Pandas Introduction

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.

Data Science: is a branch of computer science where we study how to store, use and analyze data for deriving information from it.

What Can Pandas Do? Pandas gives you answers about the data. Like:

Is there a correlation between two or more columns? What is average value? Max value? Min value? Pandas are also able to delete rows that are not relevant, or contains wrong values, like empty or NULL values. This is called cleaning the data.

In [2]:
```
#importing pandas
import pandas as pd
```

Pandas Series

A Pandas Series is like a column in a table. It is a 1-D array holding data of any type.

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

```
0    1
1    7
2    2
dtype: int64
```

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

This label can be used to access a specified value.

In [9]:
```
print(myvar[0])
```

```
1
```

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

In [11]:
```
import pandas as pd
a = [1, 7, 2]
myvar = pd.Series(a, index = ["x", "y", "z"])
print(myvar)
```

```
x    1
y    7
```

```
z    2
dtype: int64
```

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

In [12]:
```
print(myvar["y"])
```

```
7
```

In [14]:
```
# if one of the value is float that whole series dtype is float64
import pandas as pd
a=(1.4,2,3,4)
myvar=pd.Series(a)
print(myvar)
```

```
0    1.4
1    2.0
2    3.0
3    4.0
dtype: float64
```

In [15]:
```
student_name=["ram","shyam","radha","geeta","seeta"]
marks=[50,60,70,80,90]
pd.Series(marks,index=student_name)
```

Out[15]:
```
ram      50
shyam    60
radha    70
geeta    80
seeta    90
dtype: int64
```

In [16]:
```
student_name=["ram","shyam","radha","geeta","seeta"]
marks=[50,60,70,80,90]
pd.Series(marks,index=student_name,name="student result")
```

Out[16]:
```
ram      50
shyam    60
radha    70
geeta    80
seeta    90
Name: student result, dtype: int64
```

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

In [17]:
```
import pandas as pd

calories = {"day1": 420, "day2": 380, "day3": 390}

myvar = pd.Series(calories)

print(myvar)
```

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

To select only some of the items in the dictionary, use the index argument and specify only the items you want to include in the Series.

In [18]:
```
import pandas as pd
```

```
calories = {"day1": 420, "day2": 380, "day3": 390}

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

print(myvar)
```

```
day1    420
day2    380
dtype: int64
```

In [19]:
```
marks={"ram":55,"shyam":60,"radha":80}
marks_series=pd.Series(marks,name="student result")
print(marks_series)
```

```
ram      55
shyam    60
radha    80
Name: student result, dtype: int64
```

Series Attributes

In [20]:
```
marks_series.size
```

Out[20]: 3

In [22]:
```
marks_series.dtype
```

Out[22]: dtype('int64')

In [23]:
```
marks_series.name
```

Out[23]: 'student result'

In [24]:
```
marks_series.index
```

Out[24]: Index(['ram', 'shyam', 'radha'], dtype='object')

In [25]:
```
print(marks_series.values)
print(type(marks_series.values))
```

```
[55 60 80]
<class 'numpy.ndarray'>
```

In [26]:
```
print(marks_series.is_unique)

print(pd.Series([55,22,33,44,55]).is_unique)
```

```
True
False
```

In [28]:
```
# csv File with one col
subs = pd.read_csv('subs.csv')#,squeeze=True)
subs
```

Out[28]:

	Subscribers gained
0	48
1	57
2	40

	Subscribers gained
3	43
4	44
...	...
360	231
361	226
362	155
363	144
364	172

365 rows × 1 columns

In [29]:
```
type(subs)
```

Out[29]:
```
pandas.core.frame.DataFrame
```

In [30]:
```
subs = pd.read_csv(('subs.csv'),squeeze=True) # squeeze is used to convert dataframe to
subs
```

Out[30]:
```
0       48
1       57
2       40
3       43
4       44
      ...
360    231
361    226
362    155
363    144
364    172
Name: Subscribers gained, Length: 365, dtype: int64
```

In [32]:
```
type(subs)
```

Out[32]:
```
pandas.core.series.Series
```

In [38]:
```
subs = pd.read_csv(('subs.csv'),squeeze=True) # squeeze is used to converdt dataframe t
subs.describe()
```

Out[38]:
```
count    365.000000
mean     135.643836
std       62.675023
min       33.000000
25%       88.000000
50%      123.000000
75%      177.000000
max      396.000000
Name: Subscribers gained, dtype: float64
```

In [39]:
```
subs.min()
```

Out[39]:
```
33
```

In [40]:
```
subs.max()
```

Out[40]:
```
396
```

In [41]:
```
subs.median()
```

Out[41]:
```
123.0
```

In [42]:
```
subs.sum()
```

Out[42]:
```
49510
```

In [33]:
```
movies = pd.read_csv('bollywood.csv',index_col='movie',squeeze=True)
print(movies)
print(type(movies))
```

```
movie
Uri: The Surgical Strike                   Vicky Kaushal
Battalion 609                                Vicky Ahuja
The Accidental Prime Minister (film)         Anupam Kher
Why Cheat India                            Emraan Hashmi
Evening Shadows                          Mona Ambegaonkar
                                                    ...
Hum Tumhare Hain Sanam                     Shah Rukh Khan
Aankhen (2002 film)                      Amitabh Bachchan
Saathiya (film)                              Vivek Oberoi
Company (film)                                 Ajay Devgn
Awara Paagal Deewana                         Akshay Kumar
Name: lead, Length: 1500, dtype: object
<class 'pandas.core.series.Series'>
```

Series Methods

In [35]:
```
movies.head()
```

Out[35]:
```
movie
Uri: The Surgical Strike                   Vicky Kaushal
Battalion 609                                Vicky Ahuja
The Accidental Prime Minister (film)         Anupam Kher
Why Cheat India                            Emraan Hashmi
Evening Shadows                          Mona Ambegaonkar
Name: lead, dtype: object
```

In [36]:
```
movies.tail()
```

Out[36]:
```
movie
Hum Tumhare Hain Sanam      Shah Rukh Khan
Aankhen (2002 film)       Amitabh Bachchan
Saathiya (film)               Vivek Oberoi
Company (film)                  Ajay Devgn
Awara Paagal Deewana          Akshay Kumar
Name: lead, dtype: object
```

In [37]:
```
movies.head(3)
```

Out[37]:
```
movie
Uri: The Surgical Strike                 Vicky Kaushal
Battalion 609                              Vicky Ahuja
The Accidental Prime Minister (film)       Anupam Kher
Name: lead, dtype: object
```

In [43]:
```
movies.tail(3)
```

Out[43]:
```
movie
Saathiya (film)         Vivek Oberoi
Company (film)            Ajay Devgn
Awara Paagal Deewana    Akshay Kumar
Name: lead, dtype: object
```

Series Indexing

In [44]:
```
x = pd.Series([12,13,14,35,46,57,58,79,9])
x
```

Out[44]:
```
0    12
1    13
2    14
3    35
4    46
5    57
6    58
7    79
8     9
dtype: int64
```

In [46]:
```
x[3]
```

Out[46]:
```
35
```

In [47]:
```
x[0:6]
```

Out[47]:
```
0    12
1    13
2    14
3    35
4    46
5    57
dtype: int64
```

In [48]:
```
x[::-1]
```

Out[48]:
```
8     9
7    79
6    58
5    57
4    46
3    35
2    14
1    13
0    12
dtype: int64
```

In [49]:
```
movies[0]
```

Out[49]:
```
'Vicky Kaushal'
```

In [50]:
```
movies['Uri: The Surgical Strike']
```

Out[50]:
```
'Vicky Kaushal'
```

In [51]:
```
movies[-5:]
```

Out[51]:
```
movie
Hum Tumhare Hain Sanam      Shah Rukh Khan
Aankhen (2002 film)       Amitabh Bachchan
Saathiya (film)               Vivek Oberoi
Company (film)                  Ajay Devgn
Awara Paagal Deewana          Akshay Kumar
Name: lead, dtype: object
```

In [52]:
```
movies[::2]
```

Out[52]:
```
movie
Uri: The Surgical Strike                   Vicky Kaushal
The Accidental Prime Minister (film)         Anupam Kher
Evening Shadows                         Mona Ambegaonkar
Fraud Saiyaan                               Arshad Warsi
Manikarnika: The Queen of Jhansi           Kangana Ranaut
                                                     ...
Raaz (2002 film)                             Dino Morea
Waisa Bhi Hota Hai Part II                  Arshad Warsi
Kaante                                   Amitabh Bachchan
Aankhen (2002 film)                      Amitabh Bachchan
Company (film)                                 Ajay Devgn
Name: lead, Length: 750, dtype: object
```

In [53]:
```
marks_series[1] = 100
marks_series
```

Out[53]:
```
ram       55
shyam    100
radha     80
Name: student result, dtype: int64
```

In [54]:
```
movies[[0,1,3,4,5]]
```

Out[54]:
```
movie
Uri: The Surgical Strike           Vicky Kaushal
Battalion 609                        Vicky Ahuja
Why Cheat India                     Emraan Hashmi
Evening Shadows                 Mona Ambegaonkar
Soni (film)                  Geetika Vidya Ohlyan
Name: lead, dtype: object
```

In [55]:
```
list(marks_series)
```

Out[55]:
```
[55, 100, 80]
```

In [56]:
```
dict(marks_series)
```

Out[56]:
```
{'ram': 55, 'shyam': 100, 'radha': 80}
```

In [57]:
```
'2 States (2014 film)' in movies
```

Out[57]:
```
True
```

In [58]:
```
'Alia Bhatt' in movies.values
```

Out[58]:
```
True
```

In [59]:
```
# This will show the movie names in the csv.
for i in movies.index:
```

```
    print(i)
```

```
Uri: The Surgical Strike
Battalion 609
The Accidental Prime Minister (film)
Why Cheat India
Evening Shadows
Soni (film)
Fraud Saiyaan
Bombairiya
Manikarnika: The Queen of Jhansi
Thackeray (film)
Amavas
Gully Boy
Hum Chaar
Total Dhamaal
Sonchiriya
Badla (2019 film)
Mard Ko Dard Nahi Hota
Hamid (film)
Photograph (film)
Risknamaa
Mere Pyare Prime Minister
22 Yards
Kesari (film)
Notebook (2019 film)
Junglee (2019 film)
Gone Kesh
Albert Pinto Ko Gussa Kyun Aata Hai?
The Tashkent Files
Kalank
Setters (film)
Student of the Year 2
PM Narendra Modi
De De Pyaar De
India's Most Wanted (film)
Yeh Hai India
Khamoshi (2019 film)
Kabir Singh
Article 15 (film)
One Day: Justice Delivered
Hume Tumse Pyaar Kitna
Super 30 (film)
Family of Thakurganj
Batla House
Jhootha Kahin Ka
Judgementall Hai Kya
Chicken Curry Law
Arjun Patiala
Jabariya Jodi
Pranaam
The Sky Is Pink
Mission Mangal
Saaho
Dream Girl (2019 film)
Section 375
The Zoya Factor (film)
Pal Pal Dil Ke Paas
Prassthanam
P Se Pyaar F Se Faraar
Ghost (2019 film)
Bala (2019 film)
Yaaram (2019 film)
Housefull 4
Saand Ki Aankh
```

Made in China (2019 film)
Ujda Chaman
Bypass Road (film)
Satellite Shankar
Jhalki
Marjaavaan
Motichoor Chaknachoor
Keep Safe Distance (film)
Pagalpanti (2019 film)
Ramprasad Ki Tehrvi
Yeh Saali Aashiqui
Dil Bechara
Pati Patni Aur Woh (2019 film)
Commando 3 (film)
Mardaani 2
Dabangg 3
Good Newwz
Kaalakaandi
Vodka Diaries
My Birthday Song
Nirdosh
Mukkabaaz
Union Leader (film)
Love per Square Foot
Aiyaary
Padmaavat
Kuchh Bheege Alfaaz
Jaane Kyun De Yaaron
Veerey Ki Wedding
Sonu Ke Titu Ki Sweety
Hate Story 4
Dil Juunglee
3 Storeys
Raid (2018 film)
Hichki
Missing (2018 film)
Baaghi 2
October (2018 film)
Mercury (film)
Beyond the Clouds (2017 film)
Nanu Ki Jaanu
Daas Dev
Omerta (film)
Raazi
Hope Aur Hum
High Jack (film)
Khajoor Pe Atke
Parmanu: The Story of Pokhran
Bioscopewala
Bhavesh Joshi Superhero
Phamous
Race 3
Dhadak
Sanju
Saheb Biwi Aur Gangster 3
Nawabzaade
Mulk (film)
Brij Mohan Amar Rahe
Karwaan
Gold (2018 film)
Satyameva Jayate (2018 film)
Happy Phirr Bhag Jayegi
Genius (2018 Hindi film)
Yamla Pagla Deewana: Phir Se
Stree (2018 film)

Paltan (film)
Laila Majnu (2018 film)
Gali Guleiyan
Halkaa
Pataakha
Batti Gul Meter Chalu
Love Sonia
Manto (2018 film)
Ishqeria
Manmarziyaan
Mitron
Sui Dhaaga
Baazaar
Jalebi (film)
FryDay
Tumbbad
Helicopter Eela
Namaste England
Andhadhun
Badhaai Ho
5 Weddings
Kaashi in Search of Ganga
Dassehra
The Journey of Karma
Jack and Dil
Mohalla Assi
Pihu
Bhaiaji Superhit
Rajma Chawal
Zero (2018 film)
Simmba
Tiger Zinda Hai
Golmaal Again
Judwaa 2
Ok Jaanu
Coffee with D
Kaabil
Raees (film)
Thugs of Hindostan
Hind Ka Napak Ko Jawab: MSG Lion Heart 2
Running Shaadi
The Ghazi Attack
Irada (2017 film)
Rangoon (2017 Hindi film)
Wedding Anniversary
Jeena Isi Ka Naam Hai (film)
Badrinath Ki Dulhania
Trapped (2016 Hindi film)
Aa Gaya Hero
Mantra (2016 film)
Phillauri (film)
Machine (2017 film)
Bhanwarey
Anaarkali of Aarah
Naam Shabana
Hotel Salvation
Begum Jaan
Noor (film)
Ek Thi Rani Aisi Bhi
Maatr
Sarkar 3
Jattu Engineer
Half Girlfriend (film)
Meri Pyaari Bindu
Sachin: A Billion Dreams

Hindi Medium
Sweetiee Weds NRI
Dobaara: See Your Evil
Flat 211
Dear Maya
Raabta (film)
Behen Hogi Teri
Tubelight (2017 Hindi film)
Bank Chor
Ek Haseena Thi Ek Deewana Tha
Jagga Jasoos
Shab (film)
Lipstick Under My Burkha
Bachche Kachche Sachche
G Kutta Se
Indu Sarkar
Jab Harry Met Sejal
Munna Michael
Toilet: Ek Prem Katha
Mubarakan
Guest iin London
Bareilly Ki Barfi
Babumoshai Bandookbaaz
Yadvi – The Dignified Princess
Qaidi Band
Shubh Mangal Saavdhan
Raag Desh (film)
Daddy (2017 film)
Simran (film)
Lucknow Central
A Gentleman
Viceroy's House (film)
Patel Ki Punjabi Shaadi
Newton (film)
The Final Exit
Bhoomi (film)
Haseena Parkar
JD (film)
Haraamkhor
Poster Boys
Mom (film)
Chef (2017 film)
Ranchi Diaries
Babuji Ek Ticket Bambai
Rukh (film)
Secret Superstar
Aval (2017 film)
Ribbon (film)
Shaadi Mein Zaroor Aana
Jia Aur Jia
Qarib Qarib Singlle
Aksar 2
Panchlait
Tumhari Sulu
Julie 2
Kadvi Hawa
Firangi
Fukrey Returns
Monsoon Shootout
Ajji
Chalk n Duster
Rebellious Flower
Saankal
Airlift (film)
Sallu Ki Shaadi

Irudhi Suttru
Kyaa Kool Hain Hum 3
Mastizaade
Dil Jo Na Keh Saka
Jugni (2016 film)
Ghayal: Once Again
Fitoor
Sanam Re
Direct Ishq
Ishq Forever
Loveshhuda
Neerja
Aligarh (film)
Bollywood Diaries
Love Shagun
Tere Bin Laden: Dead or Alive
Jai Gangaajal
Kapoor & Sons
Rocky Handsome
Love Games (film)
Fan (film)
Nil Battey Sannata
Laal Rang
Baaghi (2016 film)
Global Baba
Shortcut Safari
The Blueberry Hunt
Santa Banta Pvt Ltd
Traffic (2016 film)
1920: London
Azhar (film)
Buddha in a Traffic Jam
Murari the Mad Gentleman
Dear Dad (film)
Phobia (2016 film)
Sarbjit (film)
Veerappan (2016 film)
Do Lafzon Ki Kahani (film)
Te3n
Udta Punjab
Khel Toh Ab Shuru Hoga
Luv U Alia
7 Hours to Go
Raman Raghav 2.0
Junooniyat
Rough Book
Dhanak
Fredrick (film)
Dil Toh Deewana Hai
Sultan (2016 film)
Brahman Naman
Great Grand Masti
Waiting (2015 film)
Ishq Click
M Cream
Madaari
Dishoom
Fever (2016 film)
Rustom (film)
Hai Apna Dil Toh Awara
Mohenjo Daro (film)
Happy Bhag Jayegi
Waarrior Savitri
A Flying Jatt
Baar Baar Dekho

Freaky Ali
Raaz: Reboot
Pink (2016 film)
Parched
Banjo (2016 film)
Days of Tafree
Wah Taj
Island City (2015 film)
Ek Kahani Julie Ki
M.S. Dhoni: The Untold Story
MSG: The Warrior Lion Heart
Devi (2016 film)
Motu Patlu: King of Kings
Anna (2016 film)
Fuddu
Saat Uchakkey
Beiimaan Love
Umrika
Shivaay
Dongari Ka Raja
Ae Dil Hai Mushkil
Tum Bin II
Rock On 2
Force 2
Dear Zindagi
Befikre
Moh Maya Money
Shorgul
Saansein
Ki & Ka
Wajah Tum Ho
Dangal (film)
Mirzya (film)
Prem Ratan Dhan Payo
Tanu Weds Manu: Returns
Tanu Weds Manu: Returns
31st October (film)
ABCD 2
Dilwale (2015 film)
Bajrangi Bhaijaan
Bajirao Mastani
Welcome Back (film)
Baby (2015 Hindi film)
Singh Is Bliing
Gabbar Is Back
Gabbar Is Back
Mumbai Can Dance Saala
Alone (2015 Hindi film)
Tevar
Sharafat Gayi Tel Lene
Dolly Ki Doli
Hawaizaada
Khamoshiyan
Rahasya
Jai Jawaan Jai Kisaan (film)
Shamitabh
Roy (film)
Badlapur (film)
Crazy Cukkad Family
Take It Easy (2015 film)
Qissa (film)
Ab Tak Chhappan 2
Dum Laga Ke Haisha
Dirty Politics (film)
MSG: The Messenger

Badmashiyaan
Coffee Bloom
Hey Bro
NH10 (film)
Hunterrr
Detective Byomkesh Bakshy!
Luckhnowi Ishq
Court (film)
Margarita with a Straw
Ek Paheli Leela
Barkhaa
Mr. X (2015 film)
NH-8 Road to Nidhivan
Dilliwali Zaalim Girlfriend
Dharam Sankat Mein
Kaagaz Ke Fools
Kuch Kuch Locha Hai
Piku
Bombay Velvet
I Love Desi
Dil Dhadakne Do
Welcome 2 Karachi
P Se PM Tak
Mere Genie Uncle
Hamari Adhuri Kahani
Miss Tanakpur Haazir Ho
Uvaa
Ishq Ke Parindey
Ishqedarriyaan
Sabki Bajegi Band
Masaan
Guddu Rangeela
Bezubaan Ishq
Aisa Yeh Jahaan
Second Hand Husband
I Love NY (2015 film)
Main Hoon Part-Time Killer
Kaun Kitne Paani Mein
Drishyam (2015 film)
Bangistan
All Is Well (2015 film)
Brothers (2015 film)
Gour Hari Dastaan
Manjhi – The Mountain Man
Thoda Lutf Thoda Ishq
Phantom (2015 film)
Hero (2015 Hindi film)
Sorry Daddy
Talvar (film)
Katti Batti
Meeruthiya Gangsters
MSG-2 The Messenger
Calendar Girls (2015 film)
Bhaag Johnny
Jazbaa
Bumper Draw
Chinar Daastaan-E-Ishq
Kis Kisko Pyaar Karoon
Pyaar Ka Punchnama 2
Wedding Pullav
Shaandaar
Titli (2014 film)
Guddu Ki Gun
The Silent Heroes
Ranbanka

Sholay
Dedh Ishqiya
Karle Pyaar Karle
Om-Dar-B-Dar
Paranthe Wali Gali
Strings of Passion
Gunday
Jai Ho (film)
Hasee Toh Phasee
Heartless (2014 film)
Ya Rab
Darr @ the Mall
One by Two (2014 film)
Babloo Happy Hai
Highway (2014 Hindi film)
Gulabi Gang (film)
Shaadi Ke Side Effects
Gulaab Gang
Queen (2014 film)
Bewakoofiyaan
Total Siyapaa
Karar: The Deal
Lakshmi (2014 film)
Ragini MMS 2
Dishkiyaoon
Ankhon Dekhi
Gang of Ghosts
Anuradha (2014 film)
W (2014 film)
O Teri
Honour Killing (film)
2 States (2014 film)
Jal (film)
Bhoothnath Returns
Main Tera Hero
Lucky Kabootar
Station (2014 film)
Youngistaan
Samrat & Co.
Purani Jeans
Kya Dilli Kya Lahore
Koyelaanchal
Manjunath (film)
Dekh Tamasha Dekh
Mastram
The Xposé
Children of War (2014 film)
Hawaa Hawaai
Kahin Hai Mera Pyar
Kuku Mathur Ki Jhand Ho Gayi
Heropanti
CityLights (2014 film)
Filmistaan
Revolver Rani
Kaanchi: The Unbreakable
Machhli Jal Ki Rani Hai
Khwaabb
Yeh Hai Bakrapur
Ek Villain
Bobby Jasoos
Humshakals
Humpty Sharma Ki Dulhania
Hate Story 2
Lekar Hum Deewana Dil
Riyasat (film)

Amit Sahni Ki List
Holiday: A Soldier Is Never Off Duty
Fugly (film)
Bazaar E Husn
Pizza (2014 film)
Entertainment (2014 film)
Singham Returns
Mardaani
Raja Natwarlal
Mary Kom (film)
Creature 3D
Finding Fanny
Khoobsurat (2014 film)
3 A.M. (2014 film)
Kick (2014 film)
Mad About Dance
Mumbhai Connection
Life Is Beautiful (2014 film)
Desi Kattey
Haider (film)
Bang Bang!
Spark (2014 film)
Daawat-e-Ishq
Balwinder Singh Famous Ho Gaya
Jigariyaa
Tamanchey
Mumbai 125 KM
Meinu Ek Ladki Chaahiye
Chaarfutiya Chhokare
Sonali Cable
Happy New Year (2014 film)
Super Nani
Ekkees Toppon Ki Salaami
Rang Rasiya
The Shaukeens
Roar: Tigers of the Sundarbans
A Decent Arrangement
Gollu Aur Pappu
Titoo MBA
Kill Dil
Ungli
Happy Ending (film)
Zed Plus
Zid (2014 film)
Action Jackson (2014 film)
Bhopal: A Prayer for Rain
Mumbai Delhi Mumbai
Badlapur Boys
Main Aur Mr. Riight
Ugly (film)
PK (film)
Dehraadun Diary
Matru Ki Bijlee Ka Mandola
Sulemani Keeda
Inkaar (2013 film)
Gangoobai
Akaash Vani
Main Krishna Hoon
Race 2
Bandook
Listen... Amaya
Special 26
Murder 3
Zila Ghaziabad
Kai Po Che!

Bloody Isshq
Saare Jahaan Se Mehnga
3G (film)
Mere Dad Ki Maruti
Jolly LLB
Saheb Biwi Aur Gangster Returns
Mai (2013 film)
Vishwaroopam
Rangrezz
Aashiqui 2
Ek Thi Daayan
Himmatwala (2013 film)
Nautanki Saala!
Jayantabhai Ki Luv Story
Commando: A One Man Army
Bombay Talkies (film)
Go Goa Gone
Gippi
Aurangzeb (film)
Ishkq in Paris
Zindagi 50-50
Yeh Jawaani Hai Deewani
Fukrey
Chhota Bheem and the Throne of Bali
Raanjhanaa
Ankur Arora Murder Case
Shortcut Romeo
Ghanchakkar (film)
Hum Hai Raahi Car Ke
Policegiri
Bhaag Milkha Bhaag
Sixteen (2013 Indian film)
B.A. Pass
Enemmy
Issaq
Bajatey Raho
Luv U Soniyo
Nasha (film)
Chor Chor Super Chor
Calapor (film)
Love in Bombay
D-Day (2013 film)
Siddharth (2013 film)
Once Upon ay Time in Mumbai Dobaara!
Madras Cafe
Satyagraha (film)
Shuddh Desi Romance
Ramaiya Vastavaiya
Chennai Express
Grand Masti
John Day (film)
Horror Story (film)
Phata Poster Nikhla Hero
Ship of Theseus (film)
The Lunchbox
Baat Bann Gayi
Boss (2013 Hindi film)
Shahid (film)
Mickey Virus
Satya 2
Rajjo
Maazii
Sooper Se Ooper
Prague (2013 film)
Wake Up India

Super Model (film)
Gori Tere Pyaar Mein
What the Fish
Jackpot (2013 film)
Table No. 21
Bullett Raja
Dhoom 3
Chashme Baddoor (2013 film)
Lootera
War Chhod Na Yaar
Chaalis Chauraasi
Ghost (2012 film)
Sadda Adda
Singh Saab the Great
Goliyon Ki Raasleela Ram-Leela
Agneepath (2012 film)
Ek Main Aur Ekk Tu
Ekk Deewana Tha
?: A Question Mark
Jodi Breakers
Tere Naal Love Ho Gaya
Staying Alive (2012 film)
Paan Singh Tomar (film)
Kahaani
Zindagi Tere Naam
Agent Vinod (2012 film)
Blood Money (2012 film)
Bumboo
Valentine's Night
Married 2 America
Chaar Din Ki Chandni
Bittoo Boss
Vicky Donor
Hate Story
Tezz
Dangerous Ishhq
Ishaqzaade
Department (film)
Fatso!
Arjun: The Warrior Prince
Life Ki Toh Lag Gayi
Shanghai (2012 film)
Ferrari Ki Sawaari
Teri Meri Kahaani (film)
Mr. Bhatti on Chutti
Yeh Khula Aasmaan
Rakhtbeej
Gangs of Wasseypur
Gangs of Wasseypur – Part 2
Cocktail (2012 film)
Gattu
Kyaa Super Kool Hain Hum
Maximum (film)
Paanch Ghantey Mien Paanch Crore
Ek Tha Tiger
Challo Driver
Shirin Farhad Ki Toh Nikal Padi
Joker (2012 film)
Aalaap (film)
Mere Dost Picture Abhi Baki Hai
Krishna Aur Kans
From Sydney with Love
Jalpari: The Desert Mermaid
Barfi!
Heroine (2012 film)

Chal Pichchur Banate Hain
Kismat Love Paisa Dilli
Jeena Hai Toh Thok Daal
OMG – Oh My God!
Aiyyaa
Chittagong (film)
Bhoot Returns
Delhi Safari
Chakravyuh (2012 film)
Student of the Year
Ajab Gazabb Love
Rush (2012 film)
1920: The Evil Returns
Sons of Ram
Ata Pata Laapata
Jab Tak Hai Jaan
Talaash: The Answer Lies Within
Login (film)
Son of Sardaar
Cigarette Ki Tarah
Dabangg 2
Players (2012 film)
Housefull 2
Bol Bachchan
English Vinglish
Impatient Vivek
Yamla Pagla Deewana
Mumbai Mast Kallander
Dhobi Ghat (film)
Turning 30
Hostel (2011 film)
Dil Toh Baccha Hai Ji
United Six
Utt Pataang
Patiala House (film)
7 Khoon Maaf
Tanu Weds Manu
F.A.L.T.U
Memories in March
Thank You (2011 film)
Angel (2011 film)
Happy Husbands (2011 film)
Teen Thay Bhai
Dum Maaro Dum (film)
Shor in the City
Zokkomon
Chalo Dilli
Aashiqui.in
Satrangee Parachute
Monica (film)
I Am (2010 Indian film)
Naughty @ 40
Haunted – 3D
Love U...Mr. Kalakaar!
Ragini MMS
Stanley Ka Dabba
Shagird (2011 film)
404 (film)
Shaitan (film)
Pyaar Ka Punchnama
Bheja Fry 2
Always Kabhi Kabhi
Double Dhamaal
Bbuddah... Hoga Terra Baap
Delhi Belly (film)

Murder 2
Chillar Party
Zindagi Na Milegi Dobara
Dear Friend Hitler
I Am Kalam
Bin Bulaye Baraati
Kucch Luv Jaisaa
Singham
Khap (film)
Bubble Gum (film)
Shabri
Phhir
Aarakshan
Chatur Singh Two Star
Sahi Dhandhe Galat Bande
Bodyguard (2011 Hindi film)
Yeh Dooriyan
Not a Love Story (2011 film)
Hum Tum Shabana
Mummy Punjabi
Mere Brother Ki Dulhan
Mausam (2011 film)
U R My Jaan
Force (2011 film)
Saheb Biwi Aur Gangster
Tere Mere Phere
Breakaway (2011 film)
Chargesheet (film)
Love Breakups Zindagi
Mujhse Fraaandship Karoge
Aazaan
Ra.One
Rockstar (2011 film)
Miley Naa Miley Hum
Tell Me O Kkhuda
Damadamm!
Ladies vs Ricky Bahl
Desi Boyz
Game (2011 film)
No One Killed Jessica
Rascals (2011 film)
The Dirty Picture
Pyaar Impossible!
Chance Pe Dance
My Friend Pinto
Veer (2010 film)
Striker (2010 film)
Rann (film)
Ishqiya
Road to Sangam
Jo Hum Chahein
Click (2010 film)
Toh Baat Pakki!
My Name Is Khan
Teen Patti (film)
Karthik Calling Karthik
Don 2
Rokkk
Aakhari Decision
Right Yaaa Wrong
Sukhmani: Hope for Life
Thanks Maa
Na Ghar Ke Na Ghaat Ke
Trump Card (film)
Shaapit

Hum Tum Aur Ghost
Well Done Abba
Tum Milo Toh Sahi
Jaane Kahan Se Aayi Hai
Prem Kaa Game
Sadiyaan
The Japanese Wife
Paathshaala
Phoonk 2
Lahore (film)
Apartment (film)
City of Gold (2010 film)
Chase (2010 film)
Housefull (2010 film)
Mittal v/s Mittal
It's a Wonderful Afterlife
Prince (2010 film)
Raavan
Bumm Bumm Bole
Kushti (film)
Kites (film)
Love Sex Aur Dhokha
Milenge Milenge
Ek Second... Jo Zindagi Badal De?
Mr. Singh Mrs. Mehta
Lamhaa
Khatta Meetha (2010 film)
Tere Bin Laden
Udaan (2010 film)
Once Upon a Time in Mumbaai
Help (film)
Peepli Live
Lafangey Parindey
Hello Darling
Antardwand
Aashayein
Soch Lo
Dabangg
Khichdi: The Movie
Red Alert: The War Within
Life Express (2010 film)
The Film Emotional Atyachar
Hisss
Crook (film)
Do Dooni Chaar
Aakrosh (2010 film)
Ramayana: The Epic
Knock Out (2010 film)
Jhootha Hi Sahi
Guzaarish (film)
Allah Ke Banday
Break Ke Baad
Khuda Kasam
Phas Gaye Re Obama
Malik Ek
A Flat (film)
No Problem (2010 film)
Band Baaja Baaraat
Kaalo
Mirch
Tees Maar Khan (2010 film)
Isi Life Mein
Toonpur Ka Super Hero
Tera Kya Hoga Johnny
Ramaa: The Saviour

I Hate Luv Storys
Dulha Mil Gaya
Anjaana Anjaani
Dunno Y... Na Jaane Kyon
Pankh
Action Replayy
3 Idiots
Luck by Chance
Love Aaj Kal
Wanted (2009 film)
Delhi-6
Raaz: The Mystery Continues
Aasma: The Sky Is the Limit
Ajab Prem Ki Ghazab Kahani
Chal Chala Chal
Billu
The Stoneman Murders
Kisse Pyaar Karoon
Dhoondte Reh Jaaoge
Karma Aur Holi
Victory (2009 film)
Kaminey
Jai Veeru
Little Zizou
Gulaal (film)
Aloo Chaat (film)
Barah Aana
Firaaq
Aa Dekhen Zara
99 (2009 film)
Ek: The Power of One
Ek Se Bure Do
Sikandar (2009 film)
Zor Lagaa Ke...Haiya!
Paying Guests
New York (2009 film)
Sankat City
Shortkut
Luck (2009 film)
Life Partner
Daddy Cool (2009 Hindi film)
Kisaan
Yeh Mera India
Aagey Se Right
Chintu Ji
Quick Gun Murugun
Fox (film)
Baabarr
Phir Kabhi
Vaada Raha
Dil Bole Hadippa!
What's Your Raashee?
Acid Factory
All the Best: Fun Begins
Wake Up Sid
Main Aurr Mrs Khanna
Blue (2009 film)
Fruit and Nut (film)
Aladin (film)
London Dreams
Jail (2009 film)
Tum Mile
Kurbaan (2009 film)
De Dana Dan
Paa (film)

Rocket Singh: Salesman of the Year
Raat Gayi Baat Gayi?
Accident on Hill Road
Chandni Chowk to China
Dostana (2008 film)
Race (2008 film)
Rab Ne Bana Di Jodi
Ghajini (2008 film)
Singh Is Kinng
Golmaal Returns
Jodhaa Akbar
Bachna Ae Haseeno
Bhoothnath
Sarkar Raj
Halla Bol
Humne Jeena Seekh Liya
Bombay to Bangkok
Tulsi (film)
Sunday (2008 film)
One Two Three
Krazzy 4
U Me Aur Hum
Sirf (film)
Tashan (film)
Anamika (2008 film)
Jimmy (2008 film)
Jannat (film)
Don Muthu Swami
Woodstock Villa
Mere Baap Pehle Aap
Summer 2007
De Taali
Haal-e-Dil
Thodi Life Thoda Magic
Thoda Pyaar Thoda Magic
Via Darjeeling
Kismat Konnection
Love Story 2050
Contract (2008 film)
Mission Istaanbul
Money Hai Toh Honey Hai
God Tussi Great Ho
Mumbai Meri Jaan
Maan Gaye Mughal-e-Azam
Rock On!!
C Kkompany
Chamku
Mukhbiir
Tahaan
1920 (film)
The Last Lear
Welcome to Sajjanpur
Saas Bahu Aur Sensex
Hari Puttar: A Comedy of Terrors
Drona (2008 film)
Hello (2008 film)
Karzzzz
Heroes (2008 film)
Roadside Romeo
Ek Vivaah... Aisa Bhi
Deshdrohi
Fashion (2008 film)
Dasvidaniya
Yuvvraaj
Oye Lucky! Lucky Oye!

Oh My God (2008 film)
Sorry Bhai!
1971 (2007 film)
Meerabai Not Out
Wafa: A Deadly Love Story
Gumnaam – The Mystery
Dil Kabaddi
Aag (2007 film)
Aap Kaa Surroor
Aggar (film)
Anwar (2007 film)
Aaja Nachle
Apne
Awarapan
Bheja Fry (film)
Bhool Bhulaiyaa
Bhram
Big Brother (2007 film)
68 Pages
Aur Pappu Paas Ho Gaya
Apna Asmaan
Black Friday (2007 film)
The Blue Umbrella (2005 film)
Blood Brothers (2007 Indian film)
Buddha Mar Gaya
Cash (2007 film)
Cheeni Kum
Chhodon Naa Yaar
Darling (2007 Indian film)
Chak De! India
Dhamaal
Goal (2007 Hindi film)
Dharm (film)
Bombay to Goa (2007 film)
Dhokha
Dil Dosti Etc
Dhol (film)
Ek Chalis Ki Last Local
Familywala
Gandhi My Father
Don't Stop Dreaming
Dus Kahaniyaan
Eklavya: The Royal Guard
Go (2007 film)
Gauri: The Unborn
Heyy Babyy
Guru (2007 film)
Honeymoon Travels Pvt. Ltd.
Jahan Jaaeyega Hamen Paaeyega
Jhoom Barabar Jhoom
Jab We Met
Good Boy Bad Boy
Fool & Final
Khoya Khoya Chand
Johnny Gaddaar
Just Married (2007 film)
Kya Love Story Hai
Kudiyon Ka Hai Zamana
Hastey Hastey
Laaga Chunari Mein Daag
Life in a... Metro
Loins of Punjab Presents
Manorama Six Feet Under
Marigold (2007 film)
MP3: Mera Pehla Pehla Pyaar

Hattrick (film)
Naqaab
Mumbai Salsa
The Namesake (film)
Namastey London
My Bollywood Bride
Nehlle Pe Dehlla
Nishabd
No Smoking (2007 film)
Om Shanti Om
Partner (2007 film)
Raqeeb
Nanhe Jaisalmer
Life Mein Kabhie Kabhiee
Risk (2007 film)
Shakalaka Boom Boom
Red Swastik
Salaam-e-Ishq: A Tribute to Love
Saawariya
Sarhad Paar
Say Salaam India
Red: The Dark Side
Shootout at Lokhandwala
Strangers (2007 Hindi film)
Swami (2007 film)
Taare Zameen Par
Ta Ra Rum Pum
Speed (2007 film)
Traffic Signal (film)
The Train (2007 film)
Showbiz (film)
Sunglass (film)
Welcome (2007 film)
36 China Town
Zamaanat
Aap Ki Khatir (2006 film)
Ahista Ahista (2006 film)
Aksar
Alag
Anthony Kaun Hai?
Apna Sapna Money Money
Ankahee (2006 film)
Yatra (2007 film)
Baabul (2006 film)
Aisa Kyon Hota Hai?
Adharm (2006 film)
Being Cyrus
Banaras (2006 film)
Bas Ek Pal
Bhagam Bhag
Chingaari
Bhoot Unkle
Chand Ke Paar Chalo (film)
Aryan: Unbreakable
Chup Chup Ke
Corporate (2006 film)
Darna Zaroori Hai
Deadline: Sirf 24 Ghante
Dil Diya Hai
Darwaaza Bandh Rakho
Eight: The Power of Shani
Dhoom 2
Dor (film)
Don (2006 Hindi film)
Family (2006 film)

Fanaa (2006 film)
Gangster (2006 film)
Golmaal: Fun Unlimited
Devaki (2005 film)
Fight Club – Members Only
Dharti Kahe Pukar Ke (2006 film)
Humko Deewana Kar Gaye
Humko Tumse Pyaar Hai
I See You (2006 film)
Jaane Hoga Kya
Jaan-E-Mann
Jawani Diwani: A Youthful Joyride
Holiday (2006 film)
Iqraar by Chance
Khosla Ka Ghosla
Kabhi Alvida Naa Kehna
Love Ke Chakkar Mein
Lage Raho Munna Bhai
Kabul Express
Jigyaasa
Krrish
Malamaal Weekly
Omkara (2006 film)
Pyaar Ke Side Effects
Naksha
Phir Hera Pheri
Pyare Mohan
Mere Jeevan Saathi (2006 film)
Prateeksha
Sacred Evil – A True Story
Rang De Basanti
Shaadi Karke Phas Gaya Yaar
Sandwich (2006 film)
Shaadi Se Pehle
Saawan... The Love Season
Shiva (2006 film)
Souten: The Other Woman
Shikhar (film)
Children of Heaven
Tathastu
The Killer (2006 film)
Umrao Jaan (2006 film)
Taxi No. 9211
Teesri Aankh: The Hidden Camera
Vivah
Utthaan
Waris Shah: Ishq Daa Waaris
Woh Lamhe...
Yun Hota Toh Kya Hota
Umar (film)
Zindaggi Rocks
Tom Dick and Harry (2006 film)
Aashiq Banaya Aapne
Anjaane (2005 film)
Apaharan
Bachke Rehna Re Baba
7½ Phere
Barsaat (2005 film)
Bewafaa (2005 film)
Black (2005 film)
Zinda (film)
Bluffmaster!
99.9 FM (film)
Bhola in Bollywood
Bhagmati (2005 film)

Blackmail (2005 film)
Bunty Aur Babli
Chaahat – Ek Nasha
Chetna: The Excitement
Chand Sa Roshan Chehra
Chocolate (2005 film)
D (film)
Deewane Huye Paagal
Bullet: Ek Dhamaka
Chehraa
Dil Jo Bhi Kahey...
Dosti: Friends Forever
Classic – Dance of Love
Dus
Elaan (2005 film)
Fareb (2005 film)
Ek Ajnabee
Fun – Can Be Dangerous Sometimes
Ek Khiladi Ek Haseena (film)
Double Cross (2005 film)
Dreams (2006 film)
Home Delivery
Garam Masala (2005 film)
Iqbal (film)
Jai Chiranjeeva
Hazaaron Khwaishein Aisi
Insan
Jo Bole So Nihaal (film)
Karam (film)
Kaal (2005 film)
Kalyug (2005 film)
Kasak (2005 film)
Hanuman (2005 film)
James (2005 film)
Kisna: The Warrior Poet
Koi Aap Sa
Khamoshh... Khauff Ki Raat
Jurm (2005 film)
Kuchh Meetha Ho Jaye
Kyaa Kool Hai Hum
Lucky: No Time for Love
Main Aisa Hi Hoon
Kyon Ki
Main Meri Patni Aur Woh
Maine Gandhi Ko Nahin Mara
Maine Pyaar Kyun Kiya?
Mangal Pandey: The Rising
Koi Mere Dil Mein Hai
Mr Prime Minister
My Brother…Nikhil
My Wife's Murder
Naina (2005 film)
Neal 'n' Nikki
No Entry
Padmashree Laloo Prasad Yadav
Page 3 (film)
Morning Raga
Parineeta (2005 film)
Pehchaan: The Face of Truth
Paheli
Rog
Pyaar Mein Twist
Salaam Namaste
Revati (film)
Sarkar (2005 film)

Sehar
Shabd (film)
Shabnam Mausi
Sheesha (2005 film)
Ramji Londonwaley
Silsiilay
Socha Na Tha
Taj Mahal: An Eternal Love Story
Tango Charlie
The Film
Vaada (film)
Vaah! Life Ho Toh Aisi!
Ssukh
Shaadi No. 1
Viruddh... Family Comes First
Waqt: The Race Against Time
Vidyaarthi
Yakeen (2005 film)
Zeher
Veer-Zaara
Main Hoon Na
Zameer: The Fire Within
Mujhse Shaadi Karogi
Dhoom
Khakee
Hum Tum
Hulchul (2004 film)
Murder (2004 film)
Yuva
Aitraaz
Aetbaar
Ab Tumhare Hawale Watan Saathiyo
Aan: Men at Work
Bardaasht
Chameli (film)
Agnipankh
Asambhav
Charas (2004 film)
Deewaar (2004 film)
Dev (2004 film)
Dil Maange More
Dil Ne Jise Apna Kahaa
Dobara
Aabra Ka Daabra
Dil Bechara Pyaar Ka Maara
Gayab
Fida
Garv: Pride & Honour
Ek Se Badhkar Ek (2004 film)
Ek Hasina Thi (film)
Girlfriend (2004 film)
Hatya (2004 film)
Hava Aney Dey
Hawas (2004 film)
Hyderabad Blues 2
Julie (2004 film)
Kaun Hai Jo Sapno Mein Aaya
Inteqam: The Perfect Game
Kis Kis Ki Kismat
Insaaf: The Justice
I Proud to Be an Indian
Khamosh Pani
Kismat (2004 film)
Lakeer – Forbidden Lines
Krishna Cottage

Kyun! Ho Gaya Na...
Madhoshi
Lakshya (film)
Ishq Hai Tumse
Maqbool
Masti (2004 film)
Meenaxi: A Tale of Three Cities
Musafir (2004 film)
Mughal-e-Azam
Muskaan
Meri Biwi Ka Jawaab Nahin
Naach (2004 film)
Netaji Subhas Chandra Bose: The Forgotten Hero
Paap
Phir Milenge
Plan (film)
Police Force: An Inside Story
Paisa Vasool
Popcorn Khao! Mast Ho Jao
Rakht
Raincoat (film)
Rudraksh (film)
Shaadi Ka Laddoo
Run (2004 film)
Rok Sako To Rok Lo
Suno Sasurjee
Swades
Taarzan: The Wonder Car
Nothing but Life
Shart: The Challenge
Tumsa Nahin Dekha: A Love Story
Vaastu Shastra (film)
Yeh Lamhe Judaai Ke
Sheen (film)
Dude Where's the Party?
Thoda Tum Badlo Thoda Hum
Koi... Mil Gaya
Kal Ho Naa Ho
Shukriya: Till Death Do Us Apart
Chalte Chalte (2003 film)
The Hero: Love Story of a Spy
Baghban (2003 film)
Main Prem Ki Diwani Hoon
LOC Kargil
Border (1997 film)
Munna Bhai M.B.B.S.
Qayamat: City Under Threat
88 Antop Hill
3 Deewarein
Aanch
Aapko Pehle Bhi Kahin Dekha Hai
Bhoot (film)
Boom (film)
Aaj Ka Andha Kanoon
Andaaz
Andaaz
Armaan (2003 film)
Chori Chori (2003 film)
Calcutta Mail
Baaz: A Bird in Danger
Basti (film)
Magic Magic 3D
Dil Ka Rishta
Darna Mana Hai
Dhoop

Dhund (2003 film)
Chura Liyaa Hai Tumne
The Bypass
Dum (2003 Hindi film)
Dil Pardesi Ho Gayaa
Ek Alag Mausam
Footpath (2003 film)
Escape from Taliban
Ek Din 24 Ghante
Gangaajal
Hawa (film)
Haasil
Ek Aur Ek Gyarah
Hungama (2003 film)
Green Card Fever
Flavors (film)
Indian Babu
Fun2shh... Dudes in the 10th Century
Inteha (2003 film)
Jaal: The Trap
Ishq Vishk
Hawayein
Jajantaram Mamantaram
Jism (2003 film)
Jhankaar Beats
Kagaar: Life on the Edge
Kash Aap Hamare Hote
Khel – No Ordinary Game
Janasheen
Kaise Kahoon Ke... Pyaar Hai
Khushi (2003 Hindi film)
Khwahish
Kucch To Hai
Kuch Naa Kaho
Main Madhuri Dixit Banna Chahti Hoon
Joggers' Park (film)
Market (2003 film)
Om (2003 film)
Out of Control (2003 film)
Mumbai Matinee
Matrubhoomi
Parwana (2003 film)
Pinjar (film)
Mumbai Se Aaya Mera Dost
Saaya (2003 film)
Samay: When Time Strikes
Nayee Padosan
Satta (film)
Sssshhh...
Praan Jaye Par Shaan Na Jaye
Raghu Romeo
Stumped (film)
Rules: Pyaar Ka Superhit Formula
Right Here Right Now (film)
Raja Bhaiya (film)
Tere Naam
Tujhe Meri Kasam
Talaash: The Hunt Begins...
Tehzeeb (2003 film)
The Pink Mirror
Yeh Dil
Xcuse Me
Raaz (2002 film)
Zameen (2003 film)
Waisa Bhi Hota Hai Part II

```
Devdas (2002 Hindi film)
Kaante
Hum Tumhare Hain Sanam
Aankhen (2002 film)
Saathiya (film)
Company (film)
Awara Paagal Deewana
```

In [60]:
```
movies.index
```

Out[60]:
```
Index(['Uri: The Surgical Strike', 'Battalion 609',
       'The Accidental Prime Minister (film)', 'Why Cheat India',
       'Evening Shadows', 'Soni (film)', 'Fraud Saiyaan', 'Bombairiya',
       'Manikarnika: The Queen of Jhansi', 'Thackeray (film)',
       ...
       'Raaz (2002 film)', 'Zameen (2003 film)', 'Waisa Bhi Hota Hai Part II',
       'Devdas (2002 Hindi film)', 'Kaante', 'Hum Tumhare Hain Sanam',
       'Aankhen (2002 film)', 'Saathiya (film)', 'Company (film)',
       'Awara Paagal Deewana'],
      dtype='object', name='movie', length=1500)
```

In [61]:
```
100 + marks_series
```

Out[61]:
```
ram      155
shyam    200
radha    180
Name: student result, dtype: int64
```

In [62]:
```
marks_series >= 100
```

Out[62]:
```
ram      False
shyam     True
radha    False
Name: student result, dtype: bool
```

In [63]:
```
# find actors who have done more than 20 movies
num_movies = movies.value_counts()
num_movies[num_movies > 20]
```

Out[63]:
```
Akshay Kumar        48
Amitabh Bachchan    45
Ajay Devgn          38
Salman Khan         31
Sanjay Dutt         26
Shah Rukh Khan      22
Emraan Hashmi       21
Name: lead, dtype: int64
```

In [64]:
```
movies.value_counts()
```

Out[64]:
```
Akshay Kumar         48
Amitabh Bachchan     45
Ajay Devgn           38
Salman Khan          31
Sanjay Dutt          26
                     ..
Sachin Tendulkar      1
Atmaram Bhende        1
Raima Sen             1
Aniket Vishwasrao     1
Attin Bhalla          1
Name: lead, Length: 566, dtype: int64
```

In [65]:
```
# Count number of day when I had more than 200 subs a day
subs[subs > 200].size
```

Out[65]:
```
59
```

In [66]:
```
#Write a Pandas program to add, subtract, multiple and divide two Pandas Series.

a = pd.Series([2, 4, 6, 8, 10])
b = pd.Series([1, 3, 5, 7, 10])

print(a+b)
print(a-b)
print(a*b)
print(a/b)
```

```
0     3
1     7
2    11
3    15
4    20
dtype: int64
0    1
1    1
2    1
3    1
4    0
dtype: int64
0      2
1     12
2     30
3     56
4    100
dtype: int64
0    2.000000
1    1.333333
2    1.200000
3    1.142857
4    1.000000
dtype: float64
```

In [67]:
```
#Write a Pandas program to compare the elements of the two Pandas Series.
#Sample Series: [2, 4, 6, 8, 10], [1, 3, 5, 7, 10]
# code here
a = pd.Series([2, 4, 6, 8, 10])
b = pd.Series([1, 3, 5, 7, 10])

print(a==b)
print(a<b)
print(a>b)
```

```
0    False
1    False
2    False
3    False
4     True
dtype: bool
0    False
1    False
2    False
3    False
4    False
dtype: bool
```

```
0     True
1     True
2     True
3     True
4    False
dtype: bool
```

what is DataFrame

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

In [68]:
```
import pandas as pd

data = {
  "calories": [420, 380, 390],
  "duration": [50, 40, 45]
}

df = pd.DataFrame(data)

print(df)
```

```
   calories  duration
0       420        50
1       380        40
2       390        45
```

In [69]:
```
print(df.loc[0]) #location 0
print(df.loc[1]) #location 1
print(df.loc[2]) #location 2
```

```
calories    420
duration     50
Name: 0, dtype: int64
calories    380
duration     40
Name: 1, dtype: int64
calories    390
duration     45
Name: 2, dtype: int64
```

In [ ]:
```
# iloc is for integer location
print(df.iloc[1]) #integer location in dataframe
```

In [70]:
```
print(df['calories'])
```

```
0    420
1    380
2    390
Name: calories, dtype: int64
```

In [71]:
```
print(df['duration'])
```

```
0    50
1    40
2    45
Name: duration, dtype: int64
```

In [72]:
```
print(df.loc[[0,1]]) # 2 rows
```

```
   calories  duration
0       420        50
```

```
1       380        40
```

In [73]:
```
df=pd.DataFrame(data,index=['day1','day2','day3'])
print(df)
```

```
      calories  duration
day1       420        50
day2       380        40
day3       390        45
```

In [74]:
```
print(df.loc['day1'])
```

```
calories    420
duration     50
Name: day1, dtype: int64
```

In [75]:
```
df[::2] # all type of indexes we can apply in dataframe same as series
```

Out[75]:

	calories	duration
day1	420	50
day3	390	45

If your data sets are stored in a file, Pandas can load them into a DataFrame.

In [76]:
```
import pandas as pd
dataset=pd.read_csv('auto-mpg.csv')
print(dataset)
```

```
      mpg  cylinders  displacement horsepower  weight  acceleration  \
0    18.0          8         307.0        130    3504          12.0
1    15.0          8         350.0        165    3693          11.5
2    18.0          8         318.0        150    3436          11.0
3    16.0          8         304.0        150    3433          12.0
4    17.0          8         302.0        140    3449          10.5
..    ...        ...           ...        ...     ...           ...
393  27.0          4         140.0         86    2790          15.6
394  44.0          4          97.0         52    2130          24.6
395  32.0          4         135.0         84    2295          11.6
396  28.0          4         120.0         79    2625          18.6
397  31.0          4         119.0         82    2720          19.4

     model year  origin                  car name
0            70       1  chevrolet chevelle malibu
1            70       1          buick skylark 320
2            70       1         plymouth satellite
3            70       1              amc rebel sst
4            70       1                ford torino
..          ...     ...                       ...
393          82       1            ford mustang gl
394          82       2                  vw pickup
395          82       1              dodge rampage
396          82       1                ford ranger
397          82       1                 chevy s-10

[398 rows x 9 columns]
```

In [77]:
```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 398 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype
```

```
---  ------        --------------  -----
 0   mpg           398 non-null    float64
 1   cylinders     398 non-null    int64
 2   displacement  398 non-null    float64
 3   horsepower    398 non-null    object
 4   weight        398 non-null    int64
 5   acceleration  398 non-null    float64
 6   model year    398 non-null    int64
 7   origin        398 non-null    int64
 8   car name      398 non-null    object
dtypes: float64(3), int64(4), object(2)
memory usage: 28.1+ KB
```

In [78]:
```
dataset.head() # print first 5 rows default
```

Out[78]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino

In [79]:
```
dataset.head(10) # print first 10 rows
```

Out[79]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino
5	15.0	8	429.0	198	4341	10.0	70	1	ford galaxie 500
6	14.0	8	454.0	220	4354	9.0	70	1	chevrolet impala
7	14.0	8	440.0	215	4312	8.5	70	1	plymouth fury iii
8	14.0	8	455.0	225	4425	10.0	70	1	pontiac catalina
9	15.0	8	390.0	190	3850	8.5	70	1	amc ambassador dpl

In [80]:
```
dataset.loc[15] #particular location
```

Out[80]:
```
mpg                          22
cylinders                     6
displacement                198
horsepower                   95
weight                     2833
acceleration               15.5
model year                   70
origin                        1
car name        plymouth duster
Name: 15, dtype: object
```

In [81]:
```
dataset.loc[5:15]
```

Out[81]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
5	15.0	8	429.0	198	4341	10.0	70	1	ford galaxie 500
6	14.0	8	454.0	220	4354	9.0	70	1	chevrolet impala
7	14.0	8	440.0	215	4312	8.5	70	1	plymouth fury iii
8	14.0	8	455.0	225	4425	10.0	70	1	pontiac catalina
9	15.0	8	390.0	190	3850	8.5	70	1	amc ambassador dpl
10	15.0	8	383.0	170	3563	10.0	70	1	dodge challenger se
11	14.0	8	340.0	160	3609	8.0	70	1	plymouth 'cuda 340
12	15.0	8	400.0	150	3761	9.5	70	1	chevrolet monte carlo
13	14.0	8	455.0	225	3086	10.0	70	1	buick estate wagon (sw)
14	24.0	4	113.0	95	2372	15.0	70	3	toyota corona mark ii
15	22.0	6	198.0	95	2833	15.5	70	1	plymouth duster

In [82]:
```
dataset.tail() #last 5 rows
```

Out[82]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
393	27.0	4	140.0	86	2790	15.6	82	1	ford mustang gl
394	44.0	4	97.0	52	2130	24.6	82	2	vw pickup

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
395	32.0	4	135.0	84	2295	11.6	82	1	dodge rampage
396	28.0	4	120.0	79	2625	18.6	82	1	ford ranger
397	31.0	4	119.0	82	2720	19.4	82	1	chevy s-10

In [83]:
```
dataset['mpg']
```

Out[83]:
```
0      18.0
1      15.0
2      18.0
3      16.0
4      17.0
       ...
393    27.0
394    44.0
395    32.0
396    28.0
397    31.0
Name: mpg, Length: 398, dtype: float64
```

In [84]:
```
dataset['mpg'].loc[4]
```

Out[84]:
```
17.0
```

In [85]:
```
dataset['mpg'].loc[10:15] # Endpoint not exculded in slicing
```

Out[85]:
```
10    15.0
11    14.0
12    15.0
13    14.0
14    24.0
15    22.0
Name: mpg, dtype: float64
```

In [86]:
```
dataset.shape  # shape is class attribute so () not provided
```

Out[86]:
```
(398, 9)
```

In [87]:
```
dataset.describe()  # without object string and ? are there in horse power and car name
```

Out[87]:

	mpg	cylinders	displacement	weight	acceleration	model year	origin
count	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000
mean	23.514573	5.454774	193.425879	2970.424623	15.568090	76.010050	1.572864
std	7.815984	1.701004	104.269838	846.841774	2.757689	3.697627	0.802055
min	9.000000	3.000000	68.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.500000	4.000000	104.250000	2223.750000	13.825000	73.000000	1.000000
50%	23.000000	4.000000	148.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	262.000000	3608.000000	17.175000	79.000000	2.000000
max	46.600000	8.000000	455.000000	5140.000000	24.800000	82.000000	3.000000

Parameters: **percentiles : *list-like of numbers, optional***

The percentiles to include in the output. All should fall between 0 and 1. The default is `[.25, .5, .75]`, which returns the 25th, 50th, and 75th percentiles.

include : *'all', list-like of dtypes or None (default), optional*

A white list of data types to include in the result. Ignored for `Series`. Here are the options:

- 'all' : All columns of the input will be included in the output.
- A list-like of dtypes : Limits the results to the provided data types. To limit the result to numeric types submit `numpy.number`. To limit it instead to object columns submit the `numpy.object` data type. Strings can also be used in the style of `select_dtypes` (e.g. `df.describe(include=['O'])`). To select pandas categorical columns, use `'category'`
- None (default) : The result will include all numeric columns.

exclude : *list-like of dtypes or None (default), optional,*

A black list of data types to omit from the result. Ignored for `Series`. Here are the options:

- A list-like of dtypes : Excludes the provided data types from the result. To exclude numeric types submit `numpy.number`. To exclude object columns submit the data type `numpy.object`. Strings can also be used in the style of `select_dtypes` (e.g. `df.describe(exclude=['O'])`). To exclude pandas categorical columns, use `'category'`
- None (default) : The result will exclude nothing.

In [88]:
```
dataset.describe(include="all") #unique top & freq only given in string not integer dat
```

Out[88]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	o
count	398.000000	398.000000	398.000000	398	398.000000	398.000000	398.000000	398.00
unique	NaN	NaN	NaN	94	NaN	NaN	NaN	
top	NaN	NaN	NaN	150	NaN	NaN	NaN	
freq	NaN	NaN	NaN	22	NaN	NaN	NaN	
mean	23.514573	5.454774	193.425879	NaN	2970.424623	15.568090	76.010050	1.57
std	7.815984	1.701004	104.269838	NaN	846.841774	2.757689	3.697627	0.80
min	9.000000	3.000000	68.000000	NaN	1613.000000	8.000000	70.000000	1.00
25%	17.500000	4.000000	104.250000	NaN	2223.750000	13.825000	73.000000	1.00
50%	23.000000	4.000000	148.500000	NaN	2803.500000	15.500000	76.000000	1.00

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	o
75%	29.000000	8.000000	262.000000	NaN	3608.000000	17.175000	79.000000	2.00
max	46.600000	8.000000	455.000000	NaN	5140.000000	24.800000	82.000000	3.00

In [89]:
```
import numpy as np
dataset.describe(include=[np.number])  # same as normal describe()
```

Out[89]:

	mpg	cylinders	displacement	weight	acceleration	model year	origin
count	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000
mean	23.514573	5.454774	193.425879	2970.424623	15.568090	76.010050	1.572864
std	7.815984	1.701004	104.269838	846.841774	2.757689	3.697627	0.802055
min	9.000000	3.000000	68.000000	1613.000000	8.000000	70.000000	1.000000
25%	17.500000	4.000000	104.250000	2223.750000	13.825000	73.000000	1.000000
50%	23.000000	4.000000	148.500000	2803.500000	15.500000	76.000000	1.000000
75%	29.000000	8.000000	262.000000	3608.000000	17.175000	79.000000	2.000000
max	46.600000	8.000000	455.000000	5140.000000	24.800000	82.000000	3.000000

In [90]:
```
dataset.describe(exclude=[np.number])
```

Out[90]:

	horsepower	car name
count	398	398
unique	94	305
top	150	ford pinto
freq	22	6

In [93]:
```
dataset[['mpg','cylinders']].describe()
```

Out[93]:

	mpg	cylinders
count	398.000000	398.000000
mean	23.514573	5.454774
std	7.815984	1.701004
min	9.000000	3.000000
25%	17.500000	4.000000
50%	23.000000	4.000000
75%	29.000000	8.000000
max	46.600000	8.000000

In [94]:
```
dataset.describe(percentiles=[.30,.45,.55])
```

Out[94]:

	mpg	cylinders	displacement	weight	acceleration	model year	origin
count	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000	398.000000
mean	23.514573	5.454774	193.425879	2970.424623	15.568090	76.010050	1.572864
std	7.815984	1.701004	104.269838	846.841774	2.757689	3.697627	0.802055
min	9.000000	3.000000	68.000000	1613.000000	8.000000	70.000000	1.000000
30%	18.000000	4.000000	112.000000	2301.000000	14.200000	73.000000	1.000000
45%	21.065000	4.000000	140.000000	2670.650000	15.000000	75.000000	1.000000
50%	23.000000	4.000000	148.500000	2803.500000	15.500000	76.000000	1.000000
55%	24.000000	6.000000	168.000000	2937.200000	15.735000	77.000000	1.000000
max	46.600000	8.000000	455.000000	5140.000000	24.800000	82.000000	3.000000

In [95]:
```
dataset[0:2].describe()
```

Out[95]:

	mpg	cylinders	displacement	weight	acceleration	model year	origin
count	2.00000	2.0	2.000000	2.000000	2.000000	2.0	2.0
mean	16.50000	8.0	328.500000	3598.500000	11.750000	70.0	1.0
std	2.12132	0.0	30.405592	133.643182	0.353553	0.0	0.0
min	15.00000	8.0	307.000000	3504.000000	11.500000	70.0	1.0
25%	15.75000	8.0	317.750000	3551.250000	11.625000	70.0	1.0
50%	16.50000	8.0	328.500000	3598.500000	11.750000	70.0	1.0
75%	17.25000	8.0	339.250000	3645.750000	11.875000	70.0	1.0
max	18.00000	8.0	350.000000	3693.000000	12.000000	70.0	1.0

In [96]:
```
dataset.corr() #correlation method
```

Out[96]:

	mpg	cylinders	displacement	weight	acceleration	model year	origin
mpg	1.000000	-0.775396	-0.804203	-0.831741	0.420289	0.579267	0.563450
cylinders	-0.775396	1.000000	0.950721	0.896017	-0.505419	-0.348746	-0.562543
displacement	-0.804203	0.950721	1.000000	0.932824	-0.543684	-0.370164	-0.609409
weight	-0.831741	0.896017	0.932824	1.000000	-0.417457	-0.306564	-0.581024
acceleration	0.420289	-0.505419	-0.543684	-0.417457	1.000000	0.288137	0.205873
model year	0.579267	-0.348746	-0.370164	-0.306564	0.288137	1.000000	0.180662
origin	0.563450	-0.562543	-0.609409	-0.581024	0.205873	0.180662	1.000000

Perfect Correlation: We can see that "mpg" and "mpg" got the number 1.000000, which makes sense, each column always has a perfect relationship with itself.

Good Correlation: "cylinders" and "displacement" got a 0.950721 correlation, which is a very good correlation, and we can predict that more cylinders means more displacement.

Bad Correlation: "model year" and "acceleration" got a 0.288137 correlation, which is a very bad correlation, meaning that we can not predict the max pulse by just looking at the duration of the work out, and vice versa.

##Scatter Matrix/Pair Plots

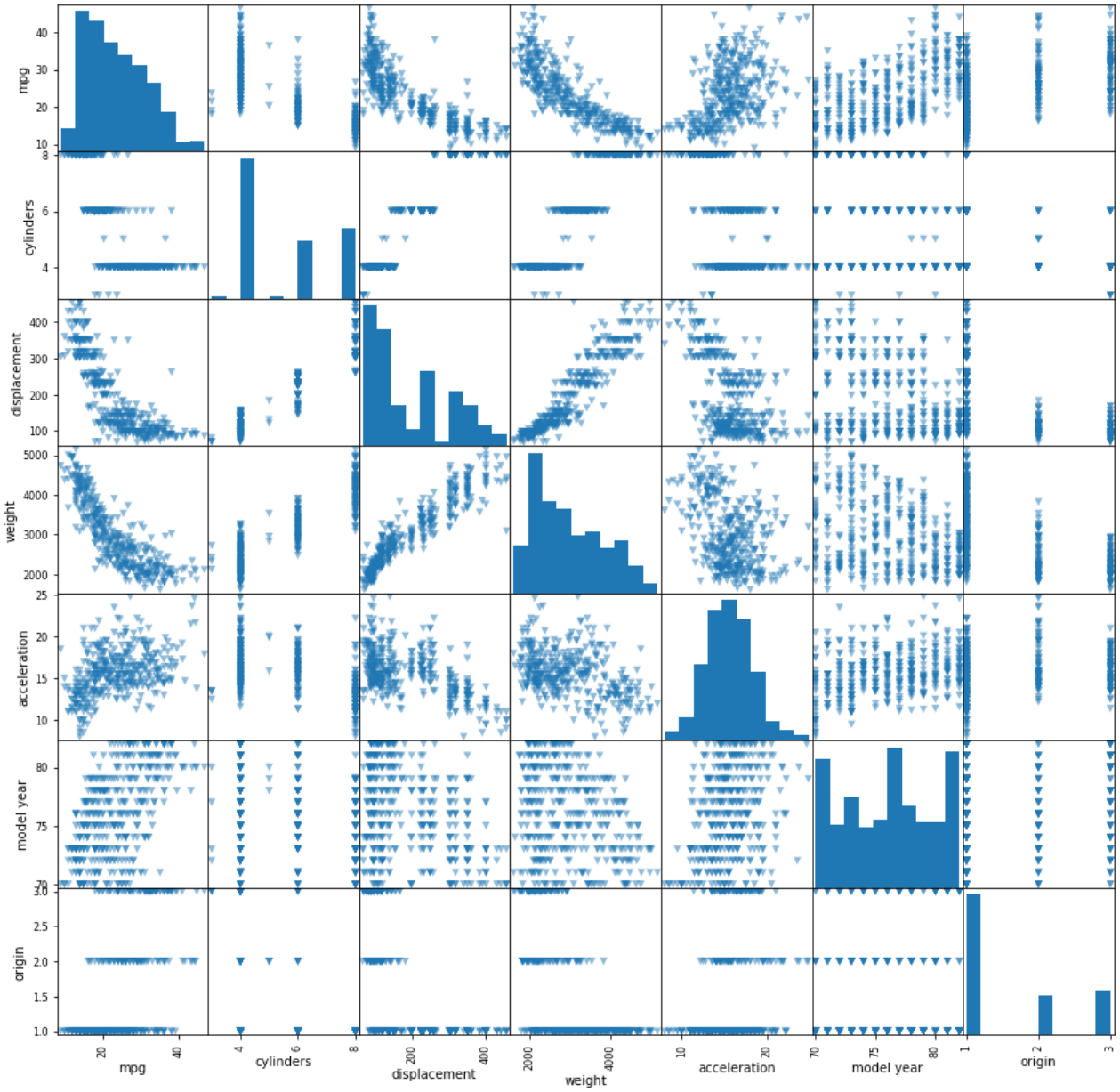
Returns a numpy.ndarray

By default, alpha=1. If you would like to form the graph plot more transparent, then you'll make alpha but 1, such as 0.5 or 0.25.

If you would like to form the graph plot less transparent, then you'll make alpha greater than 1. This solidifies the graph plot, making it less transparent and more thick and dense, so to talk .

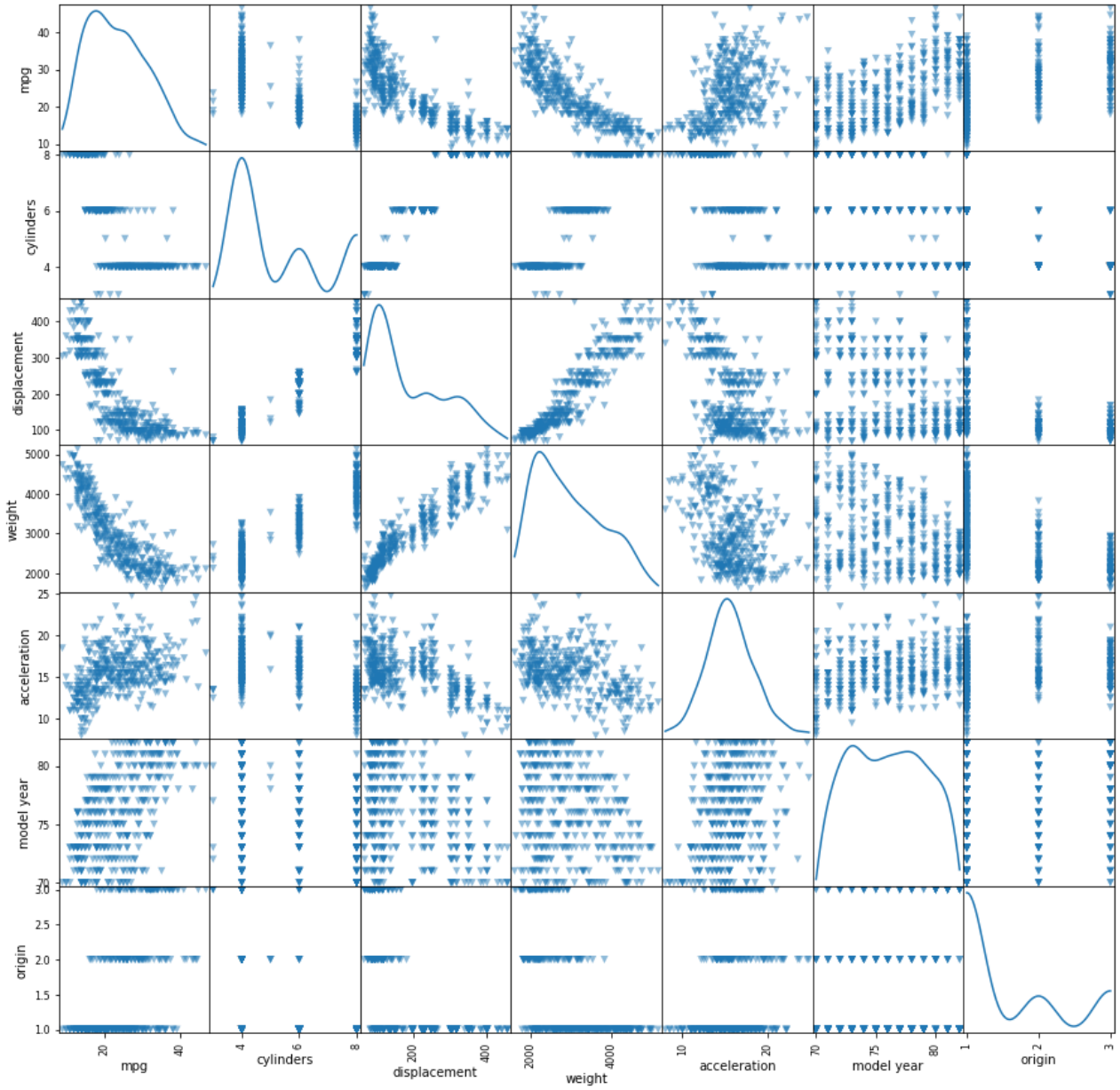
In [104…

```
import pandas as pd
import matplotlib.pyplot as plt

pd.plotting.scatter_matrix(dataset,figsize=[15,15],marker='v',alpha=0.5)
plt.show()
```



In [106…

```
import pandas as pd
import matplotlib.pyplot as plt

pd.plotting.scatter_matrix(dataset,figsize=[15,15],marker='v',alpha=0.5,diagonal='kde')
plt.show()
#kde - kernel density (Histogram change)
```



#Qualitative Data vs Quantitative Data

Quantitative data relates to information about the quantity of an object – hence it can be measured. For example, if we consider the attribute 'marks', it can be measured using a scale of measurement. Quantitative data is also termed as numeric data.

Qualitative data provides information about the quality of an object or information which cannot be measured. For example, if we consider the quality of performance of students in terms of 'Good', 'Average', and 'Poor', it falls under the category of qualitative data. Also, name or roll number of students are information that cannot be measured using some scale of measurement. So they would fall under qualitative data. Qualitative data is also called categorical data.

Quantitative Data can be analyzed by measures like mean, median, mode.

For qualitative data, we can use parallel coordinates and cross tabulation.

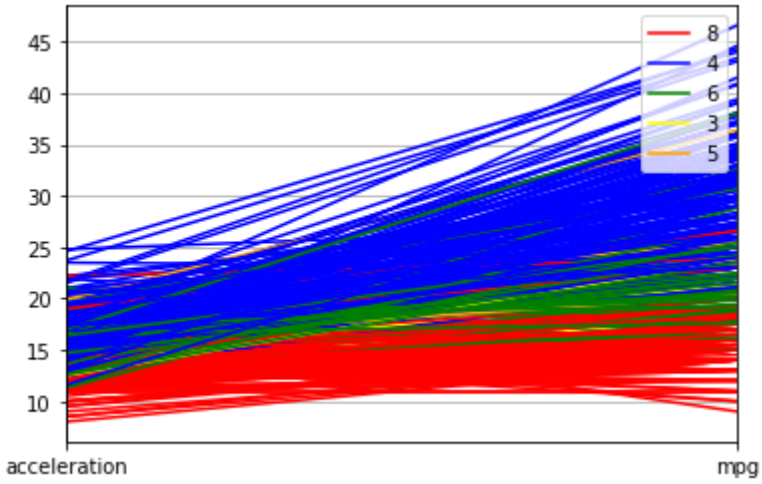
##Parallel coordinates

Parallel coordinates charts are commonly used to visualize and analyze high dimensional multivariate data. It represents each data sample as polyline connecting parallel lines where each parallel line represents an attribute of that data sample.

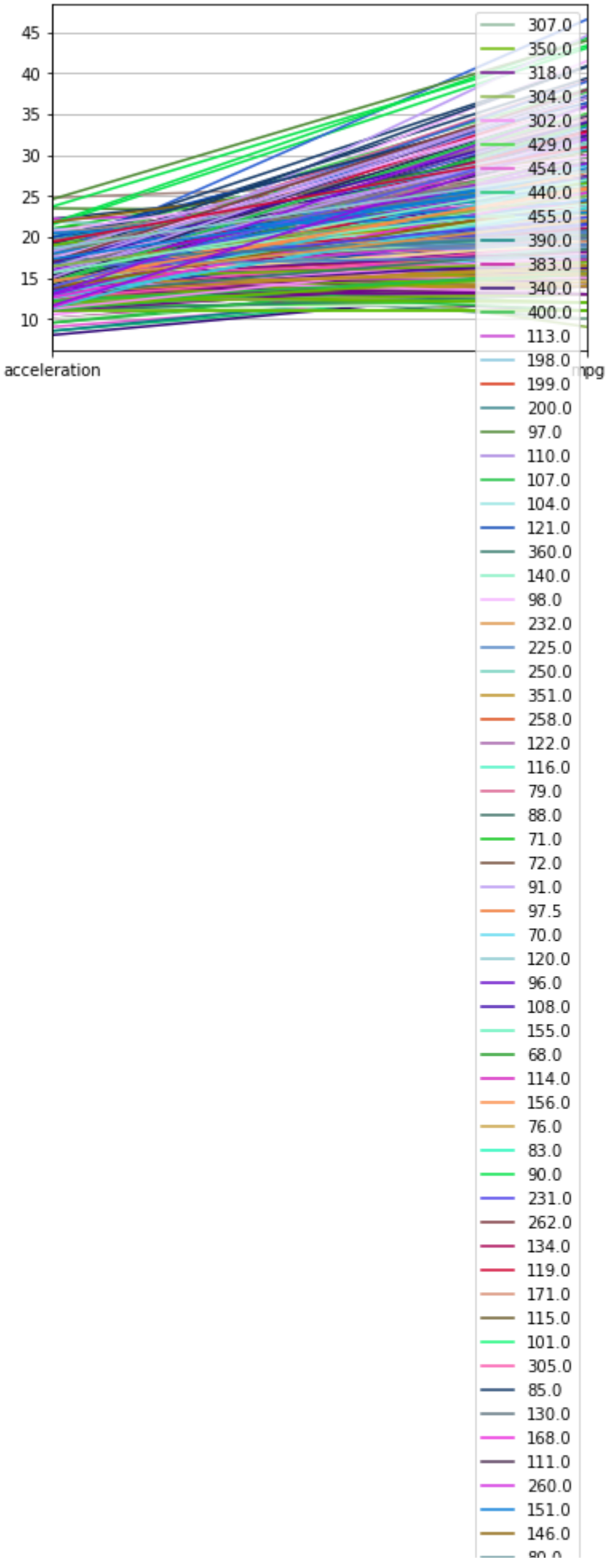
In [ ]:
```
###Attributes Qualitative(Categorical)(Nominal & Ordinal Types) & Quanititative(Numeric
```

In [107…
```
import pandas as pd
dataset=pd.read_csv('auto-mpg.csv')
```

In [108…
```
from pandas.plotting import parallel_coordinates
pll=parallel_coordinates(dataset,'cylinders',cols=["acceleration","mpg"],color=['red','
```

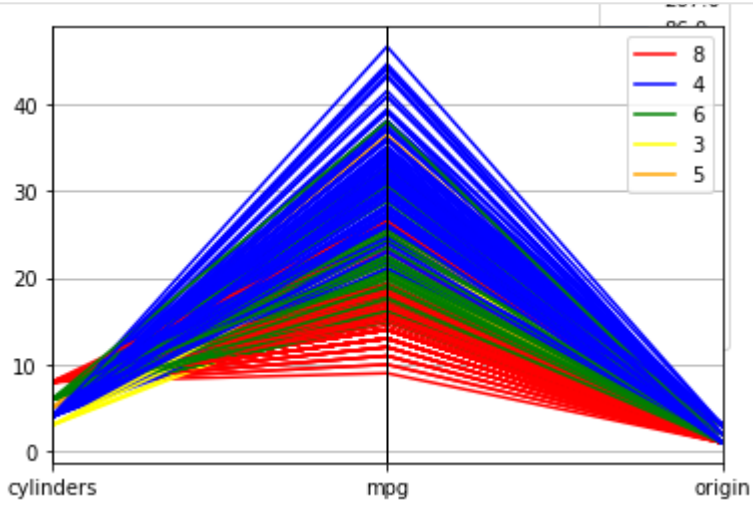


In [109…
```
#if we put displacement quantitative ..That doesn't make sense.
from pandas.plotting import parallel_coordinates
pll=parallel_coordinates(dataset,'displacement',cols=["acceleration","mpg"])
```


45
40
35
30
25
20
15
10
acceleration
mpg
307.0
350.0
318.0
304.0
302.0
429.0
454.0
440.0
455.0
390.0
383.0
340.0
400.0
113.0
198.0
199.0
200.0
97.0
110.0
107.0
104.0
121.0
360.0
140.0
98.0
232.0
225.0
250.0
351.0
258.0
122.0
116.0
79.0
88.0
71.0
72.0
91.0
97.5
70.0
120.0
96.0
108.0
155.0
68.0
114.0
156.0
76.0
83.0
90.0
231.0
262.0
134.0
119.0
171.0
115.0
101.0
305.0
85.0
130.0
168.0
111.0
260.0
151.0
146.0

In [110…

```
import matplotlib.pyplot as plt
from pandas.plotting import parallel_coordinates
pll = parallel_coordinates(dataset, 'cylinders', cols=['cylinders','mpg', 'origin'], co
plt.show()
```



##Cross Tabulation

In [111…

```
pd.crosstab(dataset['cylinders'], dataset['model year'], rownames=['cylinders'], colnam
```

Out[111…

cylinders \ model year	70	71	72	73	74	75	76	77	78	79	80	81	82
3	0	0	1	1	0	0	0	1	0	0	1	0	0
4	7	13	14	11	15	12	15	14	17	12	25	21	28
5	0	0	0	0	0	0	0	0	1	1	1	0	0
6	4	8	0	8	7	12	10	5	12	6	2	7	3
8	18	7	13	20	5	6	9	8	6	10	0	1	0

#Data Cleaning

##Missing Values

In [112…

```
import numpy as np
import pandas as pd
sales_data = pd.DataFrame({"name":["William","Emma","Sofia","Markus","Edward","Thomas",
,"region":[np.nan,"North","East",np.nan,"West","West","South",np.nan,"West","East","Sou
,"sales":[50000,52000,np.nan,np.nan,42000,72000,49000,np.nan,67000,65000,67000]
,"expenses":[42000,43000,np.nan,np.nan,38000,39000,42000,np.nan,39000,50000,45000]})
print(sales_data)
```

```
      name region    sales  expenses
0  William    NaN  50000.0   42000.0
1     Emma  North  52000.0   43000.0
2    Sofia   East      NaN       NaN
3   Markus    NaN      NaN       NaN
4   Edward   West  42000.0   38000.0
5   Thomas   West  72000.0   39000.0
6    Ethan  South  49000.0   42000.0
7      NaN    NaN      NaN       NaN
```

```
8      Arun   West  67000.0   39000.0
9     Anika   East  65000.0   50000.0
10    Paulo  South  67000.0   45000.0
```

In [113…
```
sales_data
```

Out[113…

	name	region	sales	expenses
0	William	NaN	50000.0	42000.0
1	Emma	North	52000.0	43000.0
2	Sofia	East	NaN	NaN
3	Markus	NaN	NaN	NaN
4	Edward	West	42000.0	38000.0
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
7	NaN	NaN	NaN	NaN
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

 # Methods to remove null values data

1. delete will delete complete row with NaN..It will delete data of row
2. fill with dummy data
3. replace it with mean,median values

In [126…
```
sales_data.isna()  # True when NAN
```

Out[126…

	name	region	sales	expenses
0	False	True	False	False
1	False	False	False	False
2	False	False	True	True
3	False	True	True	True
4	False	False	False	False
5	False	False	False	False
6	False	False	False	False
7	True	True	True	True
8	False	False	False	False
9	False	False	False	False
10	False	False	False	False

In [118…
```
sales_data.isna().sum()  #count the no of true in column, total 'NaN' values
```

Out[118…
```
name        1
region      3
sales       3
expenses    3
dtype: int64
```

In [117…
```
sales_data.dropna()
```

Out[117…

	name	region	sales	expenses
1	Emma	North	52000.0	43000.0
4	Edward	West	42000.0	38000.0
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

thresh specifies quantity of valid data so thresh = 2 means remove if there aren't atleast two cells with valid data (not null data)

In [119…
```
sales_data.dropna(thresh=2)
```

Out[119…

	name	region	sales	expenses
0	William	NaN	50000.0	42000.0
1	Emma	North	52000.0	43000.0
2	Sofia	East	NaN	NaN
4	Edward	West	42000.0	38000.0
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

In [120…
```
#remove all rows with any number of nulls
sales_data.dropna()
```

Out[120…

	name	region	sales	expenses
1	Emma	North	52000.0	43000.0
4	Edward	West	42000.0	38000.0

	name	region	sales	expenses
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

In [121…
```
#remove only if all values are null
sales_data.dropna(how='all')
```

Out[121…

	name	region	sales	expenses
0	William	NaN	50000.0	42000.0
1	Emma	North	52000.0	43000.0
2	Sofia	East	NaN	NaN
3	Markus	NaN	NaN	NaN
4	Edward	West	42000.0	38000.0
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

In [123…
```
#drop only if sales or expenses are null
sales_data.dropna(subset = ['sales', 'expenses'])
```

Out[123…

	name	region	sales	expenses
0	William	NaN	50000.0	42000.0
1	Emma	North	52000.0	43000.0
4	Edward	West	42000.0	38000.0
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

In [124…
```
sales_data.dropna(axis=0)
```

Out[124…

	name	region	sales	expenses
1	Emma	North	52000.0	43000.0

	name	region	sales	expenses
4	Edward	West	42000.0	38000.0
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

In [125…
```
sales_data.dropna(axis=1)
```

Out[125…

0
1
2
3
4
5
6
7
8
9
10

In [127…
```
sales_data.dropna(inplace=True)  # sales data will be deleted version
```

In [128…
```
sales_data
```

Out[128…

	name	region	sales	expenses
1	Emma	North	52000.0	43000.0
4	Edward	West	42000.0	38000.0
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

In [130…
```
# Again Taking the Sales_data because data is modified using inplace=True
import numpy as np
import pandas as pd
```

```
sales_data = pd.DataFrame({"name":["William","Emma","Sofia","Markus","Edward","Thomas",
,"region":[np.nan,"North","East",np.nan,"West","West","South",np.nan,"West","East","Sou
,"sales":[50000,52000,np.nan,np.nan,42000,72000,49000,np.nan,67000,65000,67000]
,"expenses":[42000,43000,np.nan,np.nan,38000,39000,42000,np.nan,39000,50000,45000]})
print(sales_data)
```

```
       name region    sales  expenses
0   William    NaN  50000.0   42000.0
1      Emma  North  52000.0   43000.0
2     Sofia   East      NaN       NaN
3    Markus    NaN      NaN       NaN
4    Edward   West  42000.0   38000.0
5    Thomas   West  72000.0   39000.0
6     Ethan  South  49000.0   42000.0
7       NaN    NaN      NaN       NaN
8      Arun   West  67000.0   39000.0
9     Anika   East  65000.0   50000.0
10    Paulo  South  67000.0   45000.0
```

In [131…
```
######### fill dummy data
```

In [132…
```
sales_data.fillna(0) # fill nan with 0
```

Out[132…

	name	region	sales	expenses
0	William	0	50000.0	42000.0
1	Emma	North	52000.0	43000.0
2	Sofia	East	0.0	0.0
3	Markus	0	0.0	0.0
4	Edward	West	42000.0	38000.0
5	Thomas	West	72000.0	39000.0
6	Ethan	South	49000.0	42000.0
7	0	0	0.0	0.0
8	Arun	West	67000.0	39000.0
9	Anika	East	65000.0	50000.0
10	Paulo	South	67000.0	45000.0

In [133…
```
sales_data['sales'].fillna(sales_data['sales'].mean()) # fill sales Nan with mean data
```

Out[133…
```
0     50000.0
1     52000.0
2     58000.0
3     58000.0
4     42000.0
5     72000.0
6     49000.0
7     58000.0
8     67000.0
9     65000.0
10    67000.0
Name: sales, dtype: float64
```

In [134…
```
sales_data['sales'].fillna(sales_data['sales'].median()) # fill sales Nan with median d
```

Out[134…
```
0     50000.0
1     52000.0
2     58500.0
3     58500.0
4     42000.0
5     72000.0
6     49000.0
7     58500.0
8     67000.0
9     65000.0
10    67000.0
Name: sales, dtype: float64
```

In [135…
```
import pandas as pd
dataset=pd.read_csv('auto-mpg.csv')
```

In [136…
```
dataset[dataset['horsepower']=='?']
```

Out[136…

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
32	25.0	4	98.0	?	2046	19.0	71	1	ford pinto
126	21.0	6	200.0	?	2875	17.0	74	1	ford maverick
330	40.9	4	85.0	?	1835	17.3	80	2	renault lecar deluxe
336	23.6	4	140.0	?	2905	14.3	80	1	ford mustang cobra
354	34.5	4	100.0	?	2320	15.8	81	2	renault 18i
374	23.0	4	151.0	?	3035	20.5	82	1	amc concord dl

In [137…
```
dataset = dataset[dataset['horsepower']!='?']
print(dataset)
```

```
      mpg  cylinders  displacement horsepower  weight  acceleration  \
0    18.0          8         307.0        130    3504          12.0
1    15.0          8         350.0        165    3693          11.5
2    18.0          8         318.0        150    3436          11.0
3    16.0          8         304.0        150    3433          12.0
4    17.0          8         302.0        140    3449          10.5
..    ...        ...           ...        ...     ...           ...
393  27.0          4         140.0         86    2790          15.6
394  44.0          4          97.0         52    2130          24.6
395  32.0          4         135.0         84    2295          11.6
396  28.0          4         120.0         79    2625          18.6
397  31.0          4         119.0         82    2720          19.4

     model year  origin                  car name
0            70       1  chevrolet chevelle malibu
1            70       1          buick skylark 320
2            70       1         plymouth satellite
3            70       1              amc rebel sst
4            70       1                ford torino
..          ...     ...                       ...
393          82       1            ford mustang gl
394          82       2                  vw pickup
```

```
395          82       1              dodge rampage
396          82       1                ford ranger
397          82       1                 chevy s-10

[392 rows x 9 columns]
```

In [138…
```
print(dataset) # lines is 392 now in place of 398 , 6 ? lines deleted
```

```
      mpg  cylinders  displacement horsepower  weight  acceleration  \
0    18.0          8         307.0        130    3504          12.0
1    15.0          8         350.0        165    3693          11.5
2    18.0          8         318.0        150    3436          11.0
3    16.0          8         304.0        150    3433          12.0
4    17.0          8         302.0        140    3449          10.5
..    ...        ...           ...        ...     ...           ...
393  27.0          4         140.0         86    2790          15.6
394  44.0          4          97.0         52    2130          24.6
395  32.0          4         135.0         84    2295          11.6
396  28.0          4         120.0         79    2625          18.6
397  31.0          4         119.0         82    2720          19.4

     model year  origin                  car name
0            70       1  chevrolet chevelle malibu
1            70       1          buick skylark 320
2            70       1         plymouth satellite
3            70       1              amc rebel sst
4            70       1                ford torino
..          ...     ...                       ...
393          82       1            ford mustang gl
394          82       2                  vw pickup
395          82       1              dodge rampage
396          82       1                ford ranger
397          82       1                 chevy s-10

[392 rows x 9 columns]
```

In [139…
```
dataset[dataset['horsepower']=='?'] #New Dataset with ? removed
```

Out[139…

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name

In [140…
```
dataset.drop('mpg',axis=1) # drop the whole column "mpg"
```

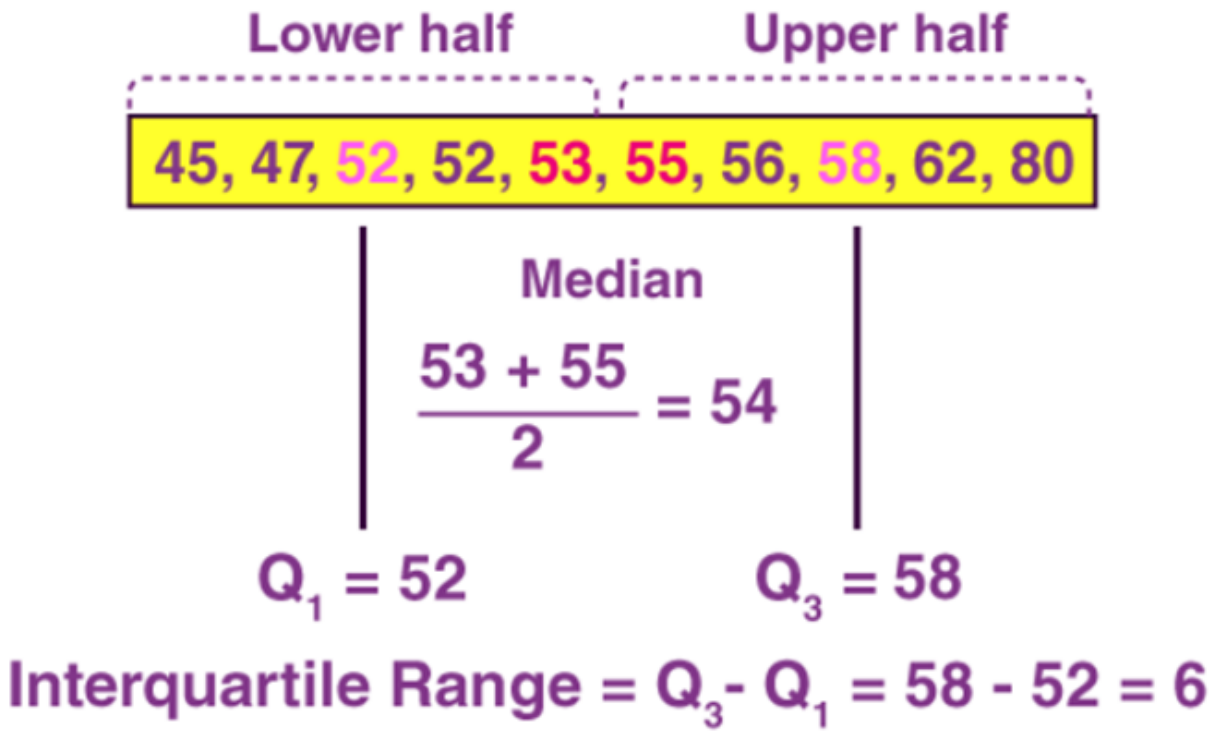
Out[140…

	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	8	302.0	140	3449	10.5	70	1	ford torino
…	…	…	…	…	…	…	…	…
393	4	140.0	86	2790	15.6	82	1	ford mustang gl
394	4	97.0	52	2130	24.6	82	2	vw pickup
395	4	135.0	84	2295	11.6	82	1	dodge rampage

	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
396	4	120.0	79	2625	18.6	82	1	ford ranger
397	4	119.0	82	2720	19.4	82	1	chevy s-10

392 rows × 8 columns

##Finding and Removing Outliers



In [144…

```
############ Outliers

# it is defined as the observation that deviated abnormally from the standard data
# IQR - Inter Quartile Range to find Outliers. (range between 25% to 75%)
# Q1-25% ,Q2-50%, Q3-75%
# IQR=Q3-Q1

# low = Q1 - 1.5*IQR
# high= Q3 + 1.5*IQR
```

In [145…

```
import pandas as pd
dataset=pd.read_csv('auto-mpg.csv')
print(dataset)
```

```
      mpg  cylinders  displacement horsepower  weight  acceleration  \
0    18.0          8         307.0        130    3504          12.0
1    15.0          8         350.0        165    3693          11.5
2    18.0          8         318.0        150    3436          11.0
3    16.0          8         304.0        150    3433          12.0
4    17.0          8         302.0        140    3449          10.5
..    ...        ...           ...        ...     ...           ...
393  27.0          4         140.0         86    2790          15.6
394  44.0          4          97.0         52    2130          24.6
395  32.0          4         135.0         84    2295          11.6
396  28.0          4         120.0         79    2625          18.6
```

```
397  31.0          4         119.0         82    2720          19.4

     model year  origin                   car name
0            70       1  chevrolet chevelle malibu
1            70       1          buick skylark 320
2            70       1         plymouth satellite
3            70       1              amc rebel sst
4            70       1                ford torino
..          ...     ...                         ...
393          82       1            ford mustang gl
394          82       2                  vw pickup
395          82       1              dodge rampage
396          82       1                ford ranger
397          82       1                 chevy s-10

[398 rows x 9 columns]
```

In [147…

```
# Function to Find the Outliers
import pandas as pd

def find_outliers(ds, col):
    quart1 = ds[col].quantile(0.25)
    quart3 = ds[col].quantile(0.75)
    IQR = quart3 - quart1 #Inter-quartile range
    low_val = quart1 - 1.5*IQR
    high_val = quart3 + 1.5*IQR
    print(low_val)
    print(high_val)
    ds = ds.loc[(ds[col] < low_val) | (ds[col] > high_val)]
    return ds

dataset = pd.read_csv('auto-mpg.csv')
find_outliers(dataset, 'acceleration')
```

```
8.8
22.2
```

Out[147…

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
7	14.0	8	440.0	215	4312	8.5	70	1	plymouth fury iii
9	15.0	8	390.0	190	3850	8.5	70	1	amc ambassador dpl
11	14.0	8	340.0	160	3609	8.0	70	1	plymouth 'cuda 340
59	23.0	4	97.0	54	2254	23.5	72	2	volkswagen type 3
299	27.2	4	141.0	71	3190	24.8	79	2	peugeot 504
326	43.4	4	90.0	48	2335	23.7	80	2	vw dasher (diesel)
394	44.0	4	97.0	52	2130	24.6	82	2	vw pickup

In [148…

```
# Function to Remove the Outliers
def remove_outliers(ds, col):
    quart1 = ds[col].quantile(0.25)
```

```
    quart3 = ds[col].quantile(0.75)
    IQR = quart3 - quart1 #Interquartile range
    low_val = quart1 - 1.5*IQR
    print(low_val)
    high_val = quart3 + 1.5*IQR
    print(high_val)
    df_out = ds.loc[(ds[col] >= low_val) & (ds[col] <= high_val)]
    return df_out

new_data = remove_outliers(dataset, 'acceleration')
print(new_data)
```

```
8.8
22.2
      mpg  cylinders  displacement horsepower  weight  acceleration  \
0    18.0          8         307.0        130    3504          12.0
1    15.0          8         350.0        165    3693          11.5
2    18.0          8         318.0        150    3436          11.0
3    16.0          8         304.0        150    3433          12.0
4    17.0          8         302.0        140    3449          10.5
..    ...        ...           ...        ...     ...           ...
392  27.0          4         151.0         90    2950          17.3
393  27.0          4         140.0         86    2790          15.6
395  32.0          4         135.0         84    2295          11.6
396  28.0          4         120.0         79    2625          18.6
397  31.0          4         119.0         82    2720          19.4

     model year  origin                  car name
0            70       1  chevrolet chevelle malibu
1            70       1          buick skylark 320
2            70       1         plymouth satellite
3            70       1              amc rebel sst
4            70       1                ford torino
..          ...     ...                       ...
392          82       1           chevrolet camaro
393          82       1            ford mustang gl
395          82       1              dodge rampage
396          82       1                ford ranger
397          82       1                 chevy s-10

[391 rows x 9 columns]
```

In [149…

```
#7 outliers removed from new_data
new_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 391 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   mpg           391 non-null    float64
 1   cylinders     391 non-null    int64
 2   displacement  391 non-null    float64
 3   horsepower    391 non-null    object
 4   weight        391 non-null    int64
 5   acceleration  391 non-null    float64
 6   model year    391 non-null    int64
 7   origin        391 non-null    int64
 8   car name      391 non-null    object
dtypes: float64(3), int64(4), object(2)
memory usage: 30.5+ KB
```

Finding Outliers and Remove Outliers in 'mpg'

In [150…
```
find_outliers(dataset, 'mpg')
```

```
0.25
46.25
```

Out[150…

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
322	46.6	4	86.0	65	2110	17.9	80	3	mazda glc

In [152…
```
# only 1 Outlier in mpg
new_data = remove_outliers(dataset, 'mpg')
print(new_data)
```

```
0.25
46.25
      mpg  cylinders  displacement horsepower  weight  acceleration  \
0    18.0          8         307.0        130    3504          12.0
1    15.0          8         350.0        165    3693          11.5
2    18.0          8         318.0        150    3436          11.0
3    16.0          8         304.0        150    3433          12.0
4    17.0          8         302.0        140    3449          10.5
..    ...        ...           ...        ...     ...           ...
393  27.0          4         140.0         86    2790          15.6
394  44.0          4          97.0         52    2130          24.6
395  32.0          4         135.0         84    2295          11.6
396  28.0          4         120.0         79    2625          18.6
397  31.0          4         119.0         82    2720          19.4

     model year  origin                  car name
0            70       1  chevrolet chevelle malibu
1            70       1          buick skylark 320
2            70       1         plymouth satellite
3            70       1              amc rebel sst
4            70       1                ford torino
..          ...     ...                         ...
393          82       1            ford mustang gl
394          82       2                  vw pickup
395          82       1              dodge rampage
396          82       1                ford ranger
397          82       1                 chevy s-10

[397 rows x 9 columns]
```

 # Drop & Remove Duplicates

In [157…
```
#drop duplicates
import pandas as pd

data = {
    "A": ["TeamA", "TeamB", "TeamB", "TeamC", "TeamA"],
    "B": [50, 40, 40, 30, 50],
    "C": [True, False, False, False, True]
}

df = pd.DataFrame(data)
print(df)
dups = df.duplicated()
```

```
       A   B      C
0  TeamA  50   True
1  TeamB  40  False
```

```
2  TeamB  40  False
3  TeamC  30  False
4  TeamA  50   True
```

In [155…
```
print(dups)
```

```
0    False
1    False
2     True
3    False
4     True
dtype: bool
```

In [158…
```
df = df.drop_duplicates()
print(df)
```

```
       A   B      C
0  TeamA  50   True
1  TeamB  40  False
3  TeamC  30  False
```

In [159…
```
df = df.reset_index(drop=True)
print(df)
```

```
       A   B      C
0  TeamA  50   True
1  TeamB  40  False
2  TeamC  30  False
```