

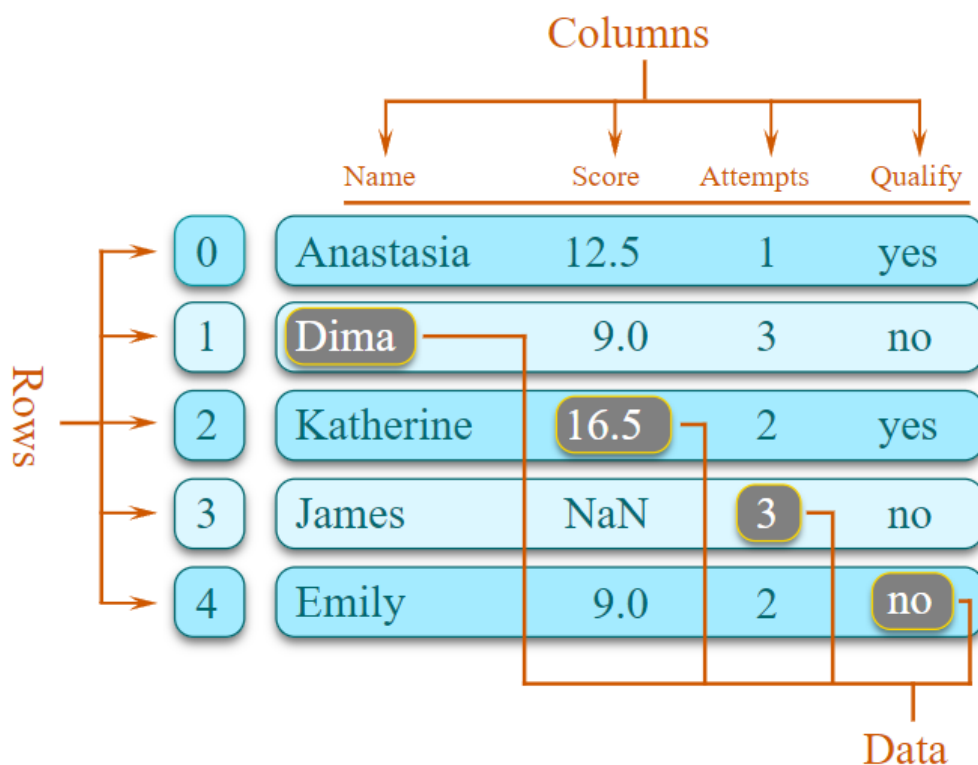
Pandas part - 02



Pandas

What is DataFrame?

A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns. Pandas DataFrame consists of three principal components, the data, rows, and columns.



The diagram illustrates the structure of a Pandas DataFrame. It features a table with 5 rows and 4 columns. The columns are labeled 'Name', 'Score', 'Attempts', and 'Qualify'. The rows are indexed from 0 to 4. The data is as follows:

	Name	Score	Attempts	Qualify
0	Anastasia	12.5	1	yes
1	Dima	9.0	3	no
2	Katherine	16.5	2	yes
3	James	NaN	3	no
4	Emily	9.0	2	no

Annotations in the diagram include:

- Columns:** A label at the top with arrows pointing to the column headers.
- Rows:** A label on the left with arrows pointing to the row indices.
- Data:** A label at the bottom right with arrows pointing to individual data cells, such as 'Dima', '16.5', '3', and 'no'.

Pandas DataFrame

```
In [ ]: import numpy as np
import pandas as pd
```

Creating DataFrame

```
In [ ]: # using lists
student_data = [
    [100, 90, 10],
    [90, 70, 7],
    [120, 100, 14],
    [80, 50, 2]
]
pd.DataFrame(student_data, columns=['iq', 'marks', 'package', ])
```

```
Out[ ]:
```

	iq	marks	package
0	100	90	10
1	90	70	7
2	120	100	14
3	80	50	2

```
In [ ]: # using dictionary
student_dict = {
    'iq': [100, 90, 80, 120, 0, 0],
    'marks': [80, 70, 100, 90, 0, 0],
    'package': [10, 7, 14, 2, 0, 0]
}

students = pd.DataFrame(student_dict)
students
```

```
Out[ ]:
```

	iq	marks	package
0	100	80	10
1	90	70	7
2	80	100	14
3	120	90	2
4	0	0	0
5	0	0	0

Read CSV

```
In [ ]: # using read_csv
movies = pd.read_csv('movies.csv')
movies.head()
```

```
Out[ ]:
```

	title_x	imdb_id	poster_path
0	Uri: The Surgical Strike	tt8291224	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/wik
1	Battalion	tt9472208	NaN https://en.wikipedia.o

609

2 The
Accidental
Prime
Minister
(film) tt6986710 <https://upload.wikimedia.org/wikipedia/en/thum...> <https://en.wikipedia.org/wiki>

3 Why
Cheat
India tt8108208 <https://upload.wikimedia.org/wikipedia/en/thum...> <https://en.wikipedia.org/w>

4 Evening
Shadows tt6028796 NaN <https://en.wikipedia.org/wi>

```
In [ ]: ipl = pd.read_csv('ipl-matches.csv')
        ipl.head()
```

```
Out[ ]: 
```

	ID	City	Date	Season	MatchNumber	Team1	Team2	Venue
0	1312200	Ahmedabad	2022-05-29	2022	Final	Rajasthan Royals	Gujarat Titans	Narendra Modi Stadium, Ahmedabad
1	1312199	Ahmedabad	2022-05-27	2022	Qualifier 2	Royal Challengers Bangalore	Rajasthan Royals	Narendra Modi Stadium, Ahmedabad
2	1312198	Kolkata	2022-05-25	2022	Eliminator	Royal Challengers Bangalore	Lucknow Super Giants	Eden Gardens, Kolkata
3	1312197	Kolkata	2022-05-24	2022	Qualifier 1	Rajasthan Royals	Gujarat Titans	Eden Gardens, Kolkata
4	1304116	Mumbai	2022-05-22	2022	70	Sunrisers Hyderabad	Punjab Kings	Wankhede Stadium, Mumbai

Shape

```
In [ ]: # shape
        ipl.shape
```

```
Out[ ]: (950, 20)
```

Column Data Types

```
In [ ]: # dtypes
        movies.dtypes
```

```
Out[ ]: title_x          object
        imdb_id         object
        poster_path     object
        wiki_link       object
        title_y         object
        original_title   object
        is_adult         int64
        year_of_release  int64
        runtime         object
        genres          object
        imdb_rating      float64
        imdb_votes       int64
        story           object
        summary         object
        tagline         object
        actors          object
        wins_nominations object
        release_date     object
        dtype: object
```

Index

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

```
Out[ ]: RangeIndex(start=0, stop=1629, step=1)
```

Column Names

```
In [ ]: # columns
        movies.columns
```

```
Out[ ]: Index(['title_x', 'imdb_id', 'poster_path', 'wiki_link', 'title_y',
              'original_title', 'is_adult', 'year_of_release', 'runtime', 'genre
              s',
              'imdb_rating', 'imdb_votes', 'story', 'summary', 'tagline', 'actor
              s',
              'wins_nominations', 'release_date'],
              dtype='object')
```

```
In [ ]: ipl.columns
```

```
Out[ ]: Index(['ID', 'City', 'Date', 'Season', 'MatchNumber', 'Team1', 'Team2',
              'Venue', 'TossWinner', 'TossDecision', 'SuperOver', 'WinningTeam',
              'WonBy', 'Margin', 'method', 'Player_of_Match', 'Team1Players',
              'Team2Players', 'Umpire1', 'Umpire2'],
              dtype='object')
```

Values

```
In [ ]: # values -> 2D numpy array
        students.values
```

```
Out[ ]: array([[100,  80, 10],
               [ 90,  70,  7],
               [ 80, 100, 14],
               [120,  90,  2],
               [  0,   0,  0],
               [  0,   0,  0]], dtype=int64)
```

```
ipl.values
```

In []:

```
Out[ ]: array([[1312200, 'Ahmedabad', '2022-05-29', ...,
        ['WP Saha', 'Shubman Gill', 'MS Wade', 'HH Pandya', 'DA Miller',
        'R Tewatia', 'Rashid Khan', 'R Sai Kishore', 'LH Ferguson', 'Yash Dayal',
        'Mohammed Shami']],
        ['CB Gaffaney', 'Nitin Menon']],
        [1312199, 'Ahmedabad', '2022-05-27', ...,
        ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D Padikkal', 'SO Het
        myer', 'R Parag', 'R Ashwin', 'TA Boult', 'YS Chahal', 'M Prasidh Krishn
        a', 'OC McCoy']],
        ['CB Gaffaney', 'Nitin Menon']],
        [1312198, 'Kolkata', '2022-05-25', ...,
        ['Q de Kock', 'KL Rahul', 'M Vohra', 'DJ Hooda', 'MP Stoinis',
        'E Lewis', 'KH Pandya', 'PVD Chameera', 'Mohsin Khan', 'Avesh Khan', 'Rav
        i Bishnoi']],
        ['J Madanagopal', 'MA Gough']],
        ...,
        [335984, 'Delhi', '2008-04-19', ...,
        ['T Kohli', 'YK Pathan', 'SR Watson', 'M Kaif', 'DS Lehmann', 'R
        A Jadeja', 'M Rawat', 'D Salunkhe', 'SK Warne', 'SK Trivedi', 'MM Pate
        l']],
        ['Aleem Dar', 'GA Pratapkumar']],
        [335983, 'Chandigarh', '2008-04-19', ...,
        ['PA Patel', 'ML Hayden', 'MEK Hussey', 'MS Dhoni', 'SK Raina',
        'JDP Oram', 'S Badrinath', 'Joginder Sharma', 'P Amarnath', 'MS Gony', 'M
        Muralitharan']],
        ['MR Benson', 'SL Shastri']],
        [335982, 'Bangalore', '2008-04-18', ...,
        ['SC Ganguly', 'BB McCullum', 'RT Ponting', 'DJ Hussey', 'Mohamm
        ad Hafeez', 'LR Shukla', 'WP Saha', 'AB Agarkar', 'AB Dinda', 'M Kartik',
        'I Sharma']],
        ['Asad Rauf', 'RE Koertzen']]], dtype=object)
```

head and tail

```
In [ ]: # head and tail
movies.head(1)
```

```
Out[ ]:   title_x  imdb_id  poster_path

Uri: The
0  Surgical  tt8291224  https://upload.wikimedia.org/wikipedia/en/thum...  https://en.wikipedia.org/wiki/U
    Strike
```

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

```
Out[ ]:   title_x  imdb_id  poster_path

1626  Sabse  tt0069204  NaN  https://en.wikipedia.org
       Bada
       Sukh

1627  Daaka  tt10833860  https://upload.wikimedia.org/wikipedia/en/thum...  https://en.
```

sample - Random Data

```
In [ ]: # sample -> random data
ipl.sample(5)
```

```
Out[ ]:
```

	ID	City	Date	Season	MatchNumber	Team1	Team2	
39	1304081	Navi Mumbai	2022-04-23	2022	35	Gujarat Titans	Kolkata Knight Riders	Dr DY Patil Sports Academy
925	336006	Bangalore	2008-05-05	2007/08	25	Royal Challengers Bangalore	Kings XI Punjab	Chinnaswamy Stadium
820	419120	Cuttack	2010-03-21	2009/10	15	Deccan Chargers	Delhi Daredevils	Bahadur Saheb Stadium
792	419148	Bangalore	2010-04-10	2009/10	43	Royal Challengers Bangalore	Kolkata Knight Riders	Chinnaswamy Stadium
696	548311	Visakhapatnam	2012-04-07	2012	6	Deccan Chargers	Chennai Super Kings	Rajasekhara Reddy Vengal Rao Cricket Ground

info - information about columns

```
In [ ]: # info
movies.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1629 entries, 0 to 1628
Data columns (total 18 columns):
#   Column              Non-Null Count  Dtype  
---  -
0   title_x              1629 non-null   object  
1   imdb_id              1629 non-null   object  
2   poster_path          1526 non-null   object  
3   wiki_link            1629 non-null   object  
4   title_y              1629 non-null   object  
5   original_title       1629 non-null   object  
6   is_adult             1629 non-null   int64   
7   year_of_release      1629 non-null   int64   
8   runtime              1629 non-null   object  
9   genres               1629 non-null   object
```

```

10  imdb_rating      1629 non-null    float64
11  imdb_votes      1629 non-null    int64
12  story           1609 non-null    object
13  summary         1629 non-null    object
14  tagline         557 non-null     object
15  actors          1624 non-null    object
16  wins_nominations 707 non-null     object
17  release_date     1522 non-null    object
dtypes: float64(1), int64(3), object(14)
memory usage: 229.2+ KB

```

describe

```

In [ ]: # describe -> mathematical summary
movies.describe()

```

```

Out[ ]:

```

	is_adult	year_of_release	imdb_rating	imdb_votes
count	1629.0	1629.000000	1629.000000	1629.000000
mean	0.0	2010.263966	5.557459	5384.263352
std	0.0	5.381542	1.567609	14552.103231
min	0.0	2001.000000	0.000000	0.000000
25%	0.0	2005.000000	4.400000	233.000000
50%	0.0	2011.000000	5.600000	1000.000000
75%	0.0	2015.000000	6.800000	4287.000000
max	0.0	2019.000000	9.400000	310481.000000

isnull - Checking Null value

```

In [ ]: # isnull
movies.isnull()

```

```

Out[ ]:

```

	title_x	imdb_id	poster_path	wiki_link	title_y	original_title	is_adult	year_of_release
0	False	False	False	False	False	False	False	False
1	False	False	True	False	False	False	False	False
2	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False
4	False	False	True	False	False	False	False	False
...
1624	False	False	False	False	False	False	False	False
1625	False	False	False	False	False	False	False	False
1626	False	False	True	False	False	False	False	False
1627	False	False	False	False	False	False	False	False
1628	False	False	False	False	False	False	False	False

1629 rows × 18 columns

Checking total null value using sum()

```
In [ ]: movies.isnull().sum()
```

```
Out[ ]: title_x          0
imdb_id              0
poster_path        103
wiki_link           0
title_y             0
original_title      0
is_adult            0
year_of_release     0
runtime             0
genres              0
imdb_rating         0
imdb_votes          0
story               20
summary             0
tagline             1072
actors              5
wins_nominations    922
release_date        107
dtype: int64
```

deduplicated

```
In [ ]: movies.duplicated().sum()
```

```
Out[ ]: 0
```

```
In [ ]: students.duplicated()
```

```
Out[ ]: 0    False
1    False
2    False
3    False
4    False
5     True
dtype: bool
```

```
In [ ]: students.duplicated().sum()
```

```
Out[ ]: 1
```

rename - rename columns name

```
In [ ]: # rename
students
```

```
Out[ ]:   iq  marks  package
0  100    80         10
1   90    70          7
2   80   100         14
3  120    90          2
4    0     0          0
5    0     0          0
```



```
In [ ]: students.rename(columns={'marks': 'percent', 'package': 'lpa'})
```

```
Out[ ]:
```

	iq	percent	lpa
0	100	80	10
1	90	70	7
2	80	100	14
3	120	90	2
4	0	0	0
5	0	0	0

For permanent change using inplace = True

```
In [ ]: # for permanent change
students.rename(columns={'marks': 'percent', 'package': 'lpa'}, inplace=True)
students
```

```
Out[ ]:
```

	iq	percent	lpa
0	100	80	10
1	90	70	7
2	80	100	14
3	120	90	2
4	0	0	0
5	0	0	0

Math Methods

```
In [ ]: # sum -> axis argument
movies.sum() # concatenate the string and sum the integer and float
```

C:\Users\dhanr\AppData\Local\Temp\ipykernel_11868\2393232322.py:2: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.

```
movies.sum() # concatenate the string and sum the integer and float
```

```
Out[ ]:
```

title_x	Uri: The Surgical StrikeBattalion 609The Accid...
imdb_id	tt8291224tt9472208tt6986710tt8108208tt6028796t...
wiki_link	https://en.wikipedia.org/wiki/Uri:_The_Surgica...
title_y	Uri: The Surgical StrikeBattalion 609The Accid...
original_title	Uri: The Surgical StrikeBattalion 609The Accid...
is_adult	0
year_of_release	3274720
runtime	1381311121211029710910414812013415314313014311...
genres	Action Drama WarWarBiography DramaCrime DramaD...
imdb_rating	9053.1
imdb_votes	8770965
summary	Indian army special forces execute a covert op...
dtype: object	

```
In [ ]: students
```

```
Out[ ]:
```

	iq	percent	lpa
0	100	80	10
1	90	70	7
2	80	100	14
3	120	90	2
4	0	0	0
5	0	0	0

sum of columns

```
In [ ]: # sum of columns
students.sum()
```

```
Out[ ]: iq          390
percent      340
lpa          33
dtype: int64
```

Sum of rows

```
In [ ]: # sum of rows
students.sum(axis=1)
```

```
Out[ ]: 0      190
1      167
2      194
3      212
4         0
5         0
dtype: int64
```

Mean

```
In [ ]: # mean of cols
students.mean()
```

```
Out[ ]: iq          65.000000
percent      56.666667
lpa          5.500000
dtype: float64
```

```
In [ ]: # mean of rows
students.mean(axis=1)
```

```
Out[ ]: 0      63.333333
1      55.666667
2      64.666667
3      70.666667
4         0.000000
5         0.000000
dtype: float64
```

min = minimum

```
In [ ]: # min of cols
```

```
students.min()
```

```
Out[ ]: iq          0
        percent    0
        lpa        0
        dtype: int64
```

```
In [ ]: # min of rows
        students.min(axis=1)
```

```
Out[ ]: 0      10
        1       7
        2     14
        3       2
        4       0
        5       0
        dtype: int64
```

Selecting cols from a DataFrame

```
In [ ]: # single cols
        movies['title_x']
```

```
Out[ ]: 0          Uri: The Surgical Strike
        1          Battalion 609
        2    The Accidental Prime Minister (film)
        3          Why Cheat India
        4          Evening Shadows
        ...
        1624      Tera Mera Saath Rahen
        1625      Yeh Zindagi Ka Safar
        1626      Sabse Bada Sukh
        1627          Daaka
        1628      Humsafar
        Name: title_x, Length: 1629, dtype: object
```

```
In [ ]: movies.columns
```

```
Out[ ]: Index(['title_x', 'imdb_id', 'poster_path', 'wiki_link', 'title_y',
              'original_title', 'is_adult', 'year_of_release', 'runtime', 'genre
              s',
              'imdb_rating', 'imdb_votes', 'story', 'summary', 'tagline', 'actor
              s',
              'wins_nominations', 'release_date'],
              dtype='object')
```

```
In [ ]: # multiple cols
        movies[['title_x', 'year_of_release', 'actors']]
```

```
Out[ ]: 
```

	title_x	year_of_release	actors
0	Uri: The Surgical Strike	2019	Vicky Kaushal Paresh Rawal Mohit Raina Yami Ga...
1	Battalion 609	2019	Vicky Ahuja Shoaib Ibrahim Shrikant Kamat Elen...
2	The Accidental Prime Minister (film)	2019	Anupam Kher Akshaye Khanna Aahana Kumra Atul S...
3	Why Cheat India	2019	Emraan Hashmi Shreya Dhanwanthary Snighdadeep ...
4	Evening Shadows	2018	Mona Ambegaonkar Ananth Narayan Mahadevan Deva...

...
1624	Tera Mera Saath Rahen	2001	Ajay Devgn Sonali Bendre Namrata Shirodkar Pre...
1625	Yeh Zindagi Ka Safar	2001	Ameesha Patel Jimmy Sheirgill Nafisa Ali Gulsh...
1626	Sabse Bada Sukh	2018	Vijay Arora Asrani Rajni Bala Kumud Damle Utpa...
1627	Daaka	2019	Gippy Grewal Zareen Khan
1628	Humsafar	2011	Fawad Khan

1629 rows × 3 columns

Selecting rows from a DataFrame

- **iloc** - searches using index position
- **loc** - searches using index labels

```
In [ ]: student_dict = {
        'name': ['nitish', 'rupesh', 'rishabh', 'amit', 'ankita', 'suresh'],
        'iq': [100, 90, 80, 120, 0, 0],
        'marks': [80, 70, 100, 90, 0, 0],
        'package': [10, 7, 14, 2, 0, 0]
    }

students = pd.DataFrame(student_dict)
students.set_index('name', inplace=True)
students
```

```
Out[ ]:      iq  marks  package
name
nitish  100     80        10
rupesh   90     70         7
rishabh  80    100        14
amit    120     90         2
ankita   0      0         0
suresh   0      0         0
```

```
In [ ]: # single row
movies.iloc[0]
```

```
Out[ ]: title_x          Uri: The Surgical Strike
imdb_id          tt8291224
poster_path      https://upload.wikimedia.org/wikipedia/en/thum...
wiki_link        https://en.wikipedia.org/wiki/Uri:_The_Surgica...
title_y          Uri: The Surgical Strike
original_title    Uri: The Surgical Strike
is_adult          0
year_of_release   2019
runtime          138
genres            Action|Drama|War
imdb_rating       8.4
imdb_votes        35112
story            Divided over five chapters  the film chronicle...
```

```

summary          Indian army special forces execute a covert op...
tagline          NaN
actors           Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga...
wins_nominations 4 wins
release_date      11 January 2019 (USA)
Name: 0, dtype: object

```

```

In [ ]: # multiple row
        movies.iloc[0:5]

```

```

Out[ ]:

```

	title_x	imdb_id	poster_path
0	Uri: The Surgical Strike	tt8291224	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/wiki...
1	Battalion 609	tt9472208	NaN https://en.wikipedia.o
2	The Accidental Prime Minister (film)	tt6986710	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/wiki...
3	Why Cheat India	tt8108208	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/w
4	Evening Shadows	tt6028796	NaN https://en.wikipedia.org/wi

```

In [ ]: movies.iloc[0:10:2]

```

```

Out[ ]:

```

	title_x	imdb_id	poster_path
0	Uri: The Surgical Strike	tt8291224	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/v
2	The Accidental Prime Minister (film)	tt6986710	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/w
4	Evening Shadows	tt6028796	NaN https://en.wikipedia.org/
6	Fraud Saiyaan	tt5013008	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.o

Manikarnika:
8 The Queen of Jhansi tt6903440 <https://upload.wikimedia.org/wikipedia/en/thum...> <https://en.wikipedia.org/v>

```
In [ ]: students.loc['nitish']
```

```
Out[ ]: iq          100  
marks        80  
package      10  
Name: nitish, dtype: int64
```

```
In [ ]: students.loc['nitish':'rishabh']
```

```
Out[ ]:      iq  marks  package  
name  
nitish  100    80     10  
rupesh   90    70      7  
rishabh  80   100    14
```

```
In [ ]: # slicing row and columns using iloc  
movies.iloc[0:3, 0:3]
```

```
Out[ ]:      title_x  imdb_id  poster_path  
0      Uri: The Surgical Strike  tt8291224  https://upload.wikimedia.org/wikipedia/en/thum...  
1      Battalion 609  tt9472208  NaN  
2      The Accidental Prime Minister (film)  tt6986710  https://upload.wikimedia.org/wikipedia/en/thum...
```

```
In [ ]: # slicing row and columns using loc  
movies.loc[0:3, 'title_x': 'poster_path']
```

```
Out[ ]:      title_x  imdb_id  poster_path  
0      Uri: The Surgical Strike  tt8291224  https://upload.wikimedia.org/wikipedia/en/thum...  
1      Battalion 609  tt9472208  NaN  
2      The Accidental Prime Minister (film)  tt6986710  https://upload.wikimedia.org/wikipedia/en/thum...  
3      Why Cheat India  tt8108208  https://upload.wikimedia.org/wikipedia/en/thum...
```

Task

```
In [ ]: ipl.head()
```

```
Out[ ]:      ID      City  Date  Season  MatchNumber  Team1  Team2  Venue  
0  1312200  Ahmedabad  2022-05-29  2022      Final  Rajasthan Royals  Gujarat Titans  Narendra Modi Stadium, Ahmedabad
```

1	1312199	Ahmedabad	2022-05-27	2022	Qualifier 2	Royal Challengers Bangalore	Rajasthan Royals	Narendra Modi Stadium, Ahmedabad
2	1312198	Kolkata	2022-05-25	2022	Eliminator	Royal Challengers Bangalore	Lucknow Super Giants	Eden Gardens, Kolkata
3	1312197	Kolkata	2022-05-24	2022	Qualifier 1	Rajasthan Royals	Gujarat Titans	Eden Gardens, Kolkata
4	1304116	Mumbai	2022-05-22	2022	70	Sunrisers Hyderabad	Punjab Kings	Wankhede Stadium, Mumbai

```
In [ ]: # find all the final winners
ipl[ipl['MatchNumber'] == 'Final']['WinningTeam', 'Season']
```

```
Out[ ]:
```

	WinningTeam	Season
0	Gujarat Titans	2022
74	Chennai Super Kings	2021
134	Mumbai Indians	2020/21
194	Mumbai Indians	2019
254	Chennai Super Kings	2018
314	Mumbai Indians	2017
373	Sunrisers Hyderabad	2016
433	Mumbai Indians	2015
492	Kolkata Knight Riders	2014
552	Mumbai Indians	2013
628	Kolkata Knight Riders	2012
702	Chennai Super Kings	2011
775	Chennai Super Kings	2009/10
835	Deccan Chargers	2009
892	Rajasthan Royals	2007/08

```
In [ ]: # how many super over finishes have occurs
ipl[ipl['SuperOver'] == 'Y'].shape[0]
```

```
Out[ ]: 14
```

```
In [ ]: # how many matches csk won in kolkata
ipl[(ipl['City'] == 'Kolkata') & (
    ipl['WinningTeam'] == 'Chennai Super Kings')].shape[0]
```

```
Out[ ]: 5
```

```
In [ ]: # toss winner is match winner in percentage
ipl[ipl['TossWinner'] == ipl['WinningTeam']].shape[0]
```

Out[]: 489

```
In [ ]: ((ipl[ipl['TossWinner'] ==  
            ipl['WinningTeam']].shape[0])/(ipl.shape[0])) * 100
```

Out[]: 51.473684210526315

```
In [ ]: movies.head(2)
```

```
Out[ ]:   title_x  imdb_id  poster_path  
  
0  Uri: The Surgical Strike  tt8291224  https://upload.wikimedia.org/wikipedia/en/thum...  https://en.wikipedia.org/wiki/L
```

```
1  Battalion 609  tt9472208  NaN  https://en.wikipedia.org/
```

```
In [ ]: # movies with rating higher than 8 and votes > 10000  
movies[(movies['imdb_rating'] > 8) & (movies['imdb_votes'] > 10000)  
       ][['title_x', 'imdb_votes', 'imdb_rating']]
```

```
Out[ ]:   title_x  imdb_votes  imdb_rating  
0  Uri: The Surgical Strike  35112  8.4  
11  Gully Boy  22440  8.2  
37  Article 15 (film)  13417  8.3  
40  Super 30 (film)  13972  8.2  
143  Tumbbad  16535  8.2  
146  Andhadhun  51615  8.4  
325  Pink (2016 film)  33902  8.2  
354  Dangal (film)  131338  8.4  
418  Masaan  19904  8.1  
426  Drishyam (2015 film)  58340  8.2  
436  Talvar (film)  26612  8.2  
469  Queen (2014 film)  56406  8.2  
536  Haider (film)  46912  8.1  
566  Ugly (film)  17483  8.1  
567  PK (film)  143605  8.1  
589  Vishwaroopam  38016  8.2  
612  Bhaag Milkha Bhaag  56205  8.2  
638  Shahid (film)  13537  8.3  
668  Paan Singh Tomar (film)  29994  8.2
```


669	Kahaani	53181	8.1
693	Gangs of Wasseypur	71636	8.2
694	Gangs of Wasseypur – Part 2	71636	8.2
709	Barfi!	70443	8.1
714	OMG – Oh My God!	46072	8.2
778	Zindagi Na Milegi Dobara	60826	8.1
869	Udaan (2010 film)	39567	8.2
912	3 Idiots	310481	8.4
930	Gulaal (film)	12799	8.1
1058	Black Friday (2007 film)	16761	8.5
1066	Chak De! India	68421	8.2
1127	Taare Zameen Par	148498	8.4
1180	Khosla Ka Ghosla	20538	8.3
1183	Lage Raho Munna Bhai	39486	8.1
1188	Omkara (2006 film)	17594	8.1
1195	Rang De Basanti	103071	8.2
1223	Black (2005 film)	31658	8.2
1252	Iqbal (film)	14864	8.1
1384	Swades	76737	8.2
1403	Munna Bhai M.B.B.S.	67148	8.1
1554	The Legend of Bhagat Singh	13455	8.1
1567	Lagaan	95686	8.1
1568	Lagaan	95686	8.1
1571	Dil Chahta Hai	62313	8.1

```
In [ ]: # Action movies with rating higher than 7.5
mask1 = movies[movies['genres'].str.split('|').apply(lambda x: 'Action' in x)]
mask1[mask1['imdb_rating'] > 7.5]
```

	title_x	imdb_id	poster_path	
0	Uri: The Surgical Strike	tt8291224	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikepe
41	Family of Thakurganj	tt8897986	https://upload.wikimedia.org/wikipedia/en/9/99...	https://en.wikepe
84	Mukkabaaz	tt7180544	https://upload.wikimedia.org/wikipedia/en/thum...	https://ei
106	Raazi	tt7098658	https://upload.wikimedia.org/wikipedia/en/thum...	htt

110	Parmanu: The Story of Pokhran	tt6826438	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi...
112	Bhavesh Joshi Superhero	tt6129302	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi...
169	The Ghazi Attack	tt6299040	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikiped...
219	Raag Desh (film)	tt6080746	https://upload.wikimedia.org/wikipedia/en/thum...	https://...
258	Irudhi Suttru	tt5310090	https://upload.wikimedia.org/wikipedia/en/f/fe...	https://en.wik...
280	Laal Rang	tt5600714		https://e...
297	Udta Punjab	tt4434004	https://upload.wikimedia.org/wikipedia/en/thum...	https://en...
354	Dangal (film)	tt5074352	https://upload.wikimedia.org/wikipedia/en/thum...	https://en...
362	Bajrangi Bhaijaan	tt3863552	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikip...
365	Baby (2015 Hindi film)	tt3848892	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikiped...
393	Detective Byomkesh Bakshi!	tt3447364	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikiped...
449	Titli (2014 film)	tt3019620	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wik...
536	Haider (film)	tt3390572	https://upload.wikimedia.org/wikipedia/en/thum...	https://en...

589	Vishwaroopam	tt2199711	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikiped
625	Madras Cafe	tt2855648	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.w
668	Paan Singh Tomar (film)	tt1620933	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi
693	Gangs of Wasseypur	tt1954470	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedia
694	Gangs of Wasseypur – Part 2	tt1954470	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedia
982	Jodhaa Akbar	tt0449994	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.w
1039	1971 (2007 film)	tt0983990	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikip
1058	Black Friday (2007 film)	tt0400234	https://upload.wikimedia.org/wikipedia/en/5/58...	https://en.wikip
1188	Omkaara (2006 film)	tt0488414	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikiped
1293	Sarkar (2005 film)	tt0432047	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikip
1294	Sehar	tt0477857	https://upload.wikimedia.org/wikipedia/en/thum...	http
1361	Lakshya (film)	tt0323013	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.w
1432	Gangaajal	tt0373856	https://upload.wikimedia.org/wikipedia/en/thum...	https://

1495	Company (film)	tt0296574	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wik
1554	The Legend of Bhagat Singh	tt0319736	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedia
1607	Nayak (2001 Hindi film)	tt0291376	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi

```
In [ ]: # another method
mask1 = movies['genres'].str.contains('Action')
mask2 = movies['imdb_rating'] > 7.5
movies[mask1 & mask2]
```

Out[]:	title_x	imdb_id	poster_path	
0	Uri: The Surgical Strike	tt8291224	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikip
41	Family of Thakurganj	tt8897986	https://upload.wikimedia.org/wikipedia/en/9/99...	https://en.wikip
84	Mukkabaaz	tt7180544	https://upload.wikimedia.org/wikipedia/en/thum...	https://en
106	Raazi	tt7098658	https://upload.wikimedia.org/wikipedia/en/thum...	htt
110	Parmanu: The Story of Pokhran	tt6826438	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi
112	Bhavesh Joshi Superhero	tt6129302	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi
169	The Ghazi Attack	tt6299040	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikiped
219	Raag Desh (film)	tt6080746	https://upload.wikimedia.org/wikipedia/en/thum...	https://
258	Irudhi Suttru	tt5310090	https://upload.wikimedia.org/wikipedia/en/f/fe...	https://en.wik

280	Laal Rang	tt5600714		NaN	https://e
297	Udta Punjab	tt4434004	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.
354	Dangal (film)	tt5074352	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.
362	Bajrangi Bhaijaan	tt3863552	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.wikip
365	Baby (2015 Hindi film)	tt3848892	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.wikiped
393	Detective Byomkesh Bakshi!	tt3447364	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.wikiped
449	Titli (2014 film)	tt3019620	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.wik
536	Haider (film)	tt3390572	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.
589	Vishwaroopam	tt2199711	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.wikiped
625	Madras Cafe	tt2855648	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.
668	Paan Singh Tomar (film)	tt1620933	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.wikiped
693	Gangs of Wasseypur	tt1954470	https://upload.wikimedia.org/wikipedia/en/thum...		https://en.wikipedia

694	Gangs of Wasseypur – Part 2	tt1954470	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedia
982	Jodhaa Akbar	tt0449994	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.w
1039	1971 (2007 film)	tt0983990	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikip
1058	Black Friday (2007 film)	tt0400234	https://upload.wikimedia.org/wikipedia/en/5/58...	https://en.wikip
1188	Omkaara (2006 film)	tt0488414	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikiped
1293	Sarkar (2005 film)	tt0432047	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikip
1294	Sehar	tt0477857	https://upload.wikimedia.org/wikipedia/en/thum...	http
1361	Lakshya (film)	tt0323013	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.w
1432	Gangaajal	tt0373856	https://upload.wikimedia.org/wikipedia/en/thum...	https://
1495	Company (film)	tt0296574	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wik
1554	The Legend of Bhagat Singh	tt0319736	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi
1607	Nayak (2001 Hindi film)	tt0291376	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi

Adding new cols

```
In [ ]: # completely new
movies['country'] = 'India'
movies.head()
```

```
Out[ ]:
```

	title_x	imdb_id	poster_path
0	Uri: The Surgical Strike	tt8291224	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/wik
1	Battalion 609	tt9472208	NaN https://en.wikipedia.o
2	The Accidental Prime Minister (film)	tt6986710	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/wiki
3	Why Cheat India	tt8108208	https://upload.wikimedia.org/wikipedia/en/thum... https://en.wikipedia.org/w
4	Evening Shadows	tt6028796	NaN https://en.wikipedia.org/wi

```
In [ ]: # from existing col
movies.isnull().sum()
```

```
Out[ ]:
```

title_x	0
imdb_id	0
poster_path	103
wiki_link	0
title_y	0
original_title	0
is_adult	0
year_of_release	0
runtime	0
genres	0
imdb_rating	0
imdb_votes	0
story	20
summary	0
tagline	1072
actors	5
wins_nominations	922
release_date	107
country	0
dtype: int64	

```
In [ ]: movies.dropna(inplace=True)
```

```
In [ ]: movies['lead actor'] = movies['actors'].str.split('|').apply(lambda x: x[0])
movies.head()
```

Out[]:	title_x	imdb_id	poster_path	
11	Gully Boy	tt2395469	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedia.c
34	Yeh Hai India	tt5525846	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedia.org/w
37	Article 15 (film)	tt10324144	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedia.org/wik
87	Aiyaary	tt6774212	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedi
96	Raid (2018 film)	tt7363076	https://upload.wikimedia.org/wikipedia/en/thum...	https://en.wikipedia.org/wiki

astype - change column type

```
In [ ]: # astype
ipl.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 950 entries, 0 to 949
Data columns (total 20 columns):
 #   Column                Non-Null Count  Dtype  
---  -
 0   ID                    950 non-null   int64  
 1   City                  899 non-null   object  
 2   Date                  950 non-null   object  
 3   Season                950 non-null   object  
 4   MatchNumber           950 non-null   object  
 5   Team1                 950 non-null   object  
 6   Team2                 950 non-null   object  
 7   Venue                 950 non-null   object  
 8   TossWinner            950 non-null   object  
 9   TossDecision          950 non-null   object  
10   SuperOver             946 non-null   object  
11   WinningTeam           946 non-null   object  
12   WonBy                 950 non-null   object  
13   Margin                932 non-null   float64 
14   method                19 non-null    object  
15   Player_of_Match       946 non-null   object  
16   Team1Players           950 non-null   object  
17   Team2Players           950 non-null   object  
18   Umpire1               950 non-null   object  
19   Umpire2               950 non-null   object
```



```
dtypes: float64(1), int64(1), object(18)
memory usage: 148.6+ KB
```

```
In [ ]: ipl['ID'] = ipl['ID'].astype('Int32')
ipl.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 950 entries, 0 to 949
Data columns (total 20 columns):
#   Column                Non-Null Count  Dtype
---  -
0   ID                    950 non-null   Int32
1   City                  899 non-null   object
2   Date                  950 non-null   object
3   Season                950 non-null   object
4   MatchNumber           950 non-null   object
5   Team1                 950 non-null   object
6   Team2                 950 non-null   object
7   Venue                 950 non-null   object
8   TossWinner            950 non-null   object
9   TossDecision          950 non-null   object
10  SuperOver             946 non-null   object
11  WinningTeam           946 non-null   object
12  WonBy                 950 non-null   object
13  Margin                932 non-null   float64
14  method                19 non-null    object
15  Player_of_Match       946 non-null   object
16  Team1Players           950 non-null   object
17  Team2Players           950 non-null   object
18  Umpire1               950 non-null   object
19  Umpire2               950 non-null   object
dtypes: Int32(1), float64(1), object(18)
memory usage: 145.8+ KB
```