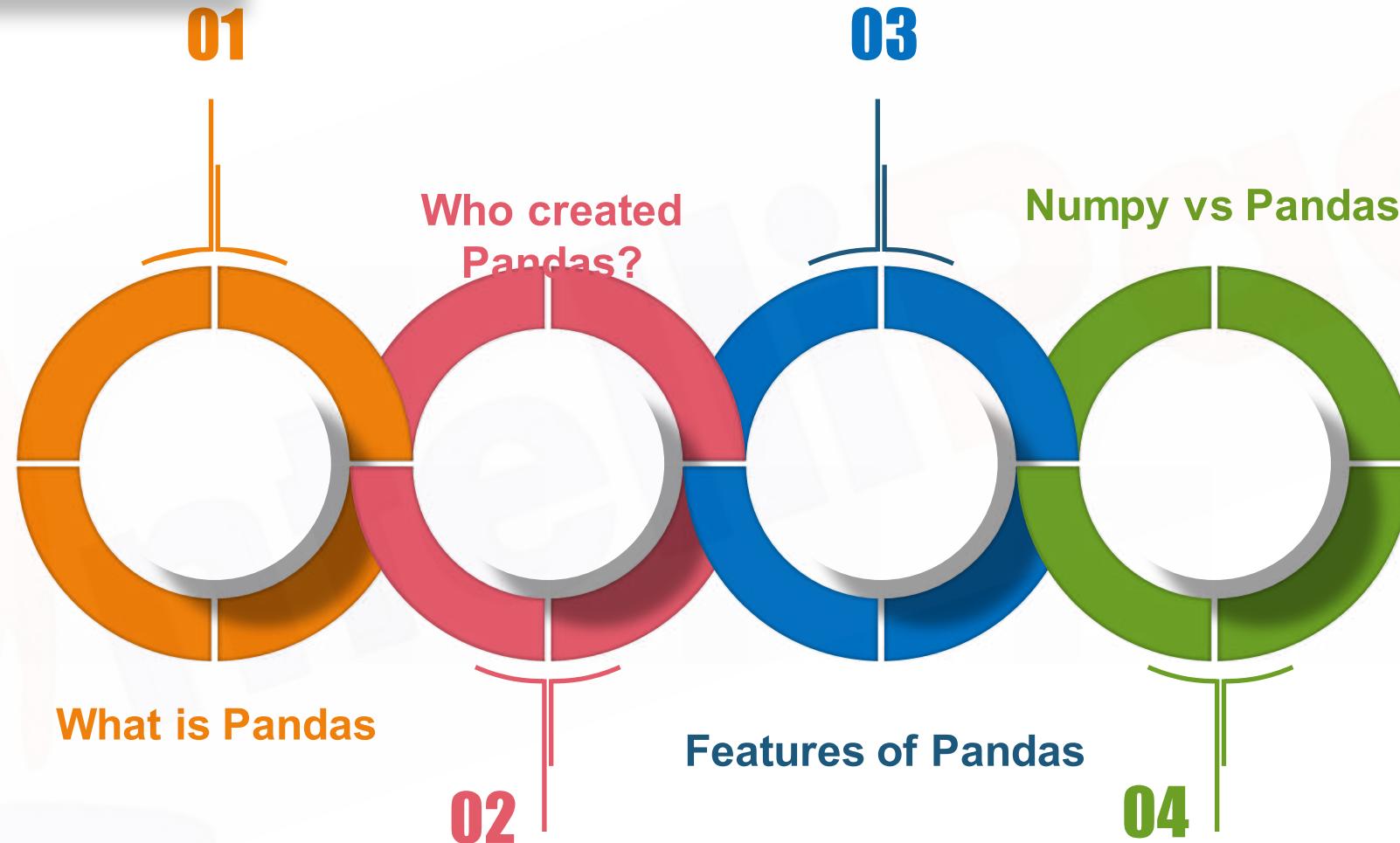
# Python Certification Course



# Introduction to Pandas



# Agenda for Today's Session



# Introduction to Pandas

## What is Pandas?

Open-source Python library

01

02

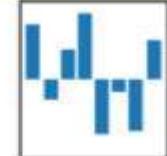
Simple yet powerful and expressive tool

03

Data Manipulation & Analysis

pandas

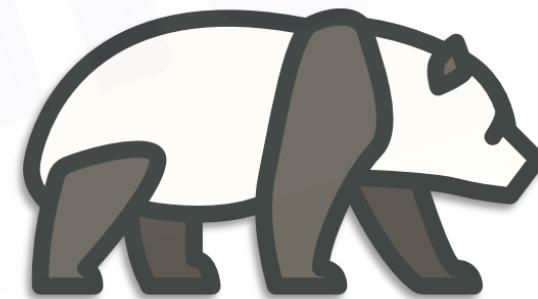
$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



# Introduction to Pandas

## Where did the name Pandas come from?

- The name Pandas is derived from the word Panel Data
- Panel Data is multi-dimensional data involving measurements over time



Person	Year	Income	Age	Sex
1	2013	20,000	23	F
1	2014	25,000	24	F
2	2013	35,000	27	M
2	2014	42,500	28	M
2	2015	50,000	29	M
3	2014	46,000	25	F

Panel Data

# Introduction to Pandas

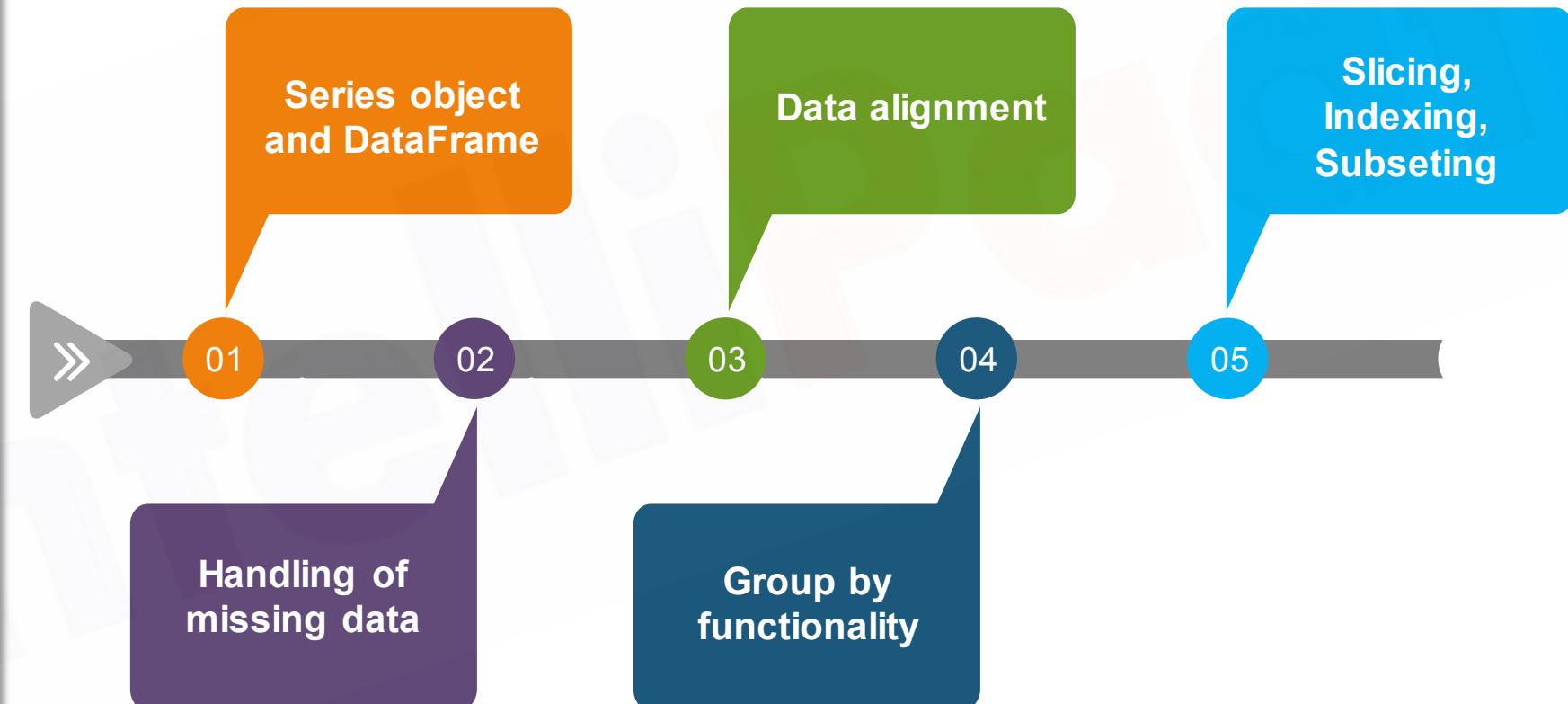
## Who created Pandas?



Created in 2015 by Wes McKinney

# Introduction to Pandas

## Features of Pandas:



# Introduction to Pandas

## Features of Pandas:

Merging and joining

Hierarchical labeling of axes

Time series-specific functionality

06

07

08

09

10

Reshaping

Robust Input Output tool

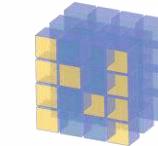
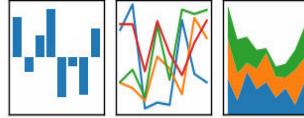


# Introduction to Pandas

## Pandas vs Numpy

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



NumPy

Pandas performs better than numpy for 500k rows or more.

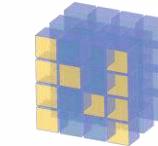
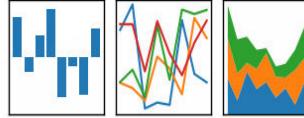
Numpy performs better for 50k rows or less.

# Introduction to Pandas

## Pandas vs Numpy

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



NumPy

Pandas Series Object is more flexible as you can define your own labeled index to index and access elements of an array

Elements in NumPy arrays are accessed by their default integer position

# Thank You

[www.intellipaat.com](http://www.intellipaat.com)



 **CALL US NOW**

India : +91-7847955955

US : 1-800-216-8930 (TOLL FREE)

[sales@intellipaat.com](mailto:sales@intellipaat.com)

A cartoon illustration of a person with brown hair and blue sunglasses, wearing a blue shirt, peeking over a dark horizontal bar from behind. The bar contains the title text.

# How to import Pandas in Python?

# Working with Pandas



## How to import Pandas in Python?

```
import pandas as pd
```

# Working with Pandas

## What kind of data does suit Pandas the most?

CUSTOMER		
NAME	DATATYPE	NULLABLE?
CUSTOMER_ID	VARCHAR	NO
FIRST_NAME	VARCHAR	NO
LAST_NAME	VARCHAR	NO
BIRTH_DAY	TIMESTAMP	NO
ADDRESS	VARCHAR	NO
ADDRESS2	VARCHAR	YES
STATE	VARCHAR	NO
ZIP_CODE	INTEGER	NO

Tabular data

	A	B	C	D
0	A0	B0	C0	D0
1	A1	B1	C1	D1
2	A2	B2	C2	D2
3	A3	B3	C3	D3
4	X0	X1	X2	X3

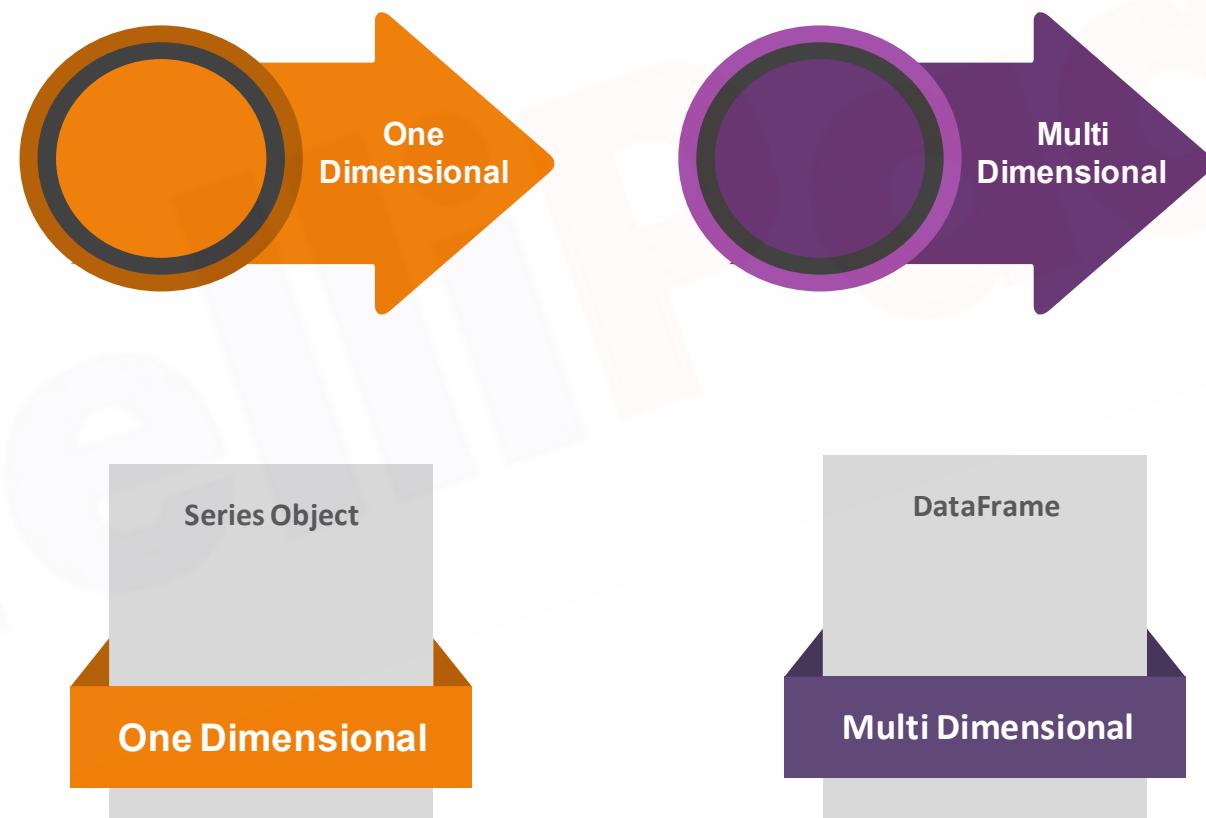
Arbitrary Matrix

	date	data
0	2018-01-01 00:00:00	52
1	2018-01-01 01:00:00	69
2	2018-01-01 02:00:00	23
3	2018-01-01 03:00:00	89
4	2018-01-01 04:00:00	53
5	2018-01-01 05:00:00	95
6	2018-01-01 06:00:00	19
7	2018-01-01 07:00:00	79
8	2018-01-01 08:00:00	33
9	2018-01-01 09:00:00	2
10	2018-01-01 10:00:00	0
11	2018-01-01 11:00:00	44
12	2018-01-01 12:00:00	45
13	2018-01-01 13:00:00	16
14	2018-01-01 14:00:00	38

Time Series Data

# Working with Pandas

## Data-set in Pandas



# Working with Pandas

## What is a series object?

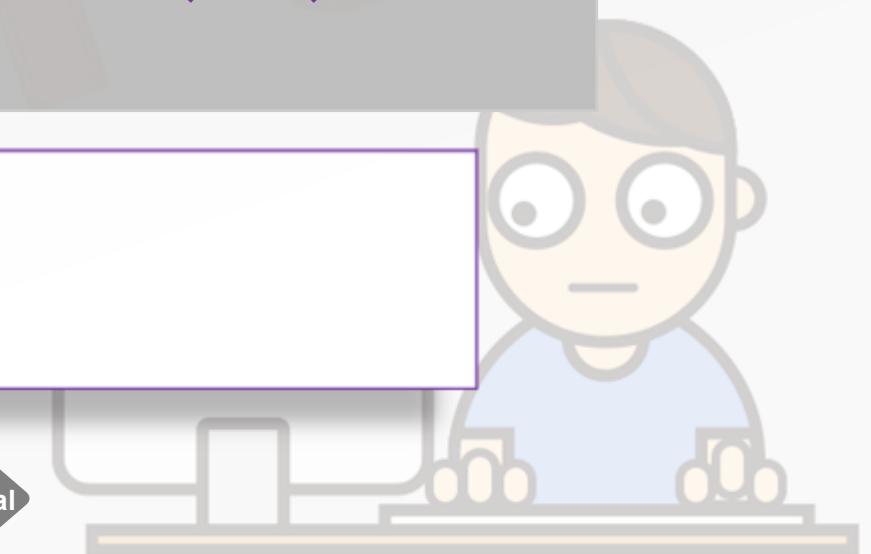
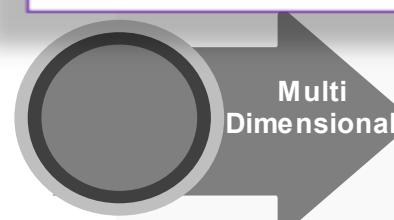
Series Object

One Dimensional

- One-dimensional labeled array
- Contains data of similar or mixed types
- Example:

```
data= [1, 2, 3, 4]  
series1 = pd.Series(data)  
series1
```

```
Out[4]: 0    1  
        1    2  
        2    3  
        3    4  
       dtype: int64
```



# Working with Pandas



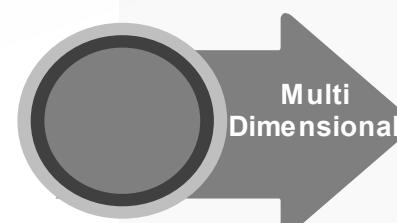
## How to check the type?

Series Object

One Dimensional

```
type(series1)
```

```
Out[13]: pandas.core.series.Series
```

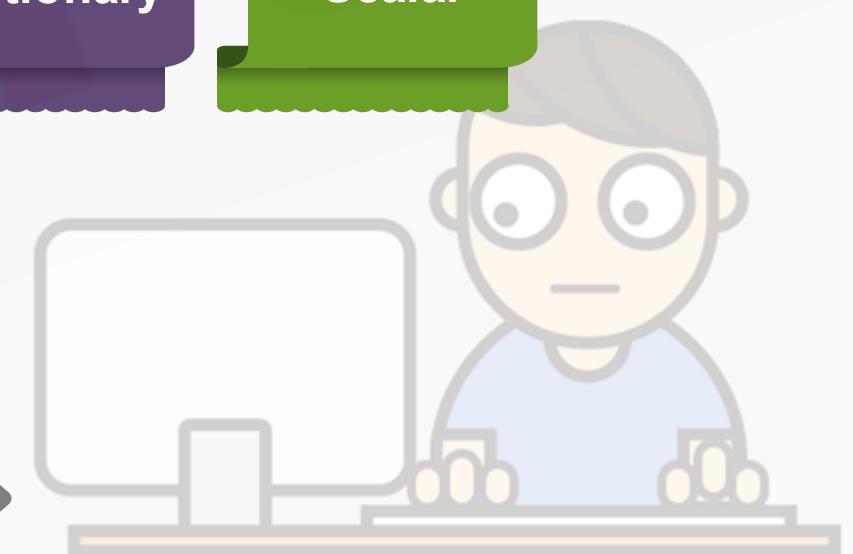


# Working with Pandas

## Create different Series Object datatypes

Series Object

One Dimensional



# Introduction to Pandas

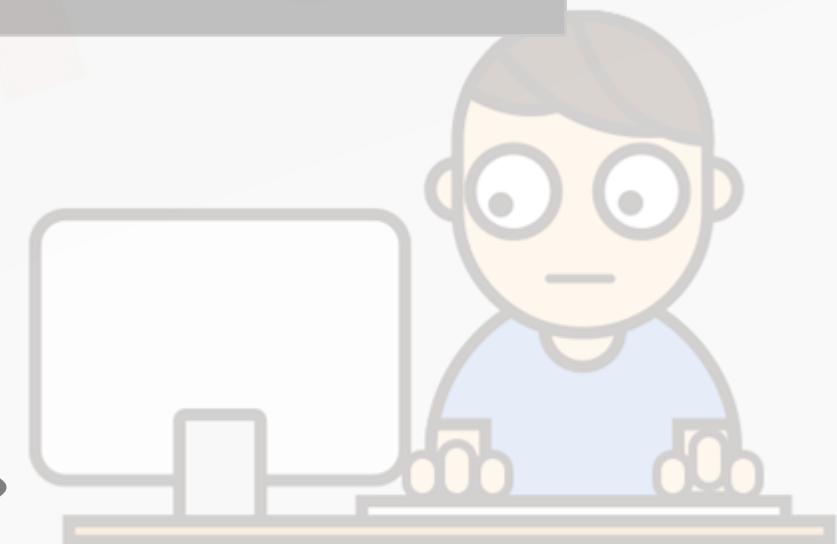


## How to create a series object?

Series Object

One Dimensional

`pd.Series(data)`



# Introduction to Pandas

## How to change the index name?

Series Object

One Dimensional

```
Out[4]: 0    1  
        1    2  
        2    3  
        3    4  
       dtype: int64
```

a  
b  
c  
d



# Introduction to Pandas

## How to change the index name?

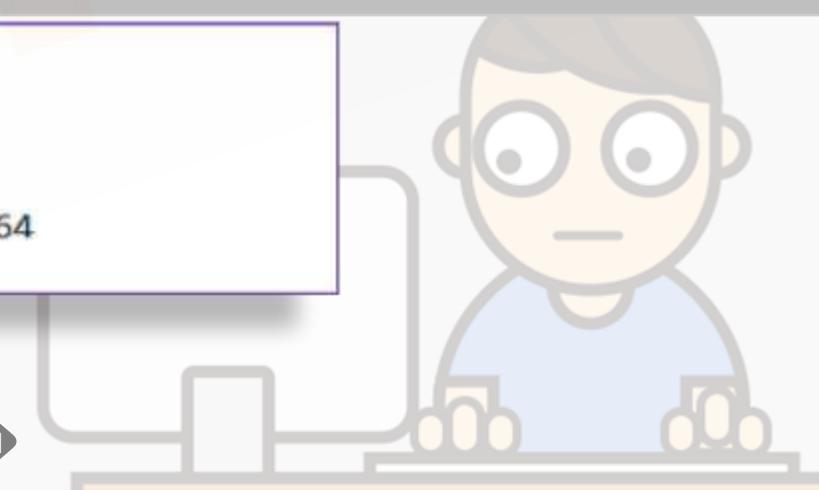
```
out[4]: 0 1  
        1 2  
        2 3  
        3 4  
      dtype: int64
```

Series Object

One Dimensional

```
series1 = pd.Series([1, 2, 3, 4]index=['a', 'b', 'c', 'd'])  
  
series1
```

```
out[12]: a 1  
         b 2  
         c 3  
         d 4  
      dtype: int64
```



# Introduction to Pandas

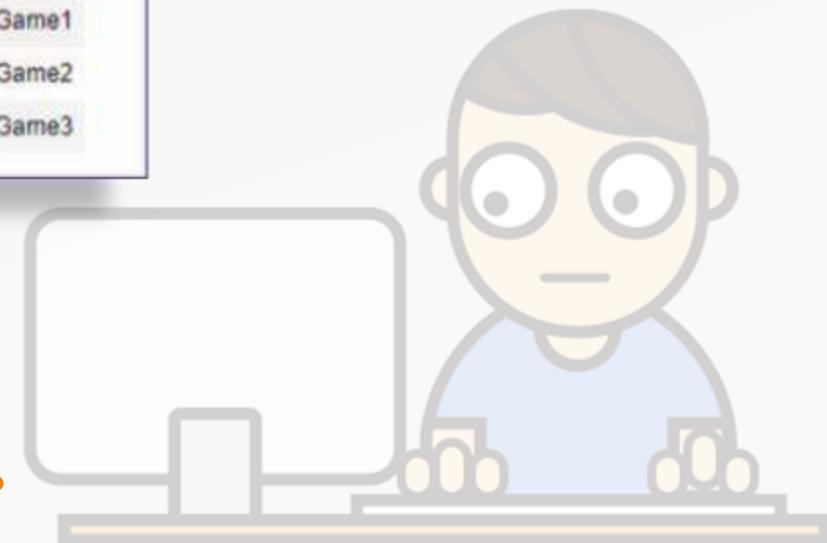
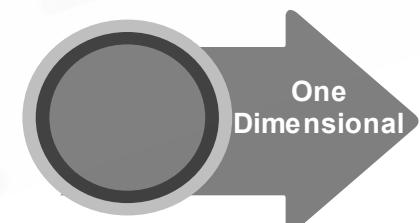
## What is a DataFrame?

DataFrame

Multi Dimensional

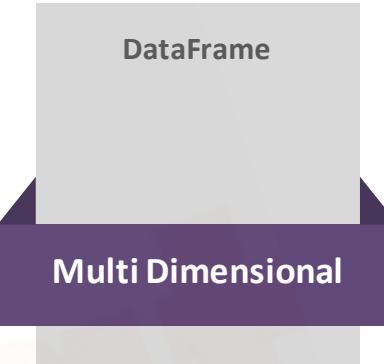
- Two-dimensional labeled data structures with columns of potentially different types
- Example:

	Player	Points	Title
0	Player1	8	Game1
1	Player2	9	Game2
2	Player3	5	Game3

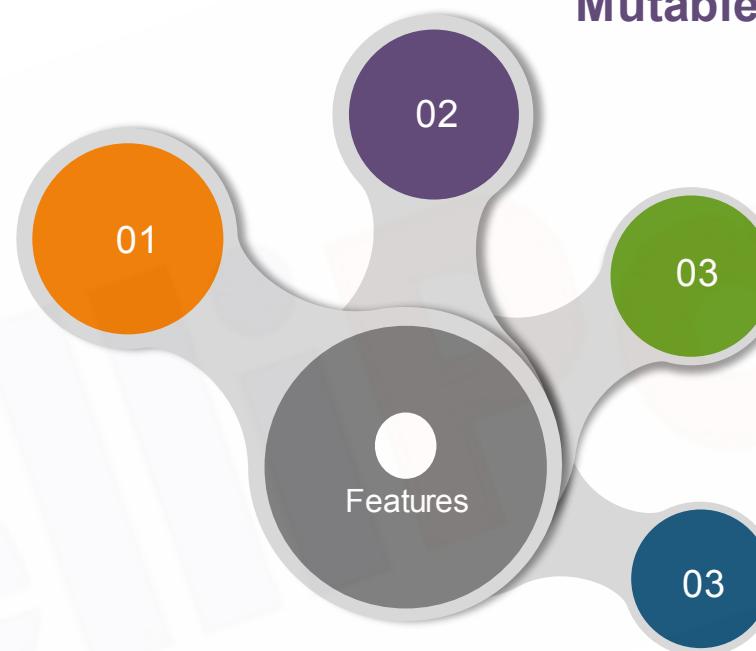


# Introduction to Pandas

## Features of DataFrame



Different  
Column  
types



Mutable Size

Labeled axes

Arithmetic  
operations on  
rows and  
columns



# Introduction to Pandas

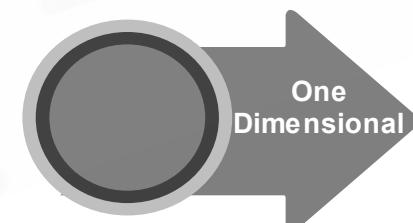


## How to create a DataFrame?

DataFrame

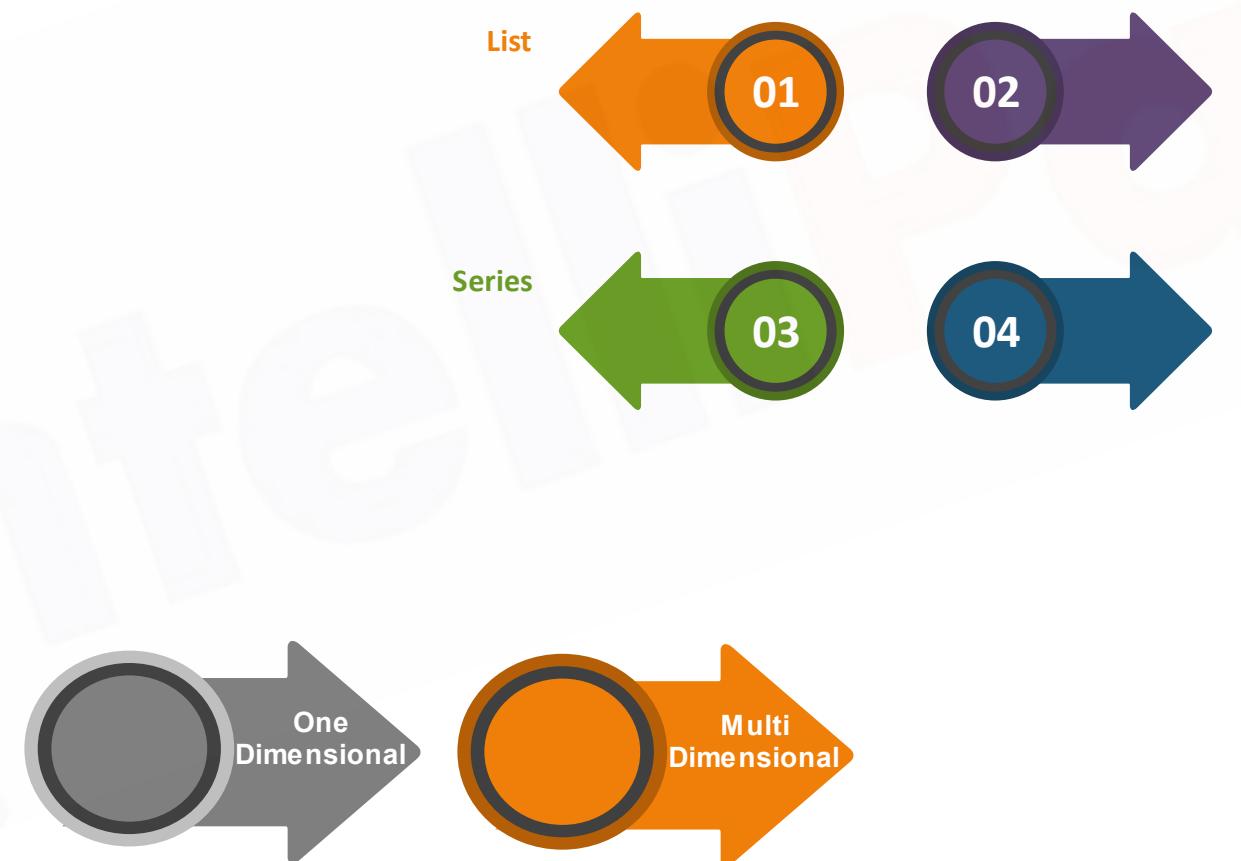
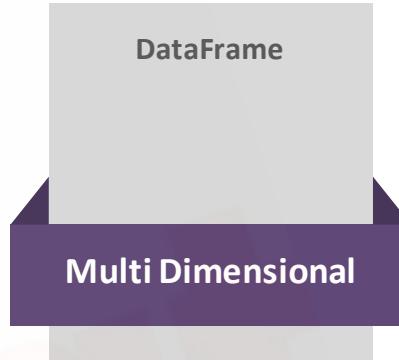
Multi Dimensional

```
pd.DataFrame(data)
```



# Introduction to Pandas

## How to create a DataFrame?



# Introduction to Pandas

## Create a DataFrame from a List

DataFrame

Multi Dimensional

```
data = [1,2,3,4,5]  
df = pd.DataFrame(data)  
  
df
```

Out[2]:	0
	0 1
	1 2
	2 3
	3 4
	4 5



# Introduction to Pandas

## Create a DataFrame from a Dictionary

DataFrame

Multi Dimensional

```
dict1 = {'fruit':['apple', 'mango', 'banana'],'count':[10,12,13]}

df = pd.DataFrame(dict1)

df
```

Out[26]:

	fruit	count
0	apple	10
1	mango	12
2	banana	13



# Introduction to Pandas

## Create a DataFrame from a Series

DataFrame

Multi Dimensional

```
data = pd.Series([6,12], index=['a','b'])  
  
df = pd.DataFrame([data])  
  
df
```

Out[4]:

	a	b
0	6	12



# Introduction to Pandas

## Create a DataFrame from a numpy ND array

DataFrame

Multi Dimensional

```
import numpy as np  
  
data= np.array([['a','b'], [6,12]])  
  
df = pd.DataFrame({'A':data[:,0], 'B':data[:,1]})  
  
df
```

	fruit	count
0	apple	10
1	mango	12
2	banana	13



# Understanding Pandas Operations with example

# Hands-on Demonstration



# Exploring the Data-set



- Dataset is based on product reviews from Amazon
- Stored in the .csv format

```
1 ,Product
_Review_Phrase,Product_Title,Website_URL,Platform,Product_Rating,Product_Category,Is_Amazon_Advantage_product,Product_Launch_Year,Product_Launch_Month,Product_Launch_Day
2 ,Amazing - Highest Selling,Ladela Bellies,http://www.amazon.com/ladela-bellies/p/itneh4kmght7tuc?pid=SkOEWXQWZ6H6,Mobile App,5,Footwear,TRUE,2016,7,12
3 ,Good - Average Selling,Bulaky vanity case Jewellery Vanity Case,http://www.amazon.com/bulaky-vanity-case-jewellery/p/1tdzy4ycfjhvtj?pid=VWMDZYRY2PP668ST,Website,3,Beauty and Personal Care ,FALSE,2016,7,12
4 ,Good - Average Selling,Roadster Men's Zipper Solid Cardigan,http://www.amazon.com/roadster-men-s-zipper-solid-cardigan/p/Itmedfy7ueukSeFsf?pid=CONESFY77SG2T6QJ,Website,3,6,Clothing ,FALSE,2016,7,12
5 ,Amazing - Highest Selling,"Camerii WM64 Elegance Analog Watch - For Men, Boys",http://www.amazon.com/camerii-wm64-elegance-analog-watch-men-boys/p/itmelydduhfcumh?pid=AUTB9Y2D2M68T,Mobile App,5,Others,TRUE,2016,7,12
6 ,Amazing - Highest Selling,"Colat COLAT MD28 Sheen Analog Watch - For Men, Women, Boys, Girls",http://www.amazon.com/colat-colat-md28-sheen-analog-watch-men-women-boys-girls/p/itmew2rx9jpyncyc?pid=AUTE2RX9H6BUDQ6A,Mobile App,5,Others,TRUE,2016,7,12
7 ,Amazing - Highest Selling,"Norlig RR-R03 Expedition Analog Watch - For Men, Boys",http://www.amazon.com/norlig-rr-r03-expedition-analog-watch-men-boys/p/itmeybzggbrqgbnf?pid=AUTB9Y2GFC2PU1AR,Mobile App,5,Others,TRUE,2016,7,12
8 ,Amazing - Highest Selling,Iyc White Casual Boots,http://www.amazon.com/Iyc-white-casual-boots/p/itmew6szlvrchacq?pid=SHOE4UK3KQDF3VF3,Mobile App,5,Footwear,TRUE,2016,7,12
9 ,Good - Average Selling,Fluid DPF-GR01 Digital Watch - For Boys,http://www.amazon.com/Fluid-dpf-gr01-digital-watch-boys/p/itmec4cgdsuhejcx?pid=AUTB4CG2Q4Q46SF,Website,3,5,Others,TRUE,2016,7,12
10 ,Great - High Selling,Bruno Manetti Cannelita Boots,http://www.amazon.com/bruno-manetti-cannelita-boots/p/itmeygk5jzcbze?pid=SKCEYK5AY8VXXH,Both,4,Footwear,TRUE,2016,7,12
```

# Importing Data-set with Pandas

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



First read the data

```
import pandas as pd  
  
Product_Review=pd.read_csv("Amazon_Products_Review.csv")
```

# Importing Data with Pandas

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Let's explore the type

```
type(Product_Review)
```

Out[31]: pandas.core.frame.DataFrame

# Importing Data with Pandas

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



For files other than CSV format

```
pd.read_table("filename")
```

```
pd.read_excel("filename")
```

```
pd.read_sql(query, connection_object)
```

```
pd.read_json(json_string)
```

# Importing Data with Pandas

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



## Read from SQL Query or Database Table

```
>>> from sqlnew import create_table  
  
>>> engine = create_table('sqlite:///memory:')  
  
>>> pd.read_sql(SELECT * FROM my_table;, new1)  
  
>>> pd.read_sql_table('my_table', new1)  
  
>>> pd.read_sql_query(SELECT * FROM my_table;, new1)
```

# Analyzing Data-set

# Basic DataFrame Functionality

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Print the first 5 rows of the DataFrame

Product\_Review.head()

In [11]: Product_Review.head()								
Unnamed: 0	Product_Review_Phrase	Product_Title	Website_URL	Platform	Product_Rating	Product_Category		
0	Amazing - Highest Selling	Ladeia Bellies	http://www.amazon.com/ladeia-bellies/p/itmeh4k...	Mobile App	5.0	Footwear		
1	Good - Average Selling	Bulaky vanity case Jewellery Vanity Case	http://www.amazon.com/bulaky-vanity-case-jewel...	Website	3.0	Beauty and Personal Care		
2	Good - Average Selling	Roadster Men's Zipper Solid Cardigan	http://www.amazon.com/roadster-men-s-zipper-so...	Website	3.6	Clothing		
3	Amazing - Highest Selling	Cameril WM64 Elegance Analog Watch - For Men	http://www.amazon.com/cameril-wm64-elegance-an...	Mobile App	5.0	Others		
4	Amazing - Highest Selling	Colat COLAT_MW20 Sheen Analog Watch - For	http://www.amazon.com/colat-colat-mw20-sheen-a...	Mobile App	5.0	Others		

# Basic DataFrame Functionality

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Print the last 5 rows of the DataFrame

Product\_Review.tail()

```
In [13]: Product_Review.tail()
```

```
Out[13]:
```

	Unnamed: 0	Product_Review_Phrase	Product_Title	Website_URL	Platform	Product_Rating	Product_Category	Index
1835	1835	Bad - Lowest Selling	Do Bhai Women Heels	http://www.amazon.com/do-bhai-women-heels/p/...	Mobile App	1.0	Footwear	1835
1836	1836	Great - High Selling	Sindhi Footwear Ballerina Belles	http://www.amazon.com/sindhi-footwear-ballerin...	Website	4.0	Footwear	1836
1837	1837	Average - Low Selling	Purple Women Heels	http://www.amazon.com/purple-women-heels/p/...	Both	2.0	Footwear	1837
1838	1838	Great - High Selling	Uberlyfe Large Vinyl Sticker	http://www.amazon.com/uberlyfe-large-vinyl-sti...	Website	4.0	Baby Care	1838
1839	1839	Amazing - Highest Selling	We Witches Comfy Hues Women Wedges	http://www.amazon.com/witches-comfy-hues-wome...	Both	5.0	Footwear	1839

# Basic DataFrame Functionality

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Print the number of rows and columns

Product\_Review.shape

In [15]: Product\_Review.shape  
Out[15]: (1840, 11)

# Basic DataFrame Functionality

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



## Information of Index, Datatype and Memory

Product\_Review.info

In [18]:	Product_Review.info
Out[18]:	<bound method DataFrame.info of
	Unnamed: 0     Product _Review_Phrase \
0	0     Amazing - Highest Selling
1	1     Good - Average Selling
2	2     Good - Average Selling
3	3     Amazing - Highest Selling
4	4     Amazing - Highest Selling
5	5     Amazing - Highest Selling
6	6     Amazing - Highest Selling
7	7     Good - Average Selling
8	8     Great - High Selling
9	9     Amazing - Highest Selling
10	10    Great - High Selling
11	11    Good - Average Selling

# Merge, Join and Concatenate

Importing Convention

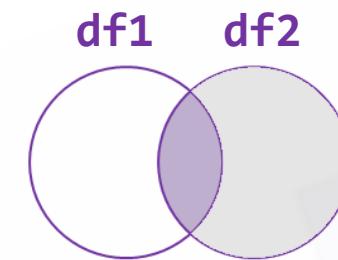
Data Analysing

Data Manipulation

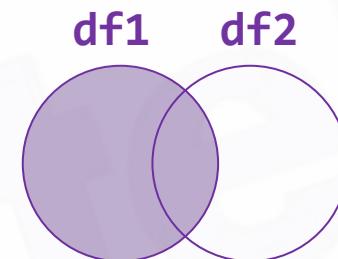
Data Visualization



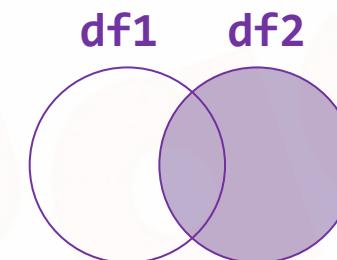
## DataFrame for Pandas Merge



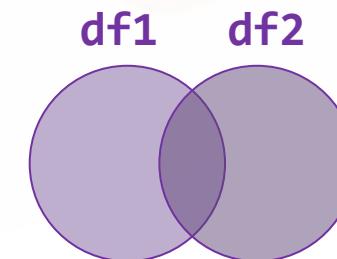
Inner Merge/  
Inner join



Left Merge/  
Left Join



Right Merge/  
Right Join



Outer Merge/  
Outer Join

# Merge, Join and Concatenate

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



## DataFrame for Pandas Merge

### DataFrame-1

```
player = ['Player1', 'Player2', 'Player3']
points = [8, 9, 5]
title = ['Game1', 'Game2', 'Game3']

df1 = pd.DataFrame({'Player': player,'Points': points,'Title': title})
df1 = df1[['Player', 'Points', 'Title']]
df1
```

```
player = ['Player1', 'Player2', 'Player3']
points = [8, 9, 5]
title = ['Game1', 'Game2', 'Game3']
df1 = pd.DataFrame({'Player': player,'Points': points,'Title': title})
df1 = df1[['Player', 'Points', 'Title']]
df1
```

	Player	Points	Title
0	Player1	8	Game1
1	Player2	9	Game2
2	Player3	5	Game3

# Merge, Join and Concatenate

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



## DataFrame for Pandas Merge

### DataFrame-2

```
player = ['Player1','Player5','Player6']
power = ['Punch','Kick', 'Elbow']
title = ['Game1','Game5','Game6']

df2 = pd.DataFrame({'Player': player, 'Power': power,'Title': title})

df2 = df2[['Player', 'Power', 'Title']]

df2
```

```
player = ['Player1','Player5','Player6']
power = ['Punch','Kick', 'Elbow']
title = ['Game1','Game5','Game6']
df2 = pd.DataFrame({'Player': player, 'Power': power,'Title': title})
df2 = df2[['Player', 'Power', 'Title']]
df2
```

	Player	Power	Title
0	Player1	Punch	Game1
1	Player5	Kick	Game5
2	Player6	Elbow	Game6

# Merge, Join and Concatenate

Importing Convention

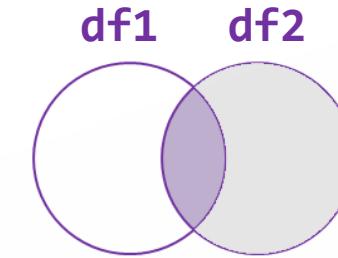
Data Analysing

Data Manipulation

Data Visualization



Inner Merge



Inner Merge

```
df1.merge(df2, on='Title', how='inner')
```

```
In [85]: df1.merge(df2, on='Title', how='inner')
```

```
Out[85]:
```

	Player_x	Points	Title	Player_y	Power
0	Player1	8	Game1	Player1	Punch

# Merge, Join and Concatenate

Importing Convention

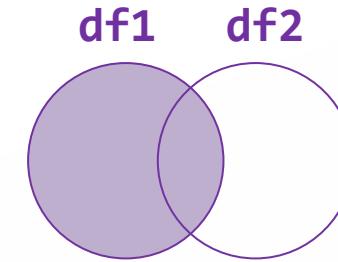
Data Analysing

Data Manipulation

Data Visualization



Left Merge



Left Merge

```
df1.merge(df2, on='Title', how='left')
```

	Player_x	Points	Title	Player_y	Power
0	Player1	8	Game1	Player1	Punch
1	Player2	9	Game2	NaN	NaN
2	Player3	5	Game3	NaN	NaN

# Merge, Join and Concatenate

Importing Convention

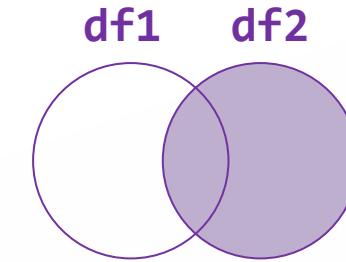
Data Analysing

Data Manipulation

Data Visualization



Right Merge



Right Merge

```
df1.merge(df2, on='Title', how='right')
```

	Player_x	Points	Title	Player_y	Power
0	Player1	8.0	Game1	Player1	Punch
1	NaN	NaN	Game5	Player5	Kick
2	NaN	NaN	Game6	Player6	Elbow

# Merge, Join and Concatenate

Importing Convention

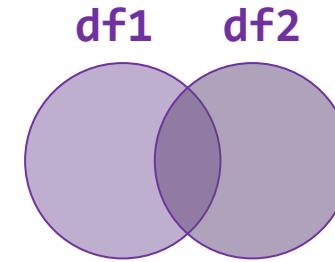
Data Analysing

Data Manipulation

Data Visualization



Outer merge



Outer Merge

```
df1.merge(df2, on='Title', how='outer')
```

	Player_x	Points	Title	Player_y	Power
0	Player1	8.0	Game1	Player1	Punch
1	Player2	9.0	Game2	NaN	NaN
2	Player3	5.0	Game3	NaN	NaN
3	NaN	NaN	Game5	Player5	Kick
4	NaN	NaN	Game6	Player6	Elbow

# Merge, Join and Concatenate

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



## DataFrame for Pandas Join

### DataFrame-1

```
player = ['Player1', 'Player2', 'Player3']
points = [8, 9, 5]
title = ['Game1', 'Game2', 'Game3']

df1 = pd.DataFrame({'Player': player,'Points': points,'Title': title})

df1.set_index('Player')

df1
```

	Points	Title
Player		
Player1	8	Game1
Player2	9	Game2
Player3	5	Game3

# Merge, Join and Concatenate

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



## DataFrame for Pandas Join

### DataFrame-2

```
player = ['Player1','Player5','Player6']

power = ['Punch','Kick', 'Elbow']

title = ['Game1','Game5','Game6']

df2 = pd.DataFrame({'Player': player, 'Power': power,'Title': title})

df2.set_index('Player')

df2
```

	Power	Title
Player		
Player1	Punch	Game1
Player5	Kick	Game5
Player6	Elbow	Game6

# Merge, Join and Concatenate

Importing Convention

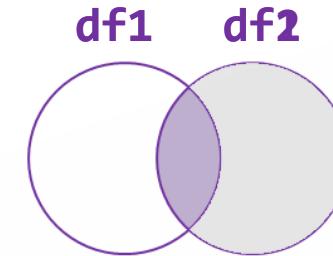
Data Analysing

Data Manipulation

Data Visualization



Inner Join



Inner Join

```
df1.join(df2, how='inner', lsuffix='_x', rsuffix='_y')
```

	Player_x	Points	Player_y	Power
Title	Game1	Player1	8	Player1
				Punch

# Merge, Join and Concatenate

Importing Convention

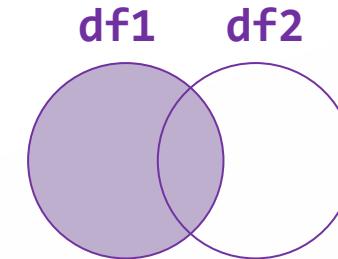
Data Analysing

Data Manipulation

Data Visualization



Left Join



Left Merge

```
df1.join(df2, how='left', lsuffix='_x', rsuffix='_y')
```

	Points	Power
Title		
Game1	8	Punch
Game2	5	NaN
Game3	9	NaN

# Merge, Join and Concatenate

Importing Convention

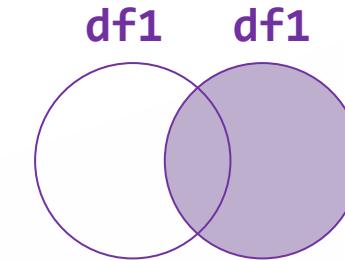
Data Analysing

Data Manipulation

Data Visualization



Right Join



right Join

```
df1.join(df2, how='left', lsuffix='_x', rsuffix='_y')
```

	Points	Power
Title		
Game1	8	Punch
Game2	5	NaN
Game3	9	NaN

# Merge, Join and Concatenate

Importing Convention

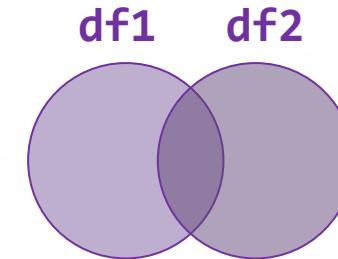
Data Analysing

Data Manipulation

Data Visualization



Outer join



Outer Join

```
df1.join(df2, how='outer', lsuffix='_x', rsuffix='_y')
```

	Points	Power
Title		
Game1	8	Punch
Game2	5	NaN
Game3	9	NaN

# Merge, Join and Concatenate

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Concatenate

Pd.concat([df1,df2])

	Player	Points	Title	Power
0	Player1	8.0	Game1	Nan
1	Player2	9.0	Game2	Nan
2	Player3	5.0	Game3	Nan
0	Player1	Nan	Game1	Punch
1	Player5	Nan	Game5	Kick
2	Player6	Nan	Game6	Elbow

# Pandas DataFrame Methods

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Mean

Product\_Review.mean()

```
Out[54]: Unnamed: 0           919.500000
          Product_Rating      3.802989
          is_Amazon_Advantage_product 0.164674
          Product_Launch_Year    2016.000000
          Product_Launch_Month   7.000000
          Product_Launch_Day     12.000000
          dtype: float64
```

# Pandas DataFrame Methods

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Median

Product\_Review.median()

```
Out[57]: Unnamed: 0          919.5
          Product_Rating      4.0
          is_Amazon_Advantage_product  0.0
          Product_Launch_Year    2016.0
          Product_Launch_Month   7.0
          Product_Launch_Day     12.0
          dtype: float64
```

# Pandas DataFrame Methods

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Standard Deviation:

`Product_Review.std()`

```
Unnamed: 0           531.306566
Product_Rating       1.263654
is_Amazon_Advantage_product 0.370987
Product_Launch_Year   0.000000
Product_Launch_Month  0.000000
Product_Launch_Day    0.000000
dtype: float64
```

# Pandas DataFrame Methods

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Maximum of each column:

Product\_Review.max()

Unnamed: 0	1839
Product_Review_Phrase	Great - High Selling
Product_Title	zDelhi.com Car Washer Z1 Ultra High Pressure W...
Website_URL	<a href="http://www.amazon.com/zyxel-vmg1312-b10a-vdsl2...">http://www.amazon.com/zyxel-vmg1312-b10a-vdsl2...</a>
Platform	Website
Product_Rating	5
Product_Category	Others
is_Amazon_Advantage_product	True
Product_Launch_Year	2016
Product_Launch_Month	7
Product_Launch_Day	12
dtype: object	

# Pandas DataFrame Methods

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Minimum of each column:

Product\_Review.min()

```
Unnamed: 0          0
Product_Review_Phrase  Amazing - Highest Selling
Product_Title        3a Autocare 3D MAT Car Mat Suzuki New Swift
Website_URL          http://www.amazon.com/3a-autocare-3d-mat-car-s...
Platform              Both
1
Product_Rating        1
Product_Category      Automotive
is_Amazon_Advantage_product False
Product_Launch_Year   2016
Product_Launch_Month 7
Product_Launch_Day    12
dtype: object
```

# Pandas DataFrame Methods

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Count of non-null values in each column:

`Product_Review.count()`

Unnamed: 0	1840
Product_Review_Phrase	1840
Product_Title	1840
Website_URL	1840
Platform	1840
Product_Rating	1840
Product_Category	1840
is_Amazon_Advantage_product	1840
Product_Launch_Year	1840
Product_Launch_Month	1840
Product_Launch_Day	1840
dtype: int64	

# Pandas DataFrame Methods

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Summary statistics for numerical column

`Product_Review.describe()`

	Unnamed: 0	Product_Rating	Product_Launch_Year	Product_Launch_Month	Product_Launch_Day
count	1840.000000	1840.000000	1840.0	1840.0	1840.0
mean	919.500000	3.802989	2016.0	7.0	12.0
std	531.306566	1.263654	0.0	0.0	0.0
min	0.000000	1.000000	2016.0	7.0	12.0
25%	459.750000	3.000000	2016.0	7.0	12.0
50%	919.500000	4.000000	2016.0	7.0	12.0
75%	1379.250000	5.000000	2016.0	7.0	12.0
max	1839.000000	5.000000	2016.0	7.0	12.0

# Pandas Mathematical Operations

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



e.g.: Divide every value in the Product Rating column by 2

```
Product_Review[“Product_Rating”] /2
```

```
Out[83]: 0    2.5
          1    1.5
          2    1.8
          3    2.5
          4    2.5
Name: Product_Rating, dtype: float64
```

# Manipulating Data-set

# DataFrame Indexing

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Selecting by Position

01

Selecting by Label

02

# DataFrame Indexing

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Selecting by Position

01

Selecting by Position

Product\_Review.iloc[:,0]

In [22]: Product\_Review.iloc[:,0]

```
Out[22]: 0      0  
         1      1  
         2      2  
         3      3  
         4      4  
         5      5  
         6      6  
         7      7  
         8      8  
         9      9  
        10     10  
        11     11  
        12     12  
        13     13  
        14     14
```

# DataFrame Indexing

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Selecting by Position

01

Selecting by Position

Product\_Review.iloc[0:5,4]

In [24]: Product\_Review.iloc[0:5,4]

Out[24]: 0 Mobile App  
1 Website  
2 Website  
3 Mobile App  
4 Mobile App  
Name: Platform, dtype: object

# DataFrame Indexing

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Selecting by Position

01

Selecting by Label

02

Selecting by Position

Product\_Review.iloc[:,:]

In [26]:	Product_Review.iloc[:,:]						
Out[26]:	Unnamed: 0	Product_Review_Phrase	Product_Title	Website_URL	Platform	Product_Rating	
0	0	Amazing - Highest Selling	Ladeia Bellies	http://www.amazon.com/ladeia-bellies/p/itmeh4k...	Mobile App	5.0	
1	1	Good - Average Selling	Bulaky vanity case Jewellery Vanity Case	http://www.amazon.com/bulaky-vanity-case-jewel...	Website	3.0	
2	2	Good - Average Selling	Roadster Men's Zipper Solid Cardigan	http://www.amazon.com/roadster-men-s-zipper-so...	Website	3.6	
3	3	Amazing - Highest Selling	Camerii WM64 Elegance Analog Watch - For Men...	http://www.amazon.com/camerii-wm64-elegance-an...	Mobile App	5.0	
4	4	Amazing - Highest Selling	Colat COLAT_MW20 Sheen Analog Watch - For Men...	http://www.amazon.com/colat-colat-mw20-sheen-a...	Mobile App	5.0	
5	5	Amazing - Highest Selling	Rorlig RR-028 Expedition Analog Watch - For M...	http://www.amazon.com/rorlig-rr-028-expedition...	Mobile App	5.0	
6	6	Amazing -	Lyc White Casual Boots	http://www.amazon.com/lyc-white-	Mobile	5.0	

# DataFrame Indexing

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Selecting by Position

01

Selecting by Label

02

Selecting by Position

Product\_Review.iloc[6:,4:]

In [28]: Product\_Review.iloc[6:,4:]

Out[28]:

	Platform	Product_Rating	Product_Category	is_Amazon_Advantage_product	Product_Launch_Year
6	Mobile App	5.0	Footwear	False	2016
7	Website	3.5	Others	True	2016
8	Both	4.0	Footwear	False	2016
9	Mobile App	5.0	Others	False	2016
10	Both	4.0	Others	True	2016
11	Website	3.5	Footwear	False	2016
12	Mobile App	5.0	Others	False	2016
13	Website	3.5	Footwear	False	2016
14	Mobile App	5.0	Others	False	2016
15	Website	3.6	Others	True	2016
16	Both	1.0	Others	False	2016

# DataFrame Indexing

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Selecting by Position

01

Selecting by Position

Selecting by Label

02

```
Produt_Reviews= Product_Reviews.iloc[:,1]  
Product_Reviews.head()
```

```
Out[108]: 0    Amazing - Highest Selling  
           1    Good - Average Selling  
           2    Good - Average Selling  
           3    Amazing - Highest Selling  
           4    Amazing - Highest Selling  
Name: Product _Review_Phrase, dtype: object
```

# DataFrame Indexing

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Selecting by Position

01

Selecting by Label

02

Selecting by label:

```
Product_Review.loc[:5,"Product_Title"]
```

```
In [33]: Product_Review.loc[:5,"Product_Title"]
Out[33]: 0 Ladela Bellies
          1 Bulaky vanity case Jewellery Vanity Case
          2 Roadster Men's Zipper Solid Cardigan
          3 Camerii WM64 Elegance Analog Watch - For Men, ...
          4 Colat COLAT_MN20 Sheen Analog Watch - For Men, ...
          5 Rorlig RR-028 Expedition Analog Watch - For M...
Name: Product_Title, dtype: object
```

# DataFrame Indexing

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Selecting by Position

01

Selecting by Label

02

Selecting by label:

```
Product_Review.loc[:5,"Product_Title","Product_Rating"]
```

```
In [36]: Product_Review.loc[:5,("Product_Title","Product_Rating"))
```

```
Out[36]:
```

	Product_Title	Product_Rating
0	Ladela Bellies	5.0
1	Bulaky vanity case Jewellery Vanity Case	3.0
2	Roadster Men's Zipper Solid Cardigan	3.6
3	Camerii WM64 Elegance Analog Watch - For Men...	5.0
4	Colat COLAT_MW20 Sheen Analog Watch - For Men...	5.0
5	Rorlig RR-028 Expedition Analog Watch - For M...	5.0

# DataFrame Setting

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Setting a value to one specific column

```
Product_Review['Platform'] = 6
```

```
Product_Review
```

Product_Review['Platform'] = 6 Product_Review					
Unnamed: 0	Product Review_Phrase	Product_Title	Website_URL	Platform	
0	Amazing - Highest Selling	Ladela Bellies	http://www.amazon.com/ladela- bellies/p/itmeh4k...	6	
1	Good - Average Selling	Bulaky vanity case Jewellery Vanity Case	http://www.amazon.com/bulaky- vanity-case-jewel...	6	
2	Good - Average Selling	Roadster Men's Zipper Solid Cardigan	http://www.amazon.com/roadster- men-s-zipper-so...	6	
3	Amazing - Highest Selling	Camerli WM54 Elegance Analog Watch - For Men...	http://www.amazon.com/camerli- wm54-elegance-an...	6	
4	Amazing - Highest Selling	Colat COLAT_MW20 Sheen Analog Watch - For Men...	http://www.amazon.com/colat-colat- mw20-sheen-a...	6	
5	Amazing - Highest Selling	Rorlig RR-028 Expedition Analog Watch - For M...	http://www.amazon.com/rorlig-rr- 028-expedition...	6	

# Applying Functions on DataFrame

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Double up all numeric values using lambda function

```
f = lambda x: x*2
```

```
df.apply(f)
```

```
df=pd.read_csv("Amazon_Products_Review.csv")
f = lambda x: x*2
df.apply(f)
df.applymap(f)
```

	Unnamed: 0	Product Review_Phrase	Product_Title	Website_URL	Platform	Product_Rating
0	0	Amazing - Highest SellingAmazing - Highest Sel...	Ladela BellesLadela Belles	http://www.amazon.com/ladela- belles/p/1tmeh4k...	Mobile AppMobile App	10.0
1	2	Good - Average SellingGood - Average Selling	Bulaky vanity case Jewellery Vanity CaseBulaky...	http://www.amazon.com/bulaky- vanity-case-jewel...	WebsiteWebsite	6.0
2	4	Good - Average SellingGood - Average Selling	Roadster Men's Zipper Sold CardiganRoadster M...	http://www.amazon.com/roadster- men-s-zipper-so...	WebsiteWebsite	7.2

# DataFrame Sorting

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



By default in ascending order.

```
Product_Review.sort_values(by='Product_Rating')
```

In [39]:	Product_Review.sort_values(by="Product_Rating")							
Out[39]:	Unnamed: 0	Product_Review_Phrase	Product_Title	Website_URL	Platform	Product_Rating	Product_Category	
	462	462	Bad - Lowest Selling	Neo Gold leaf Fancy School Art Plastic Pencil Box	<a href="http://www.amazon.com/neo-gold-leaf-fancy-scho...">http://www.amazon.com/neo-gold-leaf-fancy-scho...</a>	Both	1.0	Others
	186	186	Bad - Lowest Selling	Ploomz Fashion Women's Full Coverage Bra	<a href="http://www.amazon.com/ploomz-fashion-women-s-f...">http://www.amazon.com/ploomz-fashion-women-s-f...</a>	Both	1.0	Clothing
	1034	1034	Bad - Lowest Selling	Durian BID/32625/A/2 Leatherette 2 Seater Sofa	<a href="http://www.amazon.com/durian-bid-32625-a-2-sea...">http://www.amazon.com/durian-bid-32625-a-2-sea...</a>	Both	1.0	Others
	1030	1030	Bad - Lowest Selling	Kick Women's Leggings	<a href="http://www.amazon.com/kick-women-s-leggings/p...">http://www.amazon.com/kick-women-s-leggings/p...</a>	Both	1.0	Clothing
	...	...	Bad - Lowest	Tia by Ten on Ten Melanone	<a href="http://www.amazon.com/tia-ten-on-ten-melanone...">http://www.amazon.com/tia-ten-on-ten-melanone...</a>	...	...	...

# Pandas DataFrame Sorting

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



For descending order make **ascending=False**

```
Product_Review.sort_values('Product_Rating', ascending=False)
```

Product_Review.sort_values("Product_Rating", ascending=False)								
Unnamed: 0	_Review_Phrase	Product_Title	Website_URL	Platform	Product_Rating	Product_Category		
0	Amazing - Highest Selling	Ladeia Belles	http://www.amazon.com/ladeia-belles/pitmeh4k...	Mobile App	5.0	Footwear		
1011	Amazing - Highest Selling	Melmo Slippers	http://www.amazon.com/melmo-slippers/pitmehgt...	Website	5.0	Footwear		
976	Amazing - Highest Selling	Clovia In Aqua Women's Full Coverage Bra	http://www.amazon.com/clovia-aqua-women-s-full...	Website	5.0	Clothing		
981	Amazing - Highest Selling	Baktol Printed Men's Round Neck T-Shirt	http://www.amazon.com/baktol-printed-men-s-rou...	Website	5.0	Clothing		
989	Amazing - Highest Selling	SS Ton Professional English Willow Cricket	http://www.amazon.com/ss-ton-professional-engl...	Website	5.0	Others		

# Pandas DataFrame Ranking

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Rank the Product\_Rating column

```
Product_Review["Product_Rating"].rank()
```

Product_Review["Product_Rating"].rank()	
0	1530.5
1	426.5
2	625.0
3	1530.5
4	1530.5
5	1530.5
6	1530.5
7	590.0
8	850.0
9	1530.5
10	850.0
11	590.0
12	1530.5

# Pandas DataFrame Dropping

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Drop Product\_Rating column from the dataset

```
Product_Review.drop('Product_Rating', axis=1)
```

Product_Review.drop('Product_Rating', axis=1)						
Unnamed: 0		Product Review_Phrase	Product_Title	Website_URL	Platform	Product_Category
462	462	Bad - Lowest Selling	Neo Gold leaf Fancy School Art Plastic Pencil Box	http://www.amazon.com/neo-gold-leaf-fancy-scho...	Both	Others
1034	1034	Bad - Lowest Selling	BID/32625/A/2 Leatherette 2 Seater Sofa	Durian http://www.amazon.com/durian-bid-32625-a-2-lea...	Both	Others
1030	1030	Bad - Lowest Selling	Klick Women's Leggings	http://www.amazon.com/klick-women-s-leggings/p...	Both	Clothing
1027	1027	Bad - Lowest Selling	Designwallas Festive Pack (Set of 2) Notebook	Designwallas Festive Pack (Set of 2) Notebook http://www.amazon.com/designwallas-festive-pac...	Both	Others

# Pandas DataFrame Filtering

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Filtering the column by value:

```
filter1 = Product_Review["Product_Rating"] > 3  
  
filter1.head()
```

```
0    True  
1   False  
2    True  
3    True  
4    True  
Name: Product_Rating, dtype: bool
```

# Pandas DataFrame Filtering

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Filtering the column by value:

```
filter1 = Product_Review["Product_Rating"] > 3  
  
filtered_new = Product_Review[filter1]  
  
filtered_new.head()
```

Unnamed: 0	Product _Review_Phase	Product_Title	Website_URL	Platform	Product_Rating	Product_Category	is_Amazon_Advantage_product
0	0 Amazing - Highest Selling	Ladela Belles	http://www.amazon.com/ladela-belles/p/BLMB4L...	Mobile App	5.0	Footwear	False
2	2 Good - Average Selling	Roadster Men's Zipper Solid Cardigan	http://www.amazon.com/roadster-men-s-zipper-so...	Website	3.6	Clothing	False
3	3 Amazing - Highest Selling	Cameri WM54 Elegance Analog Watch - For Men	http://www.amazon.com/cameri-wm54-elegance-an...	Mobile App	5.0	Others	False
4	4 Amazing - Highest Selling	Colat_MW29 Sheen Analog Watch - For Men	http://www.amazon.com/colat-mw29-sheen-a...	Mobile App	5.0	Others	False
5	5 Amazing - Highest Selling	Rorig RR-028 Expedition Analog Watch	http://www.amazon.com/rorig-n-028-expedition...	Mobile App	5.0	Others	False

# Pandas DataFrame Filtering

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



Filtering the column by numeric and Boolean value:

```
filter2 = (Product_Review["Product_Rating"] > 3)  
& (Product_Review["Product_Category"] == "Footwear")  
  
filtered_review = Product_Review[filter2]  
  
filtered_review
```

Unnamed: 0	Product_Review_Phrase	Product_Title	Website_URL	Platform	Product_Rating	Product_Category	is_Amazon_Advantage_product
0	Amazing - Highest Selling	Ladela Belles	http://www.amazon.com/ladela-belles/p/1tmeh4k...	Mobile App	5.0	Footwear	False
6	Amazing - Highest Selling	Lyc White Casual Boots	http://www.amazon.com/lyc-white-casual-boots/p/...	Mobile App	5.0	Footwear	False
8	Great - High Selling	Bruno Manetti Camellia Boots	http://www.amazon.com/bruno-manetti-camellia-...	Both	4.0	Footwear	False
11	Good - Average Selling	Kielz Ladies Boots	http://www.amazon.com/kielz-ladies-boots/p/1lm...	Website	3.5	Footwear	False
13	Good - Average Selling	Kielz Ladies Boots	http://www.amazon.com/kielz-ladies-boots/p/1lm...	Website	3.5	Footwear	False

# Data-set Visualization

# Data Visualization

Importing Convention

Data Analysing

Data Manipulation

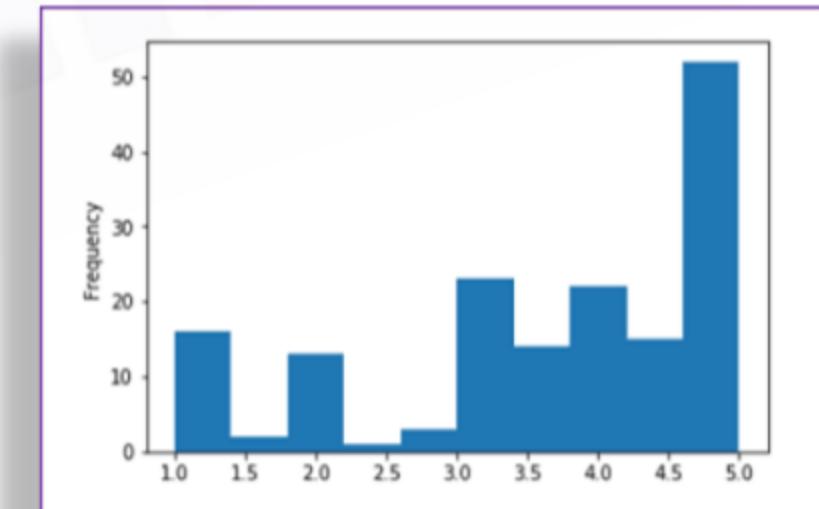
Data Visualization



## Histogram:

```
%matplotlib inline
```

```
Product_Review[Product_Review["Product_Category"]  
== "Footwear"]["Product_Rating"].plot(kind="hist")
```



# Data Visualization Using Pandas

Importing Convention

Data Analysing

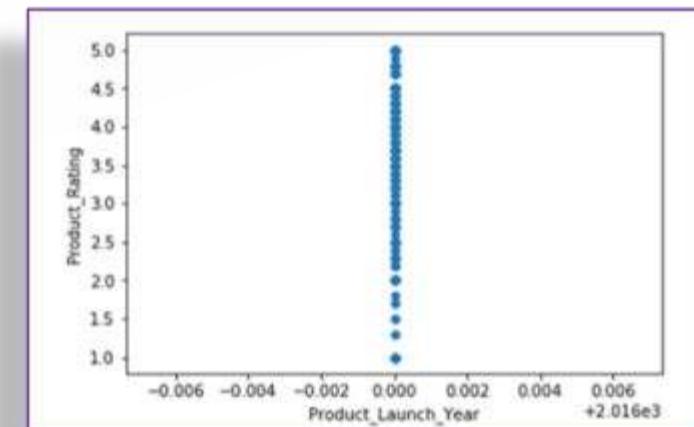
Data Manipulation

Data Visualization



Scatter plot

```
Product_Review.plot.scatter(x="Product_Launch_Year",  
y="Product_Rating")
```



# Data Visualization

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



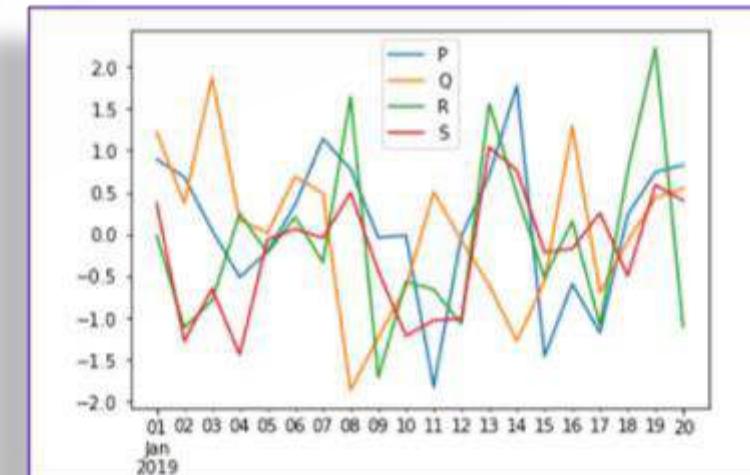
## DataFrame.plot():

```
import pandas as pd
```

```
import numpy as np
```

```
df = pd.DataFrame(np.random.randn(20,4),index=pd.date_range('1/1/2019',  
periods=20), columns=list('PQRS'))
```

```
df.plot()
```



# Data Visualization

Importing Convention

Data Analysing

Data Manipulation

Data Visualization



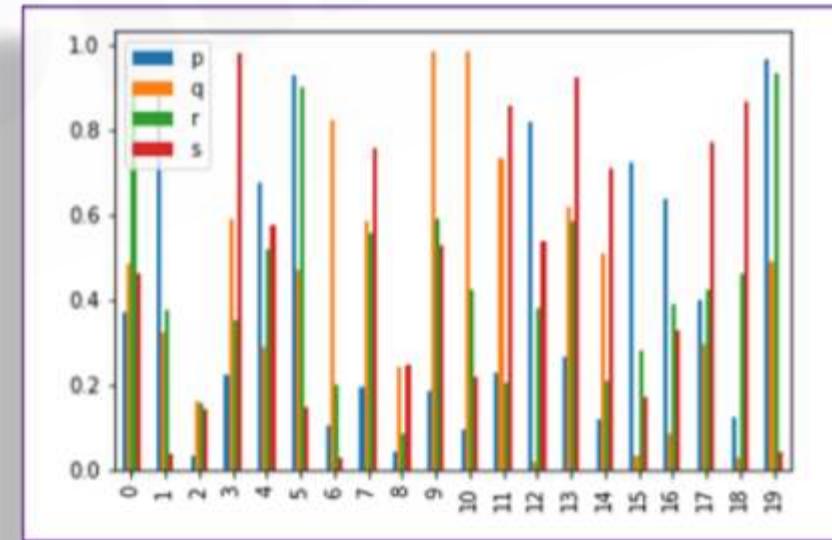
DataFrame.plot.bar():

```
import pandas as pd
```

```
import numpy as np
```

```
df = pd.DataFrame(np.random.rand(20,4),columns=['p','q','r','s'])
```

```
df.plot.bar()
```



# Data Visualization

Importing Convention

Data Analysing

Data Manipulation

Data Visualization

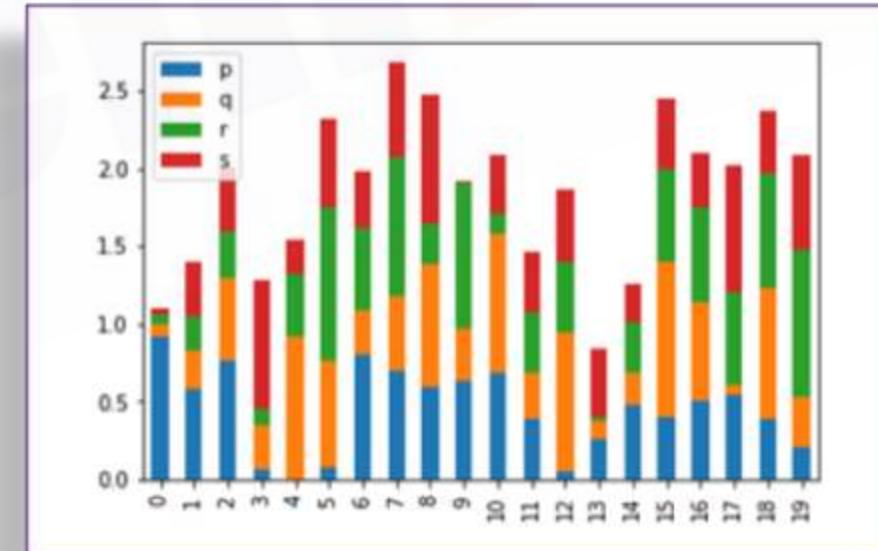


DataFrame.plot.bar(stacked=True):

```
import pandas as pd
```

```
df = pd.DataFrame(np.random.rand(20,4),columns=['p','q','r','s'])
```

```
df.plot.bar(stacked=True)
```



# Data Visualization

Importing Convention

Data Analysing

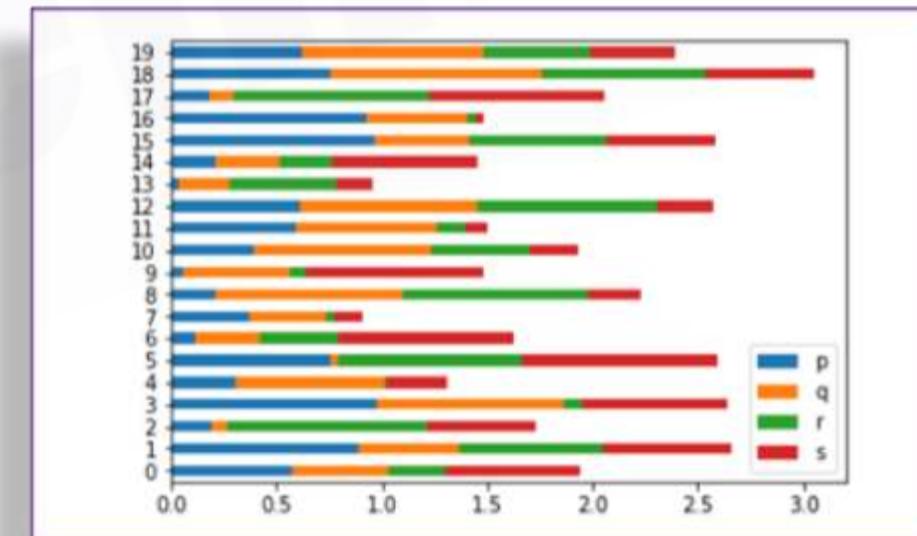
Data Manipulation

Data Visualization



DataFrame.plot.barh(stacked=True):

```
import pandas as pd  
import numpy as np  
  
df = pd.DataFrame(np.random.rand(20,4),columns=['p','q','r','s'])  
  
df.plot.barh(stacked=True)
```



# Data Visualization

Importing Convention

Data Analysing

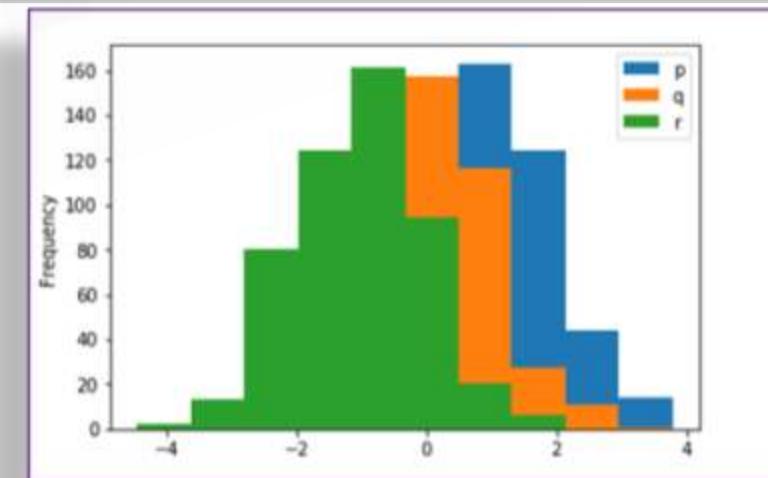
Data Manipulation

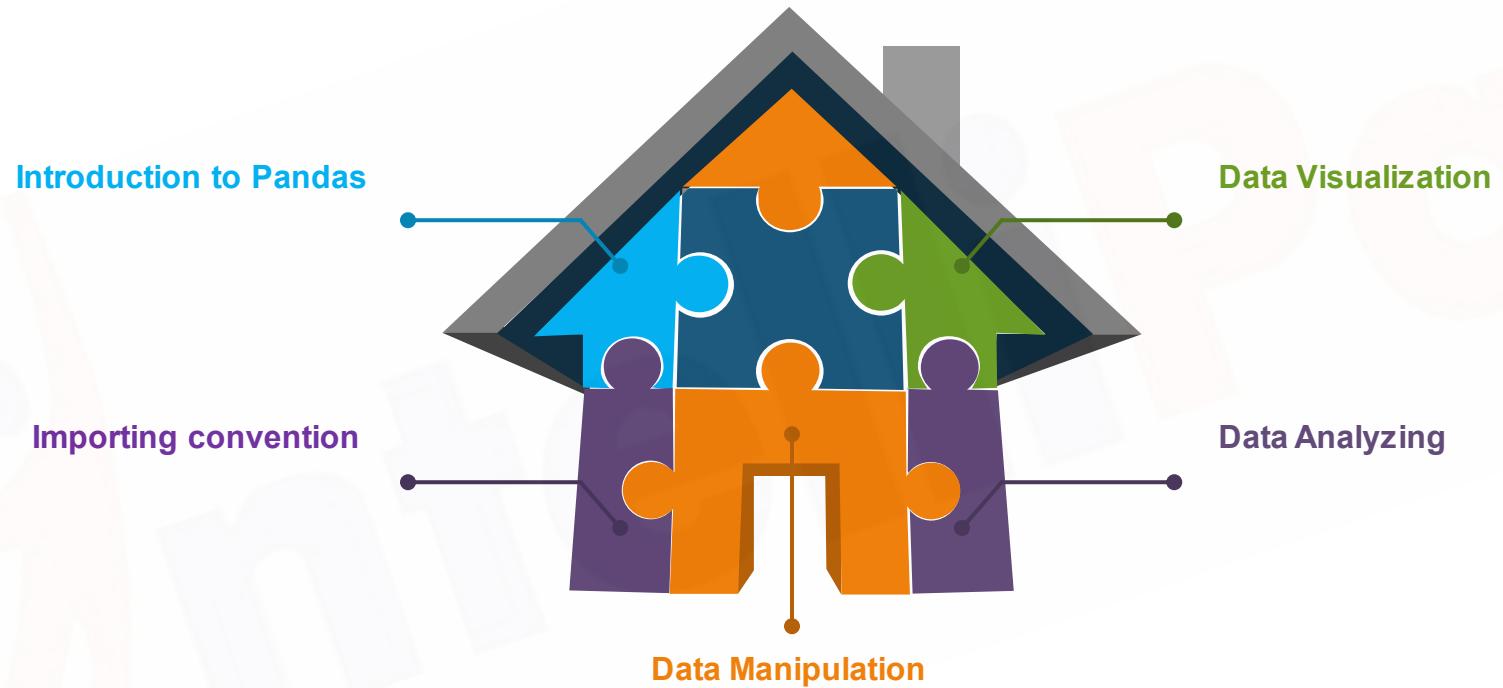
Data Visualization



DataFrame.plot.hist():

```
import pandas as pd  
  
import numpy as np  
  
df = pd.DataFrame({'p':np.random.randn(500)+1,'q':np.random.randn(500),'c':  
np.random.randn(500) - 1}, columns=['p', 'q', 'r'])  
  
df.plot.hist(bins=20)
```





# Thank You

[www.intellipaat.com](http://www.intellipaat.com)



**CALL US NOW**

India : +91-7847955955

US : 1-800-216-8930 (TOLL FREE)

[sales@intellipaat.com](mailto:sales@intellipaat.com)



**India : +91-7847955955**

**US : 1-800-216-8930 (TOLL FREE)**

**[sales@intellipaat.com](mailto:sales@intellipaat.com)**

**24X7 Chat with our Course Advisor**