pandas

July 28, 2024

0.0.1 What is Pandas

Pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.

https://pandas.pydata.org/about/index.html

0.1 Importing Pandas

[29]:
```
import pandas as pd
import numpy as np
```

0.2 Pandas Series

A Pandas Series is like a column in a table. It is a 1-D array holding data of any type. Since we did not specify an index for the data, a default one consisting of the integers 0 through N - 1 (where N is the length of the data) is created.

0.2.1 Series from lists

[30]:
```
# string datatype
countries = ['India', 'Nepal', 'Bhutan', 'Russia']

c = pd.Series(countries)
c
```

[30]:
```
0     India
1     Nepal
2    Bhutan
3    Russia
dtype: object
```

[31]:
```
# numeric datatype
runs = [87, 69, 92, 79, 84]

runs_ser = pd.Series(runs)
runs_ser
```

[31]:
```
0    87
1    69
2    92
3    79
4    84
dtype: int64
```

ser.array returns numpy array containing all the values.

[32]:
```
runs_ser.array
```

[32]:
```
<NumpyExtensionArray>
[87, 69, 92, 79, 84]
Length: 5, dtype: int64
```

[33]:
```
runs_ser.index
```

[33]:
```
RangeIndex(start=0, stop=5, step=1)
```

0.2.2 Series with custom indexing

- We can create series with custom indexing but the size should be equal.

[34]:
```
marks = [89, 78, 93, 91]
subjcets = ['maths', 'english', 'science', 'hindi']

marks_subjects = pd.Series(marks, index=subjcets)
marks_subjects
```

[34]:
```
maths      89
english    78
science    93
hindi      91
dtype: int64
```

0.2.3 Naming the Series

[35]:
```
# Dictionary
marks_dict = {
    'maths':84,
    'english':57,
    'science':89,
    'hindi':97
}

marks = pd.Series(marks_dict, name='Dilkhush\'s marks')
marks
```

[35]:
```
maths      84
english    57
science    89
hindi      97
Name: Dilkhush's marks, dtype: int64
```

0.3 Series Attributes

0.3.1 ser.size

- returns total number of non-Nan values in the series.

[36]:
```
marks
```

[36]:
```
maths      84
english    57
science    89
hindi      97
Name: Dilkhush's marks, dtype: int64
```

[37]:
```
marks.size
```

[37]:
```
4
```

0.3.2 ser.dtype

- returns the datatype of series
- datatype of series can be altered during creation.

[38]:
```
marks.dtype
```

[38]:
```
dtype('int64')
```

[39]:
```
marks = pd.Series(marks_dict, name='Dilkhush\'s Marks', dtype=int)
```

[40]:
```
marks.dtype
```

[40]:
```
dtype('int32')
```

[41]:
```
temp = pd.Series([1, 2, "Hello", 4.3])
```

[42]:
```
temp.dtype
```

[42]:
```
dtype('O')
```

[43]:
```
print(type(temp[0]))
print(type(temp[1]))
print(type(temp[2]))
```

```
print(type(temp[3]))
```

```
<class 'int'>
<class 'int'>
<class 'str'>
<class 'float'>
```

Note - NumPy Array contains values of one data type but Pandas series can contain values of different datatypes.

0.3.3 ser.name

- returns name of series

[44]:
```
marks.name
```

[44]:
```
"Dilkhush's Marks"
```

[45]:
```
temp.name
```

0.3.4 ser.index

- if the name of the index follows the python rules for variable naming then we can extract the values using ser.index

[46]:
```
marks_subjects
```

[46]:
```
maths      89
english    78
science    93
hindi      91
dtype: int64
```

[47]:
```
marks_subjects.maths
```

[47]:
```
89
```

0.3.5 ser.index

- return Index object containing indices of series in case of indexes are strings,
- if indices are numbers, RangeIndex is returned

[48]:
```
marks.index
```

[48]:
```
Index(['maths', 'english', 'science', 'hindi'], dtype='object')
```

[49]:
```
temp.index
```

[49]:
```
RangeIndex(start=0, stop=4, step=1)
```

0.3.6 ser.values

- returns ndarray containing values

[50]:
```
marks.values
```

[50]:
```
array([84, 57, 89, 97])
```

0.4 Reading Data

- We can read data using pandas inbuilt function, read_csv()
- read_csv() by default create DataFrame object, we can transform it into Series using squeeze function

Text and binary data loading functions in pandas | Function | Description | |————————|——| | read_csv | Load delimited data from a file, URL, or file-like object; use comma as default delimiter | | read_fwf | Read data in fixed-width column format (i.e., no delimiters) | | read_clipboard | Variation of read_csv that reads data from the clipboard; useful for converting tables from web pages | | read_excel | Read tabular data from an Excel XLS or XLSX file | | read_hdf | Read HDF5 files written by pandas | | read_html | Read all tables found in the given HTML document | | read_json | Read data from a JSON (JavaScript Object Notation) string representation, file, URL, or file-like object | | read_feather | Read the Feather binary file format | | read_orc | Read the Apache ORC binary file format | | read_parquet | Read the Apache Parquet binary file format | | read_pickle | Read an object stored by pandas using the Python pickle format | | read_sas | Read a SAS dataset stored in one of the SAS system's custom storage formats | | read_spss | Read a data file created by SPSS | | read_sql | Read the results of a SQL query (using SQLAlchemy) | | read_sql_table | Read a whole SQL table (using SQLAlchemy); equivalent to using a query that selects everything in that table using read_sql | | read_stata | Read a dataset from Stata file format | | read_xml | Read a table of data from an XML file |

[51]:
```
subs = pd.read_csv('subs.csv')
type(subs)
```

[51]:
```
pandas.core.frame.DataFrame
```

Note - pd.read_csv() directly reads the data into DataFrame but on that squeeze() function can be used to convert to Series

[52]:
```
subs = subs.squeeze()
```

[53]:
```
type(subs)
```

[53]:
```
pandas.core.series.Series
```

[54]:
```
subs
```

[54]:
```
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
```

[55]:
```
vk = pd.read_csv('kohli_ipl.csv', index_col='match_no', dtype=int)
vk = vk.squeeze()
vk
```

[55]:
```
match_no
1       1
2      23
3      13
4      12
5       1
       ..
211     0
212    20
213    73
214    25
215     7
Name: runs, Length: 215, dtype: int32
```

[56]:
```
movies = pd.read_csv('bollywood.csv', index_col='movie')
movies = movies.squeeze()
movies
```

[56]:
```
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
```

0.5 Series Methods

0.5.1 ser.head(n) or ser.tail(n)

- Return top or bottom n values from the series
- if n is not provided then by default n is 5

[57]:
```
subs.head()
```

[57]:
```
0    48
1    57
2    40
3    43
4    44
Name: Subscribers gained, dtype: int64
```

[58]:
```
vk.head(3)
```

[58]:
```
match_no
1     1
2    23
3    13
Name: runs, dtype: int32
```

[59]:
```
movies.tail(4)
```

[59]:
```
movie
Aankhen (2002 film)     Amitabh Bachchan
Saathiya (film)             Vivek Oberoi
Company (film)                Ajay Devgn
Awara Paagal Deewana        Akshay Kumar
Name: lead, dtype: object
```

0.5.2 ser.sample(n)

- returns randomly sampled n values.
- by default n is 1

[60]:
```
movies.sample()
```

[60]:
```
movie
Dhadak    Ishaan Khattar
Name: lead, dtype: object
```

[61]:
```
movies.sample(3)
```

[61]:
```
movie
Mad About Dance      Saahil Prem
Hotel Salvation     Adil Hussain
```

```
Why Cheat India    Emraan Hashmi
Name: lead, dtype: object
```

0.5.3 ser.value_counts()

- returns series of count of values

[62]:
```
movies.value_counts()
```

[62]:
```
lead
Akshay Kumar         48
Amitabh Bachchan     45
Ajay Devgn           38
Salman Khan          31
Sanjay Dutt          26
                     ..
Diganth               1
Parveen Kaur          1
Seema Azmi            1
Akanksha Puri         1
Edwin Fernandes       1
Name: count, Length: 566, dtype: int64
```

[63]:
```
vk.value_counts()
```

[63]:
```
runs
0     9
1     8
12    8
9     7
35    6
      ..
36    1
45    1
71    1
37    1
53    1
Name: count, Length: 78, dtype: int64
```

0.5.4 ser.sort_values(ascending = True, inplace = False)

- return sorted array on the basis of values.
- no change into original series.
- by default sort in ascencing order
- inplace is used to make permanent changes into original series

[64]:
```
vk.sort_values(ascending=False)
```

[64]:
```
match_no
128    113
126    109
123    108
164    100
120    100
       …
93       0
211      0
130      0
8        0
135      0
Name: runs, Length: 215, dtype: int32
```

[65]:
```
movies.sort_values() # Alphabetically in case of string
```

[65]:
```
movie
Qaidi Band                             Aadar Jain
Roar: Tigers of the Sundarbans      Aadil Chahal
Lipstick Under My Burkha            Aahana Kumra
Raat Gayi Baat Gayi?                Aamir Bashir
Talaash: The Answer Lies Within       Aamir Khan
                                             …
Dil Toh Deewana Hai                  Zeenat Aman
Sallu Ki Shaadi                      Zeenat Aman
Strings of Passion                   Zeenat Aman
Dunno Y… Na Jaane Kyon              Zeenat Aman
Taj Mahal: An Eternal Love Story     Zulfi Sayed
Name: lead, Length: 1500, dtype: object
```

0.5.5 ser.sort_index(ascending = True, inplace = False)

- works same as sort_values() but on index
- inplace parameter is used for making permanent changes into series

[66]:
```
movies.sort_index()
```

[66]:
```
movie
1920 (film)                   Rajniesh Duggall
1920: London                     Sharman Joshi
1920: The Evil Returns             Vicky Ahuja
1971 (2007 film)                Manoj Bajpayee
2 States (2014 film)              Arjun Kapoor
                                             …
Zindagi 50-50                      Veena Malik
Zindagi Na Milegi Dobara        Hrithik Roshan
Zindagi Tere Naam           Mithun Chakraborty
```

```
Zokkomon                         Darsheel Safary
Zor Lagaa Ke…Haiya!              Meghan Jadhav
Name: lead, Length: 1500, dtype: object
```

0.6 Mathematical Operations on Series

0.6.1 ser.count()

`Note` - count - only count all the values excluding missing values while size counts all

[67]:
```
vk.count() # total matches played by vk
```

[67]:
```
215
```

0.6.2 ser.sum()

- sum of all elements of series

[68]:
```
subs.sum() # total subscriber gained in last 365 days
```

[68]:
```
49510
```

0.6.3 ser.product()

- product of all elements of the series

[69]:
```
vk.prod()
```

[69]:
```
0
```

[70]:
```
vk.product()
```

[70]:
```
0
```

0.6.4 ser.mean()/median()/mode()/std()/var()

- apply stastical operations on series

[71]:
```
subs.mean() # average subscribers gain every day
```

[71]:
```
135.64383561643837
```

[72]:
```
vk.median() # median score
```

[72]:
```
24.0
```

[73]:
```
movies.mode() # frequent lead in movies
```

[73]:
```
0    Akshay Kumar
Name: lead, dtype: object
```

[74]:
```
subs.std()
```

[74]:
```
62.6750230372527
```

[75]:
```
vk.var()
```

[75]:
```
688.0024777222343
```

0.6.5 ser.min()/max()

- returns minimum or maximum element from the series

[76]:
```
subs.max()
```

[76]:
```
396
```

[77]:
```
vk.min()
```

[77]:
```
0
```

0.6.6 ser.describe()

- return series containing count, mean, std, min, 25%, 50%, 75% and max

[78]:
```
subs.describe()
```

[78]:
```
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
```

0.7 Indexing

- Values can be fetched using positional indexing or keyword indexing.
- Negative indexing only works if data type of index is string(object)

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

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

[80]:
```
x[0]
```

[80]:
```
12
```

[81]:
```
x[-1]
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
File c:\Program Files\Python312\Lib\site-packages\pandas\core\indexes\range.py:
 ↪413, in RangeIndex.get_loc(self, key)
    412 try:
--> 413     return self._range.index(new_key)
    414 except ValueError as err:

ValueError: -1 is not in range

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Cell In[81], line 1
----> 1 x[-1]

File c:\Program Files\Python312\Lib\site-packages\pandas\core\series.py:1111, i
 ↪Series.__getitem__(self, key)
   1108     return self._values[key]
   1110 elif key_is_scalar:
-> 1111     return self._get_value(key)
   1113 # Convert generator to list before going through hashable part
   1114 # (We will iterate through the generator there to check for slices)
   1115 if is_iterator(key):

File c:\Program Files\Python312\Lib\site-packages\pandas\core\series.py:1227, i
 ↪Series._get_value(self, label, takeable)
   1224     return self._values[label]
   1226 # Similar to Index.get_value, but we do not fall back to positional
-> 1227 loc = self.index.get_loc(label)
```

```
   1229 if is_integer(loc):
   1230     return self._values[loc]

File c:\Program Files\Python312\Lib\site-packages\pandas\core\indexes\range.py:
 ↪415, in RangeIndex.get_loc(self, key)
    413         return self._range.index(new_key)
    414     except ValueError as err:
--> 415         raise KeyError(key) from err
    416 if isinstance(key, Hashable):
    417     raise KeyError(key)

KeyError: -1
```

[ ]:
```
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
```

[ ]:
```
movies.iloc[-1]
```

[ ]:
```
'Akshay Kumar'
```

[ ]:
```
movies['Awara Paagal Deewana']
```

[ ]:
```
'Akshay Kumar'
```

0.7.1 Fancy Indexing

[ ]:
```
vk[[1,3,4,5]] # Scores of 1st, 3rd, 4th and 5th matches
```

[ ]:
```
match_no
1     1
3    13
4    12
5     1
```

```
Name: runs, dtype: int32
```

0.8 Slicing

- negative indexing can be used for slicing

[ ]:
```
vk[5:16]
```

[ ]:
```
match_no
6      9
7     34
8      0
9     21
10     3
11    10
12    38
13     3
14    11
15    50
16     2
Name: runs, dtype: int32
```

[ ]:
```
vk[-5:]
```

[ ]:
```
match_no
211     0
212    20
213    73
214    25
215     7
Name: runs, dtype: int32
```

[ ]:
```
movies[::2] # Alternet movies
```

[ ]:
```
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
```

0.9 Editing Series

[ ]:
```
marks
```

[ ]:
```
maths      84
english    57
science    89
hindi      97
Name: Dilkhush's Marks, dtype: int32
```

[ ]:
```
marks.iloc[1] = 100
```

[ ]:
```
marks
```

[ ]:
```
maths       84
english    100
science     89
hindi       97
Name: Dilkhush's Marks, dtype: int32
```

[ ]:
```
# What if an index does not exist
marks['sst'] = 87
```

[ ]:
```
marks
```

[ ]:
```
maths       84
english    100
science     89
hindi       97
sst         87
Name: Dilkhush's Marks, dtype: int32
```

[ ]:
```
# Slicing
runs_ser
```

[ ]:
```
0    87
1    69
2    92
3    79
4    84
dtype: int64
```

[ ]:
```
runs_ser[2:4] = 100
```

[ ]:
```
runs_ser
```

[ ]:
```
0     87
1     69
2    100
3    100
4     84
dtype: int64
```

[ ]:
```
# Fancy indexing
runs_ser[[0, 3, 4]] = 50
runs_ser
```

[ ]:
```
0     50
1     69
2    100
3     50
4     50
dtype: int64
```

[ ]:
```
runs_ser[[0, 3, 4]] = [0, 25, 40]
runs_ser
```

[ ]:
```
0      0
1     69
2    100
3     25
4     40
dtype: int64
```

[ ]:
```
# using index label
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
```

[ ]:
```
movies['2 States (2014 film)'] = 'Alia Bhatt'
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
```

0.10 Python functionalities on Series objects

[ ]:
```
# len/type/dir/sorted/max/min

len(subs)
```

[ ]:
```
365
```

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

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

[ ]:
```
dir(vk)
```

[ ]:
```
['T',
 '_AXIS_LEN',
 '_AXIS_ORDERS',
 '_AXIS_TO_AXIS_NUMBER',
 '_HANDLED_TYPES',
 '__abs__',
 '__add__',
 '__and__',
 '__annotations__',
 '__array__',
 '__array_priority__',
 '__array_ufunc__',
 '__bool__',
 '__class__',
 '__column_consortium_standard__',
 '__contains__',
 '__copy__',
 '__deepcopy__',
 '__delattr__',
```

```
'__delitem__',
'__dict__',
'__dir__',
'__divmod__',
'__doc__',
'__eq__',
'__finalize__',
'__float__',
'__floordiv__',
'__format__',
'__ge__',
'__getattr__',
'__getattribute__',
'__getitem__',
'__getstate__',
'__gt__',
'__hash__',
'__iadd__',
'__iand__',
'__ifloordiv__',
'__imod__',
'__imul__',
'__init__',
'__init_subclass__',
'__int__',
'__invert__',
'__ior__',
'__ipow__',
'__isub__',
'__iter__',
'__itruediv__',
'__ixor__',
'__le__',
'__len__',
'__lt__',
'__matmul__',
'__mod__',
'__module__',
'__mul__',
'__ne__',
'__neg__',
'__new__',
'__nonzero__',
'__or__',
'__pandas_priority__',
'__pos__',
'__pow__',
```

```
'__radd__',
'__rand__',
'__rdivmod__',
'__reduce__',
'__reduce_ex__',
'__repr__',
'__rfloordiv__',
'__rmatmul__',
'__rmod__',
'__rmul__',
'__ror__',
'__round__',
'__rpow__',
'__rsub__',
'__rtruediv__',
'__rxor__',
'__setattr__',
'__setitem__',
'__setstate__',
'__sizeof__',
'__str__',
'__sub__',
'__subclasshook__',
'__truediv__',
'__weakref__',
'__xor__',
'_accessors',
'_accum_func',
'_agg_examples_doc',
'_agg_see_also_doc',
'_align_for_op',
'_align_frame',
'_align_series',
'_append',
'_arith_method',
'_as_manager',
'_attrs',
'_binop',
'_cacher',
'_can_hold_na',
'_check_inplace_and_allows_duplicate_labels',
'_check_is_chained_assignment_possible',
'_check_label_or_level_ambiguity',
'_check_setitem_copy',
'_clear_item_cache',
'_clip_with_one_bound',
'_clip_with_scalar',
```

```
'_cmp_method',
'_consolidate',
'_consolidate_inplace',
'_construct_axes_dict',
'_construct_result',
'_constructor',
'_constructor_expanddim',
'_constructor_expanddim_from_mgr',
'_constructor_from_mgr',
'_data',
'_deprecate_downcast',
'_dir_additions',
'_dir_deletions',
'_drop_axis',
'_drop_labels_or_levels',
'_duplicated',
'_expanddim_from_mgr',
'_find_valid_index',
'_flags',
'_flex_method',
'_from_mgr',
'_get_axis',
'_get_axis_name',
'_get_axis_number',
'_get_axis_resolvers',
'_get_block_manager_axis',
'_get_bool_data',
'_get_cacher',
'_get_cleaned_column_resolvers',
'_get_index_resolvers',
'_get_label_or_level_values',
'_get_numeric_data',
'_get_rows_with_mask',
'_get_value',
'_get_values_tuple',
'_get_with',
'_getitem_slice',
'_gotitem',
'_hidden_attrs',
'_indexed_same',
'_info_axis',
'_info_axis_name',
'_info_axis_number',
'_init_dict',
'_init_mgr',
'_inplace_method',
'_internal_names',
```

```
'_internal_names_set',
'_is_cached',
'_is_copy',
'_is_label_or_level_reference',
'_is_label_reference',
'_is_level_reference',
'_is_mixed_type',
'_is_view',
'_is_view_after_cow_rules',
'_item_cache',
'_ixs',
'_logical_func',
'_logical_method',
'_map_values',
'_maybe_update_cacher',
'_memory_usage',
'_metadata',
'_mgr',
'_min_count_stat_function',
'_name',
'_needs_reindex_multi',
'_pad_or_backfill',
'_protect_consolidate',
'_reduce',
'_references',
'_reindex_axes',
'_reindex_indexer',
'_reindex_multi',
'_reindex_with_indexers',
'_rename',
'_replace_single',
'_repr_data_resource_',
'_repr_latex_',
'_reset_cache',
'_reset_cacher',
'_set_as_cached',
'_set_axis',
'_set_axis_name',
'_set_axis_nocheck',
'_set_is_copy',
'_set_labels',
'_set_name',
'_set_value',
'_set_values',
'_set_with',
'_set_with_engine',
'_shift_with_freq',
```

```
'_slice',
'_stat_function',
'_stat_function_ddof',
'_take_with_is_copy',
'_to_latex_via_styler',
'_typ',
'_update_inplace',
'_validate_dtype',
'_values',
'_where',
'abs',
'add',
'add_prefix',
'add_suffix',
'agg',
'aggregate',
'align',
'all',
'any',
'apply',
'argmax',
'argmin',
'argsort',
'array',
'asfreq',
'asof',
'astype',
'at',
'at_time',
'attrs',
'autocorr',
'axes',
'backfill',
'between',
'between_time',
'bfill',
'bool',
'case_when',
'clip',
'combine',
'combine_first',
'compare',
'convert_dtypes',
'copy',
'corr',
'count',
'cov',
```

```
'cummax',
'cummin',
'cumprod',
'cumsum',
'describe',
'diff',
'div',
'divide',
'divmod',
'dot',
'drop',
'drop_duplicates',
'droplevel',
'dropna',
'dtype',
'dtypes',
'duplicated',
'empty',
'eq',
'equals',
'ewm',
'expanding',
'explode',
'factorize',
'ffill',
'fillna',
'filter',
'first',
'first_valid_index',
'flags',
'floordiv',
'ge',
'get',
'groupby',
'gt',
'hasnans',
'head',
'hist',
'iat',
'idxmax',
'idxmin',
'iloc',
'index',
'infer_objects',
'info',
'interpolate',
'is_monotonic_decreasing',
```

```
'is_monotonic_increasing',
'is_unique',
'isin',
'isna',
'isnull',
'item',
'items',
'keys',
'kurt',
'kurtosis',
'last',
'last_valid_index',
'le',
'list',
'loc',
'lt',
'map',
'mask',
'max',
'mean',
'median',
'memory_usage',
'min',
'mod',
'mode',
'mul',
'multiply',
'name',
'nbytes',
'ndim',
'ne',
'nlargest',
'notna',
'notnull',
'nsmallest',
'nunique',
'pad',
'pct_change',
'pipe',
'plot',
'pop',
'pow',
'prod',
'product',
'quantile',
'radd',
'rank',
```

```
'ravel',
'rdiv',
'rdivmod',
'reindex',
'reindex_like',
'rename',
'rename_axis',
'reorder_levels',
'repeat',
'replace',
'resample',
'reset_index',
'rfloordiv',
'rmod',
'rmul',
'rolling',
'round',
'rpow',
'rsub',
'rtruediv',
'sample',
'searchsorted',
'sem',
'set_axis',
'set_flags',
'shape',
'shift',
'size',
'skew',
'sort_index',
'sort_values',
'squeeze',
'std',
'struct',
'sub',
'subtract',
'sum',
'swapaxes',
'swaplevel',
'tail',
'take',
'to_clipboard',
'to_csv',
'to_dict',
'to_excel',
'to_frame',
'to_hdf',
```

```
 'to_json',
 'to_latex',
 'to_list',
 'to_markdown',
 'to_numpy',
 'to_period',
 'to_pickle',
 'to_sql',
 'to_string',
 'to_timestamp',
 'to_xarray',
 'transform',
 'transpose',
 'truediv',
 'truncate',
 'tz_convert',
 'tz_localize',
 'unique',
 'unstack',
 'update',
 'value_counts',
 'values',
 'var',
 'view',
 'where',
 'xs']
```

[ ]:
```
sorted(subs)
```

[ ]:
```
[33,
 33,
 35,
 37,
 39,
 40,
 40,
 40,
 40,
 42,
 42,
 43,
 44,
 44,
 44,
 45,
 46,
 46,
```

48,
49,
49,
49,
49,
50,
50,
50,
51,
54,
56,
56,
56,
56,
57,
61,
62,
64,
65,
65,
66,
66,
66,
66,
67,
68,
70,
70,
70,
71,
71,
72,
72,
72,
72,
72,
73,
74,
74,
75,
76,
76,
76,
76,
77,
77,
78,

78,
78,
79,
79,
80,
80,
80,
81,
81,
82,
82,
83,
83,
83,
84,
84,
84,
85,
86,
86,
86,
87,
87,
87,
87,
88,
88,
88,
88,
88,
89,
89,
89,
90,
90,
90,
90,
91,
92,
92,
92,
93,
93,
93,
93,
95,
95,

```
96,
96,
96,
96,
97,
97,
98,
98,
99,
99,
100,
100,
100,
101,
101,
101,
102,
102,
103,
103,
104,
104,
104,
105,
105,
105,
105,
105,
105,
105,
105,
105,
108,
108,
108,
108,
108,
108,
108,
109,
109,
110,
110,
110,
111,
111,
112,
113,
```

113,
113,
114,
114,
114,
114,
115,
115,
115,
115,
117,
117,
117,
118,
118,
119,
119,
119,
119,
120,
122,
123,
123,
123,
123,
123,
124,
125,
126,
127,
128,
128,
129,
130,
131,
131,
132,
132,
134,
134,
134,
135,
135,
136,
136,
136,
137,

138,
138,
138,
139,
140,
144,
145,
146,
146,
146,
146,
147,
149,
150,
150,
150,
150,
151,
152,
152,
152,
153,
153,
153,
154,
154,
154,
155,
155,
156,
156,
156,
156,
157,
157,
157,
157,
158,
158,
159,
159,
160,
160,
160,
160,
162,
164,

166,
167,
167,
168,
170,
170,
170,
170,
171,
172,
172,
173,
173,
173,
174,
174,
175,
175,
176,
176,
177,
178,
179,
179,
180,
180,
180,
182,
183,
183,
183,
184,
184,
184,
185,
185,
185,
185,
186,
186,
186,
188,
189,
190,
190,
192,
192,

```
192,
196,
196,
196,
197,
197,
202,
202,
202,
203,
204,
206,
207,
209,
210,
210,
211,
212,
213,
214,
216,
219,
220,
221,
221,
222,
222,
224,
225,
225,
226,
227,
228,
229,
230,
231,
233,
236,
236,
237,
241,
243,
244,
245,
247,
249,
254,
```

```
254,
258,
259,
259,
261,
261,
265,
267,
268,
269,
276,
276,
290,
295,
301,
306,
312,
396]
```

[ ]:
```
min(subs)
```

[ ]: 33

[ ]:
```
max(subs)
```

[ ]: 396

[ ]:
```
# type conversion
list(marks)
```

[ ]: [84, 100, 89, 97, 87]

[ ]:
```
dict(marks)
```

[ ]: {'maths': 84, 'english': 100, 'science': 89, 'hindi': 97, 'sst': 87}

[ ]:
```
# membership operator

'2 States (2014 film)' in movies # checks index values
```

[ ]: True

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

[ ]: True

[ ]:
```
# Looping
for i in movies: # returns values
    print(i)
```

```
Vicky Kaushal
Vicky Ahuja
Anupam Kher
Emraan Hashmi
Mona Ambegaonkar
Geetika Vidya Ohlyan
Arshad Warsi
Radhika Apte
Kangana Ranaut
Nawazuddin Siddiqui
Ali Asgar
Ranveer Singh
Prit Kamani
Ajay Devgn
Sushant Singh Rajput
Amitabh Bachchan
Abhimanyu Dasani
Talha Arshad Reshi
Nawazuddin Siddiqui
Garima Agarwal
Rasika Agashe
Barun Sobti
Akshay Kumar
Zaheer Iqbal
Vidyut Jammwal
Deepika Amin
Manav Kaul
Naseeruddin Shah
Varun Dhawan
Shreyas Talpade
Tiger Shroff
Boman Irani
Ajay Devgn
Arjun Kapoor
Gavie Chahal
Prabhu Deva
Shahid Kapoor
Ayushmann Khurrana
Anupam Kher
Karanvir Bohra
Hrithik Roshan
Jimmy Sheirgill
John Abraham
```

Rishi Kapoor
Kangana Ranaut
Natalia Janoszek
Diljit Dosanjh
Sidharth Malhotra
Rajeev Khandelwal
Zaira Wasim
Akshay Kumar
Jacqueline Fernandez
Ayushmann Khurrana
Akshaye Khanna
Sonam Kapoor
Karan Deol
Sanjay Dutt
Bhavesh Kumar
Sanaya Irani
Ayushmann Khurrana
Siddhanth Kapoor
Akshay Kumar
Taapsee Pannu
Rajkummar Rao
Sunny Singh Nijjar
Neil Nitin Mukesh
Suraj Pancholi
Boman Irani
Riteish Deshmukh
Nawazuddin Siddiqui
Shahbaaz Khan
Kriti Kharbanda
Naseeruddin Shah
Vardhan Puri
Sushant Singh Rajput
Kartik Aaryan
Vidyut Jammwal
Rani Mukerji
Salman Khan
Akshay Kumar
Saif Ali Khan
Kay Kay Menon
Nora Fatehi
Ashmit Patel
Viineet Kumar
Rahul Bhat
Vicky Kaushal
Sidharth Malhotra
Deepika Padukone
Geetanjali Thapa
Akshay Anand

Pulkit Samrat
Kartik Aaryan
Lee Byford
Taapsee Pannu
Aisha Ahmed
Ajay Devgn
Rani Mukerji
Manoj Bajpayee
Tiger Shroff
Varun Dhawan
Prabhu Deva
Ishaan Khattar
Abhay Deol
Yogesh Raj Mishra
Rajkummar Rao
Alia Bhatt
Naseeruddin Shah
Sumeet Vyas
Vinay Pathak
John Abraham
Danny Denzongpa
Harshvardhan Kapoor
Jimmy Sheirgill
Anil Kapoor
Ishaan Khattar
Ranbir Kapoor
Sanjay Dutt
Dharmesh Yelande
Taapsee Pannu
Arjun Mathur
Irrfan Khan
Akshay Kumar
John Abraham
Sonakshi Sinha
Utkarsh Sharma
Dharmendra
Rajkummar Rao
Jackie Shroff
Avinash Tiwary
Manoj Bajpayee
Paoli Dam
Sanya Malhotra
Shahid Kapoor
Abhishek Bharate
Nawazuddin Siddiqui
Manish Anand
Taapsee Pannu
Jackky Bhagnani

Anushka Sharma
Radhika Apte
Rhea Chakraborty
Govinda
Sohum Shah
Kajol
Arjun Kapoor
Ayushmann Khurrana
Ayushmann Khurrana
Nargis Fakhri
Aishwarya Devan
Neil Nitin Mukesh
Shakti Kapoor
Amit Sadh
Sunny Deol
Rahul Bagga
Sunny Deol
Amyra Dastur
Shah Rukh Khan
Ranveer Singh
Salman Khan
Ajay Devgn
Varun Dhawan
Shraddha Kapoor
Sunil Grover
Hrithik Roshan
Raj Arjun
Aamir Khan
Gurmeet Ram Rahim Singh
Arsh Bajwa
Rana Daggubati
Naseeruddin Shah
Kangana Ranaut
Nana Patekar
Arbaaz Khan
Varun Dhawan
Rajkummar Rao
Govinda
Rajat Kapoor
Anushka Sharma
Kiara Advani
Shaurya Singh
Pankaj Tripathi
Taapsee Pannu
Adil Hussain
Amitabh Bachchan
Sunny Leone
Hema Malini

Raveena Tandon
Amitabh Bachchan
Amardeep Insan
Shraddha Kapoor
Ayushmann Khurrana
Sachin Tendulkar
Irrfan Khan
Himansh Kohli
Adil Hussain
Jayesh Raj
Manisha Koirala
Deepika Padukone
Rajkummar Rao
Salman Khan
Riteish Deshmukh
Shiv Darshan
Ranbir Kapoor
Ashish Bisht
Aahana Kumra
Manoj Babani
Rajveer Ankur Singh
Kirti Kulhari
Shah Rukh Khan
Tiger Shroff
Akshay Kumar
Anil Kapoor
Kartik Aaryan
Ayushmann Khurrana
Nawazuddin Siddiqui
Prisha Aneja
Aadar Jain
Ayushmann Khurrana
Kunal Kapoor
Arjun Rampal
Kangana Ranaut
Farhan Akhtar
Sidharth Malhotra
Hugh Bonneville
Rishi Kapoor
Rajkummar Rao
Kunaal Roy Kapur
Sunny Leone
Shraddha Kapoor
Rina Charaniya
Nawazuddin Siddiqui
Sunny Deol
Sridevi
Saif Ali Khan

Soundarya Sharma
Sudha Chandran
Manoj Bajpayee
Zaira Wasim
Prakash Belawadi
Kalki Koechlin
Rajkummar Rao
Richa Chadha
Irrfan Khan
Zareen Khan
Nayna Bandhopadhyay
Vidya Balan
Nishikant Kamat
Sanjay Mishra
Kapil Sharma
Pulkit Samrat
Vijay Varma
Sushama Deshpande
Richa Chadha
Prince Shah
Tanima Bhattacharya
Akshay Kumar
Zeenat Aman
Madhavan
Tusshar Kapoor
Tusshar Kapoor
Himansh Kohli
Sadhana Singh
Sunny Deol
Aditya Roy Kapoor
Rishi Kapoor
Rajniesh Duggall
Krishna Chaturvedi
Girish Taurani
Sonam Kapoor
Sukhesh Arora
Raima Sen
Anuj Sachdeva
Parthaa Akerkar
Priyanka Chopra
Sidharth Malhotra
John Abraham
Patralekhaa Paul
Shah Rukh Khan
Swara Bhaskar
Randeep Hooda
Shraddha Kapoor
Pankaj Tripathi

Jimmy Sheirgill
Kartik Elangovan
Boman Irani
Manoj Bajpayee
Sharman Joshi
Emraan Hashmi
Aanchal Dwivedi
Sanjay Singh
Arvind Swamy
Radhika Apte
Randeep Hooda
Sachiin Joshi
Kajal Aggarwal
Amitabh Bachchan
Shahid Kapoor
Hazel Croney
V. Ravichandran
Vipin Sharma
Nawazuddin Siddiqui
Pulkit Samrat
Tannishtha Chatterjee
Krrish Chhabria
Avinash Dhyani
Zeenat Aman
Salman Khan
Shashank Arora
Urvashi Rautela
Naseeruddin Shah
Sara Loren
Tom Alter
Irrfan Khan
John Abraham
Rajeev Khandelwal
Ileana D'Cruz
Sahil Anand
Hrithik Roshan
Diana Penty
Niharica Raizada
Tiger Shroff
Sidharth Malhotra
Nawazuddin Siddiqui
Emraan Hashmi
Taapsee Pannu
Tannishtha Chatterjee
Riteish Deshmukh
Yash Soni
Shreyas Talpade
Vinay Pathak

Jimmy Sharma
Sushant Singh Rajput
Ashok Insan
RJ Balaji
Saurav Chakraborty
Shashank Udapurkar
Shubham
Manoj Bajpayee
Sunny Leone
Aashish Bhatt
Ajay Devgn
Ronit Roy
Ranbir Kapoor
Neha Sharma
Farhan Akhtar
John Abraham
Alia Bhatt
Vaani Kapoor
Neha Dhupia
Jimmy Sheirgill
Sonarika Bhadoria
Amitabh Bachchan
Sharman Joshi
Aamir Khan
Harshvardhan Kapoor
Salman Khan
Kangana Ranaut
Kangana Ranaut
Soha Ali Khan
Prabhu Deva
Shah Rukh Khan
Salman Khan
Ranveer Singh
Anil Kapoor
Akshay Kumar
Akshay Kumar
Akshay Kumar
Akshay Kumar
Shakti Kapoor
Bipasha Basu
Arjun Kapoor
Zayed Khan
Malaika Arora
Naman Jain
Gurmeet Choudhary
Kay Kay Menon
Rati Agnihotri
Amitabh Bachchan

Arjun Rampal
Varun Dhawan
Swanand Kirkire
Sulabha Arya
Irrfan Khan
Nana Patekar
Ayushmann Khurrana
Anupam Kher
Gurmeet Ram Rahim Singh
Sidhant Gupta
Arjun Mathur
Ganesh Acharya
Anushka Sharma
Gulshan Devaiah
Sushant Singh Rajput
Adhyayan Suman
Vira Sathidar
Kalki Koechlin
Sunny Leone
Sara Loren
Emraan Hashmi
Auroshika Dey
Ira Dubey
Naseeruddin Shah
Vinay Pathak
Ram Kapoor
Deepika Padukone
Ranbir Kapoor
Shakti Anand
Anil Kapoor
Arshad Warsi
Meenakshi Dixit
Yash Acharya
Rajkummar Rao
Rahul Bagga
Mohit Baghel
Rishi Verma
Mimoh Chakraborty
Swara Bhaskar
Richa Chadha
Arshad Warsi
Mugdha Godse
Yashpal Sharma
Dharmendra
Sunny Deol
Smitha Gondkar
Kunal Kapoor
Ajay Devgn

Jacqueline Fernandez
Rishi Kapoor
Akshay Kumar
Vinay Pathak
Nawazuddin Siddiqui
Bhavita Anand
Saif Ali Khan
Suraj Pancholi
Shamim Khan
Irrfan Khan
Suhaas Ahuja
Jaideep Ahlawat
Charanpreet Insan
Akanksha Puri
Kunal Khemu
Aishwarya Rai Bachchan
Seema Azmi
Parveen Kaur
Kapil Sharma
Kartik Aaryan
Diganth
Shahid Kapoor
Nawazuddin Siddiqui
Kunal Khemu
Mann Bagga
Manish Paul
Sanjeev Kumar
Madhuri Dixit
Shiv Darshan
Gopi Desai
Mohinder Gujral
Zeenat Aman
Ranveer Singh
Salman Khan
Sidharth Malhotra
Adhyayan Suman
Indrapal Ahuja
Jimmy Sheirgill
Abhay Deol
Sahil Anand
Alia Bhatt
Sampat Pal Devi
Farhan Akhtar
Madhuri Dixit
Kangana Ranaut
Ayushmann Khurrana
Ali Zafar
Mahek Chahal

Monali Thakur
Sunny Leone
Harman Baweja
Sanjay Mishra
Sharman Joshi
Sachin Khedekar
Leeza Mangaldas
Pulkit Samrat
Zara Sheikh
Alia Bhatt
Purab Kohli
Amitabh Bachchan
Varun Dhawan
Arvinder Bhatti
Kanika Batra
Jackky Bhagnani
Rajeev Khandelwal
Tanuj Virwani
Vijay Raaz
Kannan Arunachalam
Anjori Alagh
Satish Kaushik
Rahul Bagga
Himesh Reshammiya
Farooq Shaikh
Makrand Deshpande
Eesha Agarwal
Siddharth Gupta
Tiger Shroff
Rajkummar Rao
Sharib Hashmi
Kangana Ranaut
Kartik Aaryan
Swara Bhaskar
Simer Motiani
Anshuman Jha
Sidharth Malhotra
Vidya Balan
Saif Ali Khan
Varun Dhawan
Jay Bhanushali
Armaan Jain
Rajesh Khanna
Vir Das
Akshay Kumar
Jimmy Sheirgill
Reshmi Ghosh
Akshay Oberoi

Akshay Kumar
Anupam Kher
Rani Mukerji
Emraan Hashmi
Priyanka Chopra
Bipasha Basu
Deepika Padukone
Sonam Kapoor
Salil Acharya
Salman Khan
Saahil Prem
Alieesa P Badresia
Manoj Amarnani
Sasha Aagha
Tabu
Hrithik Roshan
Rati Agnihotri
Aditya Roy Kapoor
Asrani
Harshvardhan Deo
Nikhil Dwivedi
Karanvir Bohra
Puru Chibber
Soha Ali Khan
Rhea Chakraborty
Shah Rukh Khan
Rekha
Anupam Kher
Randeep Hooda
Akshay Kumar
Aadil Chahal
Shabana Azmi
Dimple Kapadia
Nishant Dahiya
Ranveer Singh
Emraan Hashmi
Saif Ali Khan
Vinod Acharya
Mannara Chopra
Prabhas
Mischa Barton
Shiv Panditt
Annu Kapoor
Barun Sobti
Rahul Bhat
Aamir Khan
Adhyayan Suman
Imran Khan

Naveen Kasturia
Arjun Rampal
Sarita Joshi
Kartik Aaryan
Juhi Chawla
Saif Ali Khan
Manisha Kelkar
Farooq Shaikh
Akshay Kumar
Randeep Hooda
Vivek Oberoi
Rajkummar Rao
Akash
Vishwa Mohan Badola
Neil Nitin Mukesh
Saqib Saleem
Arshad Warsi
Jimmy Sheirgill
Asha Bhosle
Kamal Haasan
Jackky Bhagnani
Aditya Roy Kapoor
Emraan Hashmi
Ajay Devgn
Ayushmann Khurrana
Vivek Oberoi
Vidyut Jammwal
Rani Mukerji
Saif Ali Khan
Riya Vij
Arjun Kapoor
Preity Zinta
Veena Malik
Ranbir Kapoor
Pulkit Samrat
Rupa Bhimani
Dhanush
Kay Kay Menon
Neil Nitin Mukesh
Vidya Balan
Anupam Kher
Sanjay Dutt
Farhan Akhtar
Wamiqa Gabbi
Shadab Kamal
Sunil Shetty
Prateik
Tusshar Kapoor

Tanuj Virwani
Poonam Pandey
Deepak Dobriyal
Hemant Gopal
Joy Mukherjee
Rishi Kapoor
Rajesh Tailang
Akshay Kumar
John Abraham
Amitabh Bachchan
Sushant Singh Rajput
Shruti Haasan
Deepika Padukone
Vivek Oberoi
Naseeruddin Shah
Karan Kundrra
Shahid Kapoor
Aida Elkashef
Irrfan Khan
Anisa Butt
Akshay Kumar
Rajkummar Rao
Manish Paul
Puneet Singh Ratn
Kangana Ranaut
Sumit Nijhawan
Tinnu Anand
Chandan Roy Sanyal
Elisha Kriis
Ashmit Patel
Imran Khan
Manu Rishi Chadha
Naseeruddin Shah
Paresh Rawal
Saif Ali Khan
Aamir Khan
Ali Zafar
Ranveer Singh
Sharman Joshi
Naseeruddin Shah
Shiney Ahuja
Karan Sharma
Sunny Deol
Ranveer Singh
Hrithik Roshan
Kareena Kapoor
Prateik
Kiran Bhatia

Bipasha Basu
Riteish Deshmukh
Jahangir Khan
Irrfan Khan
Vidya Balan
Mithun Chakraborty
Kareena Kapoor
Manish Chaudhary
Kavin Dave
Shikhi Gupta
Archana Joglekar
Tusshar Kapoor
Pulkit Samrat
Ayushmann Khurrana
Nikhil Dwivedi
Anil Kapoor
Karisma Kapoor
Arjun Kapoor
Sanjay Dutt
Gul Panag
Yudhveer Bakoliya
Kay Kay Menon
Emraan Hashmi
Sharman Joshi
Shahid Kapoor
Anupam Kher
Anya Anand
Julia Datt
Manoj Bajpayee
Manoj Bajpayee
Saif Ali Khan
Naresh Sharma
Riteish Deshmukh
Sonu Sood
Nassar Abdulla
Salman Khan
Vickrant Mahajan
Freny Bhagat
Akshay Kumar
Omkar Das Manikpuri
Sunil Shetty
Manoj Bajpayee
Bidita Bag
Harish Chabbra
Ranbir Kapoor
Kareena Kapoor
Sagar Bhangade
Vivek Oberoi

Ravi Kishan
Paresh Rawal
Rani Mukerji
Manoj Bajpayee
J.D. Chakravarthi
Cary Elwes
Arjun Rampal
Alia Bhatt
Jackky Bhagnani
Emraan Hashmi
Vicky Ahuja
Sunidhi Chauhan
Asrani
Shah Rukh Khan
Aamir Khan
Himanshu Bhatt
Ajay Devgn
Ashok Banthia
Salman Khan
Vinod Khanna
Akshay Kumar
Amitabh Bachchan
Sridevi
Vivek Sudershan
Nafisa Ali
Mohsin
Prateik
Gul Panag
Vatsal Sheth
Ajay Devgn
Russell Geoffrey Banks
Vinay Pathak
Rishi Kapoor
Priyanka Chopra
Madhavan
Arshad Warsi
Deepti Naval
Akshay Kumar
Sumit Arora
Anay
Om Puri
Abhishek Bachchan
Sendhil Ramamurthy
Darsheel Safary
Vinay Pathak
Ishaan Manhaas
Sanjay Mishra
Ashutosh Rana

Juhi Chawla
Govinda
Mimoh Chakraborty
Tusshar Kapoor
Kainaz Motivala
Partho A. Gupte
Nana Patekar
Sara Arjun
Kalki Koechlin
Kartik Aaryan
Vinay Pathak
Lillete Dubey
Sanjay Dutt
Amitabh Bachchan
Imran Khan
Emraan Hashmi
Aarav Khanna
Hrithik Roshan
Raghuvir Yadav
Gulshan Grover
Naushaad Abbas
Shefali Shah
Ajay Devgn
Rahul Jaiswal
Apoorva Arora
Isha Koppikar
Rajniesh Duggall
Amitabh Bachchan
Sanjay Dutt
Anupam Kher
Salman Khan
Deepshika Nagpal
Deepak Dobriyal
Tusshar Kapoor
Kiron Kher
Imran Khan
Shahid Kapoor
Mikaal
John Abraham
Jimmy Sheirgill
Vinay Pathak
Vinay Virmani
Dev Anand
Zayed Khan
Saqib Saleem
Sachiin Joshi
Shah Rukh Khan
Ranbir Kapoor

Chirag Paswan
Dharmendra
Himesh Reshammiya
Ranveer Singh
Akshay Kumar
Abhishek Bachchan
Rani Mukerji
Ajay Devgn
Vidya Balan
Priyanka Chopra
Shahid Kapoor
Prateik
Salman Khan
Siddharth
Amitabh Bachchan
Vidya Balan
Paresh Rawal
Sunny Gill
Shreyas Talpade
Tabu
Shah Rukh Khan
Amitabh Bachchan
Siddhartha Gupta
Shah Rukh Khan
Tanushree Dutta
Paul Sidhu
Sunny Deol
Gurdas Maan
Master Shams
Om Puri
Vikrum Kumar
Aditya Narayan
Arshad Warsi
Boman Irani
Nana Patekar
Priyanka Chopra
Arbaaz Khan
Rekha
Rahul Bose
Nana Patekar
Sudeep
Farooq Shaikh
Rohit Roy
Sachin Khedekar
Anuj Saxena
Akshay Kumar
Hazel Croney
Sanjeev Bhaskar

Vivek Oberoi
Abhishek Bachchan
Atul Kulkarni
Rajpal Yadav
Hrithik Roshan
Nushrat Bharucha
Kareena Kapoor
Bhushan Agarwal
Prashant Narayanan
Sanjay Dutt
Akshay Kumar
Ali Zafar
Rajat Barmecha
Ajay Devgn
Bobby Deol
Omkar Das Manikpuri
Neil Nitin Mukesh
Gul Panag
Raj Singh Chaudhary
John Abraham
Barkha Madan
Salman Khan
Supriya Pathak
Sunil Shetty
Rituparna Sengupta
Ranvir Shorey
Mallika Sherawat
Emraan Hashmi
Rishi Kapoor
Ajay Devgn
Manoj Bajpayee
Sanjay Dutt
John Abraham
Hrithik Roshan
Varun Bhagwat
Deepika Padukone
Sunny Deol
Rajat Kapoor
Akanksha
Jimmy Sheirgill
Sanjay Dutt
Ranveer Singh
Aditya Srivastava
Arunoday Singh
Akshay Kumar
Akshay Oberoi
Ajay Devgn
Neil Nitin Mukesh

Sahil Khan
Imran Khan
Sushmita Sen
Priyanka Chopra
Zeenat Aman
Bipasha Basu
Akshay Kumar
Aamir Khan
Farhan Akhtar
Saif Ali Khan
Salman Khan
Waheeda Rehman
Kangana Ranaut
Hrishitaa Bhatt
Ranbir Kapoor
Asrani
Irrfan Khan
Kay Kay Menon
Arshad Warsi
Paresh Rawal
Rati Agnihotri
Harman Baweja
Shahid Kapoor
Fardeen Khan
John Abraham
Kay Kay Menon
Aftab Shivdasani
Naseeruddin Shah
Inaamulhaq
Neil Nitin Mukesh
Kunal Khemu
Bobby Deol
Anita
Parzaan Dastur
Meghan Jadhav
Shreyas Talpade
John Abraham
Kay Kay Menon
Sanjay Dutt
Mithun Chakraborty
Govinda
Sunil Shetty
Jackie Shroff
Anupam Kher
Shreyas Talpade
Rishi Kapoor
Rajendra Prasad
Arjun Rampal

Mithun Chakraborty
Mithun Chakraborty
Bobby Deol
Shahid Kapoor
Harman Baweja
Fardeen Khan
Sanjay Dutt
Ranbir Kapoor
Salman Khan
Akshay Kumar
Atmaram Bhende
Amitabh Bachchan
Salman Khan
Manoj Bajpayee
Emraan Hashmi
Lewis Tan
Akshay Kumar
Amitabh Bachchan
Ranbir Kapoor
Aamir Bashir
Farooq Shaikh
Mithun Chakraborty
Abhishek Bachchan
Saif Ali Khan
Shah Rukh Khan
Aamir Khan
Akshay Kumar
Ajay Devgn
Hrithik Roshan
Ranbir Kapoor
Amitabh Bachchan
Amitabh Bachchan
Ajay Devgn
Omkar Bhatkar
Shreyas Talpade
Manisha Koirala
Ajay Devgn
Tusshar Kapoor
Juhi Chawla
Ajay Devgn
Kay Kay Menon
Akshay Kumar
Dino Morea
Mimoh Chakraborty
Emraan Hashmi
Mithun Chakraborty
Sikander Kher
Paresh Rawal

Ahraz Ahmed
Aftab Shivdasani
Amita Pathak
Saahil Chadha
Saif Ali Khan
Kay Kay Menon
Shahid Kapoor
Harman Baweja
Adhvik Mahajan
Vivek Oberoi
Govinda
Amitabh Bachchan
Paresh Rawal
Mallika Sherawat
Arjun Rampal
Tusshar Kapoor
Bobby Deol
Sammir Dattani
Victor Banerjee
Rajniesh Duggall
Amitabh Bachchan
Shreyas Talpade
Kiron Kher
Jackie Shroff
Jayshree Arora
Sharman Joshi
Himesh Reshammiya
Sohail Khan
Saif Ali Khan
Isha Koppikar
Kamal Rashid Khan
Priyanka Chopra
Vinay Pathak
Salman Khan
Abhay Deol
Vinay Pathak
Shabana Azmi
Manoj Bajpayee
Anupam Kher
Rajesh Khanna
Dino Morea
Irrfan Khan
Amitabh Bachchan
Himesh Reshammiya
Tusshar Kapoor
Nauheed Cyrusi
Madhuri Dixit
Dharmendra

Emraan Hashmi
Vinay Pathak
Akshay Kumar
Dino Morea
Sunny Deol
Mouli Ganguly
Krishna Abhishek
Irrfan Khan
Kay Kay Menon
Pankaj Kapur
Jazzy Doe
Mona Ambegaonkar
Ajay Devgn
Amitabh Bachchan
Jimmy Sheirgill
Fardeen Khan
Shah Rukh Khan
Sanjay Dutt
John Abraham
Pankaj Kapur
Sunil Pal
Muzamil Ibrahim
Imaaduddin Shah
Sharman Joshi
Abhay Deol
Arjun Rampal
Akshaye Khanna
Rishi Kapoor
Arbaaz Khan
Amitabh Bachchan
Gautam Gupta
Mohan Azaad
Juanna Sanghvi
Mithun Chakraborty
Ranvir Shorey
Govinda
Abhishek Bachchan
Shahid Kapoor
Tusshar Kapoor
Shahid Kapoor
Shiney Ahuja
Dharmendra
Fardeen Khan
Tusshar Kapoor
Rekha
Jimmy Sheirgill
Rani Mukerji
Dharmendra

Shabana Azmi
Abhay Deol
Salman Khan
Menekka Arora
Nana Patekar
Bobby Deol
Linda Arsenio
Kal Penn
Akshay Kumar
Jason Lewis
Sanjay Dutt
Amitabh Bachchan
John Abraham
Shah Rukh Khan
Salman Khan
Rahul Khanna
Dwij Yadav
Aftab Shivdasani
Vinod Khanna
Bobby Deol
Sherlyn Chopra
Salman Khan
Rani Mukerji
Mahima Chaudhry
Mandar Jadhav
Aftab Shivdasani
Amitabh Bachchan
Jimmy Sheirgill
Manoj Bajpayee
Darsheel Safary
Saif Ali Khan
Urmila Matondkar
Kunal Khemu
Emraan Hashmi
Tushar Jalota
Jaya Bachchan
Akshay Kumar
Akshaye Khanna
Jeetendra
Akshaye Khanna
Abhay Deol
Emraan Hashmi
Akshay Kapoor
Sanjay Dutt
Bobby Darling
Aftab Shivdasani
Rekha
Amitabh Bachchan

Rati Agnihotri
Rahul Dev
Naseeruddin Shah
Urmila Matondkar
Juhi Chawla
Akshay Kumar
Sushmita Sen
Jackie Shroff
Somesh Agarwal
Sohail Khan
Kareena Kapoor
Bipasha Basu
Amitabh Bachchan
Rajit Kapoor
Emraan Hashmi
Ishrat Ali
Raj Tara
Hrithik Roshan
Ayesha Takia
Shah Rukh Khan
Amitabh Bachchan
Aamir Khan
Kangana Ranaut
Ajay Devgn
Ram Kapoor
Sunil Shetty
Ajay Devgn
Akshay Kumar
Arjun Rampal
Arjun Rampal
Aftab Shivdasani
Salman Khan
Emraan Hashmi
Onjolee Nair
Amarjeet
Anupam Kher
Amitabh Bachchan
Rishi Kapoor
Sanjay Dutt
John Abraham
Nassar Abdulla
Rekha
Paresh Rawal
Ajay Devgn
Rahul Bose
Sunny Deol
Akshay Kumar
Fardeen Khan

Akshay Kumar
Jeetendra
Sarika
Aamir Khan
Salman Khan
Govinda
Akshaye Khanna
Salman Khan
Mohit Ahlawat
Mahima Chaudhry
Ajay Devgn
Mohammad Amir Naji
Sanjay Dutt
Irrfan Khan
Aishwarya Rai Bachchan
Nana Patekar
Sunny Deol
Shahid Kapoor
Priyanshu Chatterjee
Gurdas Maan
Kangana Ranaut
Paresh Rawal
Jimmy Sheirgill
Sushmita Sen
Rakesh Bedi
Emraan Hashmi
Manisha Koirala
Ajay Devgn
Rekha
Juhi Chawla
Bobby Deol
Anil Kapoor
Amitabh Bachchan
Sanjay Dutt
Abhishek Bachchan
Shawar Ali
Ali Asgar
Tabu
Ajay Devgn
Abhishek Bachchan
Aryan Vaid
Jatin Grewal
Samir Aftab
Anil Kapoor
Randeep Hooda
Akshay Kumar
Saayli Buva
Bipasha Basu

Amitabh Bachchan
Akshay Kumar
Mithun Chakraborty
Sanjay Dutt
Arjun Rampal
Shilpa Shetty Kundra
Amitabh Bachchan
Siddharth Koirala
Fardeen Khan
Ayesha Jhulka
Aseel Adel
Vivek Oberoi
Akshay Kumar
Shreyas Talpade
Chiranjeevi
Kay Kay Menon
Akshay Kumar
Sunny Deol
John Abraham
Ajay Devgn
Kunal Khemu
Lucky Ali
Mukesh Khanna
Mohit Ahlawat
Vivek Oberoi
Aftab Shivdasani
Shilpa Shetty Kundra
Kamal Adib
Arshad Warsi
Tusshar Kapoor
Salman Khan
Ajay Devgn
Salman Khan
Rajpal Yadav
Anupam Kher
Salman Khan
Aamir Khan
Dia Mirza
Dev Anand
Sanjay Suri
Anil Kapoor
Urmila Matondkar
Uday Chopra
Anil Kapoor
Sunil Shetty
Konkona Sen Sharma
Shabana Azmi
Saif Ali Khan

Vinod Khanna
Shah Rukh Khan
Irrfan Khan
Rishi Kapoor
Saif Ali Khan
Kashmira Shah
Amitabh Bachchan
Arshad Warsi
Sanjay Dutt
Ashutosh Rana
Neha Dhupia
Madhavan
Tabu
Abhay Deol
Zulfi Sayed
Ajay Devgn
Nassar Abdulla
Arjun Rampal
Shahid Kapoor
Aarti Chhabria
Fardeen Khan
Amitabh Bachchan
Amitabh Bachchan
Vikram Aditya
Arjun Rampal
Emraan Hashmi
Shah Rukh Khan
Shah Rukh Khan
Ajay Devgn
Salman Khan
Abhishek Bachchan
Amitabh Bachchan
Saif Ali Khan
Akshaye Khanna
Mallika Sherawat
Ajay Devgn
Akshay Kumar
Amitabh Bachchan
Amitabh Bachchan
Shatrughan Sinha
Bobby Deol
Kareena Kapoor
Jimmy Sheirgill
Arjun Rampal
Jimmy Sheirgill
Amitabh Bachchan
Amitabh Bachchan
Shahid Kapoor

Salman Khan
Jackie Shroff
Naveen Bawa
Vikaas Kalantari
Tusshar Kapoor
Kareena Kapoor
Salman Khan
Sunil Shetty
Saif Ali Khan
Isha Koppikar
Akshay Kumar
Aniket Vishwasrao
Shawar Ali
Tisca Chopra
Neha Dhupia
Raqesh Bapat
Manoj Bajpayee
Mallika Sherawat
Dino Morea
Sohail Khan
Kiron Kher
Bobby Deol
Sunny Deol
Sohail Khan
Amitabh Bachchan
Bipasha Basu
Hrithik Roshan
Dino Morea
Irrfan Khan
Aftab Shivdasani
Tabu
Anil Kapoor
Prithviraj Kapoor
Aftab Shivdasani
Akshay Kumar
Antara Mali
Vinay Anand
John Abraham
Salman Khan
Sanjay Dutt
Akshay Kumar
Manisha Koirala
Akshay Kapoor
Sanjay Dutt
Ajay Devgn
Sanjay Dutt
Sanjay Suri
Abhishek Bachchan

Sunny Deol
Aftab Shivdasani
Shah Rukh Khan
Vatsal Sheth
Madhavan
Tusshar Kapoor
Emraan Hashmi
Vicky Ahuja
Shah Rukh Khan
Tarun Arora
Diwakar Pathak
Arya Babbar
Rekha
Shah Rukh Khan
Anupam Kher
Shah Rukh Khan
Sunny Deol
Amitabh Bachchan
Hrithik Roshan
Sanjay Dutt
Sanjay Dutt
Sanjay Dutt
Ajay Devgn
Atul Kulkarni
Naseeruddin Shah
Arun Bakshi
Priyanshu Chatterjee
Ajay Devgn
Amitabh Bachchan
Sudesh Berry
Akshay Kumar
Akshay Kumar
Amitabh Bachchan
Ajay Devgn
Anil Kapoor
Jackie Shroff
Sadashiv Amrapurkar
Sooraj Balaji
Arjun Rampal
Sameera Reddy
Om Puri
Amar Upadhyaya
Zayed Khan
Nawazuddin Siddiqui
Vivek Oberoi
Kapil Jhaveri
Nandita Das
Aftab Shivdasani

Manisha Koirala
Rahul Bose
Ajay Devgn
Tabu
Jimmy Sheirgill
Sanjay Dutt
Akshaye Khanna
Vikram Dasu
Reef Karim
Jaz Pandher
Rushali Arora
Ashmit Patel
Sunny Deol
Shahid Kapoor
Babbu Mann
Javed Jaffrey
Bipasha Basu
Sanjay Suri
Om Puri
Juhi Babbar
Sunny Deol
Feroz Khan
Amit Hingorani
Fardeen Khan
Himanshu Malik
Tusshar Kapoor
Aishwarya Rai Bachchan
Antara Mali
Victor Banerjee
Manisha Koirala
Attin Bhalla
Riteish Deshmukh
Rahul Bose
Tulip Joshi
Ajay Devgn
Urmila Matondkar
Abhishek Bachchan
John Abraham
Sushmita Sen
Vikas Kalantri
Raveena Tandon
Tanishaa Mukerji
Raveena Tandon
Vijay Raaz
Raveena Tandon
Tanuja
Ankit
Sadashiv Amrapurkar

```
Salman Khan
Riteish Deshmukh
Rakhee Gulzar
Shabana Azmi
Edwin Fernandes
Tusshar Kapoor
Sharman Joshi
Dino Morea
Ajay Devgn
Arshad Warsi
Shah Rukh Khan
Amitabh Bachchan
Shah Rukh Khan
Amitabh Bachchan
Vivek Oberoi
Ajay Devgn
Akshay Kumar
```

[ ]:
```
for i in movies.index: # for accessing index we have to use 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
```

[ ]:
```
# Arithmetic Operators

100 - marks
```

[ ]:
```
maths      16
english     0
science    11
hindi       3
sst        13
Name: Dilkhush's Marks, dtype: int32
```

[ ]:
```
# Relational Operators
```

```
vk[vk>50]
```

[ ]:
```
match_no
34      58
41      71
44      56
45      67
52      70
57      57
68      73
71      51
73      58
74      65
80      57
81      93
82      99
85      56
97      67
99      73
103     51
104     62
110     82
116     75
117     79
119     80
120    100
122     52
123    108
126    109
127     75
128    113
129     54
131     54
132     62
134     64
137     55
141     58
144     57
145     92
148     68
152     70
160     84
162     67
164    100
175     72
178     90
188     72
```

```
197     51
198     53
209     58
213     73
Name: runs, dtype: int32
```

0.11 Boolean Indexing on Series

[ ]:
```
# Find 50's and 100's scored by kohli

vk[(vk==50) | (vk==100)]
```

[ ]:
```
match_no
15      50
120    100
164    100
182     50
Name: runs, dtype: int32
```

[ ]:
```
# find number of ducks
vk[vk==0].size
```

[ ]:
```
9
```

[ ]:
```
# count no of days when subscriber gain is more than 200
subs[subs>200].size
```

[ ]:
```
59
```

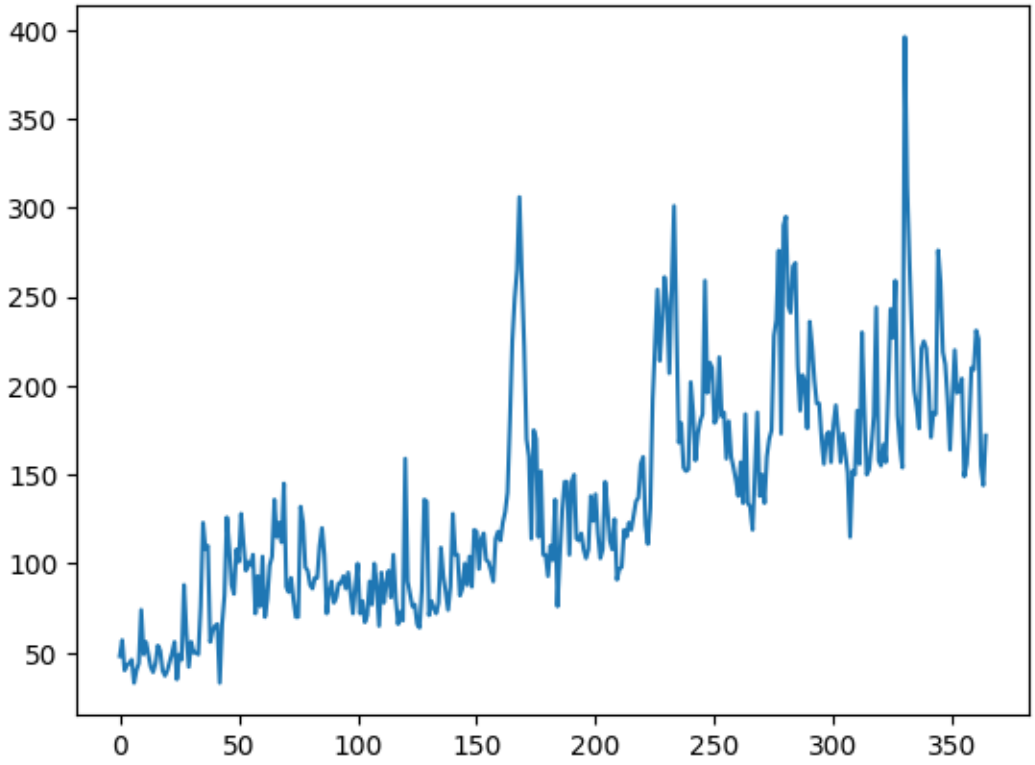
[ ]:
```
# find actors who have done more than 20 movies
num_movies = movies.value_counts()
num_movies[num_movies>20]
```

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

0.12 Plotting Graphs on Series

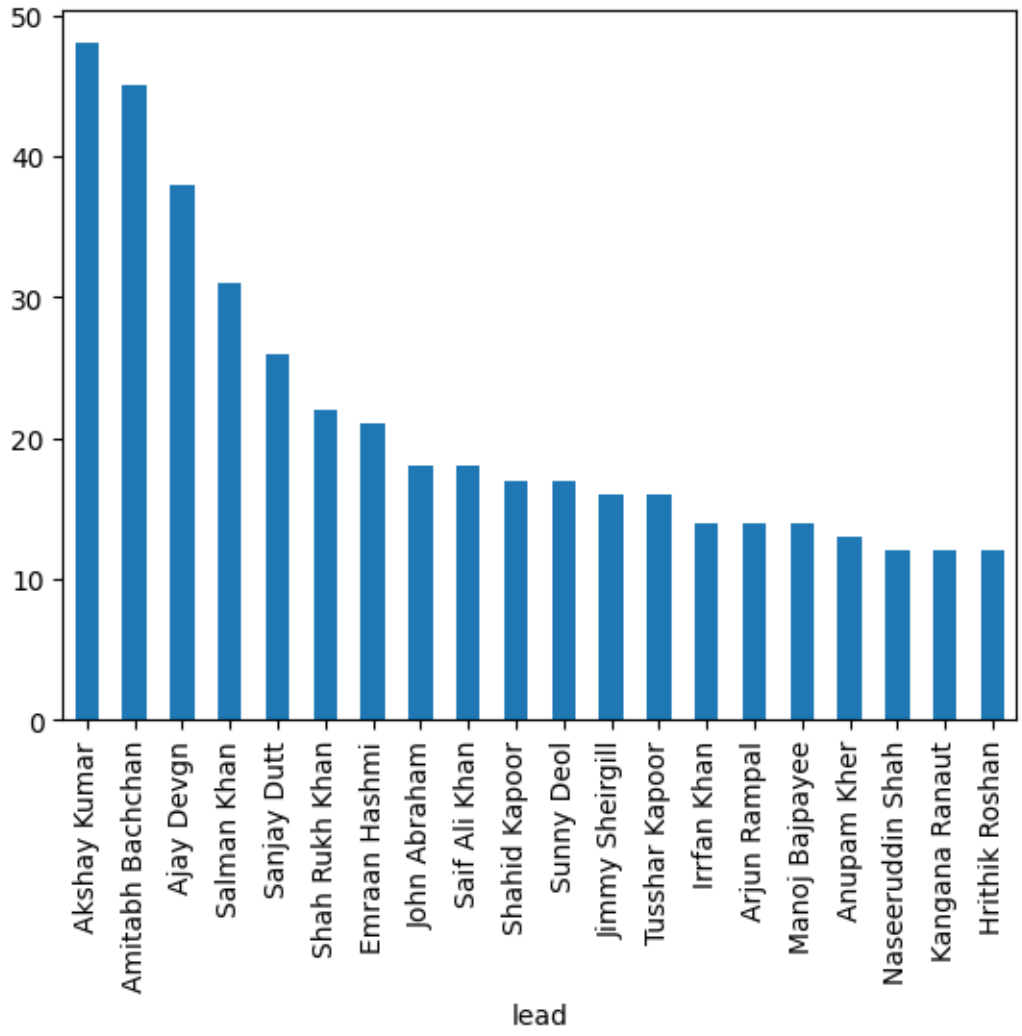
[ ]:
```
subs.plot()
```

[ ]: <Axes: >



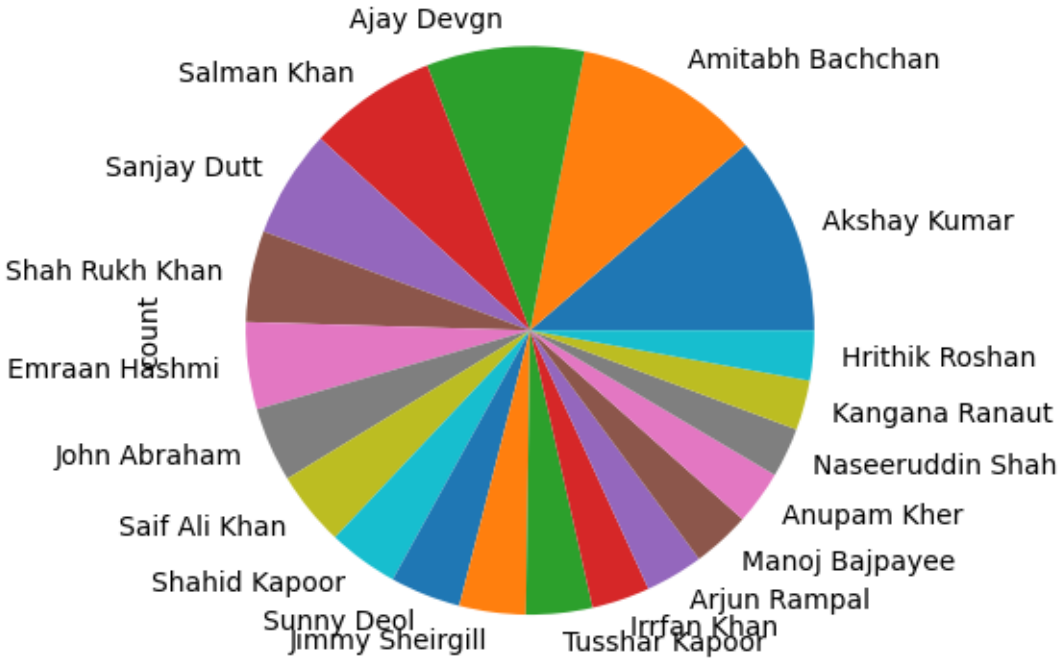
[ ]:
```
movies.value_counts().head(20).plot(kind='bar')
```

[ ]: <Axes: xlabel='lead'>



[ ]:
```
movies.value_counts().head(20).plot(kind='pie')
```

[ ]:
```
<Axes: ylabel='count'>
```



0.13 Important series methods

0.13.1 ser.astype('new_datatype')

- Change the datatype of series.
- no permanent changes.

[ ]:
```
import sys
sys.getsizeof(vk)
```

[ ]:
```
8000
```

[ ]:
```
sys.getsizeof(vk.astype('int16'))
```

[ ]:
```
7570
```

0.13.2 ser.between(start, end)

- returns boolean series including start and end.

[ ]:
```
vk[vk.between(51, 99)]
```

[ ]:
```
match_no
34     58
```

```
41     71
44     56
45     67
52     70
57     57
68     73
71     51
73     58
74     65
80     57
81     93
82     99
85     56
97     67
99     73
103    51
104    62
110    82
116    75
117    79
119    80
122    52
127    75
129    54
131    54
132    62
134    64
137    55
141    58
144    57
145    92
148    68
152    70
160    84
162    67
175    72
178    90
188    72
197    51
198    53
209    58
213    73
Name: runs, dtype: int32
```

0.13.3 ser.clip

- returns a series in which all the values are clipped in the given range.

- no change in original series

[ ]:
```
subs.clip(100, 200)
```

[ ]:
```
0      100
1      100
2      100
3      100
4      100
      ...
360    200
361    200
362    155
363    144
364    172
Name: Subscribers gained, Length: 365, dtype: int64
```

0.13.4 series.drop_duplicates(keep='first')

- return a series in which duplicated values are dropped.
- keep parameter decides which occurence to be included into series.
- by default it keeps first occurence.

[ ]:
```
temp = pd.Series([1,1,2,2,3,3,4,4])
temp
```

[ ]:
```
0    1
1    1
2    2
3    2
4    3
5    3
6    4
7    4
dtype: int64
```

[ ]:
```
temp.drop_duplicates()
```

[ ]:
```
0    1
2    2
4    3
6    4
dtype: int64
```

[ ]:
```
temp.drop_duplicates(keep='last')
```

[ ]:
```
1    1
3    2
```

```
5    3
7    4
dtype: int64
```

0.13.5 ser.duplicated()

- returns a boolean series, if values is duplicated it is true.

[ ]:
```
temp.duplicated()
```

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

[ ]:
```
temp.duplicated().sum()
```

[ ]:
```
4
```

0.13.6 ser.size

- returns total values in the series

0.13.7 ser.count()

- returns total not-NaN values

[ ]:
```
temp = pd.Series([1,2,3,np.nan,5,6,np.nan,8,np.nan,10])
temp
```

[ ]:
```
0     1.0
1     2.0
2     3.0
3     NaN
4     5.0
5     6.0
6     NaN
7     8.0
8     NaN
9    10.0
dtype: float64
```

[ ]:
```
temp.size
```

[ ]: 10

[ ]:
```
temp.count()
```

[ ]: 7

0.13.8 ser.isnull()

- returns a boolean series in which every element is checked whether is it null or not

[ ]:
```
temp.isnull().sum()
```

[ ]: 3

0.13.9 ser.dropna()

- returns a series after dropping all the Nan values.

[ ]:
```
temp.dropna()
```

[ ]:
```
0     1.0
1     2.0
2     3.0
4     5.0
5     6.0
7     8.0
9    10.0
dtype: float64
```

0.13.10 ser.fillna()

- returns a series in which NaN values are filled using some conditions

[ ]:
```
temp.fillna(0)
```

[ ]:
```
0     1.0
1     2.0
2     3.0
3     0.0
4     5.0
5     6.0
6     0.0
7     8.0
8     0.0
9    10.0
dtype: float64
```

[ ]:
```
temp.fillna(temp.mean())
```

```
0     1.0
1     2.0
2     3.0
3     5.0
4     5.0
5     6.0
6     5.0
7     8.0
8     5.0
9    10.0
dtype: float64
```

0.13.11 ser.isin(list_of_items)

- returns a boolean series which checks the elements of provided list are in the series or not

```
vk[vk.isin([49, 99])]
```

```
match_no
82    99
86    49
Name: runs, dtype: int32
```

0.13.12 ser.apply(func)

- returns series after applying the given function on all the values.

```
movies.apply(lambda x: x.split()[0].upper())
```

```
movie
Uri: The Surgical Strike                   VICKY
Battalion 609                              VICKY
The Accidental Prime Minister (film)      ANUPAM
Why Cheat India                           EMRAAN
Evening Shadows                             MONA
                                           …
Hum Tumhare Hain Sanam                      SHAH
Aankhen (2002 film)                      AMITABH
Saathiya (film)                            VIVEK
Company (film)                              AJAY
Awara Paagal Deewana                      AKSHAY
Name: lead, Length: 1500, dtype: object
```

```
subs.apply(lambda x:'good day' if x>subs.mean() else 'bad day')
```

```
0       bad day
1       bad day
2       bad day
```

```
3       bad day
4       bad day
         …
360    good day
361    good day
362    good day
363    good day
364    good day
Name: Subscribers gained, Length: 365, dtype: object
```

0.13.13 ser.copy()

- copy the series

[ ]:
```
vk
```

[ ]:
```
match_no
1       1
2      23
3      13
4      12
5       1
       ..
211     0
212    20
213    73
214    25
215     7
Name: runs, Length: 215, dtype: int32
```

[ ]:
```
new = vk.head()
new
```

[ ]:
```
match_no
1     1
2    23
3    13
4    12
5     1
Name: runs, dtype: int32
```

[ ]:
```
new[1] = 100
new
```

[ ]:
```
match_no
1    100
2     23
```

```
3     13
4     12
5      1
Name: runs, dtype: int32
```

[ ]:
```
vk
```

[ ]:
```
match_no
1      100
2       23
3       13
4       12
5        1
       …
211      0
212     20
213     73
214     25
215      7
Name: runs, Length: 215, dtype: int32
```

[ ]:
```
new = vk.head().copy()
new
```

[ ]:
```
match_no
1    100
2     23
3     13
4     12
5      1
Name: runs, dtype: int32
```

[ ]:
```
new[1] = 1
new
```

[ ]:
```
match_no
1     1
2    23
3    13
4    12
5     1
Name: runs, dtype: int32
```

[ ]:
```
vk
```

[ ]:
```
match_no
1      100
```

```
2       23
3       13
4       12
5        1
      ...
211      0
212     20
213     73
214     25
215      7
Name: runs, Length: 215, dtype: int32
```

[5]:
```
import numpy as np
import pandas as pd
```

1 DataFrame

A DataFrame represents a rectangular table of data and contains an ordered, named collection of columns, each of which can be a different value type (numeric, string, Boolean, etc.). The DataFrame has both a row and column index; it can be thought of as a dictionary of Series all sharing the same index.

1.1 Creating DataFrame

Type	Notes
2D ndarray	A matrix of data, passing optional row and column labels
Dictionary of arrays, lists, or tuples	Each sequence becomes a column in the DataFrame; all sequences must be the same length
NumPy structured/record array	Treated as the "dictionary of arrays" case
Dictionary of Series	Each value becomes a column; indexes from each Series are unioned together to form the result's row index if no explicit index is passed
Dictionary of dictionaries	Each inner dictionary becomes a column; keys are unioned to form the row index as in the "dictionary of Series" case
List of dictionaries or Series	Each item becomes a row in the DataFrame; unions of dictionary keys or Series indexes become the DataFrame's column labels
List of lists or tuples	Treated as the "2D ndarray" case
Another DataFrame	The DataFrame's indexes are used unless different ones are passed
NumPy MaskedArray	Like the "2D ndarray" case except masked values are missing in the DataFrame result

1.1.1 From Lists

[6]:
```
student_data = [
    [100, 80, 10],
    [90, 70, 7],
    [120, 100, 14],
    [80, 50, 2]
]

pd.DataFrame(student_data, columns=['iq', 'marks', 'package'])
```

[6]:

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

1.1.2 From Dictionary

[7]:
```
student_dict = {
    'name':['dilkush', 'ankit', 'neeraj', 'ritu', 'pankaj', 'pankaj'],
    'iq':[100, 90, 120, 80, 0, 0],
    'marks':[80, 70, 100, 50, 0, 0],
    'package':[10, 7, 14, 2, 0, 0]
}

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

[7]:

	name	iq	marks	package
0	dilkush	100	80	10
1	ankit	90	70	7
2	neeraj	120	100	14
3	ritu	80	50	2
4	pankaj	0	0	0
5	pankaj	0	0	0

Assigning names of index and columns

[8]:
```
students.index.name = 'ser no.'
students.columns.name = 'details'
```

[9]:
```
students
```

[9]:

details	name	iq	marks	package
ser no.				
0	dilkush	100	80	10
1	ankit	90	70	7

```
2         neeraj  120    100       14
3           ritu   80     50        2
4         pankaj    0      0        0
5         pankaj    0      0        0
```

Adding another column

[10]:
```
students['age'] = np.nan
```

[11]:
```
students
```

[11]:
```
details     name   iq  marks  package  age
ser no.
0        dilkush  100     80       10  NaN
1          ankit   90     70        7  NaN
2         neeraj  120    100       14  NaN
3           ritu   80     50        2  NaN
4         pankaj    0      0        0  NaN
5         pankaj    0      0        0  NaN
```

[12]:
```
students['age'] = 20
```

[13]:
```
students
```

[13]:
```
details     name   iq  marks  package  age
ser no.
0        dilkush  100     80       10   20
1          ankit   90     70        7   20
2         neeraj  120    100       14   20
3           ritu   80     50        2   20
4         pankaj    0      0        0   20
5         pankaj    0      0        0   20
```

[14]:
```
del students['age']
```

[15]:
```
students
```

[15]:
```
details     name   iq  marks  package
ser no.
0        dilkush  100     80       10
1          ankit   90     70        7
2         neeraj  120    100       14
3           ritu   80     50        2
4         pankaj    0      0        0
5         pankaj    0      0        0
```

1.1.3 From csv files

[16]:
```
movies = pd.read_csv('movies.csv')
movies
```

[16]:
```
                                 title_x     imdb_id  \
0                Uri: The Surgical Strike   tt8291224
1                            Battalion 609   tt9472208
2     The Accidental Prime Minister (film)   tt6986710
3                          Why Cheat India   tt8108208
4                          Evening Shadows   tt6028796
…                                      …         …
1624                 Tera Mera Saath Rahen   tt0301250
1625                  Yeh Zindagi Ka Safar   tt0298607
1626                       Sabse Bada Sukh   tt0069204
1627                                 Daaka  tt10833860
1628                               Humsafar   tt2403201

                                              poster_path  \
0     https://upload.wikimedia.org/wikipedia/en/thum…
1                                                   NaN
2     https://upload.wikimedia.org/wikipedia/en/thum…
3     https://upload.wikimedia.org/wikipedia/en/thum…
4                                                   NaN
…                                                   …
1624  https://upload.wikimedia.org/wikipedia/en/2/2b…
1625  https://upload.wikimedia.org/wikipedia/en/thum…
1626                                                NaN
1627  https://upload.wikimedia.org/wikipedia/en/thum…
1628  https://upload.wikimedia.org/wikipedia/en/thum…

                                            wiki_link  \
0     https://en.wikipedia.org/wiki/Uri:_The_Surgica…
1           https://en.wikipedia.org/wiki/Battalion_609
2     https://en.wikipedia.org/wiki/The_Accidental_P…
3         https://en.wikipedia.org/wiki/Why_Cheat_India
4         https://en.wikipedia.org/wiki/Evening_Shadows
…                                                   …
1624  https://en.wikipedia.org/wiki/Tera_Mera_Saath_…
1625  https://en.wikipedia.org/wiki/Yeh_Zindagi_Ka_S…
1626      https://en.wikipedia.org/wiki/Sabse_Bada_Sukh
1627                 https://en.wikipedia.org/wiki/Daaka
1628              https://en.wikipedia.org/wiki/Humsafar

                           title_y                original_title  is_adult  \
0          Uri: The Surgical Strike       Uri: The Surgical Strike         0
1                      Battalion 609                  Battalion 609         0
```

```
2     The Accidental Prime Minister  The Accidental Prime Minister         0
3                   Why Cheat India                Why Cheat India         0
4                   Evening Shadows                Evening Shadows         0
…                               …                              …       …
1624          Tera Mera Saath Rahen          Tera Mera Saath Rahen         0
1625           Yeh Zindagi Ka Safar           Yeh Zindagi Ka Safar         0
1626                Sabse Bada Sukh                Sabse Bada Sukh         0
1627                          Daaka                          Daaka         0
1628                        Humsafar                       Humsafar         0

      year_of_release runtime            genres  imdb_rating  imdb_votes  \
0                2019     138  Action|Drama|War          8.4       35112
1                2019     131               War          4.1          73
2                2019     112   Biography|Drama          6.1        5549
3                2019     121       Crime|Drama          6.0        1891
4                2018     102             Drama          7.3         280
…                …      …                 …          …          …
1624             2001     148             Drama          4.9         278
1625             2001     146             Drama          3.0         133
1626             2018      \N      Comedy|Drama          6.1          13
1627             2019     136            Action          7.4          38
1628             2011      35     Drama|Romance          9.0        2968

                                                  story  \
0     Divided over five chapters  the film chronicle…
1     The story revolves around a cricket match betw…
2     Based on the memoir by Indian policy analyst S…
3     The movie focuses on existing malpractices in …
4     While gay rights and marriage equality has bee…
…                                                   …
1624  Raj Dixit lives with his younger brother  Rahu…
1625  Hindi pop-star  Sarina Devan  lives a wealthy …
1626  Village born Lalloo re-locates to Bombay  and …
1627  Shinda tries robbing a bank so he can be wealt…
1628  Sara and Ashar are childhood friends who share…

                                               summary tagline  \
0     Indian army special forces execute a covert op…     NaN
1     The story of Battalion 609 revolves around a c…     NaN
2     Explores Manmohan Singh's tenure as the Prime …     NaN
3     The movie focuses on existing malpractices in …     NaN
4     Under the 'Evening Shadows'  truth often plays…     NaN
…                                                   …     …
1624  A man is torn between his handicapped brother …     NaN
1625  A singer finds out she was adopted when the ed…     NaN
1626  Village born Lalloo re-locates to Bombay  and …     NaN
1627  Shinda tries robbing a bank so he can be wealt…     NaN
```

```
1628  Ashar and Khirad are forced to get married due…     NaN

                                                  actors  \
0     Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga…
1     Vicky Ahuja|Shoaib Ibrahim|Shrikant Kamat|Elen…
2     Anupam Kher|Akshaye Khanna|Aahana Kumra|Atul S…
3     Emraan Hashmi|Shreya Dhanwanthary|Snighdadeep …
4     Mona Ambegaonkar|Ananth Narayan Mahadevan|Deva…
…                                                   …
1624  Ajay Devgn|Sonali Bendre|Namrata Shirodkar|Pre…
1625  Ameesha Patel|Jimmy Sheirgill|Nafisa Ali|Gulsh…
1626  Vijay Arora|Asrani|Rajni Bala|Kumud Damle|Utpa…
1627                          Gippy Grewal|Zareen Khan|
1628                                       Fawad Khan|

            wins_nominations              release_date
0                     4 wins     11 January 2019 (USA)
1                        NaN   11 January 2019 (India)
2                        NaN     11 January 2019 (USA)
3                        NaN     18 January 2019 (USA)
4     17 wins & 1 nomination   11 January 2019 (India)
…                       …                         …
1624                     NaN   7 November 2001 (India)
1625                     NaN  16 November 2001 (India)
1626                     NaN                       NaN
1627                     NaN     1 November 2019 (USA)
1628                     NaN     TV Series (2011-2012)

[1629 rows x 18 columns]
```

[17]:
```
ipl = pd.read_csv('ipl-matches.csv')
ipl
```

[17]:
```
          ID        City        Date   Season  MatchNumber  \
0    1312200   Ahmedabad  2022-05-29     2022        Final
1    1312199   Ahmedabad  2022-05-27     2022  Qualifier 2
2    1312198     Kolkata  2022-05-25     2022   Eliminator
3    1312197     Kolkata  2022-05-24     2022  Qualifier 1
4    1304116      Mumbai  2022-05-22     2022           70
..       …         …         …      …           …
945   335986     Kolkata  2008-04-20  2007/08            4
946   335985      Mumbai  2008-04-20  2007/08            5
947   335984       Delhi  2008-04-19  2007/08            3
948   335983  Chandigarh  2008-04-19  2007/08            2
949   335982   Bangalore  2008-04-18  2007/08            1

                          Team1                       Team2  \
```

```
0              Rajasthan Royals                Gujarat Titans
1    Royal Challengers Bangalore             Rajasthan Royals
2    Royal Challengers Bangalore          Lucknow Super Giants
3              Rajasthan Royals                Gujarat Titans
4            Sunrisers Hyderabad                  Punjab Kings
..                           …                             …
945        Kolkata Knight Riders               Deccan Chargers
946               Mumbai Indians  Royal Challengers Bangalore
947             Delhi Daredevils             Rajasthan Royals
948              Kings XI Punjab           Chennai Super Kings
949  Royal Challengers Bangalore         Kolkata Knight Riders

                                       Venue                 TossWinner  \
0              Narendra Modi Stadium, Ahmedabad             Rajasthan Royals
1              Narendra Modi Stadium, Ahmedabad             Rajasthan Royals
2                        Eden Gardens, Kolkata          Lucknow Super Giants
3                        Eden Gardens, Kolkata                Gujarat Titans
4                     Wankhede Stadium, Mumbai           Sunrisers Hyderabad
..                                          …                           …
945                              Eden Gardens               Deccan Chargers
946                          Wankhede Stadium                Mumbai Indians
947                          Feroz Shah Kotla             Rajasthan Royals
948  Punjab Cricket Association Stadium, Mohali           Chennai Super Kings
949                      M Chinnaswamy Stadium  Royal Challengers Bangalore

    TossDecision SuperOver                 WinningTeam    WonBy  Margin  \
0            bat         N               Gujarat Titans  Wickets     7.0
1          field         N             Rajasthan Royals  Wickets     7.0
2          field         N  Royal Challengers Bangalore     Runs    14.0
3          field         N               Gujarat Titans  Wickets     7.0
4            bat         N                 Punjab Kings  Wickets     5.0
..           …       …                            …      …     …
945          bat         N        Kolkata Knight Riders  Wickets     5.0
946          bat         N  Royal Challengers Bangalore  Wickets     5.0
947          bat         N             Delhi Daredevils  Wickets     9.0
948          bat         N          Chennai Super Kings     Runs    33.0
949        field         N        Kolkata Knight Riders     Runs   140.0

    method Player_of_Match                                    Team1Players  \
0      NaN       HH Pandya  ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …
1      NaN      JC Buttler  ['V Kohli', 'F du Plessis', 'RM Patidar', 'GJ …
2      NaN      RM Patidar  ['V Kohli', 'F du Plessis', 'RM Patidar', 'GJ …
3      NaN       DA Miller  ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …
4      NaN   Harpreet Brar  ['PK Garg', 'Abhishek Sharma', 'RA Tripathi', …
..     …              …                                                 …
945    NaN       DJ Hussey  ['WP Saha', 'BB McCullum', 'RT Ponting', 'SC G…
946    NaN      MV Boucher  ['L Ronchi', 'ST Jayasuriya', 'DJ Thornely', '…
```

```
947    NaN     MF Maharoof  ['G Gambhir', 'V Sehwag', 'S Dhawan', 'MK Tiwa…
948    NaN      MEK Hussey  ['K Goel', 'JR Hopes', 'KC Sangakkara', 'Yuvra…
949    NaN     BB McCullum  ['R Dravid', 'W Jaffer', 'V Kohli', 'JH Kallis…

                                          Team2Players        Umpire1  \
0    ['WP Saha', 'Shubman Gill', 'MS Wade', 'HH Pan…    CB Gaffaney
1    ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …    CB Gaffaney
2    ['Q de Kock', 'KL Rahul', 'M Vohra', 'DJ Hooda…  J Madanagopal
3    ['WP Saha', 'Shubman Gill', 'MS Wade', 'HH Pan…   BNJ Oxenford
4    ['JM Bairstow', 'S Dhawan', 'M Shahrukh Khan',…   AK Chaudhary
..                                                 …            …
945  ['AC Gilchrist', 'Y Venugopal Rao', 'VVS Laxma…      BF Bowden
946  ['S Chanderpaul', 'R Dravid', 'LRPL Taylor', '…       SJ Davis
947  ['T Kohli', 'YK Pathan', 'SR Watson', 'M Kaif'…      Aleem Dar
948  ['PA Patel', 'ML Hayden', 'MEK Hussey', 'MS Dh…      MR Benson
949  ['SC Ganguly', 'BB McCullum', 'RT Ponting', 'D…      Asad Rauf

            Umpire2
0       Nitin Menon
1       Nitin Menon
2          MA Gough
3         VK Sharma
4     NA Patwardhan
..              …
945     K Hariharan
946       DJ Harper
947  GA Pratapkumar
948      SL Shastri
949     RE Koertzen

[950 rows x 20 columns]
```

1.2 DataFrame Attributes and Methods

1.2.1 DF.head(n=5)

- returns DF containing first n rows.
- if no n specified then by default n is 5

[18]:
```
movies.head(2)
```

[18]:
```
                    title_x    imdb_id  \
0  Uri: The Surgical Strike  tt8291224
1              Battalion 609  tt9472208

                                           poster_path  \
0  https://upload.wikimedia.org/wikipedia/en/thum…
1                                                NaN
```

```
                                          wiki_link  \
0  https://en.wikipedia.org/wiki/Uri:_The_Surgica…
1        https://en.wikipedia.org/wiki/Battalion_609

                    title_y            original_title  is_adult  \
0  Uri: The Surgical Strike  Uri: The Surgical Strike         0
1             Battalion 609             Battalion 609         0

   year_of_release runtime            genres  imdb_rating  imdb_votes  \
0             2019     138  Action|Drama|War          8.4       35112
1             2019     131               War          4.1          73

                                           story  \
0  Divided over five chapters  the film chronicle…
1  The story revolves around a cricket match betw…

                                         summary tagline  \
0  Indian army special forces execute a covert op…     NaN
1  The story of Battalion 609 revolves around a c…     NaN

                                         actors wins_nominations  \
0  Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga…           4 wins
1  Vicky Ahuja|Shoaib Ibrahim|Shrikant Kamat|Elen…              NaN

              release_date
0    11 January 2019 (USA)
1  11 January 2019 (India)
```

1.2.2 DF.tail(n=5)

- same as DF.head(n), just returns last rows

[19]:
```
ipl.tail()
```

[19]:
```
          ID        City        Date   Season MatchNumber  \
945  335986     Kolkata  2008-04-20  2007/08           4
946  335985      Mumbai  2008-04-20  2007/08           5
947  335984       Delhi  2008-04-19  2007/08           3
948  335983  Chandigarh  2008-04-19  2007/08           2
949  335982   Bangalore  2008-04-18  2007/08           1

                          Team1                         Team2  \
945        Kolkata Knight Riders              Deccan Chargers
946                Mumbai Indians  Royal Challengers Bangalore
947             Delhi Daredevils             Rajasthan Royals
948              Kings XI Punjab          Chennai Super Kings
```

```
949  Royal Challengers Bangalore        Kolkata Knight Riders

                                     Venue                  TossWinner  \
945                          Eden Gardens             Deccan Chargers
946                       Wankhede Stadium              Mumbai Indians
947                       Feroz Shah Kotla            Rajasthan Royals
948  Punjab Cricket Association Stadium, Mohali         Chennai Super Kings
949                  M Chinnaswamy Stadium  Royal Challengers Bangalore

    TossDecision SuperOver                 WinningTeam    WonBy  Margin  \
945          bat         N        Kolkata Knight Riders  Wickets     5.0
946          bat         N  Royal Challengers Bangalore  Wickets     5.0
947          bat         N             Delhi Daredevils  Wickets     9.0
948          bat         N          Chennai Super Kings     Runs    33.0
949        field         N        Kolkata Knight Riders     Runs   140.0

    method Player_of_Match                                    Team1Players  \
945    NaN       DJ Hussey  ['WP Saha', 'BB McCullum', 'RT Ponting', 'SC G…
946    NaN      MV Boucher  ['L Ronchi', 'ST Jayasuriya', 'DJ Thornely', '…
947    NaN     MF Maharoof  ['G Gambhir', 'V Sehwag', 'S Dhawan', 'MK Tiwa…
948    NaN      MEK Hussey  ['K Goel', 'JR Hopes', 'KC Sangakkara', 'Yuvra…
949    NaN     BB McCullum  ['R Dravid', 'W Jaffer', 'V Kohli', 'JH Kallis…

                                     Team2Players    Umpire1  \
945  ['AC Gilchrist', 'Y Venugopal Rao', 'VVS Laxma…  BF Bowden
946  ['S Chanderpaul', 'R Dravid', 'LRPL Taylor', '…   SJ Davis
947  ['T Kohli', 'YK Pathan', 'SR Watson', 'M Kaif'…  Aleem Dar
948  ['PA Patel', 'ML Hayden', 'MEK Hussey', 'MS Dh…  MR Benson
949  ['SC Ganguly', 'BB McCullum', 'RT Ponting', 'D…  Asad Rauf

           Umpire2
945    K Hariharan
946      DJ Harper
947  GA Pratapkumar
948     SL Shastri
949    RE Koertzen
```

[20]:
```
data = {"state": ["Ohio", "Ohio", "Ohio", "Nevada", "Nevada", "Nevada"],
 "year": [2000, 2001, 2002, 2001, 2002, 2003],
 "pop": [1.5, 1.7, 3.6, 2.4, 2.9, 3.2]}

frame = pd.DataFrame(data)
frame
```

[20]:
```
    state  year  pop
0    Ohio  2000  1.5
1    Ohio  2001  1.7
```

```
2    Ohio  2002  3.6
3  Nevada  2001  2.4
4  Nevada  2002  2.9
5  Nevada  2003  3.2
```

Note - If you specify a sequence of columns, the DataFrame's columns will be arranged in that order:

[21]:
```
pd.DataFrame(data, columns=['year', 'pop', 'state'])
```

[21]:
```
   year  pop   state
0  2000  1.5    Ohio
1  2001  1.7    Ohio
2  2002  3.6    Ohio
3  2001  2.4  Nevada
4  2002  2.9  Nevada
5  2003  3.2  Nevada
```

1.2.3 DF.shape

- returns tuple containing shape of DataFrame

[22]:
```
movies.shape
```

[22]:
```
(1629, 18)
```

[23]:
```
ipl.shape
```

[23]:
```
(950, 20)
```

1.2.4 DF.dtypes

- returns series containing datatype of each column

[24]:
```
movies.dtypes
```

[24]:
```
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
```

[25]:
```
ipl.dtypes
```

[25]:
```
ID                    int64
City                 object
Date                 object
Season               object
MatchNumber          object
Team1                object
Team2                object
Venue                object
TossWinner           object
TossDecision         object
SuperOver            object
WinningTeam          object
WonBy                object
Margin              float64
method               object
Player_of_Match      object
Team1Players         object
Team2Players         object
Umpire1              object
Umpire2              object
dtype: object
```

1.2.5 DF.index

- returns RangeIndex object if index is not explicitely decided

[26]:
```
movies.index
```

[26]:
```
RangeIndex(start=0, stop=1629, step=1)
```

[27]:
```
ipl.index
```

[27]:
```
RangeIndex(start=0, stop=950, step=1)
```

1.2.6 DF.columns

- returns Index object containing all column names

[28]:
```
movies.columns
```

[28]:
```
Index(['title_x', 'imdb_id', 'poster_path', 'wiki_link', 'title_y',
       'original_title', 'is_adult', 'year_of_release', 'runtime', 'genres',
       'imdb_rating', 'imdb_votes', 'story', 'summary', 'tagline', 'actors',
       'wins_nominations', 'release_date'],
      dtype='object')
```

[29]:
```
ipl.columns
```

[29]:
```
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')
```

1.2.7 DF.values

- returns numpy ndarray containig all the values

[30]:
```
students.values
```

[30]:
```
array([['dilkush', 100, 80, 10],
       ['ankit', 90, 70, 7],
       ['neeraj', 120, 100, 14],
       ['ritu', 80, 50, 2],
       ['pankaj', 0, 0, 0],
       ['pankaj', 0, 0, 0]], dtype=object)
```

[31]:
```
ipl.values
```

[31]:
```
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 Hetmyer',
'R Parag', 'R Ashwin', 'TA Boult', 'YS Chahal', 'M Prasidh Krishna', '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', 'Ravi
Bishnoi']",
        'J Madanagopal', 'MA Gough'],
       …,
```

```
       [335984, 'Delhi', '2008-04-19', …,
        "['T Kohli', 'YK Pathan', 'SR Watson', 'M Kaif', 'DS Lehmann', 'RA
Jadeja', 'M Rawat', 'D Salunkhe', 'SK Warne', 'SK Trivedi', 'MM Patel']",
        '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', 'Mohammad
Hafeez', 'LR Shukla', 'WP Saha', 'AB Agarkar', 'AB Dinda', 'M Kartik', 'I
Sharma']",
        'Asad Rauf', 'RE Koertzen']], dtype=object)
```

1.2.8 DF.sample(n=1)

- returns randomly selected n rows from the DF

[32]:
```
ipl.sample(2)
```

[32]:
```
          ID        City        Date Season MatchNumber  \
660   548350   Bangalore  2012-05-02   2012          44
338  1082626  Chandigarh  2017-04-30   2017          36

                          Team1             Team2  \
660  Royal Challengers Bangalore   Kings XI Punjab
338              Kings XI Punjab  Delhi Daredevils

                                              Venue       TossWinner  \
660                            M Chinnaswamy Stadium  Kings XI Punjab
338  Punjab Cricket Association IS Bindra Stadium, …  Kings XI Punjab

    TossDecision SuperOver      WinningTeam    WonBy  Margin method  \
660        field         N  Kings XI Punjab  Wickets     4.0    NaN
338        field         N  Kings XI Punjab  Wickets    10.0    NaN

    Player_of_Match                                      Team1Players  \
660   Azhar Mahmood  ['MA Agarwal', 'CH Gayle', 'V Kohli', 'AB de V…
338  Sandeep Sharma  ['MJ Guptill', 'HM Amla', 'M Vohra', 'SE Marsh…

                                    Team2Players    Umpire1  \
660  ['Mandeep Singh', 'SE Marsh', 'N Saini', 'DJ H…  BF Bowden
338  ['SV Samson', 'SW Billings', 'KK Nair', 'SS Iy…   YC Barde

          Umpire2
660  C Shamshuddin
```

```
338      CK Nandan
```

1.2.9 DF.info()

- returns DF containing various informations

[33]:
```
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
```

[34]:
```
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
```

1.2.10 DF.describe()

- only applies to numerical columns,
- returns DF containing various statistical info

[35]:
```
movies.describe()
```

[35]:

	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

1.2.11 DF.isnull()

- returns boolean DF, used with sum() to count total null values in each column of DF

[36]:
```
movies.isnull().sum()
```

[36]:
```
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
```

1.2.12 DF.duplicated()

- return boolean DF

[37]:
```
students.duplicated()
```

[37]:
```
ser no.
0    False
1    False
2    False
3    False
4    False
5     True
dtype: bool
```

[38]:
```
students.duplicated().sum()
```

[38]:
```
1
```

1.2.13 DF.rename(columns={'existing_col_name':'new_col_name'}, inplace=False)

- used to rename the columns.
- inplace parameter used to make permanent changes into DF

[39]:
```
students.rename(columns={'marks':'percent', 'package':'lpa'}, inplace=True)
```

[40]:
```
students
```

[40]:

details	name	iq	percent	lpa
ser no.				
0	dilkush	100	80	10
1	ankit	90	70	7
2	neeraj	120	100	14
3	ritu	80	50	2
4	pankaj	0	0	0
5	pankaj	0	0	0

1.3 Mathematical Operations on DF

Descriptive and summary statistics | Method | Description | |———————|———————————————————————————————————————-| | count | Number of non-NA values | | describe | Compute set of summary statistics | | min, max | Compute minimum and maximum values | | argmin, argmax | Compute index locations (integers) at which minimum or maximum value is obtained, respectively; not available on DataFrame objects | | idxmin, idxmax | Compute index labels at which minimum or maximum value is obtained, respectively | | quantile | Compute sample quantile ranging from 0 to 1 (default: 0.5) | | sum | Sum of values | | mean | Mean of values | | median | Arithmetic median (50% quantile) of values | | mad | Mean absolute deviation from mean value | | prod | Product of all values | | var | Sample variance of values | | std | Sample standard deviation of values | | skew | Sample skewness (third moment) of values | | kurt | Sample kurtosis (fourth moment) of values | | cumsum | Cumulative sum of values | | cummin, cummax | Cumulative minimum or maximum of values, respectively | | cumprod | Cumulative product of values | | diff | Compute first arithmetic difference (useful for time series) | | pct_change | Compute percent changes |

[41]:
```
students = pd.DataFrame([
    [100, 80, 10],
    [90, 70, 7],
    [120, 100, 14],
    [80, 50, 2],
    [0, 0, 0],
    [0, 0, 0]
], columns=['iq', 'percent', 'lpa'])

students
```

[41]:

	iq	percent	lpa
0	100	80	10
1	90	70	7
2	120	100	14
3	80	50	2
4	0	0	0
5	0	0	0

1.3.1 DF.reindex([‘indexvalue1’, ’indexvalue2])

- used to change the order of indexes,

[42]:
```
students.reindex([1,2,5,4,0,3])
```

[42]:

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

```
3   80        50     2
```

1.3.2 DF.sum(axis=0) / DF.max(axis) / DF.min(axis) / DF.mean(axis) / DF.median(axis) / DF.var(axis) / DF.std(axis)

- various operations on values along specified axis.
- by default axis is 0
- axis = 1 -> row-wise

[43]:
```
students.sum(axis=1)
```

[43]:
```
0    190
1    167
2    234
3    132
4      0
5      0
dtype: int64
```

[44]:
```
students.max(axis=1)
```

[44]:
```
0    100
1     90
2    120
3     80
4      0
5      0
dtype: int64
```

[45]:
```
students.min()
```

[45]:
```
iq         0
percent    0
lpa        0
dtype: int64
```

[46]:
```
students.mean(axis=1)
```

[46]:
```
0    63.333333
1    55.666667
2    78.000000
3    44.000000
4     0.000000
5     0.000000
dtype: float64
```

[47]:
```
students.var()
```

[47]:
```
iq         2710.0
percent    1760.0
lpa          33.5
dtype: float64
```

[48]:
```
students.std()
```

[48]:
```
iq         52.057660
percent    41.952354
lpa         5.787918
dtype: float64
```

1.4 Selecting

Type	Notes
df[column]	Select single column or sequence of columns from the DataFrame
df.loc[rows]	Select single row or subset of rows from the DataFrame by label
df.loc[:, cols]	Select single column or subset of columns by label
df.loc[rows, cols]	Select both row(s) and column(s) by label
df.iloc[rows]	Select single row or subset of rows from the DataFrame by integer position
df.iloc[:, cols]	Select single column or subset of columns by integer position
df.iloc[rows, cols]	Select both row(s) and column(s) by integer position
df.at[row, col]	Select a single scalar value by row and column label
df.iat[row, col]	Select a single scalar value by row and column position (integers)
reindex method	Select either rows or columns by labels

1.5 Selecting Columns

1.5.1 Selecting single column from DF

[49]:
```
movies['title_x']
```

[49]:
```
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
```

[50]:
```
ipl['Venue']
```

[50]:
```
0                Narendra Modi Stadium, Ahmedabad
1                Narendra Modi Stadium, Ahmedabad
2                           Eden Gardens, Kolkata
3                           Eden Gardens, Kolkata
4                         Wankhede Stadium, Mumbai
                          …
945                                  Eden Gardens
946                               Wankhede Stadium
947                               Feroz Shah Kotla
948    Punjab Cricket Association Stadium, Mohali
949                           M Chinnaswamy Stadium
Name: Venue, Length: 950, dtype: object
```

1.5.2 Selecting Multiple Columns from DF

[51]:
```
movies[['year_of_release', 'actors', 'title_x']]
```

[51]:
```
      year_of_release                                             actors  \
0                2019  Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga…
1                2019  Vicky Ahuja|Shoaib Ibrahim|Shrikant Kamat|Elen…
2                2019  Anupam Kher|Akshaye Khanna|Aahana Kumra|Atul S…
3                2019  Emraan Hashmi|Shreya Dhanwanthary|Snighdadeep …
4                2018  Mona Ambegaonkar|Ananth Narayan Mahadevan|Deva…
…               …                                                   …
1624             2001  Ajay Devgn|Sonali Bendre|Namrata Shirodkar|Pre…
1625             2001  Ameesha Patel|Jimmy Sheirgill|Nafisa Ali|Gulsh…
1626             2018  Vijay Arora|Asrani|Rajni Bala|Kumud Damle|Utpa…
1627             2019                          Gippy Grewal|Zareen Khan|
1628             2011                                      Fawad Khan|

                                  title_x
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

[1629 rows x 3 columns]
```

[52]:
```
ipl[['Team1', 'Team2', 'WinningTeam']]
```

[52]:
```
                          Team1                       Team2  \
0               Rajasthan Royals              Gujarat Titans
1    Royal Challengers Bangalore            Rajasthan Royals
2    Royal Challengers Bangalore       Lucknow Super Giants
3               Rajasthan Royals              Gujarat Titans
4             Sunrisers Hyderabad                Punjab Kings
..                            …                           …
945        Kolkata Knight Riders              Deccan Chargers
946               Mumbai Indians  Royal Challengers Bangalore
947             Delhi Daredevils            Rajasthan Royals
948              Kings XI Punjab         Chennai Super Kings
949  Royal Challengers Bangalore       Kolkata Knight Riders

                    WinningTeam
0                 Gujarat Titans
1               Rajasthan Royals
2    Royal Challengers Bangalore
3                 Gujarat Titans
4                   Punjab Kings
..                             …
945        Kolkata Knight Riders
946  Royal Challengers Bangalore
947             Delhi Daredevils
948           Chennai Super Kings
949        Kolkata Knight Riders

[950 rows x 3 columns]
```

1.6 Selecting Rows

- iloc - searches using index positions
- loc - searches using index labels

`Note` - iloc doesn't include last index while loc includes

[53]:
```
student_dict = {
    'name':['dilkush', 'ankit', 'neeraj', 'ritu', 'pankaj', 'pankaj'],
    'iq':[100, 90, 120, 80, 0, 0],
    'marks':[80, 70, 100, 50, 0, 0],
    'package':[10, 7, 14, 2, 0, 0]
```

```
}

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

[53]:

	name	iq	marks	package
0	dilkush	100	80	10
1	ankit	90	70	7
2	neeraj	120	100	14
3	ritu	80	50	2
4	pankaj	0	0	0
5	pankaj	0	0	0

[54]:
```
students.set_index('name', inplace=True)
students
```

[54]:

name	iq	marks	package
dilkush	100	80	10
ankit	90	70	7
neeraj	120	100	14
ritu	80	50	2
pankaj	0	0	0
pankaj	0	0	0

1.6.1 Single Row using iloc

[55]:
```
movies.iloc[0]
```

[55]:
```
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
```

1.6.2 Multiple Rows using iloc

[56]:
```
movies.iloc[5:16:2]
```

[56]:
```
              title_x    imdb_id  \
5         Soni (film)  tt6078866
7          Bombairiya  tt4971258
9    Thackeray (film)  tt7777196
11          Gully Boy  tt2395469
13      Total Dhamaal  tt7639372
15  Badla (2019 film)  tt8130968

                                        poster_path  \
5   https://upload.wikimedia.org/wikipedia/en/thum…
7   https://upload.wikimedia.org/wikipedia/en/thum…
9   https://upload.wikimedia.org/wikipedia/en/thum…
11  https://upload.wikimedia.org/wikipedia/en/thum…
13  https://upload.wikimedia.org/wikipedia/en/thum…
15  https://upload.wikimedia.org/wikipedia/en/0/0c…

                                        wiki_link        title_y  \
5         https://en.wikipedia.org/wiki/Soni_(film)           Soni
7          https://en.wikipedia.org/wiki/Bombairiya     Bombairiya
9    https://en.wikipedia.org/wiki/Thackeray_(film)      Thackeray
11         https://en.wikipedia.org/wiki/Gully_Boy      Gully Boy
13     https://en.wikipedia.org/wiki/Total_Dhamaal  Total Dhamaal
15  https://en.wikipedia.org/wiki/Badla_(2019_film)          Badla

   original_title  is_adult  year_of_release runtime                   genres  \
5            Soni         0             2018      97                    Drama
7       Bombairiya         0             2019     104       Comedy|Crime|Drama
9        Thackeray         0             2019     120          Biography|Drama
11       Gully Boy         0             2019     153              Drama|Music
13  Total Dhamaal         0             2019     130  Action|Adventure|Comedy
15          Badla         0             2019     118      Crime|Drama|Mystery

    imdb_rating  imdb_votes  \
5           7.2        1595
7           4.3         295
9           5.1        2301
11          8.2       22440
13          4.3        4817
15          7.9       15499

                                                story  \
```

```
5   Soni  a young policewoman in Delhi  and her su…
7   It follows the story of Meghna  who gets embro…
9   Balasaheb Thackrey works as a cartoonist for a…
11  Gully Boy is a film about a 22-year-old boy "M…
13  Total Dhamaal is the third instalment in the D…
15  Naina Sethi a successful entrepreneur finds he…

                                                summary  \
5   While fighting crimes against women in Delhi  …
7   It follows the story of Meghna  who gets embro…
9   Biographical account of Shiv Sena Supremo  Bal…
11  A coming-of-age story based on the lives of st…
13  A group of people learn about a hidden treasur…
15  A dynamic young entrepreneur finds herself loc…

                                  tagline  \
5                                     NaN
7   They didn't mean to change the world.
9                                     NaN
11                      Apna Time Aayega!
13             The Wildest Adventure Ever
15                                    NaN

                                               actors        wins_nominations  \
5   Geetika Vidya Ohlyan|Saloni Batra|Vikas Shukla…  3 wins & 5 nominations
7   Radhika Apte|Akshay Oberoi|Siddhanth Kapoor|Ra…                     NaN
9   Nawazuddin Siddiqui|Amrita Rao|Abdul Quadir Am…                     NaN
11  Ranveer Singh|Alia Bhatt|Siddhant Chaturvedi|V…  6 wins & 3 nominations
13  Ajay Devgn|Madhuri Dixit|Anil Kapoor|Riteish D…                     NaN
15  Amitabh Bachchan|Taapsee Pannu|Amrita Singh|An…                   1 win

               release_date
5      18 January 2019 (USA)
7    18 January 2019 (India)
9    25 January 2019 (India)
11    14 February 2019 (USA)
13  22 February 2019 (India)
15     8 March 2019 (India)
```

1.6.3 Fancy Indexing using iloc

[57]:
```
movies.iloc[[0, 4, 5]]
```

[57]:
```
                    title_x    imdb_id  \
0  Uri: The Surgical Strike  tt8291224
4           Evening Shadows  tt6028796
5               Soni (film)  tt6078866
```

```
                                          poster_path  \
0  https://upload.wikimedia.org/wikipedia/en/thum…
4                                                NaN
5  https://upload.wikimedia.org/wikipedia/en/thum…

                                          wiki_link  \
0  https://en.wikipedia.org/wiki/Uri:_The_Surgica…
4      https://en.wikipedia.org/wiki/Evening_Shadows
5          https://en.wikipedia.org/wiki/Soni_(film)

                    title_y            original_title  is_adult  \
0  Uri: The Surgical Strike  Uri: The Surgical Strike         0
4           Evening Shadows           Evening Shadows         0
5                      Soni                      Soni         0

   year_of_release runtime            genres  imdb_rating  imdb_votes  \
0             2019     138  Action|Drama|War          8.4       35112
4             2018     102             Drama          7.3         280
5             2018      97             Drama          7.2        1595

                                           story  \
0  Divided over five chapters  the film chronicle…
4  While gay rights and marriage equality has bee…
5  Soni  a young policewoman in Delhi  and her su…

                                         summary tagline  \
0  Indian army special forces execute a covert op…     NaN
4  Under the 'Evening Shadows'  truth often plays…     NaN
5  While fighting crimes against women in Delhi  …     NaN

                                          actors        wins_nominations  \
0  Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga…                 4 wins
4  Mona Ambegaonkar|Ananth Narayan Mahadevan|Deva…  17 wins & 1 nomination
5  Geetika Vidya Ohlyan|Saloni Batra|Vikas Shukla…  3 wins & 5 nominations

             release_date
0    11 January 2019 (USA)
4  11 January 2019 (India)
5    18 January 2019 (USA)
```

1.6.4 Single Row using loc

[58]:
```
students.loc['ritu']
```

[58]:
```
iq         80
marks      50
package     2
Name: ritu, dtype: int64
```

1.6.5 Multiple Rows using loc

[59]:
```
students
```

[59]:
```
          iq  marks  package
name
dilkush  100     80       10
ankit     90     70        7
neeraj   120    100       14
ritu      80     50        2
pankaj     0      0        0
pankaj     0      0        0
```

[60]:
```
students.loc['dilkush':'pankaj']
```

[60]:
```
          iq  marks  package
name
dilkush  100     80       10
ankit     90     70        7
neeraj   120    100       14
ritu      80     50        2
pankaj     0      0        0
pankaj     0      0        0
```

1.6.6 Fancy Indexing using loc

[61]:
```
students.loc[['dilkush', 'neeraj', 'ritu']]
```

[61]:
```
          iq  marks  package
name
dilkush  100     80       10
neeraj   120    100       14
ritu      80     50        2
```

1.7 Selecting both Rows and Columns

[62]:
```
movies.iloc[:3, :3]
```

[62]:
```
                              title_x    imdb_id  \
0             Uri: The Surgical Strike  tt8291224
1                         Battalion 609  tt9472208
```

```
2  The Accidental Prime Minister (film)  tt6986710

                                          poster_path
0  https://upload.wikimedia.org/wikipedia/en/thum…
1                                                NaN
2  https://upload.wikimedia.org/wikipedia/en/thum…
```

[63]:
```
movies.loc[:3, 'title_x':'poster_path']
```

[63]:
```
                              title_x    imdb_id  \
0             Uri: The Surgical Strike  tt8291224
1                         Battalion 609  tt9472208
2  The Accidental Prime Minister (film)  tt6986710
3                      Why Cheat India  tt8108208

                                          poster_path
0  https://upload.wikimedia.org/wikipedia/en/thum…
1                                                NaN
2  https://upload.wikimedia.org/wikipedia/en/thum…
3  https://upload.wikimedia.org/wikipedia/en/thum…
```

1.8 Filtering a DF

[64]:
```
ipl.head(2)
```

[64]:
```
        ID       City        Date Season  MatchNumber  \
0  1312200  Ahmedabad  2022-05-29   2022        Final
1  1312199  Ahmedabad  2022-05-27   2022  Qualifier 2

                         Team1             Team2  \
0             Rajasthan Royals    Gujarat Titans
1  Royal Challengers Bangalore  Rajasthan Royals

                             Venue        TossWinner TossDecision SuperOver  \
0  Narendra Modi Stadium, Ahmedabad  Rajasthan Royals          bat         N
1  Narendra Modi Stadium, Ahmedabad  Rajasthan Royals        field         N

        WinningTeam    WonBy  Margin method Player_of_Match  \
0    Gujarat Titans  Wickets     7.0    NaN       HH Pandya
1  Rajasthan Royals  Wickets     7.0    NaN      JC Buttler

                                     Team1Players  \
0  ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …
1  ['V Kohli', 'F du Plessis', 'RM Patidar', 'GJ …

                                     Team2Players      Umpire1      Umpire2
0  ['WP Saha', 'Shubman Gill', 'MS Wade', 'HH Pan…  CB Gaffaney  Nitin Menon
```

```
1  ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …  CB Gaffaney  Nitin Menon
```

1. Find all the final winners in ipl

[65]:
```
ipl[ipl['MatchNumber']=='Final'][['Season', 'WinningTeam']]
```

[65]:

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

2. How many super over finishes have occured.

[66]:
```
ipl[ipl['SuperOver']=='Y'].shape[0]
```

[66]:
```
14
```

3. How many matches has csk won in kolkata

[67]:
```
ipl[(ipl['City']=='Kolkata') & (ipl['WinningTeam'] == 'Chennai Super Kings')].
↪shape[0]
```

[67]:
```
5
```

4. Toss winner is match winner in percentage

[68]:
```
(ipl[ipl['TossWinner'] == ipl['WinningTeam']]).shape[0]/ ipl.shape[0] * 100
```

[68]:
```
51.473684210526315
```

5. Movies with rating higher than 8 and votes>10000

[69]:
```
ipl[(movies['imdb_rating']>8) & (movies['imdb_votes']>10000)]
```

```
C:\Users\DILKHUSH\AppData\Local\Temp\ipykernel_15532\891066147.py:1:
UserWarning: Boolean Series key will be reindexed to match DataFrame index.
  ipl[(movies['imdb_rating']>8) & (movies['imdb_votes']>10000)]
```

[69]:

	ID	City	Date	Season	MatchNumber	Team1	Team2
0	1312200	Ahmedabad	2022-05-29	2022	Final	Rajasthan Royals	Gujarat Titans
11	1304109	Mumbai	2022-05-15	2022	63	Rajasthan Royals	Lucknow Super Giants
37	1304083	Mumbai	2022-04-24	2022	37	Lucknow Super Giants	Mumbai Indians
40	1304080	Mumbai	2022-04-22	2022	34	Rajasthan Royals	Delhi Capitals
143	1216502	NaN	2020-10-31	2020/21	52	Royal Challengers Bangalore	Sunrisers Hyderabad
146	1216499	Abu Dhabi	2020-10-28	2020/21	48	Royal Challengers Bangalore	Mumbai Indians
325	1082639	Chandigarh	2017-05-09	2017	49	Kings XI Punjab	Kolkata Knight Riders
354	1082608	Delhi	2017-04-17	2017	18	Delhi Daredevils	Kolkata Knight Riders
418	980929	Rajkot	2016-04-21	2016	15	Gujarat Lions	Sunrisers Hyderabad
426	980913	Delhi	2016-04-15	2016	7	Delhi Daredevils	Kings XI Punjab
436	829817	Mumbai	2015-05-19	2015	Qualifier 1	Chennai Super Kings	Mumbai Indians
469	829751	Mumbai	2015-04-25	2015	23	Mumbai Indians	Sunrisers Hyderabad
536	729309	NaN	2014-04-27	2014	16	Delhi Daredevils	Mumbai Indians
566	598060	Mumbai	2013-05-13	2013	62	Mumbai Indians	Sunrisers Hyderabad
567	598058	Jaipur	2013-05-12	2013	61	Rajasthan Royals	Chennai Super Kings
589	598034	Chennai	2013-04-28	2013	38	Chennai Super Kings	Kolkata Knight Riders
612	598012	Chennai	2013-04-13	2013	16		
638	548372	Delhi	2012-05-17	2012	67		
668	548342	Delhi	2012-04-27	2012	36		
669	548341	Pune	2012-04-26	2012	35		
693	548314	Visakhapatnam	2012-04-09	2012	9		
694	548313	Pune	2012-04-08	2012	8		
709	501264	Dharamsala	2011-05-21	2011	67		
714	501259	Mumbai	2011-05-16	2011	62		
778	419162	Mumbai	2010-04-21	2009/10	Semi Final		
869	392206	Johannesburg	2009-05-02	2009	26		
912	336021	Mumbai	2008-05-16	2007/08	38		
930	336001	Chennai	2008-05-02	2007/08	20		

```
612          Chennai Super Kings  Royal Challengers Bangalore
638             Delhi Daredevils  Royal Challengers Bangalore
668             Delhi Daredevils               Mumbai Indians
669                Pune Warriors              Deccan Chargers
693              Deccan Chargers               Mumbai Indians
694                Pune Warriors              Kings XI Punjab
709              Kings XI Punjab              Deccan Chargers
714                Pune Warriors              Deccan Chargers
778  Royal Challengers Bangalore               Mumbai Indians
869          Chennai Super Kings             Delhi Daredevils
912               Mumbai Indians        Kolkata Knight Riders
930          Chennai Super Kings             Delhi Daredevils

                                                Venue             TossWinner  \
0                       Narendra Modi Stadium, Ahmedabad       Rajasthan Royals
11                             Brabourne Stadium, Mumbai       Rajasthan Royals
37                              Wankhede Stadium, Mumbai         Mumbai Indians
40                              Wankhede Stadium, Mumbai         Delhi Capitals
143                              Sharjah Cricket Stadium    Sunrisers Hyderabad
146                                 Sheikh Zayed Stadium         Mumbai Indians
325  Punjab Cricket Association IS Bindra Stadium, …  Kolkata Knight Riders
354                                      Feroz Shah Kotla       Delhi Daredevils
418               Saurashtra Cricket Association Stadium    Sunrisers Hyderabad
426                                      Feroz Shah Kotla       Delhi Daredevils
436                                      Wankhede Stadium         Mumbai Indians
469                                      Wankhede Stadium         Mumbai Indians
536                                Sharjah Cricket Stadium         Mumbai Indians
566                                      Wankhede Stadium    Sunrisers Hyderabad
567                                 Sawai Mansingh Stadium       Rajasthan Royals
589                       MA Chidambaram Stadium, Chepauk  Kolkata Knight Riders
612                       MA Chidambaram Stadium, Chepauk    Chennai Super Kings
638                                      Feroz Shah Kotla       Delhi Daredevils
668                                      Feroz Shah Kotla         Mumbai Indians
669                             Subrata Roy Sahara Stadium        Deccan Chargers
693  Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket St…        Deccan Chargers
694                             Subrata Roy Sahara Stadium          Pune Warriors
709          Himachal Pradesh Cricket Association Stadium        Kings XI Punjab
714                             Dr DY Patil Sports Academy        Deccan Chargers
778                             Dr DY Patil Sports Academy         Mumbai Indians
869                                  New Wanderers Stadium       Delhi Daredevils
912                                      Wankhede Stadium         Mumbai Indians
930                       MA Chidambaram Stadium, Chepauk    Chennai Super Kings

    TossDecision SuperOver                 WinningTeam    WonBy  Margin  \
0            bat         N               Gujarat Titans  Wickets     7.0
11           bat         N             Rajasthan Royals     Runs    24.0
37         field         N         Lucknow Super Giants     Runs    36.0
```

```
40         field         N             Rajasthan Royals     Runs    15.0
143        field         N          Sunrisers Hyderabad  Wickets     5.0
146        field         N               Mumbai Indians  Wickets     5.0
325        field         N              Kings XI Punjab     Runs    14.0
354          bat         N        Kolkata Knight Riders  Wickets     4.0
418        field         N          Sunrisers Hyderabad  Wickets    10.0
426        field         N             Delhi Daredevils  Wickets     8.0
436          bat         N               Mumbai Indians     Runs    25.0
469          bat         N               Mumbai Indians     Runs    20.0
536          bat         N             Delhi Daredevils  Wickets     6.0
566          bat         N               Mumbai Indians  Wickets     7.0
567        field         N             Rajasthan Royals  Wickets     5.0
589        field         N          Chennai Super Kings     Runs    14.0
612        field         N          Chennai Super Kings  Wickets     4.0
638        field         N  Royal Challengers Bangalore     Runs    21.0
668        field         N             Delhi Daredevils     Runs    37.0
669          bat         N              Deccan Chargers     Runs    18.0
693          bat         N               Mumbai Indians  Wickets     5.0
694          bat         N                Pune Warriors     Runs    22.0
709        field         N              Deccan Chargers     Runs    82.0
714        field         N              Deccan Chargers  Wickets     6.0
778          bat         N               Mumbai Indians     Runs    35.0
869        field         N          Chennai Super Kings     Runs    18.0
912        field         N               Mumbai Indians  Wickets     8.0
930          bat         N             Delhi Daredevils  Wickets     8.0

    method  Player_of_Match  \
0      NaN        HH Pandya
11     NaN         TA Boult
37     NaN         KL Rahul
40     NaN       JC Buttler
143    NaN   Sandeep Sharma
146    NaN          SA Yadav
325    NaN        MM Sharma
354    NaN  NM Coulter-Nile
418    NaN          B Kumar
426    NaN         A Mishra
436    NaN       KA Pollard
469    NaN       SL Malinga
536    NaN          M Vijay
566    NaN       KA Pollard
567    NaN        SR Watson
589    NaN       MEK Hussey
612    NaN        RA Jadeja
638    NaN         CH Gayle
668    NaN         V Sehwag
669    NaN         CL White
```

```
693    NaN        RG Sharma
694    NaN       MN Samuels
709    NaN         S Dhawan
714    NaN         A Mishra
778    NaN       KA Pollard
869    NaN        SB Jakati
912    NaN       SM Pollock
930    NaN         V Sehwag

                                          Team1Players  \
0    ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …
11   ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …
37   ['Q de Kock', 'KL Rahul', 'MK Pandey', 'MP Sto…
40   ['JC Buttler', 'D Padikkal', 'SV Samson', 'SO …
143  ['JR Philippe', 'D Padikkal', 'V Kohli', 'AB d…
146  ['JR Philippe', 'D Padikkal', 'V Kohli', 'AB d…
325  ['MJ Guptill', 'M Vohra', 'SE Marsh', 'WP Saha…
354  ['SV Samson', 'SW Billings', 'KK Nair', 'SS Iy…
418  ['AJ Finch', 'BB McCullum', 'SK Raina', 'KD Ka…
426  ['Q de Kock', 'SS Iyer', 'SV Samson', 'P Negi'…
436  ['DR Smith', 'MEK Hussey', 'F du Plessis', 'SK…
469  ['LMP Simmons', 'PA Patel', 'UBT Chand', 'RG S…
536  ['Q de Kock', 'M Vijay', 'JP Duminy', 'KP Piet…
566  ['DR Smith', 'SR Tendulkar', 'KD Karthik', 'RG…
567  ['R Dravid', 'AM Rahane', 'JP Faulkner', 'SV S…
589  ['WP Saha', 'MEK Hussey', 'SK Raina', 'MS Dhon…
612  ['MEK Hussey', 'M Vijay', 'SK Raina', 'S Badri…
638  ['UBT Chand', 'DA Warner', 'Y Venugopal Rao', …
668  ['DPMD Jayawardene', 'V Sehwag', 'KP Pietersen…
669  ['MK Pandey', 'JD Ryder', 'SC Ganguly', 'RV Ut…
693  ['PA Patel', 'S Dhawan', 'B Chipli', 'DT Chris…
694  ['JD Ryder', 'SC Ganguly', 'MN Samuels', 'RV U…
709  ['PC Valthaty', 'AC Gilchrist', 'SE Marsh', 'K…
714  ['JD Ryder', 'MK Pandey', 'SC Ganguly', 'RV Ut…
778  ['JH Kallis', 'R Dravid', 'KP Pietersen', 'RV …
869  ['M Vijay', 'ML Hayden', 'SK Raina', 'S Badrin…
912  ['ST Jayasuriya', 'SR Tendulkar', 'RV Uthappa'…
930  ['PA Patel', 'SP Fleming', 'S Vidyut', 'MS Dho…

                                          Team2Players                Umpire1  \
0    ['WP Saha', 'Shubman Gill', 'MS Wade', 'HH Pan…            CB Gaffaney
11   ['Q de Kock', 'KL Rahul', 'A Badoni', 'DJ Hood…              PG Pathak
37   ['Ishan Kishan', 'RG Sharma', 'D Brevis', 'SA …              M Erasmus
40   ['PP Shaw', 'DA Warner', 'SN Khan', 'RR Pant',…          NA Patwardhan
143  ['DA Warner', 'WP Saha', 'MK Pandey', 'KS Will…  KN Ananthapadmanabhan
146  ['Q de Kock', 'Ishan Kishan', 'SA Yadav', 'SS …              UV Gandhe
325  ['SP Narine', 'CA Lynn', 'G Gambhir', 'RV Utha…         A Nand Kishore
```

354	['G Gambhir', 'C de Grandhomme', 'RV Uthappa',…	Nitin Menon
418	['DA Warner', 'S Dhawan', 'MC Henriques', 'EJG…	K Bharatan
426	['M Vijay', 'M Vohra', 'SE Marsh', 'DA Miller'…	S Ravi
436	['LMP Simmons', 'PA Patel', 'RG Sharma', 'KA P…	HDPK Dharmasena
469	['S Dhawan', 'DA Warner', 'KL Rahul', 'NV Ojha…	HDPK Dharmasena
536	['RG Sharma', 'AP Tare', 'CJ Anderson', 'AT Ra…	Aleem Dar
566	['PA Patel', 'S Dhawan', 'GH Vihari', 'CL Whit…	AK Chaudhary
567	['MEK Hussey', 'M Vijay', 'SK Raina', 'MS Dhon…	HDPK Dharmasena
589	['MS Bisla', 'G Gambhir', 'BB McCullum', 'JH K…	Aleem Dar
612	['CH Gayle', 'MA Agarwal', 'V Kohli', 'AB de V…	Asad Rauf
638	['CH Gayle', 'TM Dilshan', 'V Kohli', 'AB de V…	HDPK Dharmasena
668	['AC Blizzard', 'SR Tendulkar', 'RG Sharma', '…	Aleem Dar
669	['PA Patel', 'S Dhawan', 'CL White', 'KC Sanga…	S Ravi
693	['TL Suman', 'RE Levi', 'RG Sharma', 'AT Rayud…	AK Chaudhary
694	['AC Gilchrist', 'PC Valthaty', 'Mandeep Singh…	S Das
709	['S Dhawan', 'DB Ravi Teja', 'JP Duminy', 'CL …	Asad Rauf
714	['S Sohal', 'S Dhawan', 'KC Sangakkara', 'JP D…	S Ravi
778	['S Dhawan', 'SR Tendulkar', 'AM Nayar', 'AT R…	BR Doctrove
869	['G Gambhir', 'DA Warner', 'AB de Villiers', '…	DJ Harper
912	['Salman Butt', 'A Chopra', 'SC Ganguly', 'DJ …	BR Doctrove
930	['G Gambhir', 'V Sehwag', 'AB de Villiers', 'S…	BF Bowden

	Umpire2
0	Nitin Menon
11	Tapan Sharma
37	HAS Khalid
40	Nitin Menon
143	K Srinivasan
146	CB Gaffaney
325	S Ravi
354	CK Nandan
418	HDPK Dharmasena
426	C Shamshuddin
436	RK Illingworth
469	CB Gaffaney
536	VA Kulkarni
566	SJA Taufel
567	CK Nandan
589	SJA Taufel
612	AK Chaudhary
638	C Shamshuddin
668	BNJ Oxenford
669	RJ Tucker
693	JD Cloete
694	SJA Taufel
709	AM Saheba
714	SK Tarapore

```
778        RB Tiffin
869      RE Koertzen
912        DJ Harper
930      K Hariharan
```

6. Action movies with rating higher than 7.5

[70]:
```
movies[(movies['genres'].str.contains('Action')) & (movies['imdb_rating']>7.5)]
```

[70]:
```
                            title_x    imdb_id  \
0          Uri: The Surgical Strike  tt8291224
41             Family of Thakurganj  tt8897986
84                         Mukkabaaz  tt7180544
106                           Raazi  tt7098658
110   Parmanu: The Story of Pokhran  tt6826438
112         Bhavesh Joshi Superhero  tt6129302
169                The Ghazi Attack  tt6299040
219                Raag Desh (film)  tt6080746
258                   Irudhi Suttru  tt5310090
280                       Laal Rang  tt5600714
297                     Udta Punjab  tt4434004
354                   Dangal (film)  tt5074352
362               Bajrangi Bhaijaan  tt3863552
365          Baby (2015 Hindi film)  tt3848892
393      Detective Byomkesh Bakshy!  tt3447364
449                Titli (2014 film)  tt3019620
536                    Haider (film)  tt3390572
589                     Vishwaroopam  tt2199711
625                      Madras Cafe  tt2855648
668         Paan Singh Tomar (film)  tt1620933
693               Gangs of Wasseypur  tt1954470
694     Gangs of Wasseypur - Part 2  tt1954470
982                     Jodhaa Akbar  tt0449994
1039               1971 (2007 film)  tt0983990
1058       Black Friday (2007 film)  tt0400234
1188             Omkara (2006 film)  tt0488414
1293             Sarkar (2005 film)  tt0432047
1294                          Sehar  tt0477857
1361                 Lakshya (film)  tt0323013
1432                      Gangaajal  tt0373856
1495                 Company (film)  tt0296574
1554     The Legend of Bhagat Singh  tt0319736
1607         Nayak (2001 Hindi film)  tt0291376

                                                poster_path  \
0     https://upload.wikimedia.org/wikipedia/en/thum…
41    https://upload.wikimedia.org/wikipedia/en/9/99…
```

```
84    https://upload.wikimedia.org/wikipedia/en/thum…
106   https://upload.wikimedia.org/wikipedia/en/thum…
110   https://upload.wikimedia.org/wikipedia/en/thum…
112   https://upload.wikimedia.org/wikipedia/en/thum…
169   https://upload.wikimedia.org/wikipedia/en/thum…
219   https://upload.wikimedia.org/wikipedia/en/thum…
258   https://upload.wikimedia.org/wikipedia/en/f/fe…
280                                                NaN
297   https://upload.wikimedia.org/wikipedia/en/thum…
354   https://upload.wikimedia.org/wikipedia/en/thum…
362   https://upload.wikimedia.org/wikipedia/en/thum…
365   https://upload.wikimedia.org/wikipedia/en/thum…
393   https://upload.wikimedia.org/wikipedia/en/thum…
449   https://upload.wikimedia.org/wikipedia/en/thum…
536   https://upload.wikimedia.org/wikipedia/en/thum…
589   https://upload.wikimedia.org/wikipedia/en/thum…
625   https://upload.wikimedia.org/wikipedia/en/thum…
668   https://upload.wikimedia.org/wikipedia/en/thum…
693   https://upload.wikimedia.org/wikipedia/en/thum…
694   https://upload.wikimedia.org/wikipedia/en/thum…
982   https://upload.wikimedia.org/wikipedia/en/thum…
1039  https://upload.wikimedia.org/wikipedia/en/thum…
1058  https://upload.wikimedia.org/wikipedia/en/5/58…
1188  https://upload.wikimedia.org/wikipedia/en/thum…
1293  https://upload.wikimedia.org/wikipedia/en/thum…
1294  https://upload.wikimedia.org/wikipedia/en/thum…
1361  https://upload.wikimedia.org/wikipedia/en/thum…
1432  https://upload.wikimedia.org/wikipedia/en/thum…
1495  https://upload.wikimedia.org/wikipedia/en/thum…
1554  https://upload.wikimedia.org/wikipedia/en/thum…
1607  https://upload.wikimedia.org/wikipedia/en/thum…

                                              wiki_link  \
0     https://en.wikipedia.org/wiki/Uri:_The_Surgica…
41    https://en.wikipedia.org/wiki/Family_of_Thakur…
84              https://en.wikipedia.org/wiki/Mukkabaaz
106                 https://en.wikipedia.org/wiki/Raazi
110   https://en.wikipedia.org/wiki/Parmanu:_The_Sto…
112   https://en.wikipedia.org/wiki/Bhavesh_Joshi_Su…
169   https://en.wikipedia.org/wiki/The_Ghazi_Attack…
219              https://en.wikipedia.org/wiki/Raagdesh
258         https://en.wikipedia.org/wiki/Saala_Khadoos
280             https://en.wikipedia.org/wiki/Laal_Rang
297           https://en.wikipedia.org/wiki/Udta_Punjab
354         https://en.wikipedia.org/wiki/Dangal_(film)
362     https://en.wikipedia.org/wiki/Bajrangi_Bhaijaan
365   https://en.wikipedia.org/wiki/Baby_(2015_Hindi…
```

```
393   https://en.wikipedia.org/wiki/Detective_Byomke…
449     https://en.wikipedia.org/wiki/Titli_(2014_film)
536         https://en.wikipedia.org/wiki/Haider_(film)
589   https://en.wikipedia.org/wiki/Vishwaroop_(Hind…
625           https://en.wikipedia.org/wiki/Madras_Cafe
668   https://en.wikipedia.org/wiki/Paan_Singh_Tomar…
693    https://en.wikipedia.org/wiki/Gangs_of_Wasseypur
694   https://en.wikipedia.org/wiki/Gangs_of_Wasseyp…
982          https://en.wikipedia.org/wiki/Jodhaa_Akbar
1039    https://en.wikipedia.org/wiki/1971_(2007_film)
1058  https://en.wikipedia.org/wiki/Black_Friday_(20…
1188   https://en.wikipedia.org/wiki/Omkara_(2006_film)
1293   https://en.wikipedia.org/wiki/Sarkar_(2005_film)
1294                https://en.wikipedia.org/wiki/Sehar
1361      https://en.wikipedia.org/wiki/Lakshya_(film)
1432           https://en.wikipedia.org/wiki/Gangaajal
1495      https://en.wikipedia.org/wiki/Company_(film)
1554  https://en.wikipedia.org/wiki/The_Legend_of_Bh…
1607  https://en.wikipedia.org/wiki/Nayak_(2001_Hind…

                           title_y                original_title  is_adult  \
0          Uri: The Surgical Strike       Uri: The Surgical Strike         0
41             Family of Thakurganj           Family of Thakurganj         0
84                      The Brawler                       Mukkabaaz         0
106                          Raazi                           Raazi         0
110   Parmanu: The Story of Pokhran  Parmanu: The Story of Pokhran         0
112         Bhavesh Joshi Superhero        Bhavesh Joshi Superhero         0
169                The Ghazi Attack               The Ghazi Attack         0
219                       Raag Desh                      Raag Desh         0
258                   Saala Khadoos                  Saala Khadoos         0
280                       Laal Rang                      Laal Rang         0
297                     Udta Punjab                    Udta Punjab         0
354                          Dangal                         Dangal         0
362              Bajrangi Bhaijaan              Bajrangi Bhaijaan         0
365                            Baby                           Baby         0
393     Detective Byomkesh Bakshy!     Detective Byomkesh Bakshy!         0
449                          Titli                          Titli         0
536                         Haider                         Haider         0
589                    Vishwaroopam                   Vishwaroopam         0
625                     Madras Cafe                    Madras Cafe         0
668                Paan Singh Tomar               Paan Singh Tomar         0
693              Gangs of Wasseypur             Gangs of Wasseypur         0
694              Gangs of Wasseypur             Gangs of Wasseypur         0
982                    Jodhaa Akbar                   Jodhaa Akbar         0
1039                           1971                           1971         0
1058                  Black Friday                   Black Friday         0
1188                         Omkara                         Omkara         0
```

```
1293                        Sarkar                        Sarkar         0
1294                         Sehar                         Sehar         0
1361                       Lakshya                       Lakshya         0
1432                     Gangaajal                     Gangaajal         0
1495                       Company                       Company         0
1554    The Legend of Bhagat Singh    The Legend of Bhagat Singh         0
1607          Nayak: The Real Hero          Nayak: The Real Hero         0

      year_of_release runtime                  genres  imdb_rating  \
0                2019     138         Action|Drama|War          8.4
41               2019     127             Action|Drama          9.4
84               2017     154       Action|Drama|Sport          8.1
106              2018     138    Action|Drama|Thriller          7.8
110              2018     129     Action|Drama|History          7.7
112              2018     154             Action|Drama          7.6
169              2017     116      Action|Thriller|War          7.6
219              2017     135     Action|Drama|History          8.3
258              2016     109       Action|Drama|Sport          7.6
280              2016     147       Action|Crime|Drama          8.0
297              2016     148       Action|Crime|Drama          7.8
354              2016     161   Action|Biography|Drama          8.4
362              2015     163      Action|Comedy|Drama          8.0
365              2015     159          Action|Thriller          8.0
393              2015     139  Action|Mystery|Thriller          7.6
449              2014     116    Action|Drama|Thriller          7.6
536              2014     160       Action|Crime|Drama          8.1
589              2013     148          Action|Thriller          8.2
625              2013     130    Action|Drama|Thriller          7.7
668              2012     135   Action|Biography|Crime          8.2
693              2012     321      Action|Comedy|Crime          8.2
694              2012     321      Action|Comedy|Crime          8.2
982              2008     213     Action|Drama|History          7.6
1039             2007     160         Action|Drama|War          7.9
1058             2004     143       Action|Crime|Drama          8.5
1188             2006     155       Action|Crime|Drama          8.1
1293             2005     124       Action|Crime|Drama          7.6
1294             2005     125       Action|Crime|Drama          7.8
1361             2004     186     Action|Drama|Romance          7.9
1432             2003     157       Action|Crime|Drama          7.8
1495             2002     155       Action|Crime|Drama          8.0
1554             2002     155   Action|Biography|Drama          8.1
1607             2001     187    Action|Drama|Thriller          7.8

      imdb_votes                                                story  \
0          35112  Divided over five chapters  the film chronicle…
41           895  The film is based on small town of North India…
84          5434  A boxer (Shravan) belonging to upper cast  tra…
```

```
106        20289  Hidayat Khan is the son of an Indian freedom f…
110        18292  Captain Ashwat Raina's efforts to turn India i…
112         4928  Bhavesh Joshi Superhero is an action film abou…
169        10332  In 1971  amid rising tensions between India an…
219          341  A period film based on the historic 1945 India…
258        10507  An under-fire boxing coach  Prabhu is transfer…
280         3741  The friendship of two men is tested when thing…
297        23995  What on earth can a rock star  a migrant labor…
354       131338  Biopic of Mahavir Singh Phogat  who taught wre…
362        65877  A little mute girl from a Pakistani village ge…
365        49426  The country is perpetually under threat from t…
393        14674  CALCUTTA 1943 A WAR - A MYSTERY - and A DETECT…
449         3677  In the badlands of Delhi's dystopic underbelly…
536        46912  Vishal Bhardwaj's adaptation of William Shakes…
589        38016  Vishwanathan  a Kathak dance teacher in New Yo…
625        21393  An Indian Intelligence agent (portrayed by Joh…
668        29994  Paan Singh Tomar is a Hindi-language film  bas…
693        71636  Shahid Khan is exiled after impersonating the …
694        71636  Shahid Khan is exiled after impersonating the …
982        27541  Jodhaa Akbar is a sixteenth century love story…
1039        1121  Based on true facts  the film revolves around …
1058       16761  A dramatic presentation of the bomb blasts tha…
1188       17594  Advocate Raghunath Mishra has arranged the mar…
1293       14694  Meet Subhash Nagre - a wealthy and influential…
1294        1861  At the tender age of 8 Ajay Kumar is traumatiz…
1361       18777  Karan is a lazy good-for-nothing who lives on …
1432       14295  An SP Amit Kumar who is given charge of Tezpur…
1495       13474  Mallik is a henchman of Aslam Bhai  a Mumbai u…
1554       13455  Bhagat was born in British India during the ye…
1607       12522  Employed as a camera-man at a popular televisi…

                                                    summary  \
0     Indian army special forces execute a covert op…
41    The film is based on small town of North India…
84    A boxer struggles to make his mark in the boxi…
106   A Kashmiri woman agrees to marry a Pakistani a…
110   Ashwat Raina and his teammates arrive in Pokhr…
112   The origin story of Bhavesh Joshi  an Indian s…
169   A Pakistani submarine  Ghazi plans to secretly…
219   A period film based on the historic 1945 India…
258   The story of a former boxer who quits boxing f…
280   The friendship of two men is tested when thing…
297   A story that revolves around drug abuse in the…
354   Former wrestler Mahavir Singh Phogat and his t…
362   An Indian man with a magnanimous heart takes a…
365   An elite counter-intelligence unit learns of a…
393   While investigating the disappearance of a che…
```

```
449   A Hindi feature film set in the lower depths o…
536   A young man returns to Kashmir after his fathe…
589   When a classical dancer's suspecting wife sets…
625   An Indian intelligence agent journeys to a war…
668   The story of Paan Singh Tomar  an Indian athle…
693   A clash between Sultan and Shahid Khan leads t…
694   A clash between Sultan and Shahid Khan leads t…
982   A sixteenth century love story about a marriag…
1039  Based on true facts  the film revolves around …
1058  Black Friday is a film about the investigation…
1188  A politically-minded enforcer's misguided trus…
1293  The authority of a man  who runs a parallel go…
1294  Ajay Kumar the newly appointed  honest SSP of …
1361  An aimless  jobless  irresponsible grown man j…
1432  An IPS officer motivates and leads a dysfuncti…
1495  A small-time gangster named Chandu teams up wi…
1554  The story of a young revolutionary who raised …
1607  A man accepts a challenge by the chief ministe…

                                                 tagline  \
0                                                    NaN
41                                                   NaN
84                                                   NaN
106                             An incredible true story
110   1998| India: one secret operation| six Indians…
112            This year| justice will have a new name.
169                       The war you did not know about
219                                                  NaN
258                                                  NaN
280   Every job good or bad| must be done with honesty.
297                                                  NaN
354       You think our girls are any lesser than boys?
362                                                  NaN
365           History Is Made By Those Who Give A Damn!
393                                Expect The Unexpected
449                         Daring| Desireable| Dangerous
536                                                  NaN
589                                                  NaN
625                                                  NaN
668                                                  NaN
693                                                  NaN
694                                                  NaN
982                                                  NaN
1039                              Honor the heroes…
1058                 The story of the Bombay bomb blasts
1188                                                 NaN
1293  'There are no Rights and Wrongs. Only Power' -…
```

```
1294                                                  NaN
1361  It took him 24 years and 18000 feet to find hi…
1432                                                  NaN
1495                           A law & order enterprise
1554                                                  NaN
1607                                Fight the power

                                                   actors  \
0     Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga…
41    Jimmy Sheirgill|Mahie Gill|Nandish Singh|Prana…
84    Viineet Kumar|Jimmy Sheirgill|Zoya Hussain|Rav…
106   Alia Bhatt|Vicky Kaushal|Rajit Kapoor|Shishir …
110   John Abraham|Boman Irani|Diana Penty|Anuja Sat…
112   Harshvardhan Kapoor|Priyanshu Painyuli|Ashish …
169   Rana Daggubati|Kay Kay Menon|Atul Kulkarni|Om …
219   Kunal Kapoor|Amit Sadh|Mohit Marwah|Kenneth De…
258   Madhavan|Ritika Singh|Mumtaz Sorcar|Nassar|Rad…
280   Randeep Hooda|Akshay Oberoi|Rajniesh Duggall|P…
297   Shahid Kapoor|Alia Bhatt|Kareena Kapoor|Diljit…
354   Aamir Khan|Fatima Sana Shaikh|Sanya Malhotra|S…
362   Salman Khan|Harshaali Malhotra|Nawazuddin Sidd…
365   Akshay Kumar|Danny Denzongpa|Rana Daggubati|Ta…
393   Sushant Singh Rajput|Anand Tiwari|Neeraj Kabi|…
449   Nawazuddin Siddiqui|Niharika Singh|Anil George…
536   Tabu|Shahid Kapoor|Shraddha Kapoor|Kay Kay Men…
589   Kamal Haasan|Rahul Bose|Shekhar Kapur|Pooja Ku…
625   John Abraham|Nargis Fakhri|Raashi Khanna|Praka…
668                                       Irrfan Khan|
693   Manoj Bajpayee|Richa Chadha|Nawazuddin Siddiqu…
694   Manoj Bajpayee|Richa Chadha|Nawazuddin Siddiqu…
982   Hrithik Roshan|Aishwarya Rai Bachchan|Sonu Soo…
1039        Manoj Bajpayee|Ravi Kishan|Deepak Dobriyal|
1058  Kay Kay Menon|Pavan Malhotra|Aditya Srivastava…
1188  Ajay Devgn|Saif Ali Khan|Vivek Oberoi|Kareena …
1293  Amitabh Bachchan|Abhishek Bachchan|Kay Kay Men…
1294  Arshad Warsi|Pankaj Kapur|Mahima Chaudhry|Sush…
1361  Hrithik Roshan|Preity Zinta|Amitabh Bachchan|O…
1432  Ajay Devgn|Gracy Singh|Mohan Joshi|Yashpal Sha…
1495  Ajay Devgn|Mohanlal|Manisha Koirala|Seema Bisw…
1554  Ajay Devgn|Sushant Singh|D. Santosh|Akhilendra…
1607  Anil Kapoor|Rani Mukerji|Amrish Puri|Johnny Le…

              wins_nominations              release_date
0                      4 wins      11 January 2019 (USA)
41                        NaN       19 July 2019 (India)
84      3 wins & 6 nominations      12 January 2018 (USA)
106   21 wins & 26 nominations          11 May 2018 (USA)
```

```
110                       NaN          25 May 2018 (USA)
112               2 nominations           1 June 2018 (USA)
169       1 win & 7 nominations     17 February 2017 (USA)
219                       NaN       28 July 2017 (India)
258      9 wins & 2 nominations      29 January 2016 (USA)
280                       NaN      22 April 2016 (India)
297    11 wins & 19 nominations          17 June 2016 (USA)
354     23 wins & 4 nominations     21 December 2016 (USA)
362    25 wins & 13 nominations          17 July 2015 (USA)
365                     1 win    23 January 2015 (India)
393                       NaN         3 April 2015 (USA)
449      4 wins & 5 nominations         20 June 2014 (USA)
536    28 wins & 24 nominations       2 October 2014 (USA)
589     5 wins & 11 nominations    25 January 2013 (India)
625    10 wins & 10 nominations     23 August 2013 (India)
668    10 wins & 11 nominations          2 March 2012 (USA)
693    12 wins & 43 nominations  2 August 2012 (Singapore)
694    12 wins & 43 nominations  2 August 2012 (Singapore)
982    32 wins & 21 nominations     15 February 2008 (USA)
1039                    1 win       9 March 2007 (India)
1058              3 nominations    9 February 2007 (India)
1188   19 wins & 20 nominations          28 July 2006 (USA)
1293    2 wins & 10 nominations         1 July 2005 (India)
1294                      NaN       29 July 2005 (India)
1361    4 wins & 10 nominations         18 June 2004 (USA)
1432    4 wins & 29 nominations     29 August 2003 (India)
1495    16 wins & 9 nominations      15 April 2002 (India)
1554    11 wins & 5 nominations        7 June 2002 (India)
1607              2 nominations   7 September 2001 (India)
```

7. write a function that can return the track record of 2 teams against each other.

[ ]:

1.9 Adding new columns

1.9.1 Adding completely new column

[71]:
```
movies['country'] = 'India'
```

[72]:
```
movies.head(2)
```

[72]:
```
                    title_x    imdb_id  \
0  Uri: The Surgical Strike  tt8291224
1              Battalion 609  tt9472208

                                          poster_path  \
```

```
0  https://upload.wikimedia.org/wikipedia/en/thum…
1                                                NaN

                                          wiki_link  \
0  https://en.wikipedia.org/wiki/Uri:_The_Surgica…
1       https://en.wikipedia.org/wiki/Battalion_609

                   title_y           original_title  is_adult  \
0  Uri: The Surgical Strike  Uri: The Surgical Strike         0
1            Battalion 609            Battalion 609         0

   year_of_release runtime           genres  imdb_rating  imdb_votes  \
0             2019     138  Action|Drama|War          8.4       35112
1             2019     131              War          4.1          73

                                           story  \
0  Divided over five chapters  the film chronicle…
1  The story revolves around a cricket match betw…

                                         summary tagline  \
0  Indian army special forces execute a covert op…     NaN
1  The story of Battalion 609 revolves around a c…     NaN

                                          actors wins_nominations  \
0  Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga…           4 wins
1  Vicky Ahuja|Shoaib Ibrahim|Shrikant Kamat|Elen…              NaN

             release_date country
0    11 January 2019 (USA)   India
1  11 January 2019 (India)   India
```

1.9.2 Adding columns from existing

[73]:
```
movies.dropna(inplace=True)
```

[74]:
```
movies['lead actor'] = movies['actors'].str.split('|').apply(lambda x:x[0])
```

[75]:
```
movies.head(2)
```

[75]:
```
          title_x    imdb_id  \
11      Gully Boy  tt2395469
34  Yeh Hai India  tt5525846

                                         poster_path  \
11  https://upload.wikimedia.org/wikipedia/en/thum…
34  https://upload.wikimedia.org/wikipedia/en/thum…
```

```
                                    wiki_link        title_y original_title  \
11     https://en.wikipedia.org/wiki/Gully_Boy      Gully Boy      Gully Boy
34  https://en.wikipedia.org/wiki/Yeh_Hai_India  Yeh Hai India  Yeh Hai India

    is_adult  year_of_release runtime                  genres  imdb_rating  \
11         0             2019     153             Drama|Music          8.2
34         0             2017     128  Action|Adventure|Drama          5.7

    imdb_votes                                                story  \
11       22440  Gully Boy is a film about a 22-year-old boy "M…
34         169  Yeh Hai India  follows the story of a 25 years…

                                             summary  \
11  A coming-of-age story based on the lives of st…
34  Yeh Hai India  follows the story of a 25 years…

                    tagline  \
11        Apna Time Aayega!
34  A Film for Every Indian

                                                 actors        wins_nominations  \
11  Ranveer Singh|Alia Bhatt|Siddhant Chaturvedi|V…  6 wins & 3 nominations
34   Gavie Chahal|Mohan Agashe|Mohan Joshi|Lom Harsh|   2 wins & 1 nomination

              release_date country     lead actor
11  14 February 2019 (USA)   India  Ranveer Singh
34     24 May 2019 (India)   India   Gavie Chahal
```

1.10 Important DataFrame Functions

1.10.1 DF['col'].astype('new_datatype')

[76]:
```
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
```

[77]:
```
ipl['ID'] = ipl['ID'].astype('int32')
```

[78]:
```
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: float64(1), int32(1), object(18)
memory usage: 144.9+ KB
```

[79]:
```
ipl['Season'] = ipl['Season'].astype('category')
```

[80]:
```
ipl['Team1'] = ipl['Team1'].astype('category')
ipl['Team2'] = ipl['Team2'].astype('category')
ipl['WinningTeam'] = ipl['WinningTeam'].astype('category')
```

[81]:
```
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    category
 4   MatchNumber      950 non-null    object
 5   Team1            950 non-null    category
 6   Team2            950 non-null    category
 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    category
 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: category(4), float64(1), int32(1), object(14)
memory usage: 121.6+ KB
```

1.10.2 DF.value_counts

- Counts the frequency of values in the DF.
- Mostly used in Series rather than DF

[82]:
```
a = pd.Series([1,1,1,2,2,3])
a.value_counts()
```

[82]:
```
1    3
2    2
3    1
Name: count, dtype: int64
```

[83]:
```
marks = pd.DataFrame([
    [100, 80, 10],
    [90, 70, 7],
    [120, 100, 14],
    [80, 70, 14],
    [80, 70, 14]
], columns=['iq', 'marks', 'package'])

marks
```

[83]:

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

[84]:
```
marks.value_counts()
```

[84]:
```
iq   marks  package
80   70     14         2
90   70     7          1
100  80     10         1
120  100    14         1
Name: count, dtype: int64
```

[85]:
```
ipl = pd.read_csv('ipl-matches.csv')
ipl.head(2)
```

[85]:

	ID	City	Date	Season	MatchNumber	Team1	Team2	Venue	TossWinner	TossDecision	SuperOver	WinningTeam	WonBy	Margin	method	Player_of_Match	Team1Players
0	1312200	Ahmedabad	2022-05-29	2022	Final	Rajasthan Royals	Gujarat Titans	Narendra Modi Stadium, Ahmedabad	Rajasthan Royals	bat	N	Gujarat Titans	Wickets	7.0	NaN	HH Pandya	['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …
1	1312199	Ahmedabad	2022-05-27	2022	Qualifier 2	Royal Challengers Bangalore	Rajasthan Royals	Narendra Modi Stadium, Ahmedabad	Rajasthan Royals	field	N	Rajasthan Royals	Wickets	7.0	NaN	JC Buttler	

```
1  ['V Kohli', 'F du Plessis', 'RM Patidar', 'GJ …

                                        Team2Players      Umpire1      Umpire2
0  ['WP Saha', 'Shubman Gill', 'MS Wade', 'HH Pan…  CB Gaffaney  Nitin Menon
1  ['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …  CB Gaffaney  Nitin Menon
```

1. Find which player has won most POTM in finals and qualifiers.

[86]:
```
ipl[~ipl['MatchNumber'].str.isdigit()]['Player_of_Match'].value_counts()
```

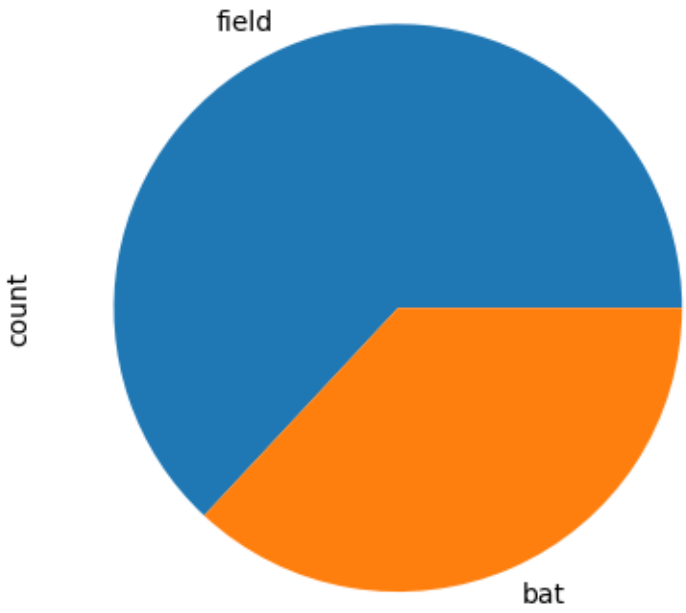
[86]:
```
Player_of_Match
KA Pollard           3
F du Plessis         3
SK Raina             3
A Kumble             2
MK Pandey            2
YK Pathan            2
M Vijay              2
JJ Bumrah            2
AB de Villiers       2
SR Watson            2
HH Pandya            1
Harbhajan Singh      1
A Nehra              1
V Sehwag             1
UT Yadav             1
MS Bisla             1
BJ Hodge             1
MEK Hussey           1
MS Dhoni             1
CH Gayle             1
MM Patel             1
DE Bollinger         1
AC Gilchrist         1
RG Sharma            1
DA Warner            1
MC Henriques         1
JC Buttler           1
RM Patidar           1
DA Miller            1
VR Iyer              1
SP Narine            1
RD Gaikwad           1
TA Boult             1
MP Stoinis           1
KS Williamson        1
RR Pant              1
```

```
SA Yadav              1
Rashid Khan           1
AD Russell            1
KH Pandya             1
KV Sharma             1
NM Coulter-Nile       1
Washington Sundar     1
BCJ Cutting           1
M Ntini               1
Name: count, dtype: int64
```

2. Toss decision plot

[87]:
```
ipl['TossDecision'].value_counts().plot(kind='pie')
```

[87]:
```
<Axes: ylabel='count'>
```



3. How many matches each team has played

[88]:
```
ipl['Team1'].value_counts() + ipl['Team2'].value_counts()
```

[88]:
```
Chennai Super Kings      208
Deccan Chargers           75
Delhi Capitals            63
```

```
Delhi Daredevils               161
Gujarat Lions                   30
Gujarat Titans                  16
Kings XI Punjab                190
Kochi Tuskers Kerala            14
Kolkata Knight Riders          223
Lucknow Super Giants            15
Mumbai Indians                 231
Pune Warriors                   46
Punjab Kings                    28
Rajasthan Royals               192
Rising Pune Supergiant          16
Rising Pune Supergiants         14
Royal Challengers Bangalore    226
Sunrisers Hyderabad            152
Name: count, dtype: int64
```

1.10.3 DF.sort_values('col', ascending=True, na_position='last', inplace=False)

- Sorts the Series or DataFrame based on values

[89]:
```
x = pd.Series([12, 14, 1, 56, 89])
x
```

[89]:
```
0    12
1    14
2     1
3    56
4    89
dtype: int64
```

[90]:
```
x.sort_values()
```

[90]:
```
2     1
0    12
1    14
3    56
4    89
dtype: int64
```

[91]:
```
movies = pd.read_csv('movies.csv')
movies.head(2)
```

[91]:
```
                    title_x    imdb_id  \
0  Uri: The Surgical Strike  tt8291224
1             Battalion 609  tt9472208
```

```
                                          poster_path  \
0  https://upload.wikimedia.org/wikipedia/en/thum…
1                                                NaN

                                           wiki_link  \
0  https://en.wikipedia.org/wiki/Uri:_The_Surgica…
1       https://en.wikipedia.org/wiki/Battalion_609

                    title_y            original_title  is_adult  \
0  Uri: The Surgical Strike  Uri: The Surgical Strike         0
1             Battalion 609             Battalion 609         0

   year_of_release runtime            genres  imdb_rating  imdb_votes  \
0             2019     138  Action|Drama|War          8.4       35112
1             2019     131               War          4.1          73

                                             story  \
0  Divided over five chapters  the film chronicle…
1  The story revolves around a cricket match betw…

                                           summary tagline  \
0  Indian army special forces execute a covert op…     NaN
1  The story of Battalion 609 revolves around a c…     NaN

                                           actors wins_nominations  \
0  Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga…           4 wins
1  Vicky Ahuja|Shoaib Ibrahim|Shrikant Kamat|Elen…              NaN

              release_date
0    11 January 2019 (USA)
1  11 January 2019 (India)
```

[92]:
```
movies.sort_values('title_x')
```

[92]:
```
                       title_x    imdb_id  \
1498        16 December (film)  tt0313844
1021              1920 (film)  tt1301698
287              1920: London  tt5638500
723     1920: The Evil Returns  tt2222550
1039          1971 (2007 film)  tt0983990
…                          …        …
778   Zindagi Na Milegi Dobara  tt1562872
670          Zindagi Tere Naam  tt2164702
756                   Zokkomon  tt1605790
939      Zor Lagaa Ke…Haiya!   tt1479857
1623                  Zubeidaa  tt0255713
```

```
                                          poster_path  \
1498  https://upload.wikimedia.org/wikipedia/en/thum…
1021  https://upload.wikimedia.org/wikipedia/en/thum…
287   https://upload.wikimedia.org/wikipedia/en/thum…
723   https://upload.wikimedia.org/wikipedia/en/e/e7…
1039  https://upload.wikimedia.org/wikipedia/en/thum…
…                                                   …
778   https://upload.wikimedia.org/wikipedia/en/thum…
670   https://upload.wikimedia.org/wikipedia/en/thum…
756   https://upload.wikimedia.org/wikipedia/en/thum…
939   https://upload.wikimedia.org/wikipedia/en/thum…
1623  https://upload.wikimedia.org/wikipedia/en/thum…

                                            wiki_link  \
1498   https://en.wikipedia.org/wiki/16_December_(film)
1021          https://en.wikipedia.org/wiki/1920_(film)
287           https://en.wikipedia.org/wiki/1920_London
723   https://en.wikipedia.org/wiki/1920:_The_Evil_R…
1039     https://en.wikipedia.org/wiki/1971_(2007_film)
…                                                   …
778   https://en.wikipedia.org/wiki/Zindagi_Na_Mileg…
670     https://en.wikipedia.org/wiki/Zindagi_Tere_Naam
756             https://en.wikipedia.org/wiki/Zokkomon
939   https://en.wikipedia.org/wiki/Zor_Lagaa_Ke…H…
1623            https://en.wikipedia.org/wiki/Zubeidaa

                     title_y            original_title  is_adult  \
1498                  16-Dec                     16-Dec         0
1021                    1920                       1920         0
287                1920 London                1920 London         0
723         1920: Evil Returns         1920: Evil Returns         0
1039                    1971                       1971         0
…                          …                          …       …
778   Zindagi Na Milegi Dobara   Zindagi Na Milegi Dobara         0
670          Zindagi Tere Naam          Zindagi Tere Naam         0
756                  Zokkomon                   Zokkomon         0
939     Zor Lagaa Ke… Haiya!    Zor Lagaa Ke… Haiya!         0
1623                 Zubeidaa                   Zubeidaa         0

      year_of_release runtime                    genres  imdb_rating  \
1498             2002     158          Action|Thriller          6.9
1021             2008     138   Horror|Mystery|Romance          6.4
287              2016     120           Horror|Mystery          4.1
723              2012     124     Drama|Horror|Romance          4.8
1039             2007     160         Action|Drama|War          7.9
…                …          …                       …          …
778              2011     155             Comedy|Drama          8.1
```

```
670              2012     120                   Romance          4.7
756              2011     109         Action|Adventure          4.0
939              2009      \N      Comedy|Drama|Family          6.4
1623             2001     153  Biography|Drama|History          6.2

      imdb_votes                                                story  \
1498        1091  16 December  1971 was the day when India won t…
1021        2588  A devotee of Bhagwan Shri Hanuman  Arjun Singh…
287         1373  Shivangi (Meera Chopra) lives in London with h…
723         1587  This story revolves around a famous poet who m…
1039        1121  Based on true facts  the film revolves around …
…           …                                                    …
778        60826  Three friends decide to turn their fantasy vac…
670           27  Mr. Singh  an elderly gentleman  relates to hi…
756          274  After the passing of his parents in an acciden…
939           46  A tree narrates the story of four Mumbai-based…
1623        1384  The film begins with Riyaz (Rajat Kapoor)  Zub…

                                                      summary  \
1498  Indian intelligence agents race against time t…
1021  After forsaking his family and religion  a hus…
287   After her husband is possessed by an evil spir…
723   This story revolves around a famous poet who m…
1039  Based on true facts  the film revolves around …
…                                                        …
778   Three friends decide to turn their fantasy vac…
670   Mr. Singh  an elderly gentleman  relates to hi…
756   An orphan is abused and abandoned  believed to…
939   Children build a tree-house to spy on a beggar…
1623  Zubeidaa  an aspiring Muslim actress  marries …

                                                     tagline  \
1498                                                     NaN
1021  A Love Made in Heaven…A Revenge Born in Hell…
287                                   Fear strikes again
723                                   Possession is back
1039                            Honor the heroes…
…                                                        …
778                                                      NaN
670                                                      NaN
756                      Betrayal. Friendship. Bravery.
939                                                      NaN
1623                            The Story of a Princess

                                                      actors  \
1498  Danny Denzongpa|Gulshan Grover|Milind Soman|Di…
1021  Rajniesh Duggall|Adah Sharma|Anjori Alagh|Raj …
```

```
287   Sharman Joshi|Meera Chopra|Vishal Karwal|Suren…
723   Vicky Ahuja|Tia Bajpai|Irma Jämhammar|Sharad K…
1039        Manoj Bajpayee|Ravi Kishan|Deepak Dobriyal|
…                                                   …
778   Hrithik Roshan|Farhan Akhtar|Abhay Deol|Katrin…
670   Mithun Chakraborty|Ranjeeta Kaur|Priyanka Meht…
756   Darsheel Safary|Anupam Kher|Manjari Fadnnis|Ti…
939   Meghan Jadhav|Mithun Chakraborty|Riya Sen|Seem…
1623  Karisma Kapoor|Rekha|Manoj Bajpayee|Rajit Kapo…

              wins_nominations               release_date
1498             2 nominations      22 March 2002 (India)
1021                        NaN  12 September 2008 (India)
287                         NaN           6 May 2016 (USA)
723                         NaN    2 November 2012 (India)
1039                    1 win       9 March 2007 (India)
…                           …                          …
778   30 wins & 22 nominations       15 July 2011 (India)
670                     1 win      16 March 2012 (India)
756                         NaN      22 April 2011 (India)
939                         NaN                       NaN
1623   3 wins & 13 nominations    19 January 2001 (India)

[1629 rows x 18 columns]
```

[93]:
```
students = pd.DataFrame(
    {
        'name':['nitish','ankit','rupesh',np.nan,'mrityunjay',np.
 ↪nan,'rishabh',np.nan,'aditya',np.nan],
        'college':['bit','iit','vit',np.nan,np.nan,'vlsi','ssit',np.nan,np.
 ↪nan,'git'],
        'branch':['eee','it','cse',np.nan,'me','ce','civ','cse','bio',np.nan],
        'cgpa':[6.66,8.25,6.41,np.nan,5.6,9.0,7.4,10,7.4,np.nan],
        'package':[4,5,6,np.nan,6,7,8,9,np.nan,np.nan]

    }
)

students
```

[93]:
```
         name college branch   cgpa  package
0      nitish     bit    eee   6.66      4.0
1       ankit     iit     it   8.25      5.0
2      rupesh     vit    cse   6.41      6.0
3         NaN     NaN    NaN    NaN      NaN
4  mrityunjay     NaN     me   5.60      6.0
5         NaN    vlsi     ce   9.00      7.0
```

```
6     rishabh    ssit    civ   7.40      8.0
7         NaN     NaN    cse  10.00      9.0
8      aditya     NaN    bio   7.40      NaN
9         NaN     git    NaN    NaN      NaN
```

[94]:
```
students.sort_values('name', na_position='first', inplace=False)
```

[94]:
```
         name college branch   cgpa  package
3         NaN     NaN    NaN    NaN      NaN
5         NaN    vlsi     ce   9.00      7.0
7         NaN     NaN    cse  10.00      9.0
9         NaN     git    NaN    NaN      NaN
8      aditya     NaN    bio   7.40      NaN
1       ankit     iit     it   8.25      5.0
4  mrityunjay     NaN     me   5.60      6.0
0      nitish     bit    eee   6.66      4.0
6     rishabh    ssit    civ   7.40      8.0
2      rupesh     vit    cse   6.41      6.0
```

1.10.4 DF.sort_values(list_of_columns, ascending=list)

[95]:
```
movies.sort_values(['year_of_release', 'title_x'], ascending=[True, False])
```

[95]:
```
                                  title_x     imdb_id  \
1623                              Zubeidaa   tt0255713
1625                  Yeh Zindagi Ka Safar   tt0298607
1622         Yeh Teraa Ghar Yeh Meraa Ghar   tt0298606
1620              Yeh Raaste Hain Pyaar Ke   tt0292740
1573                   Yaadein (2001 film)   tt0248617
…                                       …         …
37                       Article 15 (film)  tt10324144
46                           Arjun Patiala   tt7881524
10                                 Amavas    tt8396186
26    Albert Pinto Ko Gussa Kyun Aata Hai?   tt4355838
21                               22 Yards    tt9496212

                                              poster_path  \
1623  https://upload.wikimedia.org/wikipedia/en/thum…
1625  https://upload.wikimedia.org/wikipedia/en/thum…
1622  https://upload.wikimedia.org/wikipedia/en/thum…
1620  https://upload.wikimedia.org/wikipedia/en/thum…
1573  https://upload.wikimedia.org/wikipedia/en/thum…
…                                                   …
37    https://upload.wikimedia.org/wikipedia/en/thum…
46    https://upload.wikimedia.org/wikipedia/en/thum…
10    https://upload.wikimedia.org/wikipedia/en/thum…
26    https://upload.wikimedia.org/wikipedia/en/thum…
```

```
21    https://upload.wikimedia.org/wikipedia/en/thum…

                                              wiki_link  \
1623            https://en.wikipedia.org/wiki/Zubeidaa
1625  https://en.wikipedia.org/wiki/Yeh_Zindagi_Ka_S…
1622  https://en.wikipedia.org/wiki/Yeh_Teraa_Ghar_Y…
1620  https://en.wikipedia.org/wiki/Yeh_Raaste_Hain_…
1573  https://en.wikipedia.org/wiki/Yaadein_(2001_film)
…                                                   …
37      https://en.wikipedia.org/wiki/Article_15_(film)
46          https://en.wikipedia.org/wiki/Arjun_Patiala
10                https://en.wikipedia.org/wiki/Amavas
26    https://en.wikipedia.org/wiki/Albert_Pinto_Ko_…
21              https://en.wikipedia.org/wiki/22_Yards

                                  title_y  \
1623                             Zubeidaa
1625                  Yeh Zindagi Ka Safar
1622         Yeh Teraa Ghar Yeh Meraa Ghar
1620              Yeh Raaste Hain Pyaar Ke
1573                               Yaadein…
…                                       …
37                             Article 15
46                           Arjun Patiala
10                                 Amavas
26    Albert Pinto Ko Gussa Kyun Aata Hai?
21                                 22 Yards

                           original_title  is_adult  year_of_release runtime  \
1623                             Zubeidaa         0             2001     153
1625                  Yeh Zindagi Ka Safar         0             2001     146
1622         Yeh Teraa Ghar Yeh Meraa Ghar         0             2001     175
1620              Yeh Raaste Hain Pyaar Ke         0             2001     149
1573                               Yaadein…         0             2001     171
…                                       …       …                …     …
37                             Article 15         0             2019     130
46                           Arjun Patiala         0             2019     107
10                                 Amavas         0             2019     134
26    Albert Pinto Ko Gussa Kyun Aata Hai?         0             2019     100
21                                 22 Yards         0             2019     126

                        genres  imdb_rating  imdb_votes  \
1623  Biography|Drama|History          6.2        1384
1625                    Drama          3.0         133
1622             Comedy|Drama          5.7         704
1620            Drama|Romance          4.0         607
1573   Drama|Musical|Romance          4.4        3034
```

```
…                        …           …          …
37                 Crime|Drama         8.3      13417
46               Action|Comedy         4.1        676
10            Horror|Thriller         2.8        235
26                       Drama         4.8         56
21                       Sport         5.3        124

                                                  story  \
1623  The film begins with Riyaz (Rajat Kapoor)  Zub…
1625  Hindi pop-star  Sarina Devan  lives a wealthy …
1622  In debt; Dayashankar Pandey is forced to go to…
1620  Two con artistes and car thieves  Vicky (Ajay …
1573  Raj Singh Puri is best friends with L.K. Malho…
…                                                   …
37    In the rural heartlands of India  an upright p…
46    Arjun Patiala(Diljit Dosanjh)has recently been…
10    Far away from the bustle of the city  a young …
26    Albert leaves his house one morning without te…
21    A dramatic portrayal of a victorious tale of a…

                                                summary  \
1623  Zubeidaa  an aspiring Muslim actress  marries …
1625  A singer finds out she was adopted when the ed…
1622  In debt; Dayashankar Pandey is forced to go to…
1620  Two con artistes and car thieves  Vicky (Ajay …
1573  Raj Singh Puri is best friends with L.K. Malho…
…                                                   …
37    In the rural heartlands of India  an upright p…
46    This spoof comedy narrates the story of a cop …
10    The lives of a couple turn into a nightmare  a…
26    Albert Pinto goes missing one day and his girl…
21    A dramatic portrayal of a victorious tale of a…

                                    tagline  \
1623                The Story of a Princess
1625                                   NaN
1622                                   NaN
1620  Love is a journey… not a destination
1573                 memories to cherish…
…                                       …
37    Farq Bahut Kar Liya| Ab Farq Laayenge.
46                                     NaN
10                                     NaN
26                                     NaN
21                                     NaN

                                                actors  \
```

```
1623  Karisma Kapoor|Rekha|Manoj Bajpayee|Rajit Kapo…
1625  Ameesha Patel|Jimmy Sheirgill|Nafisa Ali|Gulsh…
1622  Sunil Shetty|Mahima Chaudhry|Paresh Rawal|Saur…
1620  Ajay Devgn|Madhuri Dixit|Preity Zinta|Vikram G…
1573  Jackie Shroff|Hrithik Roshan|Kareena Kapoor|Am…
…                                                   …
37    Ayushmann Khurrana|Nassar|Manoj Pahwa|Kumud Mi…
46    Diljit Dosanjh|Kriti Sanon|Varun Sharma|Ronit …
10    Ali Asgar|Vivan Bhatena|Nargis Fakhri|Sachiin …
26                            Manav Kaul|Nandita Das|
21    Barun Sobti|Rajit Kapur|Panchhi Bora|Kartikey …

             wins_nominations              release_date
1623  3 wins & 13 nominations   19 January 2001 (India)
1625                      NaN  16 November 2001 (India)
1622           1 nomination    12 October 2001 (India)
1620                      NaN    10 August 2001 (India)
1573           1 nomination      27 June 2001 (India)
…                         …                         …
37                     1 win        28 June 2019 (USA)
46                        NaN       26 July 2019 (USA)
10                        NaN   8 February 2019 (India)
26                        NaN     12 April 2019 (India)
21                        NaN     15 March 2019 (India)

[1629 rows x 18 columns]
```

1.10.5 Ser['col'].rank(ascending = False)

[96]:
```
batsman = pd.read_csv('batsman_runs_ipl.csv')
batsman.head()
```

[96]:
```
           batter  batsman_run
0  A Ashish Reddy          280
1        A Badoni          161
2      A Chandila            4
3        A Chopra           53
4     A Choudhary           25
```

[97]:
```
batsman['batsman_rank'] = batsman['batsman_run'].rank(ascending=False)
```

[98]:
```
batsman.sort_values('batsman_rank')
```

[98]:
```
           batter  batsman_run  batsman_rank
569       V Kohli         6634           1.0
462      S Dhawan         6244           2.0
130     DA Warner         5883           3.0
```

```
430     RG Sharma         5881           4.0
493      SK Raina         5536           5.0
..            …           …            …
512   SS Cottrell            0         594.0
466     S Kaushik            0         594.0
203     IC Pandey            0         594.0
467       S Ladda            0         594.0
468  S Lamichhane            0         594.0

[605 rows x 3 columns]
```

1.10.6 DF.sort_index(ascending=True)

[99]:
```
marks = {
    'maths':67,
    'english':57,
    'science':89,
    'hindi':100
}

marks_series = pd.Series(marks)
marks_series
```

[99]:
```
maths       67
english     57
science     89
hindi      100
dtype: int64
```

[100]:
```
marks_series.sort_index(ascending=False)
```

[100]:
```
science     89
maths       67
hindi      100
english     57
dtype: int64
```

[101]:
```
movies.sort_index(ascending=False)
```

[101]:
```
                                 title_x     imdb_id  \
1628                             Humsafar   tt2403201
1627                                Daaka  tt10833860
1626                    Sabse Bada Sukh    tt0069204
1625                 Yeh Zindagi Ka Safar   tt0298607
1624               Tera Mera Saath Rahen    tt0301250
…                                     …          …
4                        Evening Shadows    tt6028796
```

```
3                          Why Cheat India   tt8108208
2     The Accidental Prime Minister (film)   tt6986710
1                            Battalion 609   tt9472208
0                 Uri: The Surgical Strike   tt8291224

                                            poster_path  \
1628  https://upload.wikimedia.org/wikipedia/en/thum…
1627  https://upload.wikimedia.org/wikipedia/en/thum…
1626                                                NaN
1625  https://upload.wikimedia.org/wikipedia/en/thum…
1624  https://upload.wikimedia.org/wikipedia/en/2/2b…
…                                                   …
4                                                   NaN
3     https://upload.wikimedia.org/wikipedia/en/thum…
2     https://upload.wikimedia.org/wikipedia/en/thum…
1                                                   NaN
0     https://upload.wikimedia.org/wikipedia/en/thum…

                                             wiki_link  \
1628            https://en.wikipedia.org/wiki/Humsafar
1627               https://en.wikipedia.org/wiki/Daaka
1626     https://en.wikipedia.org/wiki/Sabse_Bada_Sukh
1625  https://en.wikipedia.org/wiki/Yeh_Zindagi_Ka_S…
1624  https://en.wikipedia.org/wiki/Tera_Mera_Saath_…
…                                                   …
4         https://en.wikipedia.org/wiki/Evening_Shadows
3         https://en.wikipedia.org/wiki/Why_Cheat_India
2     https://en.wikipedia.org/wiki/The_Accidental_P…
1           https://en.wikipedia.org/wiki/Battalion_609
0     https://en.wikipedia.org/wiki/Uri:_The_Surgica…

                           title_y                 original_title  is_adult  \
1628                       Humsafar                       Humsafar         0
1627                          Daaka                          Daaka         0
1626                Sabse Bada Sukh                Sabse Bada Sukh         0
1625           Yeh Zindagi Ka Safar           Yeh Zindagi Ka Safar         0
1624          Tera Mera Saath Rahen          Tera Mera Saath Rahen         0
…                                 …                              …       …
4                   Evening Shadows                Evening Shadows         0
3                   Why Cheat India                Why Cheat India         0
2     The Accidental Prime Minister  The Accidental Prime Minister         0
1                       Battalion 609                  Battalion 609         0
0            Uri: The Surgical Strike       Uri: The Surgical Strike         0

      year_of_release runtime            genres  imdb_rating  imdb_votes  \
1628             2011      35     Drama|Romance          9.0        2968
1627             2019     136            Action          7.4          38
```

```
1626             2018      \N      Comedy|Drama          6.1          13
1625             2001     146             Drama          3.0         133
1624             2001     148             Drama          4.9         278
…                …     …               …          …          …
4                2018     102             Drama          7.3         280
3                2019     121       Crime|Drama          6.0        1891
2                2019     112   Biography|Drama          6.1        5549
1                2019     131               War          4.1          73
0                2019     138  Action|Drama|War          8.4       35112

                                                    story  \
1628  Sara and Ashar are childhood friends who share…
1627  Shinda tries robbing a bank so he can be wealt…
1626  Village born Lalloo re-locates to Bombay  and …
1625  Hindi pop-star  Sarina Devan  lives a wealthy …
1624  Raj Dixit lives with his younger brother  Rahu…
…                                                   …
4     While gay rights and marriage equality has bee…
3     The movie focuses on existing malpractices in …
2     Based on the memoir by Indian policy analyst S…
1     The story revolves around a cricket match betw…
0     Divided over five chapters  the film chronicle…

                                                 summary tagline  \
1628  Ashar and Khirad are forced to get married due…     NaN
1627  Shinda tries robbing a bank so he can be wealt…     NaN
1626  Village born Lalloo re-locates to Bombay  and …     NaN
1625  A singer finds out she was adopted when the ed…     NaN
1624  A man is torn between his handicapped brother …     NaN
…                                                   …     …
4     Under the 'Evening Shadows'  truth often plays…     NaN
3     The movie focuses on existing malpractices in …     NaN
2     Explores Manmohan Singh's tenure as the Prime …     NaN
1     The story of Battalion 609 revolves around a c…     NaN
0     Indian army special forces execute a covert op…     NaN

                                                   actors  \
1628                                         Fawad Khan|
1627                               Gippy Grewal|Zareen Khan|
1626  Vijay Arora|Asrani|Rajni Bala|Kumud Damle|Utpa…
1625  Ameesha Patel|Jimmy Sheirgill|Nafisa Ali|Gulsh…
1624  Ajay Devgn|Sonali Bendre|Namrata Shirodkar|Pre…
…                                                   …
4     Mona Ambegaonkar|Ananth Narayan Mahadevan|Deva…
3     Emraan Hashmi|Shreya Dhanwanthary|Snighdadeep …
2     Anupam Kher|Akshaye Khanna|Aahana Kumra|Atul S…
1     Vicky Ahuja|Shoaib Ibrahim|Shrikant Kamat|Elen…
```

```
0     Vicky Kaushal|Paresh Rawal|Mohit Raina|Yami Ga…

             wins_nominations              release_date
1628                      NaN     TV Series (2011-2012)
1627                      NaN     1 November 2019 (USA)
1626                      NaN                       NaN
1625                      NaN  16 November 2001 (India)
1624                      NaN   7 November 2001 (India)
…                         …                         …
4     17 wins & 1 nomination   11 January 2019 (India)
3                         NaN     18 January 2019 (USA)
2                         NaN     11 January 2019 (USA)
1                         NaN   11 January 2019 (India)
0                     4 wins      11 January 2019 (USA)

[1629 rows x 18 columns]
```

1.10.7 DF.set_index('col', inplace=False)

- Sets given column as index

[102]:
```
batsman.set_index('batter', inplace=True)
batsman
```

[102]:
```
                batsman_run  batsman_rank
batter
A Ashish Reddy          280         166.5
A Badoni                161         226.0
A Chandila                4         535.0
A Chopra                 53         329.0
A Choudhary              25         402.5
…                       …             …
Yash Dayal                0         594.0
Yashpal Singh            47         343.0
Younis Khan               3         547.5
Yuvraj Singh           2754          27.0
Z Khan                  117         256.0

[605 rows x 2 columns]
```

1.10.8 DF.reset_index(inplace=False)

- resets the index.
- mostly used for transforming series into dataframe

[103]:
```
batsman.reset_index(inplace=True)
```

[104]:
```
batsman
```

[104]:
```
              batter  batsman_run  batsman_rank
0    A Ashish Reddy          280         166.5
1          A Badoni          161         226.0
2        A Chandila            4         535.0
3          A Chopra           53         329.0
4       A Choudhary           25         402.5
..              …           …           …
600      Yash Dayal            0         594.0
601   Yashpal Singh           47         343.0
602     Younis Khan            3         547.5
603    Yuvraj Singh         2754          27.0
604          Z Khan          117         256.0

[605 rows x 3 columns]
```

Note- How to replace existing index without loosing?

[105]:
```
batsman.reset_index().set_index('batsman_rank')
```

[105]:
```
               index          batter  batsman_run
batsman_rank
166.5              0  A Ashish Reddy          280
226.0              1        A Badoni          161
535.0              2      A Chandila            4
329.0              3        A Chopra           53
402.5              4     A Choudhary           25
…                  …              …           …
594.0            600      Yash Dayal            0
343.0            601   Yashpal Singh           47
547.5            602     Younis Khan            3
27.0             603    Yuvraj Singh         2754
256.0            604          Z Khan          117

[605 rows x 3 columns]
```

[106]:
```
marks_series.reset_index()
```

[106]:
```
      index    0
0     maths   67
1   english   57
2   science   89
3     hindi  100
```

1.10.9 DF.rename(columns={'existing_name':'new_name'}, inplace=True)

[107]:
```
movies.set_index('title_x', inplace=True)
```

[108]:
```
movies.rename(columns={'imdb_id':'imdb', 'poster_path':'link'}, inplace=True)
```

[109]:
```
movies
```

[109]:
```
                                          imdb  \
title_x
Uri: The Surgical Strike                tt8291224
Battalion 609                           tt9472208
The Accidental Prime Minister (film)    tt6986710
Why Cheat India                         tt8108208
Evening Shadows                         tt6028796
…                                             …
Tera Mera Saath Rahen                   tt0301250
Yeh Zindagi Ka Safar                    tt0298607
Sabse Bada Sukh                         tt0069204
Daaka                                  tt10833860
Humsafar                                tt2403201

                                                   link  \
title_x
Uri: The Surgical Strike
https://upload.wikimedia.org/wikipedia/en/thum…
Battalion 609
NaN
The Accidental Prime Minister (film)
https://upload.wikimedia.org/wikipedia/en/thum…
Why Cheat India
https://upload.wikimedia.org/wikipedia/en/thum…
Evening Shadows
NaN
…
…
Tera Mera Saath Rahen
https://upload.wikimedia.org/wikipedia/en/2/2b…
Yeh Zindagi Ka Safar
https://upload.wikimedia.org/wikipedia/en/thum…
Sabse Bada Sukh
NaN
Daaka
https://upload.wikimedia.org/wikipedia/en/thum…
Humsafar
https://upload.wikimedia.org/wikipedia/en/thum…
```

```
wiki_link  \
title_x
Uri: The Surgical Strike
https://en.wikipedia.org/wiki/Uri:_The_Surgica…
Battalion 609
https://en.wikipedia.org/wiki/Battalion_609
The Accidental Prime Minister (film)
https://en.wikipedia.org/wiki/The_Accidental_P…
Why Cheat India
https://en.wikipedia.org/wiki/Why_Cheat_India
Evening Shadows
https://en.wikipedia.org/wiki/Evening_Shadows
…
…
Tera Mera Saath Rahen
https://en.wikipedia.org/wiki/Tera_Mera_Saath_…
Yeh Zindagi Ka Safar
https://en.wikipedia.org/wiki/Yeh_Zindagi_Ka_S…
Sabse Bada Sukh
https://en.wikipedia.org/wiki/Sabse_Bada_Sukh
Daaka
https://en.wikipedia.org/wiki/Daaka
Humsafar
https://en.wikipedia.org/wiki/Humsafar

                                                                  title_y  \
title_x
Uri: The Surgical Strike                    Uri: The Surgical Strike
Battalion 609                                         Battalion 609
The Accidental Prime Minister (film)  The Accidental Prime Minister
Why Cheat India                                     Why Cheat India
Evening Shadows                                     Evening Shadows
…                                                                 …
Tera Mera Saath Rahen                          Tera Mera Saath Rahen
Yeh Zindagi Ka Safar                            Yeh Zindagi Ka Safar
Sabse Bada Sukh                                   Sabse Bada Sukh
Daaka                                                       Daaka
Humsafar                                                 Humsafar

                                                    original_title  is_adult  \
title_x
Uri: The Surgical Strike                    Uri: The Surgical Strike         0
Battalion 609                                         Battalion 609         0
The Accidental Prime Minister (film)  The Accidental Prime Minister         0
Why Cheat India                                     Why Cheat India         0
Evening Shadows                                     Evening Shadows         0
…                                                                 …       …
```

```
Tera Mera Saath Rahen                          Tera Mera Saath Rahen         0
Yeh Zindagi Ka Safar                            Yeh Zindagi Ka Safar         0
Sabse Bada Sukh                                      Sabse Bada Sukh         0
Daaka                                                           Daaka        0
Humsafar                                                      Humsafar       0

                                    year_of_release runtime  \
title_x
Uri: The Surgical Strike                       2019     138
Battalion 609                                  2019     131
The Accidental Prime Minister (film)           2019     112
Why Cheat India                                2019     121
Evening Shadows                                2018     102
…                                               …     …
Tera Mera Saath Rahen                          2001     148
Yeh Zindagi Ka Safar                           2001     146
Sabse Bada Sukh                                2018      \N
Daaka                                          2019     136
Humsafar                                       2011      35

                                                genres  imdb_rating  \
title_x
Uri: The Surgical Strike                  Action|Drama|War          8.4
Battalion 609                                        War          4.1
The Accidental Prime Minister (film)    Biography|Drama          6.1
Why Cheat India                             Crime|Drama          6.0
Evening Shadows                                  Drama          7.3
…                                                 …           …
Tera Mera Saath Rahen                            Drama          4.9
Yeh Zindagi Ka Safar                             Drama          3.0
Sabse Bada Sukh                         Comedy|Drama          6.1
Daaka                                          Action          7.4
Humsafar                                Drama|Romance          9.0

                                    imdb_votes  \
title_x
Uri: The Surgical Strike                 35112
Battalion 609                               73
The Accidental Prime Minister (film)      5549
Why Cheat India                           1891
Evening Shadows                            280
…                                           …
Tera Mera Saath Rahen                      278
Yeh Zindagi Ka Safar                       133
Sabse Bada Sukh                             13
Daaka                                       38
Humsafar                                  2968
```

```
  story  \
title_x
Uri: The Surgical Strike              Divided over five chapters  the film
chronicle…
Battalion 609                         The story revolves around a cricket match
betw…
The Accidental Prime Minister (film)  Based on the memoir by Indian policy
analyst S…
Why Cheat India                       The movie focuses on existing malpractices
in …
Evening Shadows                       While gay rights and marriage equality has
bee…
…
…
Tera Mera Saath Rahen                 Raj Dixit lives with his younger brother
Rahu…
Yeh Zindagi Ka Safar                  Hindi pop-star  Sarina Devan  lives a
wealthy …
Sabse Bada Sukh                       Village born Lalloo re-locates to Bombay
and …
Daaka                                 Shinda tries robbing a bank so he can be
wealt…
Humsafar                              Sara and Ashar are childhood friends who
share…

summary  \
title_x
Uri: The Surgical Strike              Indian army special forces execute a
covert op…
Battalion 609                         The story of Battalion 609 revolves around
a c…
The Accidental Prime Minister (film)  Explores Manmohan Singh's tenure as the
Prime …
Why Cheat India                       The movie focuses on existing malpractices
in …
Evening Shadows                       Under the 'Evening Shadows'  truth often
plays…
…
…
Tera Mera Saath Rahen                 A man is torn between his handicapped
brother …
Yeh Zindagi Ka Safar                  A singer finds out she was adopted when
the ed…
Sabse Bada Sukh                       Village born Lalloo re-locates to Bombay
and …
Daaka                                 Shinda tries robbing a bank so he can be
```

```
wealt…
Humsafar                                Ashar and Khirad are forced to get married
due…

                                      tagline  \
title_x
Uri: The Surgical Strike                 NaN
Battalion 609                            NaN
The Accidental Prime Minister (film)     NaN
Why Cheat India                          NaN
Evening Shadows                          NaN
…                                         …
Tera Mera Saath Rahen                    NaN
Yeh Zindagi Ka Safar                     NaN
Sabse Bada Sukh                          NaN
Daaka                                    NaN
Humsafar                                 NaN

 actors  \
title_x
Uri: The Surgical Strike              Vicky Kaushal|Paresh Rawal|Mohit
Raina|Yami Ga…
Battalion 609                         Vicky Ahuja|Shoaib Ibrahim|Shrikant
Kamat|Elen…
The Accidental Prime Minister (film)  Anupam Kher|Akshaye Khanna|Aahana
Kumra|Atul S…
Why Cheat India                       Emraan Hashmi|Shreya
Dhanwanthary|Snighdadeep …
Evening Shadows                       Mona Ambegaonkar|Ananth Narayan
Mahadevan|Deva…
…
…
Tera Mera Saath Rahen                 Ajay Devgn|Sonali Bendre|Namrata
Shirodkar|Pre…
Yeh Zindagi Ka Safar                  Ameesha Patel|Jimmy Sheirgill|Nafisa
Ali|Gulsh…
Sabse Bada Sukh                       Vijay Arora|Asrani|Rajni Bala|Kumud
Damle|Utpa…
Daaka                                                          Gippy
Grewal|Zareen Khan|
Humsafar
Fawad Khan|

                                           wins_nominations  \
title_x
Uri: The Surgical Strike                             4 wins
Battalion 609                                           NaN
```

```
The Accidental Prime Minister (film)                    NaN
Why Cheat India                                         NaN
Evening Shadows                        17 wins & 1 nomination
…                                                         …
Tera Mera Saath Rahen                                   NaN
Yeh Zindagi Ka Safar                                    NaN
Sabse Bada Sukh                                         NaN
Daaka                                                   NaN
Humsafar                                                NaN

                                                   release_date
title_x
Uri: The Surgical Strike                    11 January 2019 (USA)
Battalion 609                               11 January 2019 (India)
The Accidental Prime Minister (film)        11 January 2019 (USA)
Why Cheat India                             18 January 2019 (USA)
Evening Shadows                             11 January 2019 (India)
…                                                             …
Tera Mera Saath Rahen                        7 November 2001 (India)
Yeh Zindagi Ka Safar                        16 November 2001 (India)
Sabse Bada Sukh                                              NaN
Daaka                                        1 November 2019 (USA)
Humsafar                                     TV Series (2011-2012)

[1629 rows x 17 columns]
```

1.10.10 Ser.unique()

- Returns ndarray array containing unique values

[110]:
```
temp = pd.Series([1,1,2,2,3,3,4,4,5,5, np.nan,np.nan])
temp
```

[110]:
```
0     1.0
1     1.0
2     2.0
3     2.0
4     3.0
5     3.0
6     4.0
7     4.0
8     5.0
9     5.0
10    NaN
11    NaN
dtype: float64
```

[111]:
```
temp.unique()
```

[111]:
```
array([ 1.,  2.,  3.,  4.,  5., nan])
```

1.10.11 DF.unique()

- returns the total number of values excluding nan values
- `Note` - uniques includes nan values while nunique don't count them.

[112]:
```
temp.nunique()
```

[112]:
```
5
```

1.10.12 DF['col'].isnull()

- checks every value of Series or DF whether it is null or not

[113]:
```
students[students['name'].isnull()]
```

[113]:

	name	college	branch	cgpa	package
3	NaN	NaN	NaN	NaN	NaN
5	NaN	vlsi	ce	9.0	7.0
7	NaN	NaN	cse	10.0	9.0
9	NaN	git	NaN	NaN	NaN

1.10.13 DF['col'].notnull()

- works same as isnull()
- only difference is that it returns True if values is not null otherwise False

[114]:
```
students[students['name'].notnull()]
```

[114]:

	name	college	branch	cgpa	package
0	nitish	bit	eee	6.66	4.0
1	ankit	iit	it	8.25	5.0
2	rupesh	vit	cse	6.41	6.0
4	mrityunjay	NaN	me	5.60	6.0
6	rishabh	ssit	civ	7.40	8.0
8	aditya	NaN	bio	7.40	NaN

1.10.14 DF['col'].hasnans

- used for checking the Nan in whole DF or series

[115]:
```
students['name'].hasnans
```

[115]:
```
True
```

1.10.15 DF['col'].dropna(how='any', inplace=False)

- Drops the whole rows in which Nan is present.
- How parameter defines which row should be dropped.

1. if how is any it means if any of the values is nan then drop whole row
2. if how is all than row will be dropped if all values are nan

[116]:
```
students
```

[116]:

	name	college	branch	cgpa	package
0	nitish	bit	eee	6.66	4.0
1	ankit	iit	it	8.25	5.0
2	rupesh	vit	cse	6.41	6.0
3	NaN	NaN	NaN	NaN	NaN
4	mrityunjay	NaN	me	5.60	6.0
5	NaN	vlsi	ce	9.00	7.0
6	rishabh	ssit	civ	7.40	8.0
7	NaN	NaN	cse	10.00	9.0
8	aditya	NaN	bio	7.40	NaN
9	NaN	git	NaN	NaN	NaN

[117]:
```
students.dropna()
```

[117]:

	name	college	branch	cgpa	package
0	nitish	bit	eee	6.66	4.0
1	ankit	iit	it	8.25	5.0
2	rupesh	vit	cse	6.41	6.0
6	rishabh	ssit	civ	7.40	8.0

[118]:
```
students.dropna(how='all')
```

[118]:

	name	college	branch	cgpa	package
0	nitish	bit	eee	6.66	4.0
1	ankit	iit	it	8.25	5.0
2	rupesh	vit	cse	6.41	6.0
4	mrityunjay	NaN	me	5.60	6.0
5	NaN	vlsi	ce	9.00	7.0
6	rishabh	ssit	civ	7.40	8.0
7	NaN	NaN	cse	10.00	9.0
8	aditya	NaN	bio	7.40	NaN
9	NaN	git	NaN	NaN	NaN

[119]:
```
students.dropna(subset=['name', 'college'])
```

[119]:

	name	college	branch	cgpa	package
0	nitish	bit	eee	6.66	4.0
1	ankit	iit	it	8.25	5.0
2	rupesh	vit	cse	6.41	6.0

```
6  rishabh    ssit    civ  7.40      8.0
```

1.10.16 DF['col'].fillna()

- Handling missing values

[120]:
```
students['name'].fillna('unknown')
```

[120]:
```
0        nitish
1         ankit
2        rupesh
3       unknown
4    mrityunjay
5       unknown
6       rishabh
7       unknown
8        aditya
9       unknown
Name: name, dtype: object
```

[121]:
```
students['package'].fillna(students['package'].mean())
```

[121]:
```
0    4.000000
1    5.000000
2    6.000000
3    6.428571
4    6.000000
5    7.000000
6    8.000000
7    9.000000
8    6.428571
9    6.428571
Name: package, dtype: float64
```

Note- There is another way to handle filling missing values, we can fill by previous value or upcoming value

[122]:
```
students['name'].bfill()
```

[122]:
```
0        nitish
1         ankit
2        rupesh
3    mrityunjay
4    mrityunjay
5       rishabh
6       rishabh
7        aditya
8        aditya
```

```
9           NaN
Name: name, dtype: object
```

[123]:
```
students['name'].ffill()
```

[123]:
```
0        nitish
1         ankit
2        rupesh
3        rupesh
4    mrityunjay
5    mrityunjay
6       rishabh
7       rishabh
8        aditya
9        aditya
Name: name, dtype: object
```

1.10.17 DF.drop_duplicates(keep='first')

- droppes duplicated rows
- keep parameter tells which occurrence should be included.

[124]:
```
temp = pd.Series([1,1,1,2,3,3,4,4])
temp.drop_duplicates()
```

[124]:
```
0    1
3    2
4    3
6    4
dtype: int64
```

[125]:
```
marks = pd.DataFrame([
    [100,80,10],
    [90,70,7],
    [120,100,14],
    [80,70,14],
    [80,70,14]
], columns=['iq', 'marks', 'package'])
marks
```

[125]:

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

[126]:
```
marks.duplicated().sum()
```

[126]:
```
1
```

[127]:
```
marks.drop_duplicates(keep='last')
```

[127]:

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

Find the last match played by virat kohli in Delhi

[128]:
```
ipl.head(2)
```

[128]:

	ID	City	Date	Season	MatchNumber	Team1	Team2	Venue	TossWinner	TossDecision	SuperOver	WinningTeam	WonBy	Margin	method	Player_of_Match	Team1Players	Team2Players	Umpire1	Umpire2
0	1312200	Ahmedabad	2022-05-29	2022	Final	Rajasthan Royals	Gujarat Titans	Narendra Modi Stadium, Ahmedabad	Rajasthan Royals	bat	N	Gujarat Titans	Wickets	7.0	NaN	HH Pandya	['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …	['WP Saha', 'Shubman Gill', 'MS Wade', 'HH Pan…	CB Gaffaney	Nitin Menon
1	1312199	Ahmedabad	2022-05-27	2022	Qualifier 2	Royal Challengers Bangalore	Rajasthan Royals	Narendra Modi Stadium, Ahmedabad	Rajasthan Royals	field	N	Rajasthan Royals	Wickets	7.0	NaN	JC Buttler	['V Kohli', 'F du Plessis', 'RM Patidar', 'GJ …	['YBK Jaiswal', 'JC Buttler', 'SV Samson', 'D …	CB Gaffaney	Nitin Menon

[129]:
```
ipl['all_players'] = ipl['Team1Players'] + ipl['Team2Players']
```

[130]:
```
def did_kohli_play(players_list):
    return 'V Kohli' in players_list

ipl['did_kohli_played'] = ipl['all_players'].apply(did_kohli_play)
```

[131]:
```
ipl[(ipl['City'] == 'Delhi') & (ipl['did_kohli_played'])].
↪drop_duplicates(subset=['City', 'did_kohli_played'], keep='first')
```

[131]:
```
           ID   City        Date Season MatchNumber           Team1  \
208  1178421  Delhi  2019-04-28   2019          46  Delhi Capitals

                          Team2                 Venue      TossWinner  \
208  Royal Challengers Bangalore  Arun Jaitley Stadium  Delhi Capitals

    TossDecision  … WonBy Margin method  Player_of_Match  \
208          bat  …  Runs   16.0    NaN         S Dhawan

                                          Team1Players  \
208  ['PP Shaw', 'S Dhawan', 'SS Iyer', 'RR Pant', …

                                          Team2Players       Umpire1  \
208  ['PA Patel', 'V Kohli', 'AB de Villiers', 'S D…  BNJ Oxenford

                   Umpire2                                         all_players  \
208  KN Ananthapadmanabhan  ['PP Shaw', 'S Dhawan', 'SS Iyer', 'RR Pant', …

    did_kohli_played
208             True

[1 rows x 22 columns]
```

1.10.18 DF.drop(index=[], columns=[])

- dropping specific rows

[132]:
```
temp = pd.Series([10,2,3,16,45,78,10])
temp
```

[132]:
```
0    10
1     2
2     3
3    16
4    45
5    78
6    10
dtype: int64
```

[133]:
```
temp.drop(index=[0,6])
```

[133]:
```
1     2
2     3
3    16
```

```
4    45
5    78
dtype: int64
```

[134]:
```
students.drop(columns=['branch', 'cgpa'])
```

[134]:

	name	college	package
0	nitish	bit	4.0
1	ankit	iit	5.0
2	rupesh	vit	6.0
3	NaN	NaN	NaN
4	mrityunjay	NaN	6.0
5	NaN	vlsi	7.0
6	rishabh	ssit	8.0
7	NaN	NaN	9.0
8	aditya	NaN	NaN
9	NaN	git	NaN

1.10.19 DF.apply(func)

- apply function on every value of the series or DF

[135]:
```
temp = pd.Series([10, 20, 30, 40, 50])
temp
```

[135]:
```
0    10
1    20
2    30
3    40
4    50
dtype: int64
```

[136]:
```
def sigmoid(val):
    return 1/1+np.exp(-val)

temp.apply(sigmoid)
```

[136]:
```
0    1.000045
1    1.000000
2    1.000000
3    1.000000
4    1.000000
dtype: float64
```

[2]:
```
import numpy as np
import pandas as pd
```

[3]:
```
movies = pd.read_csv('imdb-top-1000.csv')
movies.head(1)
```

[3]:
```
                Series_Title Released_Year  Runtime  Genre  IMDB_Rating  \
0  The Shawshank Redemption          1994      142  Drama          9.3

          Director        Star1  No_of_Votes       Gross  Metascore
0  Frank Darabont  Tim Robbins      2343110  28341469.0       80.0
```

[4]:
```
genres = movies.groupby('Genre')
```

[5]:
```
genres
```

[5]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x000001CB1B0423F0>
```

1.10.20 Applying builtin aggregation functions on groupby objects

[6]:
```
genres.min()
```

[6]:
```
                              Series_Title Released_Year  Runtime  \
Genre
Action                                 300          1924       45
Adventure            2001: A Space Odyssey          1925       88
Animation                            Akira          1940       71
Biography                12 Years a Slave          1928       93
Comedy                (500) Days of Summer          1921       68
Crime                        12 Angry Men          1931       80
Drama                                1917          1925       64
Family          E.T. the Extra-Terrestrial          1971      100
Fantasy       Das Cabinet des Dr. Caligari          1920       76
Film-Noir                Shadow of a Doubt          1941      100
Horror                               Alien          1933       71
Mystery                       Dark City          1938       96
Thriller                Wait Until Dark          1967      108
Western    Il buono, il brutto, il cattivo          1965      132

           IMDB_Rating               Director                    Star1  \
Genre
Action             7.6       Abhishek Chaubey               Aamir Khan
Adventure          7.6         Akira Kurosawa               Aamir Khan
Animation          7.6            Adam Elliot            Adrian Molina
Biography          7.6             Adam McKay            Adrien Brody
Comedy             7.6  Alejandro G. Iñárritu               Aamir Khan
Crime              7.6         Akira Kurosawa               Ajay Devgn
Drama              7.6             Aamir Khan               Abhay Deol
Family             7.8             Mel Stuart             Gene Wilder
Fantasy            7.9             F.W. Murnau             Max Schreck
```

```
Film-Noir          7.8       Alfred Hitchcock           Humphrey Bogart
Horror             7.6     Alejandro Amenábar           Anthony Perkins
Mystery            7.6            Alex Proyas  Bernard-Pierre Donnadieu
Thriller           7.8          Terence Young            Audrey Hepburn
Western            7.8         Clint Eastwood            Clint Eastwood

           No_of_Votes        Gross  Metascore
Genre
Action           25312       3296.0       33.0
Adventure        29999      61001.0       41.0
Animation        25229     128985.0       61.0
Biography        27254      21877.0       48.0
Comedy           26337       1305.0       45.0
Crime            27712       6013.0       47.0
Drama            25088       3600.0       28.0
Family          178731    4000000.0       67.0
Fantasy          57428  337574718.0        NaN
Film-Noir        59556     449191.0       94.0
Horror           27007      89029.0       46.0
Mystery          33982    1035953.0       52.0
Thriller         27733   17550741.0       81.0
Western          65659    5321508.0       69.0
```

1. find the top 3 genres by total earning

[10]:
```
movies.groupby('Genre').sum()['Gross'].sort_values(ascending=False).head(3)
```

[10]:
```
Genre
Drama     3.540997e+10
Action    3.263226e+10
Comedy    1.566387e+10
Name: Gross, dtype: float64
```

[11]:
```
# Another way for same
movies.groupby('Genre')['Gross'].sum().sort_values(ascending=False).head(3)
```

[11]:
```
Genre
Drama     3.540997e+10
Action    3.263226e+10
Comedy    1.566387e+10
Name: Gross, dtype: float64
```

2. Find the genre with highest avg IMDB rating

[12]:
```
movies.groupby('Genre')['IMDB_Rating'].mean().sort_values(ascending=False).
↪head(1)
```

[12]:
```
Genre
Western    8.35
Name: IMDB_Rating, dtype: float64
```

3. Find director with most popularity

[17]:
```
movies.groupby('Director')['No_of_Votes'].sum().sort_values(ascending=False).
↪head(1)
```

[17]:
```
Director
Christopher Nolan    11578345
Name: No_of_Votes, dtype: int64
```

4. Find the highest rating of movies in each genre.

[18]:
```
movies.groupby('Genre')['IMDB_Rating'].max()
```

[18]:
```
Genre
Action       9.0
Adventure    8.6
Animation    8.6
Biography    8.9
Comedy       8.6
Crime        9.2
Drama        9.3
Family       7.8
Fantasy      8.1
Film-Noir    8.1
Horror       8.5
Mystery      8.4
Thriller     7.8
Western      8.8
Name: IMDB_Rating, dtype: float64
```

5. Find the number of movies done by each actor

[19]:
```
# Using groupby()
movies.groupby('Star1')['Series_Title'].count().sort_values(ascending=False)
```

[19]:
```
Star1
Tom Hanks             12
Robert De Niro        11
Clint Eastwood        10
Al Pacino             10
Leonardo DiCaprio      9
                      ..
Glen Hansard           1
Giuseppe Battiston     1
```

```
Giulietta Masina       1
Gerardo Taracena       1
Ömer Faruk Sorak       1
Name: Series_Title, Length: 660, dtype: int64
```

[20]:
```
# Using value_counts()
movies['Star1'].value_counts()
```

[20]:
```
Star1
Tom Hanks          12
Robert De Niro     11
Al Pacino          10
Clint Eastwood     10
Humphrey Bogart     9
                   ..
Preity Zinta        1
Javier Bardem       1
Ki-duk Kim          1
Vladimir Garin      1
Robert Donat        1
Name: count, Length: 660, dtype: int64
```

1.11 Attributes and Methods

1.11.1 len(DF.groupy('col'))

- Returns total number of groups.

[21]:
```
len(movies.groupby('Genre'))
```

[21]:
```
14
```

[23]:
```
movies['Genre'].nunique()
```

[23]:
```
14
```

1.11.2 DF.groupby('col').size()

- Returns series where index is group name and value is the values belongs to that group.
- output is same as DF['col'].value_counts()

[24]:
```
movies.groupby('Genre').size()
```

[24]:
```
Genre
Action       172
Adventure     72
Animation     82
Biography     88
```

```
Comedy       155
Crime        107
Drama        289
Family         2
Fantasy        2
Film-Noir      3
Horror        11
Mystery       12
Thriller       1
Western        4
dtype: int64
```

[25]:
```
movies.groupby('Released_Year').size()
```

[25]:
```
Released_Year
1920     1
1921     1
1922     1
1924     1
1925     2
        ..
2017    22
2018    19
2019    23
2020     6
PG       1
Length: 100, dtype: int64
```

1.11.3 DF.groupby('Col').first()/last()/nth(n)

- returns DF containing first or last or nth indexed values from each group.
- If any group has less values than n index than it ignores that group

[26]:
```
genres = movies.groupby('Genre')
genres.first()
```

[26]:
```
                              Series_Title Released_Year  Runtime  \
Genre
Action                     The Dark Knight          2008      152
Adventure                     Interstellar          2014      169
Animation    Sen to Chihiro no kamikakushi          2001      125
Biography                Schindler's List          1993      195
Comedy                       Gisaengchung          2019      132
Crime                       The Godfather          1972      175
Drama             The Shawshank Redemption          1994      142
Family          E.T. the Extra-Terrestrial          1982      115
Fantasy       Das Cabinet des Dr. Caligari          1920       76
```

```
Film-Noir                    The Third Man          1949      104
Horror                              Psycho          1960      109
Mystery                           Memento           2000      113
Thriller                Wait Until Dark             1967      108
Western    Il buono, il brutto, il cattivo          1966      161

           IMDB_Rating              Director                Star1  \
Genre
Action             9.0     Christopher Nolan       Christian Bale
Adventure          8.6     Christopher Nolan  Matthew McConaughey
Animation          8.6        Hayao Miyazaki        Daveigh Chase
Biography          8.9      Steven Spielberg          Liam Neeson
Comedy             8.6          Bong Joon Ho         Kang-ho Song
Crime              9.2  Francis Ford Coppola        Marlon Brando
Drama              9.3       Frank Darabont          Tim Robbins
Family             7.8      Steven Spielberg         Henry Thomas
Fantasy            8.1          Robert Wiene        Werner Krauss
Film-Noir          8.1            Carol Reed         Orson Welles
Horror             8.5      Alfred Hitchcock      Anthony Perkins
Mystery            8.4     Christopher Nolan           Guy Pearce
Thriller           7.8         Terence Young       Audrey Hepburn
Western            8.8          Sergio Leone       Clint Eastwood

           No_of_Votes        Gross  Metascore
Genre
Action         2303232  534858444.0       84.0
Adventure      1512360  188020017.0       74.0
Animation       651376   10055859.0       96.0
Biography      1213505   96898818.0       94.0
Comedy          552778   53367844.0       96.0
Crime          1620367  134966411.0      100.0
Drama          2343110   28341469.0       80.0
Family          372490  435110554.0       91.0
Fantasy          57428  337574718.0        NaN
Film-Noir       158731     449191.0       97.0
Horror          604211   32000000.0       97.0
Mystery        1125712   25544867.0       80.0
Thriller         27733   17550741.0       81.0
Western         688390    6100000.0       90.0
```

[27]:
```
genres.last()
```

[27]:
```
                          Series_Title Released_Year  Runtime  \
Genre
Action                Escape from Alcatraz          1979      112
Adventure                Kelly's Heroes          1970      144
Animation                The Jungle Book          1967       78
```

```
Biography                      Midnight Express          1978      121
Comedy                  Breakfast at Tiffany's          1961      115
Crime                             The 39 Steps          1935       86
Drama                                 Lifeboat          1944       97
Family     Willy Wonka & the Chocolate Factory          1971      100
Fantasy                                Nosferatu          1922       94
Film-Noir                   Shadow of a Doubt          1943      108
Horror                             The Others          2001      101
Mystery                          Lost Highway          1997      134
Thriller                       Wait Until Dark          1967      108
Western                 The Outlaw Josey Wales          1976      135

           IMDB_Rating             Director             Star1  No_of_Votes  \
Genre
Action             7.6          Don Siegel     Clint Eastwood       121731
Adventure          7.6      Brian G. Hutton     Clint Eastwood        45338
Animation          7.6  Wolfgang Reitherman        Phil Harris       166409
Biography          7.6          Alan Parker         Brad Davis        73662
Comedy             7.6        Blake Edwards     Audrey Hepburn       166544
Crime              7.6     Alfred Hitchcock       Robert Donat        51853
Drama              7.6     Alfred Hitchcock  Tallulah Bankhead        26471
Family             7.8           Mel Stuart       Gene Wilder       178731
Fantasy            7.9          F.W. Murnau        Max Schreck        88794
Film-Noir          7.8     Alfred Hitchcock      Teresa Wright        59556
Horror             7.6   Alejandro Amenábar      Nicole Kidman       337651
Mystery            7.6          David Lynch       Bill Pullman       131101
Thriller           7.8        Terence Young     Audrey Hepburn        27733
Western            7.8       Clint Eastwood     Clint Eastwood        65659

                Gross  Metascore
Genre
Action      43000000.0       76.0
Adventure    1378435.0       50.0
Animation  141843612.0       65.0
Biography   35000000.0       59.0
Comedy     679874270.0       76.0
Crime      302787539.0       93.0
Drama      852142728.0       78.0
Family       4000000.0       67.0
Fantasy    445151978.0        NaN
Film-Noir  123353292.0       94.0
Horror      96522687.0       74.0
Mystery      3796699.0       52.0
Thriller    17550741.0       81.0
Western     31800000.0       69.0
```

[28]:
```
genres.nth(6) # 7th values is at 6th index
```

[28]:
```
                                    Series_Title Released_Year  Runtime  \
16   Star Wars: Episode V - The Empire Strikes Back          1980      124
27                                           Se7en          1995      127
32                          It's a Wonderful Life          1946      130
66                                         WALL·E          2008       98
83                             The Great Dictator          1940      125
102                                   Braveheart          1995      178
118                           North by Northwest          1959      136
420                                       Sleuth          1972      138
724                                     Get Out           2017      104

         Genre  IMDB_Rating              Director             Star1  \
16      Action          8.7        Irvin Kershner       Mark Hamill
27       Crime          8.6         David Fincher    Morgan Freeman
32       Drama          8.6           Frank Capra     James Stewart
66   Animation          8.4        Andrew Stanton         Ben Burtt
83      Comedy          8.4       Charles Chaplin   Charles Chaplin
102  Biography          8.3            Mel Gibson        Mel Gibson
118  Adventure          8.3      Alfred Hitchcock        Cary Grant
420    Mystery          8.0  Joseph L. Mankiewicz  Laurence Olivier
724     Horror          7.7          Jordan Peele    Daniel Kaluuya

     No_of_Votes        Gross  Metascore
16       1159315  290475067.0       82.0
27       1445096  100125643.0       65.0
32        405801   82385199.0       89.0
66        999790  223808164.0       95.0
83        203150     288475.0        NaN
102       959181   75600000.0       68.0
118       299198   13275000.0       98.0
420        44748    4081254.0        NaN
724       492851  176040665.0       85.0
```

1.11.4 DF.groupby('col').get_group('group')

- returns DF containing specific values belonging to that column
- same output by column specific filtering

[29]:
```
genres.get_group('Horror')
```

[29]:
```
                Series_Title Released_Year  Runtime   Genre  IMDB_Rating  \
49                    Psycho          1960      109  Horror          8.5
75                     Alien          1979      117  Horror          8.4
271                The Thing          1982      109  Horror          8.1
419              The Exorcist          1973      122  Horror          8.0
544  Night of the Living Dead          1968       96  Horror          7.9
707             The Innocents          1961      100  Horror          7.8
```

```
724                  Get Out          2017      104  Horror          7.7
844                Halloween          1978       91  Horror          7.7
876        The Invisible Man          1933       71  Horror          7.7
932                      Saw          2004      103  Horror          7.6
948              The Others           2001      101  Horror          7.6

                Director             Star1  No_of_Votes        Gross  Metascore
49     Alfred Hitchcock    Anthony Perkins       604211   32000000.0       97.0
75         Ridley Scott   Sigourney Weaver       787806   78900000.0       89.0
271      John Carpenter       Kurt Russell       371271   13782838.0       57.0
419    William Friedkin      Ellen Burstyn       362393  232906145.0       81.0
544    George A. Romero        Duane Jones       116557      89029.0       89.0
707         Jack Clayton       Deborah Kerr        27007    2616000.0       88.0
724         Jordan Peele     Daniel Kaluuya       492851  176040665.0       85.0
844      John Carpenter   Donald Pleasence       233106   47000000.0       87.0
876         James Whale       Claude Rains        30683  298791505.0       87.0
932           James Wan         Cary Elwes       379020   56000369.0       46.0
948  Alejandro Amenábar      Nicole Kidman       337651   96522687.0       74.0
```

1.11.5 DF.groupby('col').groups

- Returns dictionary having group names as keys and index values belonging to that group as values in dictionary

[31]:
```
genres.groups
```

[31]:
```
{'Action': [2, 5, 8, 10, 13, 14, 16, 29, 30, 31, 39, 42, 44, 55, 57, 59, 60, 63,
68, 72, 106, 109, 129, 130, 134, 140, 142, 144, 152, 155, 160, 161, 166, 168,
171, 172, 177, 181, 194, 201, 202, 216, 217, 223, 224, 236, 241, 262, 275, 294,
308, 320, 325, 326, 331, 337, 339, 340, 343, 345, 348, 351, 353, 356, 357, 362,
368, 369, 375, 376, 390, 410, 431, 436, 473, 477, 479, 482, 488, 493, 496, 502,
507, 511, 532, 535, 540, 543, 564, 569, 570, 573, 577, 582, 583, 602, 605, 608,
615, 623, …], 'Adventure': [21, 47, 93, 110, 114, 116, 118, 137, 178, 179,
191, 193, 209, 226, 231, 247, 267, 273, 281, 300, 301, 304, 306, 323, 329, 361,
366, 377, 402, 406, 415, 426, 458, 470, 497, 498, 506, 513, 514, 537, 549, 552,
553, 566, 576, 604, 609, 618, 638, 647, 675, 681, 686, 692, 711, 713, 739, 755,
781, 797, 798, 851, 873, 884, 912, 919, 947, 957, 964, 966, 984, 991],
'Animation': [23, 43, 46, 56, 58, 61, 66, 70, 101, 135, 146, 151, 158, 170, 197,
205, 211, 213, 219, 229, 230, 242, 245, 246, 270, 330, 332, 358, 367, 378, 386,
389, 394, 395, 399, 401, 405, 409, 469, 499, 510, 516, 518, 522, 578, 586, 592,
595, 596, 599, 633, 640, 643, 651, 665, 672, 694, 728, 740, 741, 744, 756, 758,
761, 771, 783, 796, 799, 822, 828, 843, 875, 891, 892, 902, 906, 920, 956, 971,
976, 986, 992], 'Biography': [7, 15, 18, 35, 38, 54, 102, 107, 131, 139, 147,
157, 159, 173, 176, 212, 215, 218, 228, 235, 243, 263, 276, 282, 290, 298, 317,
328, 338, 342, 346, 359, 360, 365, 372, 373, 385, 411, 416, 418, 424, 429, 484,
525, 536, 542, 545, 575, 579, 587, 600, 606, 614, 622, 632, 635, 644, 649, 650,
657, 671, 673, 684, 729, 748, 753, 757, 759, 766, 770, 779, 809, 810, 815, 820,
```

```
831, 849, 858, 877, 882, 897, 910, 915, 923, 940, 949, 952, 987], 'Comedy': [19,
26, 51, 52, 64, 78, 83, 95, 96, 112, 117, 120, 127, 128, 132, 153, 169, 183,
192, 204, 207, 208, 214, 221, 233, 238, 240, 250, 251, 252, 256, 261, 266, 277,
284, 311, 313, 316, 318, 322, 327, 374, 379, 381, 392, 396, 403, 413, 414, 417,
427, 435, 445, 446, 449, 455, 459, 460, 463, 464, 466, 471, 472, 475, 481, 490,
494, 500, 503, 509, 526, 528, 530, 531, 533, 538, 539, 541, 547, 557, 558, 562,
563, 565, 574, 591, 593, 594, 598, 613, 626, 630, 660, 662, 667, 679, 680, 683,
687, 701, …], 'Crime': [1, 3, 4, 6, 22, 25, 27, 28, 33, 37, 41, 71, 77, 79,
86, 87, 103, 108, 111, 113, 123, 125, 133, 136, 162, 163, 164, 165, 180, 186,
187, 189, 198, 222, 232, 239, 255, 257, 287, 288, 299, 305, 335, 363, 364, 380,
384, 397, 437, 438, 441, 442, 444, 450, 451, 465, 474, 480, 485, 487, 505, 512,
519, 520, 523, 527, 546, 556, 560, 584, 597, 603, 607, 611, 621, 639, 653, 664,
669, 676, 695, 708, 723, 762, 763, 767, 775, 791, 795, 802, 811, 823, 827, 833,
885, 895, 921, 922, 926, 938, …], 'Drama': [0, 9, 11, 17, 20, 24, 32, 34, 36,
40, 45, 50, 53, 62, 65, 67, 73, 74, 76, 80, 82, 84, 85, 88, 89, 90, 91, 92, 94,
97, 98, 99, 100, 104, 105, 121, 122, 124, 126, 138, 141, 143, 148, 149, 150,
154, 156, 167, 174, 175, 182, 184, 185, 188, 190, 195, 196, 199, 200, 203, 206,
210, 225, 227, 234, 237, 244, 248, 249, 253, 254, 258, 259, 260, 264, 265, 268,
269, 272, 274, 278, 279, 280, 283, 285, 286, 289, 291, 292, 293, 295, 296, 297,
302, 303, 307, 310, 312, 314, 315, …], 'Family': [688, 698], 'Fantasy': [321,
568], 'Film-Noir': [309, 456, 712], 'Horror': [49, 75, 271, 419, 544, 707, 724,
844, 876, 932, 948], 'Mystery': [69, 81, 119, 145, 220, 393, 420, 714, 829, 899,
959, 961], 'Thriller': [700], 'Western': [12, 48, 115, 691]}
```

1.11.6 DF.groupby('col').describe()

- applies specific statistical functions to numeric columns of each group

[32]:
```
genres.describe()
```

[32]:

	Runtime							
	count	mean	std	min	25%	50%	75%	max
Genre								
Action	172.0	129.046512	28.500706	45.0	110.75	127.5	143.25	321.0
Adventure	72.0	134.111111	33.317320	88.0	109.00	127.0	149.00	228.0
Animation	82.0	99.585366	14.530471	71.0	90.00	99.5	106.75	137.0
Biography	88.0	136.022727	25.514466	93.0	120.00	129.0	146.25	209.0
Comedy	155.0	112.129032	22.946213	68.0	96.00	106.0	124.50	188.0
Crime	107.0	126.392523	27.689231	80.0	106.50	122.0	141.50	229.0
Drama	289.0	124.737024	27.740490	64.0	105.00	121.0	137.00	242.0
Family	2.0	107.500000	10.606602	100.0	103.75	107.5	111.25	115.0
Fantasy	2.0	85.000000	12.727922	76.0	80.50	85.0	89.50	94.0
Film-Noir	3.0	104.000000	4.000000	100.0	102.00	104.0	106.00	108.0
Horror	11.0	102.090909	13.604812	71.0	98.00	103.0	109.00	122.0
Mystery	12.0	119.083333	14.475423	96.0	110.75	117.5	130.25	138.0
Thriller	1.0	108.000000	NaN	108.0	108.00	108.0	108.00	108.0
Western	4.0	148.250000	17.153717	132.0	134.25	148.0	162.00	165.0

	IMDB_Rating		…	Gross		Metascore	\
	count	mean	…	75%	max	count	
Genre			…				
Action	172.0	7.949419	…	2.674437e+08	936662225.0	143.0	
Adventure	72.0	7.937500	…	1.998070e+08	874211619.0	64.0	
Animation	82.0	7.930488	…	2.520612e+08	873839108.0	75.0	
Biography	88.0	7.938636	…	9.829924e+07	753585104.0	79.0	
Comedy	155.0	7.901290	…	8.107809e+07	886752933.0	125.0	
Crime	107.0	8.016822	…	7.102163e+07	790482117.0	87.0	
Drama	289.0	7.957439	…	1.164461e+08	924558264.0	241.0	
Family	2.0	7.800000	…	3.273329e+08	435110554.0	2.0	
Fantasy	2.0	8.000000	…	4.182577e+08	445151978.0	0.0	
Film-Noir	3.0	7.966667	…	6.273068e+07	123353292.0	3.0	
Horror	11.0	7.909091	…	1.362817e+08	298791505.0	11.0	
Mystery	12.0	7.975000	…	1.310949e+08	474203697.0	8.0	
Thriller	1.0	7.800000	…	1.755074e+07	17550741.0	1.0	
Western	4.0	8.350000	…	1.920000e+07	31800000.0	4.0	

	mean	std	min	25%	50%	75%	max
Genre							
Action	73.419580	12.421252	33.0	65.00	74.0	82.00	98.0
Adventure	78.437500	12.345393	41.0	69.75	80.5	87.25	100.0
Animation	81.093333	8.813646	61.0	75.00	82.0	87.50	96.0
Biography	76.240506	11.028187	48.0	70.50	76.0	84.50	97.0
Comedy	78.720000	11.829160	45.0	72.00	79.0	88.00	99.0
Crime	77.080460	13.099102	47.0	69.50	77.0	87.00	100.0
Drama	79.701245	12.744687	28.0	72.00	82.0	89.00	100.0
Family	79.000000	16.970563	67.0	73.00	79.0	85.00	91.0
Fantasy	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Film-Noir	95.666667	1.527525	94.0	95.00	96.0	96.50	97.0
Horror	80.000000	15.362291	46.0	77.50	87.0	88.50	97.0
Mystery	79.125000	18.604435	52.0	65.25	77.0	98.50	100.0
Thriller	81.000000	NaN	81.0	81.00	81.0	81.00	81.0
Western	78.250000	9.032349	69.0	72.75	77.0	82.50	90.0

[14 rows x 40 columns]

1.11.7

1.11.8 DF.groupby('col').sample(n, replace=False)

- Returns n random samples from each group
- if the total values in a group is less than n then replace should be true because if samples is less than n then there will be error
- by default n is 1

[33]:
```
genres.sample(2, replace=True)
```

[33]:

	Series_Title	Released_Year	Runtime	Genre	IMDB_Rating	Director	Star1	No_of_Votes
59	Avengers: Endgame	2019	181	Action	8.4	Anthony Russo	Joe Russo	809955
106	Aliens	1986	137	Action	8.3	James Cameron	Sigourney Weaver	652719
798	Interstate 60: Episodes of the Road	2002	116	Adventure	7.7	Bob Gale	James Marsden	29999
470	Hunt for the Wilderpeople	2016	101	Adventure	7.9	Taika Waititi	Sam Neill	111483
330	Zootopia	2016	108	Animation	8.0	Byron Howard	Rich Moore	434143
43	The Lion King	1994	88	Animation	8.5	Roger Allers	Rob Minkoff	942045
18	Hamilton	2020	160	Biography	8.6	Thomas Kail	Lin-Manuel Miranda	55291
940	Finding Neverland	2004	106	Biography	7.6	Marc Forster	Johnny Depp	198677
687	The King of Comedy	1982	109	Comedy	7.8	Martin Scorsese	Robert De Niro	88511
379	Yeopgijeogin geunyeo	2001	137	Comedy	8.0	Jae-young Kwak	Tae-Hyun Cha	45403
6	Pulp Fiction	1994	154	Crime	8.9	Quentin Tarantino	John Travolta	1826188
288	Cool Hand Luke	1967	127	Crime	8.1	Stuart Rosenberg	Paul Newman	161984
32	It's a Wonderful Life	1946	130	Drama	8.6	Frank Capra	James Stewart	405801
825	Fried Green Tomatoes	1991	130	Drama	7.7	Jon Avnet	Kathy Bates	66941
698	Willy Wonka & the Chocolate Factory	1971	100	Family				
688	E.T. the Extra-Terrestrial	1982	115	Family				
321	Das Cabinet des Dr. Caligari	1920	76	Fantasy				
568	Nosferatu	1922	94	Fantasy				
309	The Third Man	1949	104	Film-Noir				
456	The Maltese Falcon	1941	100	Film-Noir				
844	Halloween	1978	91	Horror				
707	The Innocents	1961	100	Horror				
829	Spoorloos	1988	107	Mystery				
829	Spoorloos	1988	107	Mystery				
700	Wait Until Dark	1967	108	Thriller				
700	Wait Until Dark	1967	108	Thriller				
691	The Outlaw Josey Wales	1976	135	Western				
691	The Outlaw Josey Wales	1976	135	Western				

698	7.8	Mel Stuart	Gene Wilder	178731
688	7.8	Steven Spielberg	Henry Thomas	372490
321	8.1	Robert Wiene	Werner Krauss	57428
568	7.9	F.W. Murnau	Max Schreck	88794
309	8.1	Carol Reed	Orson Welles	158731
456	8.0	John Huston	Humphrey Bogart	148928
844	7.7	John Carpenter	Donald Pleasence	233106
707	7.8	Jack Clayton	Deborah Kerr	27007
829	7.7	George Sluizer	Bernard-Pierre Donnadieu	33982
829	7.7	George Sluizer	Bernard-Pierre Donnadieu	33982
700	7.8	Terence Young	Audrey Hepburn	27733
700	7.8	Terence Young	Audrey Hepburn	27733
691	7.8	Clint Eastwood	Clint Eastwood	65659
691	7.8	Clint Eastwood	Clint Eastwood	65659

	Gross	Metascore
59	858373000.0	78.0
106	85160248.0	84.0
798	174381905.0	NaN
470	5202582.0	81.0
330	341268248.0	78.0
43	422783777.0	88.0
18	440984783.0	90.0
940	51680613.0	67.0
687	2500000.0	73.0
379	772721890.0	NaN
6	107928762.0	94.0
288	16217773.0	92.0
32	82385199.0	89.0
825	82418501.0	64.0
698	4000000.0	67.0
688	435110554.0	91.0
321	337574718.0	NaN
568	445151978.0	NaN
309	449191.0	97.0
456	2108060.0	96.0
844	47000000.0	87.0
707	2616000.0	88.0
829	367916835.0	NaN
829	367916835.0	NaN
700	17550741.0	81.0
700	17550741.0	81.0
691	31800000.0	69.0
691	31800000.0	69.0

1.11.9 DF.groupby('col').nunique()

- unique values in each column of each group in form of DF

[35]:
```
genres.nunique()
```

[35]:

Genre	Series_Title	Released_Year	Runtime	IMDB_Rating	Director	Star1	\
Action	172	61	78	15	123	121	
Adventure	72	49	58	10	59	59	
Animation	82	35	41	11	51	77	
Biography	88	44	56	13	76	72	
Comedy	155	72	70	11	113	133	
Crime	106	56	65	14	86	85	
Drama	289	83	95	14	211	250	
Family	2	2	2	1	2	2	
Fantasy	2	2	2	2	2	2	
Film-Noir	3	3	3	3	3	3	
Horror	11	11	10	8	10	11	
Mystery	12	11	10	8	10	11	
Thriller	1	1	1	1	1	1	
Western	4	4	4	4	2	2	

Genre	No_of_Votes	Gross	Metascore
Action	172	172	50
Adventure	72	72	33
Animation	82	82	29
Biography	88	88	40
Comedy	155	155	44
Crime	107	107	39
Drama	288	287	52
Family	2	2	2
Fantasy	2	2	0
Film-Noir	3	3	3
Horror	11	11	9
Mystery	12	12	7
Thriller	1	1	1
Western	4	4	4

1.12 passing aggregation methods as dict

1.12.1 DF.goupby('col').agg({'col':'agg_func'})

- Apply specific aggregation functions of specific columns of each group
- We can apply more than one agg func on one column.

[36]:
```
genres.agg({
    'Runtime':'mean',
    'IMDB_Rating':'mean',
    'No_of_Votes':'sum',
    'Gross':'sum',
    'Metascore':'max'
})
```

[36]:

Genre	Runtime	IMDB_Rating	No_of_Votes	Gross	Metascore
Action	129.046512	7.949419	72282412	3.263226e+10	98.0
Adventure	134.111111	7.937500	22576163	9.496922e+09	100.0
Animation	99.585366	7.930488	21978630	1.463147e+10	96.0
Biography	136.022727	7.938636	24006844	8.276358e+09	97.0
Comedy	112.129032	7.901290	27620327	1.566387e+10	99.0
Crime	126.392523	8.016822	33533615	8.452632e+09	100.0
Drama	124.737024	7.957439	61367304	3.540997e+10	100.0
Family	107.500000	7.800000	551221	4.391106e+08	91.0
Fantasy	85.000000	8.000000	146222	7.827267e+08	NaN
Film-Noir	104.000000	7.966667	367215	1.259105e+08	97.0
Horror	102.090909	7.909091	3742556	1.034649e+09	97.0
Mystery	119.083333	7.975000	4203004	1.256417e+09	100.0
Thriller	108.000000	7.800000	27733	1.755074e+07	81.0
Western	148.250000	8.350000	1289665	5.822151e+07	90.0

[38]:
```
genres.agg({
    'Runtime':['min', 'mean'],
    'IMDB_Rating':'mean',
    'No_of_Votes':['sum', 'max'],
    'Gross':'sum',
    'Metascore':'max'
})
```

[38]:

	Runtime		IMDB_Rating	No_of_Votes		Gross	\
	min	mean	mean	sum	max	sum	
Genre							
Action	45	129.046512	7.949419	72282412	2303232	3.263226e+10	
Adventure	88	134.111111	7.937500	22576163	1512360	9.496922e+09	
Animation	71	99.585366	7.930488	21978630	999790	1.463147e+10	
Biography	93	136.022727	7.938636	24006844	1213505	8.276358e+09	
Comedy	68	112.129032	7.901290	27620327	939631	1.566387e+10	
Crime	80	126.392523	8.016822	33533615	1826188	8.452632e+09	
Drama	64	124.737024	7.957439	61367304	2343110	3.540997e+10	
Family	100	107.500000	7.800000	551221	372490	4.391106e+08	
Fantasy	76	85.000000	8.000000	146222	88794	7.827267e+08	
Film-Noir	100	104.000000	7.966667	367215	158731	1.259105e+08	
Horror	71	102.090909	7.909091	3742556	787806	1.034649e+09	

```
Mystery        96  119.083333    7.975000     4203004  1129894  1.256417e+09
Thriller      108  108.000000    7.800000       27733    27733  1.755074e+07
Western       132  148.250000    8.350000     1289665   688390  5.822151e+07
```

```
          Metascore
                max
Genre
Action         98.0
Adventure     100.0
Animation      96.0
Biography      97.0
Comedy         99.0
Crime         100.0
Drama         100.0
Family         91.0
Fantasy         NaN
Film-Noir      97.0
Horror         97.0
Mystery       100.0
Thriller       81.0
Western        90.0
```

Highest rated movie from each group

[41]:
```
temp = pd.DataFrame(columns=movies.columns)
for group, data in genres:
    temp = temp.append(data[data['IMDB_Rating'] == data['IMDB_Rating'].max()])
temp
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_20716\2112507160.py in ?()
      1 temp = pd.DataFrame(columns=movies.columns)
      2 for group, data in genres:
----> 3     temp = temp.append(data[data['IMDB_Rating'] == data['IMDB_Rating'].
 ↪max()])
      4 temp

c:\Program Files\Python312\Lib\site-packages\pandas\core\generic.py in ?(self,␣
 ↪name)
   6289             and name not in self._accessors
   6290             and self._info_axis.
 ↪_can_hold_identifiers_and_holds_name(name)
   6291         ):
   6292             return self[name]
-> 6293         return object.__getattribute__(self, name)
```

```
AttributeError: 'DataFrame' object has no attribute 'append'
```

1.13 Split Apply Combine

1.13.1 DF.groupby('col').apply(func, include_groups=False)

[45]:
```
genres.apply(np.minimum.reduce, include_groups=False)
```

[45]:

Genre	Series_Title	Released_Year	Runtime	\
Action	300	1924	45	
Adventure	2001: A Space Odyssey	1925	88	
Animation	Akira	1940	71	
Biography	12 Years a Slave	1928	93	
Comedy	(500) Days of Summer	1921	68	
Crime	12 Angry Men	1931	80	
Drama	1917	1925	64	
Family	E.T. the Extra-Terrestrial	1971	100	
Fantasy	Das Cabinet des Dr. Caligari	1920	76	
Film-Noir	Shadow of a Doubt	1941	100	
Horror	Alien	1933	71	
Mystery	Dark City	1938	96	
Thriller	Wait Until Dark	1967	108	
Western	Il buono, il brutto, il cattivo	1965	132	

Genre	IMDB_Rating	Director	Star1	\
Action	7.6	Abhishek Chaubey	Aamir Khan	
Adventure	7.6	Akira Kurosawa	Aamir Khan	
Animation	7.6	Adam Elliot	Adrian Molina	
Biography	7.6	Adam McKay	Adrien Brody	
Comedy	7.6	Alejandro G. Iñárritu	Aamir Khan	
Crime	7.6	Akira Kurosawa	Ajay Devgn	
Drama	7.6	Aamir Khan	Abhay Deol	
Family	7.8	Mel Stuart	Gene Wilder	
Fantasy	7.9	F.W. Murnau	Max Schreck	
Film-Noir	7.8	Alfred Hitchcock	Humphrey Bogart	
Horror	7.6	Alejandro Amenábar	Anthony Perkins	
Mystery	7.6	Alex Proyas	Bernard-Pierre Donnadieu	
Thriller	7.8	Terence Young	Audrey Hepburn	
Western	7.8	Clint Eastwood	Clint Eastwood	

Genre	No_of_Votes	Gross	Metascore
Action	25312	3296.0	NaN
Adventure	29999	61001.0	NaN
Animation	25229	128985.0	NaN

```
Biography        27254      21877.0        NaN
Comedy           26337       1305.0        NaN
Crime            27712       6013.0        NaN
Drama            25088       3600.0        NaN
Family          178731    4000000.0       67.0
Fantasy          57428  337574718.0        NaN
Film-Noir        59556     449191.0       94.0
Horror           27007      89029.0       46.0
Mystery          33982    1035953.0        NaN
Thriller         27733   17550741.0       81.0
Western          65659    5321508.0       69.0
```

[46]:
```
def foo(group):
    return group['Series_Title'].str.startswith('A').sum()
```

[48]:
```
genres.apply(foo, include_groups=False)
```

[48]:
```
Genre
Action       10
Adventure     2
Animation     2
Biography     9
Comedy       14
Crime         4
Drama        21
Family        0
Fantasy       0
Film-Noir     0
Horror        1
Mystery       0
Thriller      0
Western       0
dtype: int64
```

Find ranking of each movie in the group according to IMDB rating

[50]:
```
def rank_movie(group):
    group['genre_rank'] = group['IMDB_Rating'].rank(ascending=False)
    return group
```

[51]:
```
genres.apply(rank_movie, include_groups=False)
```

[51]:
```
                                                Series_Title Released_Year  \
Genre
Action    2                                  The Dark Knight          2008
          5       The Lord of the Rings: The Return of the King          2003
          8                                          Inception          2010
```

```
          10   The Lord of the Rings: The Fellowship of the Ring          2001
          13                The Lord of the Rings: The Two Towers          2002
…                                                          …           …
Thriller 700                                    Wait Until Dark          1967
Western  12                    Il buono, il brutto, il cattivo          1966
          48                       Once Upon a Time in the West          1968
          115                        Per qualche dollaro in più          1965
          691                            The Outlaw Josey Wales          1976

                Runtime  IMDB_Rating          Director              Star1  \
Genre
Action   2          152          9.0  Christopher Nolan     Christian Bale
          5          201          8.9      Peter Jackson        Elijah Wood
          8          148          8.8  Christopher Nolan  Leonardo DiCaprio
          10         178          8.8      Peter Jackson        Elijah Wood
          13         179          8.7      Peter Jackson        Elijah Wood
…                  …            …                …                 …
Thriller 700        108          7.8      Terence Young     Audrey Hepburn
Western  12         161          8.8       Sergio Leone     Clint Eastwood
          48         165          8.5       Sergio Leone        Henry Fonda
          115        132          8.3       Sergio Leone     Clint Eastwood
          691        135          7.8     Clint Eastwood     Clint Eastwood

                No_of_Votes        Gross  Metascore  genre_rank
Genre
Action   2          2303232  534858444.0       84.0         1.0
          5          1642758  377845905.0       94.0         2.0
          8          2067042  292576195.0       74.0         3.5
          10         1661481  315544750.0       92.0         3.5
          13         1485555  342551365.0       87.0         6.0
…                      …          …          …          …
Thriller 700          27733   17550741.0       81.0         1.0
Western  12         688390    6100000.0       90.0         1.0
          48         302844    5321508.0       80.0         2.0
          115        232772   15000000.0       74.0         3.0
          691         65659   31800000.0       69.0         4.0

[1000 rows x 10 columns]
```

Find normalized IMDB rating group wise

[52]:
```
def normalizer(group):
    group['norm_rating'] = (group['IMDB_Rating'] - group['IMDB_Rating'].min())/
 ↪(group['IMDB_Rating'].max() - group['IMDB_Rating'].min())
    return group
```

1.14 Groupby on multiple columns

[55]:
```
duo = movies.groupby(['Director', 'Star1'])
duo
```

[55]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x000001CB1EA0D0D0>
```

[57]:
```
duo.size()
```

[57]:
```
Director             Star1
Aamir Khan           Amole Gupte       1
Aaron Sorkin         Eddie Redmayne    1
Abdellatif Kechiche  Léa Seydoux       1
Abhishek Chaubey     Shahid Kapoor     1
Abhishek Kapoor      Amit Sadh         1
                                      ..
Zaza Urushadze       Lembit Ulfsak     1
Zoya Akhtar          Hrithik Roshan    1
                     Vijay Varma       1
Çagan Irmak          Çetin Tekindor    1
Ömer Faruk Sorak     Cem Yilmaz        1
Length: 898, dtype: int64
```

[58]:
```
duo.get_group(('Aaron Sorkin', 'Eddie Redmayne'))
```

[58]:
```
                    Series_Title Released_Year  Runtime  Genre  IMDB_Rating  \
612  The Trial of the Chicago 7          2020      129  Drama          7.8

         Director           Star1  No_of_Votes        Gross  Metascore
612  Aaron Sorkin  Eddie Redmayne        89896  853090410.0       77.0
```

Find most earning actor - director combo

[59]:
```
duo['Gross'].sum().sort_values(ascending=False).head(1)
```

[59]:
```
Director        Star1
Akira Kurosawa  Toshirô Mifune    2.999877e+09
Name: Gross, dtype: float64
```

Find the bes actor-genre combo based on metascore

[62]:
```
movies.groupby(['Star1', 'Genre'])['Metascore'].mean().reset_index().
↪sort_values('Metascore',ascending=False).head(1)
```

[62]:
```
              Star1  Genre  Metascore
230  Ellar Coltrane  Drama      100.0
```

1.15 Exercises

[63]:
```
ipl = pd.read_csv('deliveries1.csv')
ipl.head(2)
```

[63]:
```
   match_id  inning         batting_team                bowling_team  over  \
0         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1
1         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1

   ball    batsman non_striker    bowler  is_super_over  …  bye_runs  \
0     1  DA Warner    S Dhawan  TS Mills              0  …         0
1     2  DA Warner    S Dhawan  TS Mills              0  …         0

   legbye_runs  noball_runs  penalty_runs  batsman_runs  extra_runs  \
0            0            0             0             0           0
1            0            0             0             0           0

   total_runs  player_dismissed dismissal_kind fielder
0           0               NaN            NaN     NaN
1           0               NaN            NaN     NaN

[2 rows x 21 columns]
```

[64]:
```
ipl.shape
```

[64]:
```
(179078, 21)
```

1. Find the top 10 batsman in terms of runs

[65]:
```
ipl.groupby('batsman')['batsman_runs'].sum().sort_values(ascending=False).
↪head(10)
```

[65]:
```
batsman
V Kohli           5434
SK Raina          5415
RG Sharma         4914
DA Warner         4741
S Dhawan          4632
CH Gayle          4560
MS Dhoni          4477
RV Uthappa        4446
AB de Villiers    4428
G Gambhir         4223
Name: batsman_runs, dtype: int64
```

2. Find the batsman with max no of sixes

[71]:
```
six = ipl[ipl['batsman_runs'] == 6]
six.groupby('batsman')['batsman_runs'].sum().sort_values(ascending=False).
 ↪head(1)
```

[71]:
```
batsman
CH Gayle    1962
Name: batsman_runs, dtype: int64
```

3. Find batsman with most number of 4s and 6s in last 5 overs

[77]:
```
last_overs = ipl[ipl['over'] > 15]
last_overs[(last_overs['batsman_runs'] == 4) | (last_overs['batsman_runs']==6)].
 ↪groupby('batsman')['batsman'].count().sort_values(ascending=False).head(1)
```

[77]:
```
batsman
MS Dhoni    340
Name: batsman, dtype: int64
```

4. Virat Kohli's runs against all the teams

[79]:
```
temp_df = ipl[ipl['batsman'] == 'V Kohli']
temp_df.groupby('bowling_team')['batsman_runs'].sum().reset_index()
```

[79]:

	bowling_team	batsman_runs
0	Chennai Super Kings	749
1	Deccan Chargers	306
2	Delhi Capitals	66
3	Delhi Daredevils	763
4	Gujarat Lions	283
5	Kings XI Punjab	636
6	Kochi Tuskers Kerala	50
7	Kolkata Knight Riders	675
8	Mumbai Indians	628
9	Pune Warriors	128
10	Rajasthan Royals	370
11	Rising Pune Supergiant	83
12	Rising Pune Supergiants	188
13	Sunrisers Hyderabad	509

5. Highest score of any batsman

[80]:
```
def highest(batsman):
    temp_df = ipl[ipl['batsman'] == batsman]
    return temp_df.groupby('match_id')['batsman_runs'].sum().
 ↪sort_values(ascending=False).head(1).values[0]
```

[81]:
```
highest('V Kohli')
```

[81]:
```
113
```

[1]:
```
import numpy as np
import pandas as pd
```

[69]:
```
courses = pd.read_csv('courses.csv')
students = pd.read_csv('students.csv')
nov = pd.read_csv('reg-month1.csv')
dec = pd.read_csv('reg-month2.csv')
```

[5]:
```
nov
```

[5]:

	student_id	course_id
0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1
6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7
14	13	3
15	24	4
16	21	1
17	16	5
18	23	3
19	17	7
20	23	6
21	25	1
22	19	2
23	25	10
24	3	3

[6]:
```
dec
```

[6]:

	student_id	course_id
0	3	5
1	16	7
2	12	10
3	12	1
4	14	9

5	7	7
6	7	2
7	16	3
8	17	10
9	11	8
10	14	6
11	12	5
12	12	7
13	18	8
14	1	10
15	1	9
16	2	5
17	7	6
18	22	5
19	22	6
20	23	9
21	23	5
22	14	4
23	14	1
24	11	10
25	42	9
26	50	8
27	38	1

1.16 Concating DataFrames

[9]:
```
regs = pd.concat([nov, dec], ignore_index=True)
regs
```

[9]:

	student_id	course_id
0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1
6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7
14	13	3
15	24	4
16	21	1

```
17          16          5
18          23          3
19          17          7
20          23          6
21          25          1
22          19          2
23          25         10
24           3          3
25           3          5
26          16          7
27          12         10
28          12          1
29          14          9
30           7          7
31           7          2
32          16          3
33          17         10
34          11          8
35          14          6
36          12          5
37          12          7
38          18          8
39           1         10
40           1          9
41           2          5
42           7          6
43          22          5
44          22          6
45          23          9
46          23          5
47          14          4
48          14          1
49          11         10
50          42          9
51          50          8
52          38          1
```

1.16.1 Multiindex DataFrame

[10]:
```
pd.concat([nov, dec], keys=['Nov', 'Dec'])
```

[10]:
```
        student_id  course_id
Nov 0           23          1
    1           15          5
    2           18          6
    3           23          4
    4           16          9
```

	5	18	1
	6	1	1
	7	7	8
	8	22	3
	9	15	1
	10	19	4
	11	1	6
	12	7	10
	13	11	7
	14	13	3
	15	24	4
	16	21	1
	17	16	5
	18	23	3
	19	17	7
	20	23	6
	21	25	1
	22	19	2
	23	25	10
	24	3	3
Dec	0	3	5
	1	16	7
	2	12	10
	3	12	1
	4	14	9
	5	7	7
	6	7	2
	7	16	3
	8	17	10
	9	11	8
	10	14	6
	11	12	5
	12	12	7
	13	18	8
	14	1	10
	15	1	9
	16	2	5
	17	7	6
	18	22	5
	19	22	6
	20	23	9
	21	23	5
	22	14	4
	23	14	1
	24	11	10
	25	42	9
	26	50	8

```
    27          38          1
```

[11]:
```
pd.concat([nov, dec], axis=1)
```

[11]:

	student_id	course_id	student_id	course_id
0	23.0	1.0	3	5
1	15.0	5.0	16	7
2	18.0	6.0	12	10
3	23.0	4.0	12	1
4	16.0	9.0	14	9
5	18.0	1.0	7	7
6	1.0	1.0	7	2
7	7.0	8.0	16	3
8	22.0	3.0	17	10
9	15.0	1.0	11	8
10	19.0	4.0	14	6
11	1.0	6.0	12	5
12	7.0	10.0	12	7
13	11.0	7.0	18	8
14	13.0	3.0	1	10
15	24.0	4.0	1	9
16	21.0	1.0	2	5
17	16.0	5.0	7	6
18	23.0	3.0	22	5
19	17.0	7.0	22	6
20	23.0	6.0	23	9
21	25.0	1.0	23	5
22	19.0	2.0	14	4
23	25.0	10.0	14	1
24	3.0	3.0	11	10
25	NaN	NaN	42	9
26	NaN	NaN	50	8
27	NaN	NaN	38	1

1.17 Join

1.17.1 Inner Join

[12]:
```
students.merge(regs, how='inner', on='student_id')
```

[12]:

	student_id	name	partner	course_id
0	1	Kailash Harjo	23	1
1	1	Kailash Harjo	23	6
2	1	Kailash Harjo	23	10
3	1	Kailash Harjo	23	9
4	2	Esha Butala	1	5
5	3	Parveen Bhalla	3	3

6	3	Parveen Bhalla	3	5
7	7	Tarun Thaker	9	8
8	7	Tarun Thaker	9	10
9	7	Tarun Thaker	9	7
10	7	Tarun Thaker	9	2
11	7	Tarun Thaker	9	6
12	11	David Mukhopadhyay	20	7
13	11	David Mukhopadhyay	20	8
14	11	David Mukhopadhyay	20	10
15	12	Radha Dutt	19	10
16	12	Radha Dutt	19	1
17	12	Radha Dutt	19	5
18	12	Radha Dutt	19	7
19	13	Munni Varghese	24	3
20	14	Pranab Natarajan	22	9
21	14	Pranab Natarajan	22	6
22	14	Pranab Natarajan	22	4
23	14	Pranab Natarajan	22	1
24	15	Preet Sha	16	5
25	15	Preet Sha	16	1
26	16	Elias Dodiya	25	9
27	16	Elias Dodiya	25	5
28	16	Elias Dodiya	25	7
29	16	Elias Dodiya	25	3
30	17	Yasmin Palan	7	7
31	17	Yasmin Palan	7	10
32	18	Fardeen Mahabir	13	6
33	18	Fardeen Mahabir	13	1
34	18	Fardeen Mahabir	13	8
35	19	Qabeel Raman	12	4
36	19	Qabeel Raman	12	2
37	21	Seema Kota	15	1
38	22	Yash Sethi	21	3
39	22	Yash Sethi	21	5
40	22	Yash Sethi	21	6
41	23	Chhavi Lachman	18	1
42	23	Chhavi Lachman	18	4
43	23	Chhavi Lachman	18	3
44	23	Chhavi Lachman	18	6
45	23	Chhavi Lachman	18	9
46	23	Chhavi Lachman	18	5
47	24	Radhika Suri	17	4
48	25	Shashank D'Alia	2	1
49	25	Shashank D'Alia	2	10

1.17.2 Left Join

[13]:
```
courses.merge(regs, how='left', on='course_id')
```

[13]:

	course_id	course_name	price	student_id
0	1	python	2499	23.0
1	1	python	2499	18.0
2	1	python	2499	1.0
3	1	python	2499	15.0
4	1	python	2499	21.0
5	1	python	2499	25.0
6	1	python	2499	12.0
7	1	python	2499	14.0
8	1	python	2499	38.0
9	2	sql	3499	19.0
10	2	sql	3499	7.0
11	3	data analysis	4999	22.0
12	3	data analysis	4999	13.0
13	3	data analysis	4999	23.0
14	3	data analysis	4999	3.0
15	3	data analysis	4999	16.0
16	4	machine learning	9999	23.0
17	4	machine learning	9999	19.0
18	4	machine learning	9999	24.0
19	4	machine learning	9999	14.0
20	5	tableau	2499	15.0
21	5	tableau	2499	16.0
22	5	tableau	2499	3.0
23	5	tableau	2499	12.0
24	5	tableau	2499	2.0
25	5	tableau	2499	22.0
26	5	tableau	2499	23.0
27	6	power bi	1899	18.0
28	6	power bi	1899	1.0
29	6	power bi	1899	23.0
30	6	power bi	1899	14.0
31	6	power bi	1899	7.0
32	6	power bi	1899	22.0
33	7	ms sxcel	1599	11.0
34	7	ms sxcel	1599	17.0
35	7	ms sxcel	1599	16.0
36	7	ms sxcel	1599	7.0
37	7	ms sxcel	1599	12.0
38	8	pandas	1099	7.0
39	8	pandas	1099	11.0
40	8	pandas	1099	18.0
41	8	pandas	1099	50.0

42	9	plotly	699	16.0
43	9	plotly	699	14.0
44	9	plotly	699	1.0
45	9	plotly	699	23.0
46	9	plotly	699	42.0
47	10	pyspark	2499	7.0
48	10	pyspark	2499	25.0
49	10	pyspark	2499	12.0
50	10	pyspark	2499	17.0
51	10	pyspark	2499	1.0
52	10	pyspark	2499	11.0
53	11	Numpy	699	NaN
54	12	C++	1299	NaN

1.17.3 Right Join

[15]:
```
temp = pd.DataFrame({
    'student_id':[26, 27, 28],
    'name':['Nitish', 'Ankit', 'Rahul'],
    'partner':[28, 26, 27]
})

students = pd.concat([students, temp], ignore_index=True)
```

[18]:
```
students.merge(regs, how='right', on='student_id')
```

[18]:

	student_id	name	partner	course_id
0	23	Chhavi Lachman	18.0	1
1	15	Preet Sha	16.0	5
2	18	Fardeen Mahabir	13.0	6
3	23	Chhavi Lachman	18.0	4
4	16	Elias Dodiya	25.0	9
5	18	Fardeen Mahabir	13.0	1
6	1	Kailash Harjo	23.0	1
7	7	Tarun Thaker	9.0	8
8	22	Yash Sethi	21.0	3
9	15	Preet Sha	16.0	1
10	19	Qabeel Raman	12.0	4
11	1	Kailash Harjo	23.0	6
12	7	Tarun Thaker	9.0	10
13	11	David Mukhopadhyay	20.0	7
14	13	Munni Varghese	24.0	3
15	24	Radhika Suri	17.0	4
16	21	Seema Kota	15.0	1
17	16	Elias Dodiya	25.0	5
18	23	Chhavi Lachman	18.0	3
19	17	Yasmin Palan	7.0	7

20	23	Chhavi Lachman	18.0	6
21	25	Shashank D'Alia	2.0	1
22	19	Qabeel Raman	12.0	2
23	25	Shashank D'Alia	2.0	10
24	3	Parveen Bhalla	3.0	3
25	3	Parveen Bhalla	3.0	5
26	16	Elias Dodiya	25.0	7
27	12	Radha Dutt	19.0	10
28	12	Radha Dutt	19.0	1
29	14	Pranab Natarajan	22.0	9
30	7	Tarun Thaker	9.0	7
31	7	Tarun Thaker	9.0	2
32	16	Elias Dodiya	25.0	3
33	17	Yasmin Palan	7.0	10
34	11	David Mukhopadhyay	20.0	8
35	14	Pranab Natarajan	22.0	6
36	12	Radha Dutt	19.0	5
37	12	Radha Dutt	19.0	7
38	18	Fardeen Mahabir	13.0	8
39	1	Kailash Harjo	23.0	10
40	1	Kailash Harjo	23.0	9
41	2	Esha Butala	1.0	5
42	7	Tarun Thaker	9.0	6
43	22	Yash Sethi	21.0	5
44	22	Yash Sethi	21.0	6
45	23	Chhavi Lachman	18.0	9
46	23	Chhavi Lachman	18.0	5
47	14	Pranab Natarajan	22.0	4
48	14	Pranab Natarajan	22.0	1
49	11	David Mukhopadhyay	20.0	10
50	42	NaN	NaN	9
51	50	NaN	NaN	8
52	38	NaN	NaN	1

1.17.4 Outer Join

[20]:
```
students.merge(regs, how='outer', on='student_id').tail(10)
```

[20]:

	student_id	name	partner	course_id
53	23	Chhavi Lachman	18.0	5.0
54	24	Radhika Suri	17.0	4.0
55	25	Shashank D'Alia	2.0	1.0
56	25	Shashank D'Alia	2.0	10.0
57	26	Nitish	28.0	NaN
58	27	Ankit	26.0	NaN
59	28	Rahul	27.0	NaN
60	38	NaN	NaN	1.0

```
61          42              NaN      NaN        9.0
62          50              NaN      NaN        8.0
```

1. Find total revenue generated

[22]:
```
regs.merge(courses, how='inner', on='course_id')['price'].sum()
```

[22]:
```
154247
```

2. Find month by month revenue

[30]:
```
temp = pd.concat([nov, dec], keys=['Nov', 'Dec']).reset_index()
temp.merge(courses, how='inner', on='course_id').groupby('level_0')['price'].
 ↪sum()
```

[30]:
```
level_0
Dec    65072
Nov    89175
Name: price, dtype: int64
```

3. Print the registration table with cols name, course, price

[32]:
```
students.merge(regs, on='student_id').merge(courses, on='course_id')[['name',␣
 ↪'course_name', 'price']]
```

[32]:

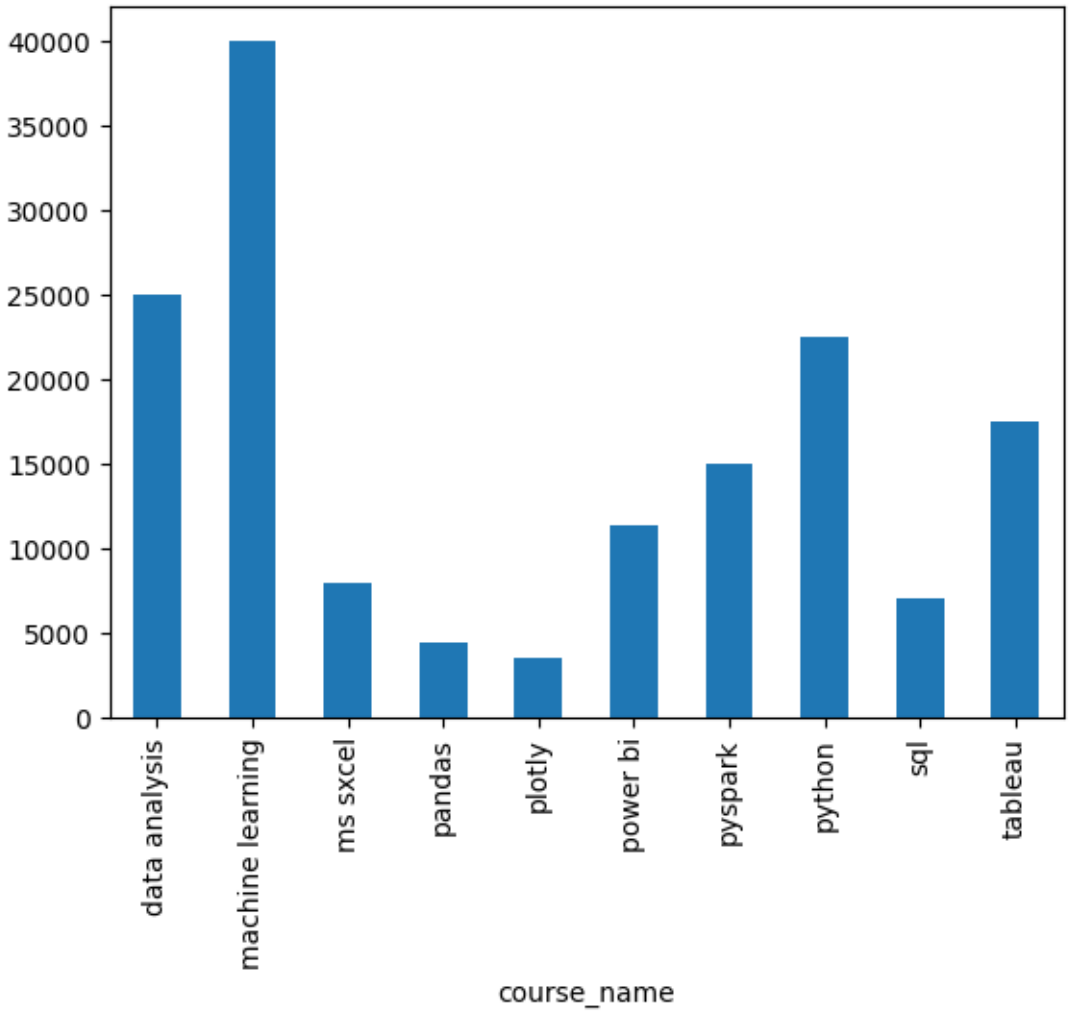
	name	course_name	price
0	Kailash Harjo	python	2499
1	Kailash Harjo	power bi	1899
2	Kailash Harjo	pyspark	2499
3	Kailash Harjo	plotly	699
4	Esha Butala	tableau	2499
5	Parveen Bhalla	data analysis	4999
6	Parveen Bhalla	tableau	2499
7	Tarun Thaker	pandas	1099
8	Tarun Thaker	pyspark	2499
9	Tarun Thaker	ms sxcel	1599
10	Tarun Thaker	sql	3499
11	Tarun Thaker	power bi	1899
12	David Mukhopadhyay	ms sxcel	1599
13	David Mukhopadhyay	pandas	1099
14	David Mukhopadhyay	pyspark	2499
15	Radha Dutt	pyspark	2499
16	Radha Dutt	python	2499
17	Radha Dutt	tableau	2499
18	Radha Dutt	ms sxcel	1599
19	Munni Varghese	data analysis	4999
20	Pranab Natarajan	plotly	699
21	Pranab Natarajan	power bi	1899

22	Pranab Natarajan	machine learning	9999
23	Pranab Natarajan	python	2499
24	Preet Sha	tableau	2499
25	Preet Sha	python	2499
26	Elias Dodiya	plotly	699
27	Elias Dodiya	tableau	2499
28	Elias Dodiya	ms sxcel	1599
29	Elias Dodiya	data analysis	4999
30	Yasmin Palan	ms sxcel	1599
31	Yasmin Palan	pyspark	2499
32	Fardeen Mahabir	power bi	1899
33	Fardeen Mahabir	python	2499
34	Fardeen Mahabir	pandas	1099
35	Qabeel Raman	machine learning	9999
36	Qabeel Raman	sql	3499
37	Seema Kota	python	2499
38	Yash Sethi	data analysis	4999
39	Yash Sethi	tableau	2499
40	Yash Sethi	power bi	1899
41	Chhavi Lachman	python	2499
42	Chhavi Lachman	machine learning	9999
43	Chhavi Lachman	data analysis	4999
44	Chhavi Lachman	power bi	1899
45	Chhavi Lachman	plotly	699
46	Chhavi Lachman	tableau	2499
47	Radhika Suri	machine learning	9999
48	Shashank D'Alia	python	2499
49	Shashank D'Alia	pyspark	2499

4. Plot bar chart for revenue/courses

[40]:
```
regs.merge(courses, on='course_id').groupby('course_name')['price'].sum().
↪plot(kind='bar')
```

[40]:
```
<Axes: xlabel='course_name'>
```



5. Find students who enrolled in both the months

[43]:
```
common_students = np.intersect1d(nov['student_id'], dec['student_id'])
students[students['student_id'].isin(common_students)]
```

[43]:

	student_id	name	partner
0	1	Kailash Harjo	23
2	3	Parveen Bhalla	3
6	7	Tarun Thaker	9
10	11	David Mukhopadhyay	20
15	16	Elias Dodiya	25
16	17	Yasmin Palan	7
17	18	Fardeen Mahabir	13
21	22	Yash Sethi	21
22	23	Chhavi Lachman	18

6. Find course that got no enrollment

[46]:
```
course_id = np.setdiff1d(courses['course_id'], regs['course_id'])
courses[courses['course_id'].isin(course_id)]
```

[46]:

	course_id	course_name	price
10	11	Numpy	699
11	12	C++	1299

7. Find students who did not enrolled into any of the courses

[48]:
```
student_id = np.setdiff1d(students['student_id'], regs['student_id'])
students[students['student_id'].isin(student_id)]
```

[48]:

	student_id	name	partner
3	4	Marlo Dugal	14
4	5	Kusum Bahri	6
5	6	Lakshmi Contractor	10
7	8	Radheshyam Dey	5
8	9	Nitika Chatterjee	4
9	10	Aayushman Sant	8
19	20	Hanuman Hegde	11
25	26	Nitish	28
26	27	Ankit	26
27	28	Rahul	27

8. Print student name and partner name for all enrolled students

[52]:
```
students.merge(students, left_on='student_id', right_on='partner')[['name_x',␣
↪'name_y']]
```

[52]:

	name_x	name_y
0	Kailash Harjo	Esha Butala
1	Esha Butala	Shashank D'Alia
2	Parveen Bhalla	Parveen Bhalla
3	Marlo Dugal	Nitika Chatterjee
4	Kusum Bahri	Radheshyam Dey
5	Lakshmi Contractor	Kusum Bahri
6	Tarun Thaker	Yasmin Palan
7	Radheshyam Dey	Aayushman Sant
8	Nitika Chatterjee	Tarun Thaker
9	Aayushman Sant	Lakshmi Contractor
10	David Mukhopadhyay	Hanuman Hegde
11	Radha Dutt	Qabeel Raman
12	Munni Varghese	Fardeen Mahabir
13	Pranab Natarajan	Marlo Dugal
14	Preet Sha	Seema Kota
15	Elias Dodiya	Preet Sha

```
16        Yasmin Palan        Radhika Suri
17     Fardeen Mahabir      Chhavi Lachman
18        Qabeel Raman         Radha Dutt
19       Hanuman Hegde  David Mukhopadhyay
20          Seema Kota         Yash Sethi
21          Yash Sethi    Pranab Natarajan
22      Chhavi Lachman       Kailash Harjo
23        Radhika Suri      Munni Varghese
24     Shashank D'Alia        Elias Dodiya
25              Nitish              Ankit
26               Ankit              Rahul
27               Rahul             Nitish
```

9. Find top 3 students who did most number enrollments

[56]:
```
top3students= regs.value_counts('student_id').head(3).reset_index()
top3students.merge(students, on='student_id')
```

[56]:
```
   student_id  count            name  partner
0          23      6  Chhavi Lachman       18
1           7      5    Tarun Thaker        9
2           1      4   Kailash Harjo       23
```

[63]:
```
regs.merge(students, on='student_id').groupby(['student_id', 'name'])['name'].
↪count().sort_values(ascending=False).head(3)
```

[63]:
```
student_id  name
23          Chhavi Lachman    6
7           Tarun Thaker      5
1           Kailash Harjo     4
Name: name, dtype: int64
```

10. Find top 3 students who spent most amount of money on courses

[68]:
```
regs.merge(students, on='student_id').merge(courses, on='course_id').
↪groupby(['student_id', 'name'])['price'].sum().sort_values(ascending=False).
↪head(3)
```

[68]:
```
student_id  name
23          Chhavi Lachman      22594
14          Pranab Natarajan    15096
19          Qabeel Raman        13498
Name: price, dtype: int64
```

1.18 Practise

1. Find top 3 stadiums with highest sixes/match ratio

[71]:
```
matches = pd.read_csv('matches.csv')
delivery = pd.read_csv('deliveries.csv')
```

[74]:
```
matches.head(2)
```

[74]:
```
   id  season       city        date                 team1  \
0   1    2017  Hyderabad  2017-04-05  Sunrisers Hyderabad
1   2    2017       Pune  2017-04-06       Mumbai Indians

                         team2                 toss_winner toss_decision  \
0  Royal Challengers Bangalore  Royal Challengers Bangalore         field
1       Rising Pune Supergiant       Rising Pune Supergiant         field

   result  dl_applied                  winner  win_by_runs  win_by_wickets  \
0  normal           0     Sunrisers Hyderabad           35               0
1  normal           0  Rising Pune Supergiant            0               7

  player_of_match                                      venue         umpire1  \
0    Yuvraj Singh  Rajiv Gandhi International Stadium, Uppal     AY Dandekar
1       SPD Smith    Maharashtra Cricket Association Stadium  A Nand Kishore

    umpire2  umpire3
0  NJ Llong      NaN
1    S Ravi      NaN
```

[75]:
```
delivery.head(2)
```

[75]:
```
   match_id  inning         batting_team                 bowling_team  over  \
0         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1
1         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1

   ball    batsman non_striker    bowler  is_super_over  …  bye_runs  \
0     1  DA Warner    S Dhawan  TS Mills              0  …         0
1     2  DA Warner    S Dhawan  TS Mills              0  …         0

   legbye_runs  noball_runs  penalty_runs  batsman_runs  extra_runs  \
0            0            0             0             0           0
1            0            0             0             0           0

   total_runs  player_dismissed dismissal_kind fielder
0           0               NaN            NaN     NaN
1           0               NaN            NaN     NaN

[2 rows x 21 columns]
```

[81]:
```
temp = delivery.merge(matches, left_on='match_id', right_on='id')
temp.head(2)
```

[81]:
```
   match_id  inning         batting_team                bowling_team  over  \
0         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1
1         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1

   ball    batsman non_striker    bowler  is_super_over  …  result  \
0     1  DA Warner    S Dhawan  TS Mills              0  …  normal
1     2  DA Warner    S Dhawan  TS Mills              0  …  normal

   dl_applied              winner  win_by_runs  win_by_wickets  \
0           0  Sunrisers Hyderabad           35               0
1           0  Sunrisers Hyderabad           35               0

   player_of_match                                      venue      umpire1  \
0     Yuvraj Singh  Rajiv Gandhi International Stadium, Uppal  AY Dandekar
1     Yuvraj Singh  Rajiv Gandhi International Stadium, Uppal  AY Dandekar

    umpire2 umpire3
0  NJ Llong     NaN
1  NJ Llong     NaN

[2 rows x 39 columns]
```

[84]:
```
six = temp[temp['batsman_runs'] == 6]
six.head(2)
```

[84]:
```
    match_id  inning         batting_team                bowling_team  over  \
10         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     2
47         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     8

    ball       batsman non_striker       bowler  is_super_over  …  result  \
10     4     DA Warner    S Dhawan  A Choudhary              0  …  normal
47     4  MC Henriques    S Dhawan      TM Head              0  …  normal

    dl_applied              winner  win_by_runs  win_by_wickets  \
10           0  Sunrisers Hyderabad           35               0
47           0  Sunrisers Hyderabad           35               0

    player_of_match                                      venue      umpire1  \
10     Yuvraj Singh  Rajiv Gandhi International Stadium, Uppal  AY Dandekar
47     Yuvraj Singh  Rajiv Gandhi International Stadium, Uppal  AY Dandekar

     umpire2 umpire3
10  NJ Llong     NaN
47  NJ Llong     NaN

[2 rows x 39 columns]
```

[88]:
```
num_sixes = six.groupby('venue')['venue'].count()
```

[89]:
```
num_matches = matches['venue'].value_counts()
```

[90]:
```
(num_sixes/num_matches).sort_values(ascending=False).head(3)
```

[90]:
```
venue
Holkar Cricket Stadium     17.600000
M Chinnaswamy Stadium      13.227273
Sharjah Cricket Stadium    12.666667
dtype: float64
```

2. Find orange cap holder of all the seasons

[95]:
```
temp.groupby(['season', 'batsman'])['batsman_runs'].sum().reset_index().
↪sort_values('batsman_runs', ascending=False).
↪drop_duplicates(subset=['season']).sort_values('season
                                                                                ␣
↪                                                                               ␣
↪       ')
```

[95]:

	season	batsman	batsman_runs
1383	2016	V Kohli	973
910	2013	MEK Hussey	733
684	2012	CH Gayle	733
1088	2014	RV Uthappa	660
1422	2017	DA Warner	641
446	2010	SR Tendulkar	618
115	2008	SE Marsh	616
502	2011	CH Gayle	608
229	2009	ML Hayden	572
1148	2015	DA Warner	562

2 Multiindex

[1]:
```
import numpy as np
import pandas as pd
```

2.0.1 Series -> 1D DataFrame -> 2D

[3]:
```
index_val = [('cse',2019), ('cse',2020), ('cse',2021), ('cse',2022),␣
↪('ece',2019), ('ece',2020), ('ece',2021), ('ece',2022)]
a = pd.Series([1,2,3,4,5,6,7,8], index=index_val)
a
```

[3]:
```
(cse, 2019)    1
(cse, 2020)    2
(cse, 2021)    3
(cse, 2022)    4
(ece, 2019)    5
(ece, 2020)    6
(ece, 2021)    7
(ece, 2022)    8
dtype: int64
```

2.0.2 The Problem

[5]:
```
a['cse']
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File c:\Program Files\Python312\Lib\site-packages\pandas\core\indexes\base.py:
 ↪3802, in Index.get_loc(self, key)
   3801 try:
-> 3802     return self._engine.get_loc(casted_key)
   3803 except KeyError as err:

File index.pyx:153, in pandas._libs.index.IndexEngine.get_loc()

File index.pyx:182, in pandas._libs.index.IndexEngine.get_loc()

File pandas\\_libs\\hashtable_class_helper.pxi:7081, in pandas._libs.hashtable.
 ↪PyObjectHashTable.get_item()

File pandas\\_libs\\hashtable_class_helper.pxi:7089, in pandas._libs.hashtable.
 ↪PyObjectHashTable.get_item()

KeyError: 'cse'

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Cell In[5], line 1
----> 1 a['cse']

File c:\Program Files\Python312\Lib\site-packages\pandas\core\series.py:1111, i
 ↪Series.__getitem__(self, key)
   1108     return self._values[key]
   1110 elif key_is_scalar:
-> 1111     return self._get_value(key)
   1113 # Convert generator to list before going through hashable part
   1114 # (We will iterate through the generator there to check for slices)
```

```
   1115 if is_iterator(key):

File c:\Program Files\Python312\Lib\site-packages\pandas\core\series.py:1227, i
 ↪Series._get_value(self, label, takeable)
   1224     return self._values[label]
   1226 # Similar to Index.get_value, but we do not fall back to positional
-> 1227 loc = self.index.get_loc(label)
   1229 if is_integer(loc):
   1230     return self._values[loc]

File c:\Program Files\Python312\Lib\site-packages\pandas\core\indexes\base.py:
 ↪3809, in Index.get_loc(self, key)
   3804     if isinstance(casted_key, slice) or (
   3805         isinstance(casted_key, abc.Iterable)
   3806         and any(isinstance(x, slice) for x in casted_key)
   3807     ):
   3808         raise InvalidIndexError(key)
-> 3809     raise KeyError(key) from err
   3810 except TypeError:
   3811     # If we have a listlike key, _check_indexing_error will raise
   3812     #  InvalidIndexError. Otherwise we fall through and re-raise
   3813     #  the TypeError.
   3814     self._check_indexing_error(key)

KeyError: 'cse'
```

2.0.3 The Solution - Multiindex

2.1 Creating Multiindex Series

1. pd.MultiIndex.from_tuples()

[6]:
```
index_val = [('cse',2019), ('cse',2020), ('cse',2021), ('cse',2022),
 ↪('ece',2019), ('ece',2020), ('ece',2021), ('ece',2022)]
multiindex = pd.MultiIndex.from_tuples(index_val)
multiindex
```

[6]:
```
MultiIndex([('cse', 2019),
            ('cse', 2020),
            ('cse', 2021),
            ('cse', 2022),
            ('ece', 2019),
            ('ece', 2020),
            ('ece', 2021),
            ('ece', 2022)],
           )
```

2. pd.MultiIndex.from_product()

[7]:
```
pd.MultiIndex.from_product([['cse', 'ece'], [2019, 2020, 2021, 2022]])
```

[7]:
```
MultiIndex([('cse', 2019),
            ('cse', 2020),
            ('cse', 2021),
            ('cse', 2022),
            ('ece', 2019),
            ('ece', 2020),
            ('ece', 2021),
            ('ece', 2022)],
           )
```

level inside MultiIndex object

[8]:
```
multiindex.levels
```

[8]:
```
FrozenList([['cse', 'ece'], [2019, 2020, 2021, 2022]])
```

[9]:
```
multiindex.levels[0]
```

[9]:
```
Index(['cse', 'ece'], dtype='object')
```

[10]:
```
multiindex.levels[1]
```

[10]:
```
Index([2019, 2020, 2021, 2022], dtype='int64')
```

2.1.1 Creating Series with MultiIndex object

[13]:
```
s = pd.Series([1,2,3,4,5,6,7,8], index=multiindex)
s
```

[13]:
```
cse  2019    1
     2020    2
     2021    3
     2022    4
ece  2019    5
     2020    6
     2021    7
     2022    8
dtype: int64
```

[14]:
```
# Fetching items
s['cse']
```

[14]:
```
2019    1
2020    2
2021    3
```

```
2022    4
dtype: int64
```

[15]:
```
s['cse',2020]
```

[15]:
```
2
```

2.2 MultiIndex DataFrame

[16]:
```
branch_df = pd.DataFrame(
    [
        [1,2],
        [3,4],
        [5,6],
        [7,8],
        [9,10],
        [11,12],
        [13,14],
        [15,16]
    ],
    index=multiindex,
    columns=['avg_pkg', 'students']
)

branch_df
```

[16]:

		avg_pkg	students
cse	2019	1	2
	2020	3	4
	2021	5	6
	2022	7	8
ece	2019	9	10
	2020	11	12
	2021	13	14
	2022	15	16

[17]:
```
branch_df.loc['cse']
```

[17]:

	avg_pkg	students
2019	1	2
2020	3	4
2021	5	6
2022	7	8

[23]:
```
branch_df.loc['cse', 2019]
```

[23]:
```
avg_pkg     1
students    2
Name: (cse, 2019), dtype: int64
```

[24]:
```
branch_df['avg_pkg']
```

[24]:
```
cse  2019     1
     2020     3
     2021     5
     2022     7
ece  2019     9
     2020    11
     2021    13
     2022    15
Name: avg_pkg, dtype: int64
```

Another perspective

[25]:
```
branch_df2 = pd.DataFrame(
    [
        [1,2,0,0],
        [3,4,0,0],
        [5,6,0,0],
        [7,8,0,0]
    ],
    index= [2019, 2020, 2021, 2022],
    columns= pd.MultiIndex.from_product([['delhi', 'mumbai'], ['avg_pkg','students']])
)

branch_df2
```

[25]:

	delhi		mumbai	
	avg_pkg	students	avg_pkg	students
2019	1	2	0	0
2020	3	4	0	0
2021	5	6	0	0
2022	7	8	0	0

[26]:
```
branch_df2['delhi']['avg_pkg']
```

[26]:
```
2019    1
2020    3
2021    5
2022    7
Name: avg_pkg, dtype: int64
```

[29]:
```
branch_df2.loc[2022]
```

[29]:
```
delhi   avg_pkg     7
        students    8
mumbai  avg_pkg     0
        students    0
Name: 2022, dtype: int64
```

MultiIndex DF in terms of both cols and index

[30]:
```
branch_df3 = pd.DataFrame(
    [
        [1,2,0,0],
        [3,4,0,0],
        [5,6,0,0],
        [7,8,0,0],
        [9,10,0,0],
        [11,12,0,0],
        [13,14,0,0],
        [15,16,0,0]
    ],
    index=multiindex,
    columns= pd.MultiIndex.from_product([['delhi', 'mumbai'], ['avg_pkg','␣
 ↪'students']])
)

branch_df3
```

[30]:

		delhi		mumbai	
		avg_pkg	students	avg_pkg	students
cse	2019	1	2	0	0
	2020	3	4	0	0
	2021	5	6	0	0
	2022	7	8	0	0
ece	2019	9	10	0	0
	2020	11	12	0	0
	2021	13	14	0	0
	2022	15	16	0	0

2.3 Stacking and Unstacking

- Stacking - last level of col transform into last level of index.
- Unstacking - last level of index transform into last level of col.

[34]:
```
branch_df3.stack(future_stack=True)
```

[34]:
```
                    delhi  mumbai
cse 2019 avg_pkg        1       0
         students       2       0
     2020 avg_pkg        3       0
         students       4       0
     2021 avg_pkg        5       0
         students       6       0
     2022 avg_pkg        7       0
         students       8       0
ece 2019 avg_pkg        9       0
         students      10       0
     2020 avg_pkg       11       0
         students      12       0
     2021 avg_pkg       13       0
         students      14       0
     2022 avg_pkg       15       0
         students      16       0
```

[35]:
```
branch_df3.unstack()
```

[35]:
```
      delhi                                          mumbai                \
    avg_pkg                students                  avg_pkg
       2019 2020 2021 2022     2019 2020 2021 2022    2019 2020 2021 2022
cse       1    3    5    7        2    4    6    8       0    0    0    0
ece       9   11   13   15       10   12   14   16       0    0    0    0


    students
        2019 2020 2021 2022
cse        0    0    0    0
ece        0    0    0    0
```

`Note` - MultiIndex DataFrame is also DataFrame so all the operations which can be performed on Normal DataFrame can also be performed on the MultiIndex DataFrame

2.3.1 Extracting Single Row

[39]:
```
branch_df3
```

[39]:
```
           delhi           mumbai
         avg_pkg students avg_pkg students
cse 2019       1        2       0        0
    2020       3        4       0        0
    2021       5        6       0        0
    2022       7        8       0        0
ece 2019       9       10       0        0
    2020      11       12       0        0
```

```
          2021      13        14        0         0
          2022      15        16        0         0
```

[38]:
```
branch_df3.loc[('cse',2020)]
```

[38]:
```
delhi   avg_pkg     3
        students    4
mumbai  avg_pkg     0
        students    0
Name: (cse, 2020), dtype: int64
```

[51]:
```
branch_df3.iloc[1]
```

[51]:
```
delhi   avg_pkg     3
        students    4
mumbai  avg_pkg     0
        students    0
Name: (cse, 2020), dtype: int64
```

2.3.2 Extracting Multiple Rows

[43]:
```
branch_df3.loc[('cse',2019):('ece',2020):2]
```

[43]:
```
            delhi           mumbai
          avg_pkg students avg_pkg students
cse 2019        1        2       0        0
    2021        5        6       0        0
ece 2019        9       10       0        0
```

[52]:
```
branch_df3.iloc[0:5:2]
```

[52]:
```
            delhi           mumbai
          avg_pkg students avg_pkg students
cse 2019        1        2       0        0
    2021        5        6       0        0
ece 2019        9       10       0        0
```

2.3.3 Extracting Columns

[56]:
```
branch_df3['delhi']['avg_pkg']
```

[56]:
```
cse  2019     1
     2020     3
     2021     5
     2022     7
ece  2019     9
```

```
     2020    11
     2021    13
     2022    15
Name: avg_pkg, dtype: int64
```

[57]:
```
branch_df3.iloc[:, 0]
```

[57]:
```
cse  2019     1
     2020     3
     2021     5
     2022     7
ece  2019     9
     2020    11
     2021    13
     2022    15
Name: (delhi, avg_pkg), dtype: int64
```

2.3.4 Extracting Rows and Cols

[58]:
```
branch_df3.iloc[[0,4], [1,2]]
```

[58]:

	delhi	mumbai
	students	avg_pkg
cse 2019	2	0
ece 2019	10	0

[59]:
```
branch_df3.iloc[[2,6], [0,2]]
```

[59]:

	delhi	mumbai
	avg_pkg	avg_pkg
cse 2021	5	0
ece 2021	13	0

2.3.5 Sort_index(ascending=True)

[60]:
```
branch_df3.sort_index(ascending=False)
```

[60]:

	delhi		mumbai	
	avg_pkg	students	avg_pkg	students
ece 2022	15	16	0	0
2021	13	14	0	0
2020	11	12	0	0
2019	9	10	0	0
cse 2022	7	8	0	0
2021	5	6	0	0
2020	3	4	0	0

```
          2019       1        2       0        0
```

[61]:
```
branch_df3.sort_index(ascending=[True, False])
```

[61]:
```
           delhi           mumbai
         avg_pkg students avg_pkg students
cse 2022       7        8       0        0
    2021       5        6       0        0
    2020       3        4       0        0
    2019       1        2       0        0
ece 2022      15       16       0        0
    2021      13       14       0        0
    2020      11       12       0        0
    2019       9       10       0        0
```

[62]:
```
branch_df3.sort_index(level=0, ascending=False)
```

[62]:
```
           delhi           mumbai
         avg_pkg students avg_pkg students
ece 2022      15       16       0        0
    2021      13       14       0        0
    2020      11       12       0        0
    2019       9       10       0        0
cse 2022       7        8       0        0
    2021       5        6       0        0
    2020       3        4       0        0
    2019       1        2       0        0
```

DF.transpose()

[63]:
```
branch_df3.transpose()
```

[63]:
```
                cse                   ece
               2019 2020 2021 2022 2019 2020 2021 2022
delhi  avg_pkg    1    3    5    7    9   11   13   15
       students   2    4    6    8   10   12   14   16
mumbai avg_pkg    0    0    0    0    0    0    0    0
       students   0    0    0    0    0    0    0    0
```

DF.swaplevel(axis=0)

[70]:
```
branch_df3.swaplevel(axis=1)
```

[70]:
```
         avg_pkg students avg_pkg students
           delhi    delhi  mumbai   mumbai
cse 2019       1        2       0        0
    2020       3        4       0        0
    2021       5        6       0        0
```

```
     2022       7        8       0       0
ece  2019       9       10       0       0
     2020      11       12       0       0
     2021      13       14       0       0
     2022      15       16       0       0
```

2.3.6 Long Vs Wide Data

Wide format is where we have a single row for every data point with multiple columns to hold the values of various attributes.

Long format is where, for each data point we have as many rows as the number of attributes and each row contains the value of a particular attribute for a given data point.

2.4 DF.melt()

DF.melt(id_vars='cols_to_keep_as_it_is', var_name='name', value_name='name')

- Wide to Long

[72]:
```
pd.DataFrame({'cse':[120]}).melt()
```

[72]:
```
  variable  value
0      cse    120
```

[75]:
```
temp = pd.DataFrame({'cse':[120], 'ece':[100], 'mech':[50]})
temp
```

[75]:
```
   cse  ece  mech
0  120  100    50
```

[76]:
```
temp.melt()
```

[76]:
```
  variable  value
0      cse    120
1      ece    100
2     mech     50
```

[77]:
```
temp.melt(var_name='branch', value_name='num_students')
```

[77]:
```
  branch  num_students
0    cse           120
1    ece           100
2   mech            50
```

[80]:
```
temp = pd.DataFrame(
    {
        'branch':['cse','ece','mech'],
        '2020':[100,150,60],
        '2021':[120,130,80],
        '2022':[150,140,70]
    }
)

temp
```

[80]:

	branch	2020	2021	2022
0	cse	100	120	150
1	ece	150	130	140
2	mech	60	80	70

[81]:
```
temp.melt(id_vars=['branch'],var_name='year',value_name='students')
```

[81]:

	branch	year	students
0	cse	2020	100
1	ece	2020	150
2	mech	2020	60
3	cse	2021	120
4	ece	2021	130
5	mech	2021	80
6	cse	2022	150
7	ece	2022	140
8	mech	2022	70

[96]:
```
death = pd.read_csv('time_series_covid19_deaths_global.csv')
confirm = pd.read_csv('time_series_covid19_confirmed_global.csv')

death.head(2)
```

[96]:

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	\
0	NaN	Afghanistan	33.93911	67.709953	0	0	
1	NaN	Albania	41.15330	20.168300	0	0	

	1/24/20	1/25/20	1/26/20	1/27/20	…	12/24/22	12/25/22	12/26/22	\
0	0	0	0	0	…	7845	7846	7846	
1	0	0	0	0	…	3595	3595	3595	

	12/27/22	12/28/22	12/29/22	12/30/22	12/31/22	1/1/23	1/2/23
0	7846	7846	7847	7847	7849	7849	7849
1	3595	3595	3595	3595	3595	3595	3595

[2 rows x 1081 columns]

[97]:
```
death = death.melt(id_vars=['Province/State','Country/Region', 'Lat', 'Long'],␣
↪var_name='date', value_name='num_deaths')
print(death.shape)
death.head(2)
```

```
(311253, 6)
```

[97]:

	Province/State	Country/Region	Lat	Long	date	num_deaths
0	NaN	Afghanistan	33.93911	67.709953	1/22/20	0
1	NaN	Albania	41.15330	20.168300	1/22/20	0

[98]:
```
confirm = confirm.melt(id_vars=['Province/State','Country/Region', 'Lat',␣
↪'Long'], var_name='date', value_name='num_confirmed')
print(confirm.shape)
confirm.head(2)
```

```
(311253, 6)
```

[98]:

	Province/State	Country/Region	Lat	Long	date	num_confirmed
0	NaN	Afghanistan	33.93911	67.709953	1/22/20	0
1	NaN	Albania	41.15330	20.168300	1/22/20	0

[100]:
```
confirm.merge(death, on=['Province/State','Country/Region', 'Lat', 'Long',␣
↪'date'])[['Country/Region', 'date', 'num_confirmed', 'num_deaths']]
```

[100]:

	Country/Region	date	num_confirmed	num_deaths
0	Afghanistan	1/22/20	0	0
1	Albania	1/22/20	0	0
2	Algeria	1/22/20	0	0
3	Andorra	1/22/20	0	0
4	Angola	1/22/20	0	0
…	…	…	…	…
311248	West Bank and Gaza	1/2/23	703228	5708
311249	Winter Olympics 2022	1/2/23	535	0
311250	Yemen	1/2/23	11945	2159
311251	Zambia	1/2/23	334661	4024
311252	Zimbabwe	1/2/23	259981	5637

[311253 rows x 4 columns]

2.5 Pivot Table

The pivot table takes simple column-wise data as input, and groups the entries into a two-dimensional table that provides a multidimensional summarization of the data.

[2]:
```
import numpy as np
import pandas as pd
import seaborn as sns
```

[3]:
```
df = sns.load_dataset('tips')
df
```

[3]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4
..	…	…	…	…	…	…	…
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

[244 rows x 7 columns]

[4]:
```
df.groupby(['sex', 'smoker'], observed=False)[['total_bill']].mean().unstack()
```

[4]:

	total_bill	
smoker	Yes	No
sex		
Male	22.284500	19.791237
Female	17.977879	18.105185

[5]:
```
df.pivot_table(index='sex', columns='smoker', values='total_bill',␣
↪observed=False)
```

[5]:

smoker	Yes	No
sex		
Male	22.284500	19.791237
Female	17.977879	18.105185

Multidimensional Pivot Tables

[8]:
```
df.pivot_table(index=['sex', 'smoker'], columns=['day', 'time'],␣
↪aggfunc={'size':'mean', 'tip':'max', 'total_bill':'sum'}, observed=False)
```

[8]:

		size				tip			\
day		Thur		Fri		Sat	Sun	Thur	
time		Lunch	Dinner	Lunch	Dinner	Dinner	Dinner	Lunch	
sex	smoker								
Male	Yes	2.300000	NaN	1.666667	2.4	2.629630	2.600000	5.00	
	No	2.500000	NaN	NaN	2.0	2.656250	2.883721	6.70	
Female	Yes	2.428571	NaN	2.000000	2.0	2.200000	2.500000	5.00	
	No	2.500000	2.0	3.000000	2.0	2.307692	3.071429	5.17	

```
                                                  total_bill                \
day                      Fri           Sat    Sun       Thur           Fri
time           Dinner Lunch Dinner Dinner Dinner      Lunch Dinner  Lunch
sex    smoker
Male   Yes        NaN  2.20   4.73  10.00    6.5     191.71   0.00  34.16
       No         NaN   NaN   3.50   9.00    6.0     369.73   0.00   0.00
Female Yes        NaN  3.48   4.30   6.50    4.0     134.53   0.00  39.78
       No         3.0  3.00   3.25   4.67    5.2     381.58  18.78  15.98


day                     Sat           Sun
time           Dinner Lunch  Dinner Lunch  Dinner
sex    smoker
Male   Yes     129.46   0.0  589.62   0.0  392.12
       No       34.95   0.0  637.73   0.0  877.34
Female Yes      48.80   0.0  304.00   0.0   66.16
       No       22.75   0.0  247.05   0.0  291.54
```

Margins

- Additional Rows and Cols containing sum of rows and cols

[10]:
```
df.pivot_table(index='sex', columns='smoker', values='total_bill',␣
↪aggfunc='sum', observed=False, margins=True)
```

[10]:
```
smoker      Yes       No      All
sex
Male    1337.07  1919.75  3256.82
Female   593.27   977.68  1570.95
All     1930.34  2897.43  4827.77
```

Plotting Graphs

[23]:
```
df = pd.read_csv('expense_data.csv')
df
```

[23]:
```
                 Date               Account        Category  Subcategory  \
0      3/2/2022 10:11  CUB - online payment            Food          NaN
1      3/2/2022 10:11  CUB - online payment           Other          NaN
2      3/1/2022 19:50  CUB - online payment            Food          NaN
3      3/1/2022 18:56  CUB - online payment  Transportation          NaN
4      3/1/2022 18:22  CUB - online payment            Food          NaN
..               …                  …              …           …
272  11/22/2021 14:16  CUB - online payment            Food          NaN
273  11/22/2021 14:16  CUB - online payment            Food          NaN
274  11/21/2021 17:07  CUB - online payment  Transportation          NaN
275  11/21/2021 15:50  CUB - online payment            Food          NaN
276  11/21/2021 13:30  CUB - online payment           Other          NaN
```

```
                   Note     INR Income/Expense  Note.1  Amount Currency  \
0               Brownie    50.0        Expense     NaN    50.0      INR
1     To lended people   300.0        Expense     NaN   300.0      INR
2                Dinner    78.0        Expense     NaN    78.0      INR
3                 Metro    30.0        Expense     NaN    30.0      INR
4                Snacks    67.0        Expense     NaN    67.0      INR
..                  …     …            …     …      …      …
272              Dinner    90.0        Expense     NaN    90.0      INR
273  Lunch with company    97.0        Expense     NaN    97.0      INR
274              Rapido   130.0        Expense     NaN   130.0      INR
275               Lunch   875.0        Expense     NaN   875.0      INR
276      Got from gobi  2000.0         Income     NaN  2000.0      INR

     Account.1
0         50.0
1        300.0
2         78.0
3         30.0
4         67.0
..         …
272       90.0
273       97.0
274      130.0
275      875.0
276     2000.0

[277 rows x 11 columns]
```

[24]:
```
df['Category'].value_counts()
```

[24]:
```
Category
Food                156
Other                60
Transportation       31
Apparel               7
Household             6
Allowance             6
Social Life           5
Education             1
Salary                1
Self-development      1
Beauty                1
Gift                  1
Petty cash            1
Name: count, dtype: int64
```

[25]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 277 entries, 0 to 276
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Date            277 non-null    object
 1   Account         277 non-null    object
 2   Category        277 non-null    object
 3   Subcategory     0 non-null      float64
 4   Note            273 non-null    object
 5   INR             277 non-null    float64
 6   Income/Expense  277 non-null    object
 7   Note.1          0 non-null      float64
 8   Amount          277 non-null    float64
 9   Currency        277 non-null    object
 10  Account.1       277 non-null    float64
dtypes: float64(5), object(6)
memory usage: 23.9+ KB
```

[26]:
```
df['Date'] = pd.to_datetime(df['Date'])

df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 277 entries, 0 to 276
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Date            277 non-null    datetime64[ns]
 1   Account         277 non-null    object
 2   Category        277 non-null    object
 3   Subcategory     0 non-null      float64
 4   Note            273 non-null    object
 5   INR             277 non-null    float64
 6   Income/Expense  277 non-null    object
 7   Note.1          0 non-null      float64
 8   Amount          277 non-null    float64
 9   Currency        277 non-null    object
 10  Account.1       277 non-null    float64
dtypes: datetime64[ns](1), float64(5), object(5)
memory usage: 23.9+ KB
```

[27]:
```
df['month'] = df['Date'].dt.month_name()
```

[28]:
```
df.head(2)
```

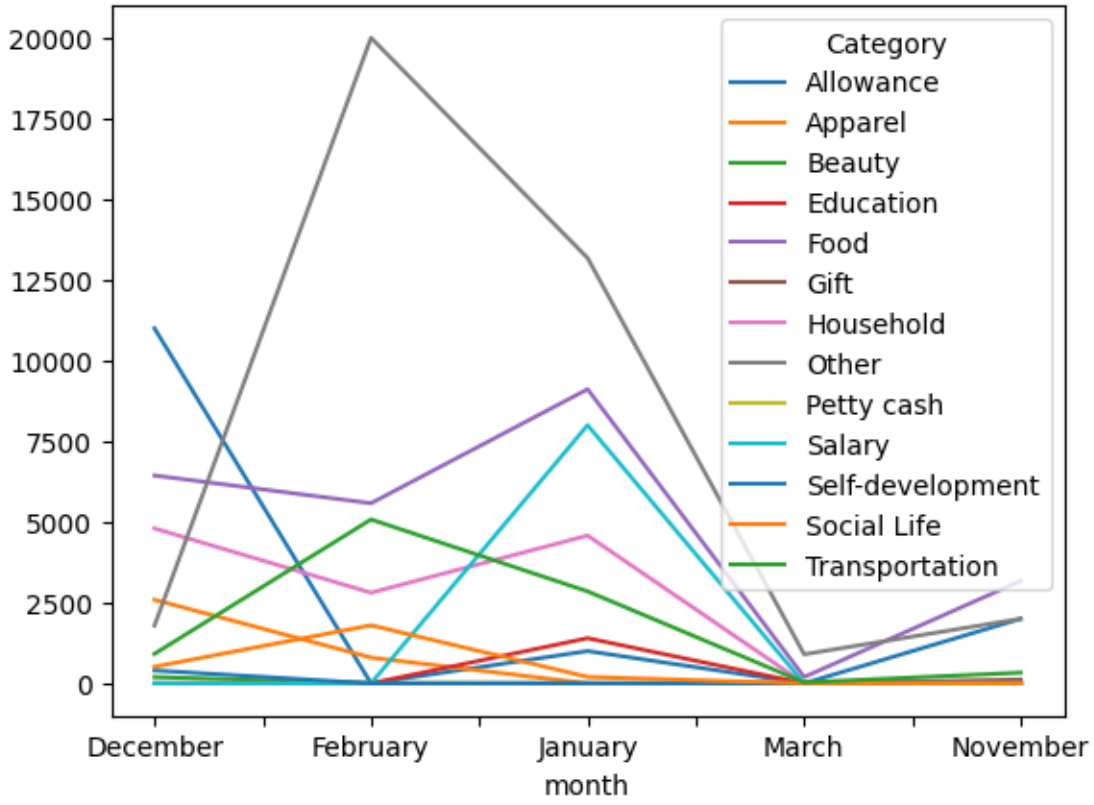
[28]:
```
                  Date               Account Category  Subcategory  \
0 2022-03-02 10:11:00  CUB - online payment     Food          NaN
1 2022-03-02 10:11:00  CUB - online payment    Other          NaN

                Note    INR Income/Expense  Note.1  Amount Currency  Account.1  \
0           Brownie   50.0        Expense     NaN    50.0      INR       50.0
1  To lended people  300.0        Expense     NaN   300.0      INR      300.0

   month
0  March
1  March
```

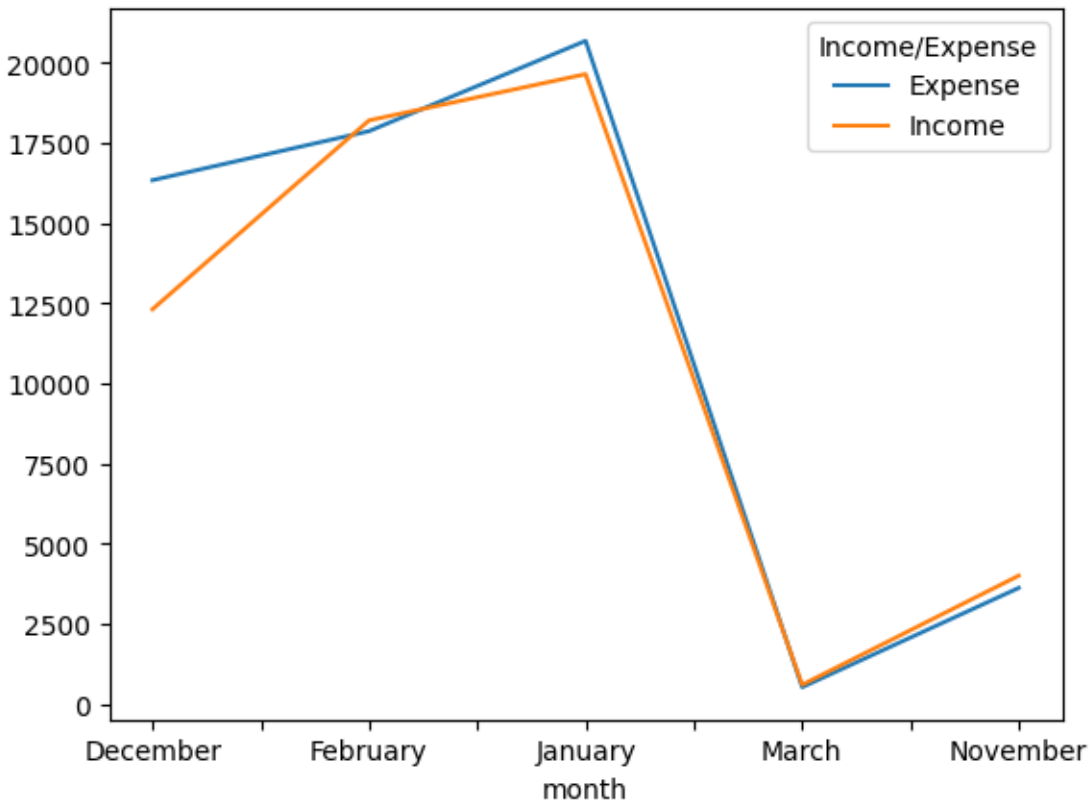
[29]:
```
df.pivot_table(index='month', columns='Category', values='INR', aggfunc='sum',␣
↪fill_value=0).plot()
```

[29]:
```
<Axes: xlabel='month'>
```



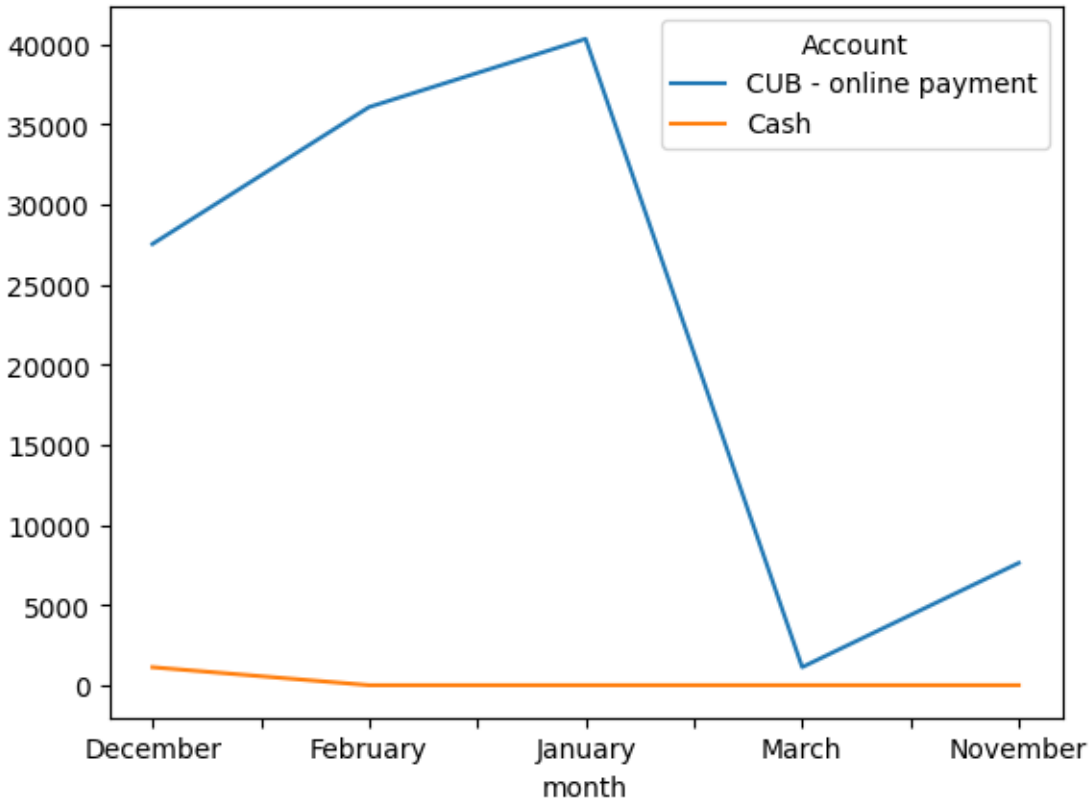
[30]:
```
df.pivot_table(index='month', columns='Income/Expense', values='INR',␣
↪aggfunc='sum', fill_value=0).plot()
```

[30]:
```
<Axes: xlabel='month'>
```



[32]:
```
df.pivot_table(index='month', columns='Account', values='INR', aggfunc='sum',␣
↪fill_value=0).plot()
```

[32]: <Axes: xlabel='month'>



[1]:
```
import numpy as np
import pandas as pd
```

2.5.1 Vectorized Operations

[3]:
```
a = np.array([1,2,3,4])
a * 4
```

[3]:
```
array([ 4,  8, 12, 16])
```

The problem with vectorized operations.

[4]:
```
s = ['cat', 'mat', None, 'rat']
[i.startswith('c') for i in s]
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[4], line 2
      1 s = ['cat', 'mat', None, 'rat']
----> 2 [i.startswith('c') for i in s]
```

```
AttributeError: 'NoneType' object has no attribute 'startswith'
```

How Pandas solves this issue?

- .str is string accessor.
- it is fast and optimized in compare to traditional python code

[5]:
```
s = pd.Series(['cat', 'mat', None, 'rat'])

s.str.startswith('c')
```

[5]:
```
0     True
1    False
2     None
3    False
dtype: object
```

[6]:
```
df = pd.read_csv('titanic.csv')
df.head(2)
```

[6]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1

                                                 Name     Sex   Age  SibSp  \
0                             Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1

   Parch     Ticket     Fare Cabin Embarked
0      0  A/5 21171   7.2500   NaN        S
1      0   PC 17599  71.2833   C85        C
```

[7]:
```
df['Name']
```

[7]:
```
0                                Braund, Mr. Owen Harris
1      Cumings, Mrs. John Bradley (Florence Briggs Th…
2                                 Heikkinen, Miss. Laina
3           Futrelle, Mrs. Jacques Heath (Lily May Peel)
4                               Allen, Mr. William Henry
                              …
886                               Montvila, Rev. Juozas
887                        Graham, Miss. Margaret Edith
888            Johnston, Miss. Catherine Helen "Carrie"
889                               Behr, Mr. Karl Howell
890                                 Dooley, Mr. Patrick
Name: Name, Length: 891, dtype: object
```

2.6 Common Functions

2.6.1 DF['col'].str.upper()

- Change to uppercase

[8]:
```
df['Name'].str.upper()
```

[8]:
```
0                                 BRAUND, MR. OWEN HARRIS
1       CUMINGS, MRS. JOHN BRADLEY (FLORENCE BRIGGS TH…
2                                  HEIKKINEN, MISS. LAINA
3            FUTRELLE, MRS. JACQUES HEATH (LILY MAY PEEL)
4                                ALLEN, MR. WILLIAM HENRY
                              …
886                                MONTVILA, REV. JUOZAS
887                         GRAHAM, MISS. MARGARET EDITH
888              JOHNSTON, MISS. CATHERINE HELEN "CARRIE"
889                                BEHR, MR. KARL HOWELL
890                                  DOOLEY, MR. PATRICK
Name: Name, Length: 891, dtype: object
```

2.6.2 DF['col'].str.lower()

- change to lowercase

[9]:
```
df['Name'].str.lower()
```

[9]:
```
0                                 braund, mr. owen harris
1       cumings, mrs. john bradley (florence briggs th…
2                                  heikkinen, miss. laina
3            futrelle, mrs. jacques heath (lily may peel)
4                                allen, mr. william henry
                              …
886                                montvila, rev. juozas
887                         graham, miss. margaret edith
888              johnston, miss. catherine helen "carrie"
889                                behr, mr. karl howell
890                                  dooley, mr. patrick
Name: Name, Length: 891, dtype: object
```

2.6.3 DF['col'].str.capitalize()

- change first letter of the first word into uppercase

[10]:
```
df['Name'].str.capitalize()
```

[10]:
```
0                                 Braund, mr. owen harris
1       Cumings, mrs. john bradley (florence briggs th…
2                                  Heikkinen, miss. laina
```

```
3           Futrelle, mrs. jacques heath (lily may peel)
4                               Allen, mr. william henry
                               …
886                                 Montvila, rev. juozas
887                          Graham, miss. margaret edith
888              Johnston, miss. catherine helen "carrie"
889                                 Behr, mr. karl howell
890                                   Dooley, mr. patrick
Name: Name, Length: 891, dtype: object
```

2.6.4 DF['col'].str.title()

- Change first letter of every word into uppercase

[11]:
```
df['Name'].str.title()
```

[11]:
```
0                                Braund, Mr. Owen Harris
1      Cumings, Mrs. John Bradley (Florence Briggs Th…
2                                 Heikkinen, Miss. Laina
3           Futrelle, Mrs. Jacques Heath (Lily May Peel)
4                               Allen, Mr. William Henry
                               …
886                                 Montvila, Rev. Juozas
887                          Graham, Miss. Margaret Edith
888              Johnston, Miss. Catherine Helen "Carrie"
889                                 Behr, Mr. Karl Howell
890                                   Dooley, Mr. Patrick
Name: Name, Length: 891, dtype: object
```

2.6.5 DF['col'].str.len()

- Returns length of the string

[12]:
```
df['Name'].str.len()
```

[12]:
```
0      23
1      51
2      22
3      44
4      24
       ..
886    21
887    28
888    40
889    21
890    19
Name: Name, Length: 891, dtype: int64
```

2.6.6 str.strip('char')

- Remove unwanted trailing characters

[20]:
```
"---,,--,-Khush-   ---,,,--".strip('-, ')
```

[20]:
```
'Khush'
```

[21]:
```
df['Name'].str.strip()
```

[21]:
```
0                                Braund, Mr. Owen Harris
1      Cumings, Mrs. John Bradley (Florence Briggs Th…
2                                 Heikkinen, Miss. Laina
3           Futrelle, Mrs. Jacques Heath (Lily May Peel)
4                               Allen, Mr. William Henry
                              …
886                                Montvila, Rev. Juozas
887                         Graham, Miss. Margaret Edith
888             Johnston, Miss. Catherine Helen "Carrie"
889                                Behr, Mr. Karl Howell
890                                  Dooley, Mr. Patrick
Name: Name, Length: 891, dtype: object
```

2.6.7 DF['col'].str.split('char')

- Splits the string based on the char

[27]:
```
df['lastname'] = df['Name'].str.split(',').str.get(0)
df.head()
```

[27]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked   lastname
0      0         A/5 21171   7.2500   NaN        S     Braund
1      0          PC 17599  71.2833   C85        C    Cumings
2      0  STON/O2. 3101282   7.9250   NaN        S  Heikkinen
```

```
3      0            113803  53.1000  C123        S   Futrelle
4      0            373450   8.0500   NaN        S      Allen
```

[34]:
```
df[['title', 'firstname']] = df['Name'].str.split(',').str.get(1).str.strip().
  ↪str.split(' ', n=1, expand=True)
df['title'].value_counts()
```

[34]:
```
title
Mr.          517
Miss.        182
Mrs.         125
Master.       40
Dr.            7
Rev.           6
Mlle.          2
Major.         2
Col.           2
the            1
Capt.          1
Ms.            1
Sir.           1
Lady.          1
Mme.           1
Don.           1
Jonkheer.      1
Name: count, dtype: int64
```

2.6.8 DF['col'].str.replace('existing_str', 'new_str')

- Replace the string

[36]:
```
df['title'] = df['title'].str.replace('Ms.', 'Miss.')
df['title'] = df['title'].str.replace('Mlle.', 'Miss.')
```

[37]:
```
df['title'].value_counts()
```

[37]:
```
title
Mr.          517
Miss.        185
Mrs.         125
Master.       40
Dr.            7
Rev.           6
Major.         2
Col.           2
Don.           1
Mme.           1
```

```
Lady.          1
Sir.           1
Capt.          1
the            1
Jonkheer.      1
Name: count, dtype: int64
```

2.7 Filtering

2.7.1 DF['col'].str.startswith('char')

- Check each string whether it starts with specific char or not

[39]:
```
df[df['firstname'].str.startswith('A')]
```

[39]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	lastname	title	firstname
13	14	0	3	Andersson, Mr. Anders Johan	male	39.0	1	5	347082	31.2750	NaN	S	Andersson	Mr.	Anders Johan
22	23	1	3	McGowan, Miss. Anna "Annie"	female	15.0	0	0	330923	8.0292	NaN	Q	McGowan	Miss.	Anna "Annie"
35	36	0	1	Holverson, Mr. Alexander Oskar	male	42.0	1	0	113789	52.0000	NaN	S	Holverson	Mr.	Alexander Oskar
38	39	0	3	Vander Planke, Miss. Augusta Maria	female	18.0	2	0	345764	18.0000	NaN	S	Vander Planke	Miss.	Augusta Maria
61	62	1	1	Icard, Miss. Amelie	female	38.0	0	0	113572	80.0000	B28	NaN	Icard	Miss.	Amelie
..	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
842	843	1	1	Serepeca, Miss. Augusta	female	30.0	0	0	113798	31.0000	NaN	C	Serepeca	Miss.	Augusta
845	846	0	3	Abbing, Mr. Anthony	male	42.0	0	0	C.A. 5547	7.5500	NaN	S	Abbing	Mr.	Anthony
866	867	1	2	Duran y More, Miss. Asuncion	female	27.0	1	0	SC/PARIS 2149	13.8583	NaN	C	Duran y More	Miss.	Asuncion
875	876	1	3	Najib, Miss. Adele Kiamie "Jane"	female	15.0	0	0	2667	7.2250	NaN	C	Najib	Miss.	Adele Kiamie "Jane"
876	877	0	3	Gustafsson, Mr. Alfred Ossian	male	20.0	0	0	7534	9.8458	NaN	S	Gustafsson	Mr.	Alfred Ossian

```
842       Serepeca  Miss.              Augusta
845         Abbing    Mr.              Anthony
866   Duran y More  Miss.             Asuncion
875          Najib  Miss.  Adele Kiamie "Jane"
876     Gustafsson    Mr.        Alfred Ossian

[95 rows x 15 columns]
```

2.7.2 DF['col'].str.endswith('char')

[41]:
```
df[df['firstname'].str.endswith('D')]
```

[41]:
```
     PassengerId  Survived  Pclass                      Name   Sex  Age  \
168          169         0       1       Baumann, Mr. John D  male  NaN
629          630         0       3  O'Connell, Mr. Patrick D  male  NaN

     SibSp  Parch    Ticket     Fare Cabin Embarked   lastname title  \
168      0      0  PC 17318  25.9250   NaN        S    Baumann   Mr.
629      0      0    334912   7.7333   NaN        Q  O'Connell   Mr.

     firstname
168     John D
629  Patrick D
```

2.7.3 DF['col'].str.isdigit()

- True if str is consisting of only digits

[43]:
```
df[df['firstname'].str.isdigit()]
```

[43]:
```
Empty DataFrame
Columns: [PassengerId, Survived, Pclass, Name, Sex, Age, SibSp, Parch, Ticket,
Fare, Cabin, Embarked, lastname, title, firstname]
Index: []
```

2.7.4 Slicing

[51]:
```
df['Name'].str[::-1]
```

[51]:
```
0                               sirraH newO .rM ,dnuarB
1      )reyahT sggirB ecnerolF( yeldarB nhoJ .srM ,sg…
2                                 aniaL .ssiM ,nenikkieH
3           )leeP yaM yliL( htaeH seuqcaJ .srM ,ellertuF
4                               yrneH mailliW .rM ,nellA
                              …
886                                sazouJ .veR ,alivtnoM
887                          htidE teragraM .ssiM ,maharG
```

```
888             "eirraC" neleH enirehtaC .ssiM ,notsnhoJ
889                                 llewoH lraK .rM ,rheB
890                                   kcirtaP .rM ,yelooD
Name: Name, Length: 891, dtype: object
```

2.8 Applying Regular Expressions

Any firstname contains John in both cases

[47]:
```
df[df['firstname'].str.contains('john', case=False)]
```

[47]:

	PassengerId	Survived	Pclass	\
1	2	1	1	
41	42	0	2	
45	46	0	3	
98	99	1	2	
112	113	0	3	
117	118	0	2	
160	161	0	3	
162	163	0	3	
165	166	1	3	
168	169	0	1	
188	189	0	3	
212	213	0	3	
226	227	1	2	
227	228	0	3	
324	325	0	3	
328	329	1	3	
401	402	0	3	
418	419	0	2	
467	468	0	1	
527	528	0	1	
548	549	0	3	
549	550	1	2	
550	551	1	1	
563	564	0	3	
572	573	1	1	
574	575	0	3	
581	582	1	1	
583	584	0	1	
586	587	0	2	
594	595	0	2	
613	614	0	3	
624	625	0	3	
657	658	0	3	
694	695	0	1	
698	699	0	1	

700	701	1	1
733	734	0	2
760	761	0	3
765	766	1	1
818	819	0	3
822	823	0	1
825	826	0	3
848	849	0	2
864	865	0	2

	Name	Sex	Age	SibSp	\
1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	
41	Turpin, Mrs. William John Robert (Dorothy Ann …	female	27.0	1	
45	Rogers, Mr. William John	male	NaN	0	
98	Doling, Mrs. John T (Ada Julia Bone)	female	34.0	0	
112	Barton, Mr. David John	male	22.0	0	
117	Turpin, Mr. William John Robert	male	29.0	1	
160	Cribb, Mr. John Hatfield	male	44.0	0	
162	Bengtsson, Mr. John Viktor	male	26.0	0	
165	Goldsmith, Master. Frank John William "Frankie"	male	9.0	0	
168	Baumann, Mr. John D	male	NaN	0	
188	Bourke, Mr. John	male	40.0	1	
212	Perkin, Mr. John Henry	male	22.0	0	
226	Mellors, Mr. William John	male	19.0	0	
227	Lovell, Mr. John Hall ("Henry")	male	20.5	0	
324	Sage, Mr. George John Jr	male	NaN	8	
328	Goldsmith, Mrs. Frank John (Emily Alice Brown)	female	31.0	1	
401	Adams, Mr. John	male	26.0	0	
418	Matthews, Mr. William John	male	30.0	0	
467	Smart, Mr. John Montgomery	male	56.0	0	
527	Farthing, Mr. John	male	NaN	0	
548	Goldsmith, Mr. Frank John	male	33.0	1	
549	Davies, Master. John Morgan Jr	male	8.0	1	
550	Thayer, Mr. John Borland Jr	male	17.0	0	
563	Simmons, Mr. John	male	NaN	0	
572	Flynn, Mr. John Irwin ("Irving")	male	36.0	0	
574	Rush, Mr. Alfred George John	male	16.0	0	
581	Thayer, Mrs. John Borland (Marian Longstreth M…	female	39.0	1	
583	Ross, Mr. John Hugo	male	36.0	0	
586	Jarvis, Mr. John Denzil	male	47.0	0	
594	Chapman, Mr. John Henry	male	37.0	1	
613	Horgan, Mr. John	male	NaN	0	
624	Bowen, Mr. David John "Dai"	male	21.0	0	
657	Bourke, Mrs. John (Catherine)	female	32.0	1	
694	Weir, Col. John	male	60.0	0	
698	Thayer, Mr. John Borland	male	49.0	1	
700	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	female	18.0	1	

733	Berriman, Mr. William John	male	23.0	0			
760	Garfirth, Mr. John	male	NaN	0			
765	Hogeboom, Mrs. John C (Anna Andrews)	female	51.0	1			
818	Holm, Mr. John Fredrik Alexander	male	43.0	0			
822	Reuchlin, Jonkheer. John George	male	38.0	0			
825	Flynn, Mr. John	male	NaN	0			
848	Harper, Rev. John	male	28.0	0			
864	Gill, Mr. John William	male	24.0	0			

	Parch	Ticket	Fare	Cabin	Embarked	lastname	title	\
1	0	PC 17599	71.2833	C85	C	Cumings	Mrs.	
41	0	11668	21.0000	NaN	S	Turpin	Mrs.	
45	0	S.C./A.4. 23567	8.0500	NaN	S	Rogers	Mr.	
98	1	231919	23.0000	NaN	S	Doling	Mrs.	
112	0	324669	8.0500	NaN	S	Barton	Mr.	
117	0	11668	21.0000	NaN	S	Turpin	Mr.	
160	1	371362	16.1000	NaN	S	Cribb	Mr.	
162	0	347068	7.7750	NaN	S	Bengtsson	Mr.	
165	2	363291	20.5250	NaN	S	Goldsmith	Master.	
168	0	PC 17318	25.9250	NaN	S	Baumann	Mr.	
188	1	364849	15.5000	NaN	Q	Bourke	Mr.	
212	0	A/5 21174	7.2500	NaN	S	Perkin	Mr.	
226	0	SW/PP 751	10.5000	NaN	S	Mellors	Mr.	
227	0	A/5 21173	7.2500	NaN	S	Lovell	Mr.	
324	2	CA. 2343	69.5500	NaN	S	Sage	Mr.	
328	1	363291	20.5250	NaN	S	Goldsmith	Mrs.	
401	0	341826	8.0500	NaN	S	Adams	Mr.	
418	0	28228	13.0000	NaN	S	Matthews	Mr.	
467	0	113792	26.5500	NaN	S	Smart	Mr.	
527	0	PC 17483	221.7792	C95	S	Farthing	Mr.	
548	1	363291	20.5250	NaN	S	Goldsmith	Mr.	
549	1	C.A. 33112	36.7500	NaN	S	Davies	Master.	
550	2	17421	110.8833	C70	C	Thayer	Mr.	
563	0	SOTON/OQ 392082	8.0500	NaN	S	Simmons	Mr.	
572	0	PC 17474	26.3875	E25	S	Flynn	Mr.	
574	0	A/4. 20589	8.0500	NaN	S	Rush	Mr.	
581	1	17421	110.8833	C68	C	Thayer	Mrs.	
583	0	13049	40.1250	A10	C	Ross	Mr.	
586	0	237565	15.0000	NaN	S	Jarvis	Mr.	
594	0	SC/AH 29037	26.0000	NaN	S	Chapman	Mr.	
613	0	370377	7.7500	NaN	Q	Horgan	Mr.	
624	0	54636	16.1000	NaN	S	Bowen	Mr.	
657	1	364849	15.5000	NaN	Q	Bourke	Mrs.	
694	0	113800	26.5500	NaN	S	Weir	Col.	
698	1	17421	110.8833	C68	C	Thayer	Mr.	
700	0	PC 17757	227.5250	C62 C64	C	Astor	Mrs.	
733	0	28425	13.0000	NaN	S	Berriman	Mr.	

760	0	358585	14.5000	NaN	S	Garfirth	Mr.
765	0	13502	77.9583	D11	S	Hogeboom	Mrs.
818	0	C 7075	6.4500	NaN	S	Holm	Mr.
822	0	19972	0.0000	NaN	S	Reuchlin	Jonkheer.
825	0	368323	6.9500	NaN	Q	Flynn	Mr.
848	1	248727	33.0000	NaN	S	Harper	Rev.
864	0	233866	13.0000	NaN	S	Gill	Mr.

	firstname
1	John Bradley (Florence Briggs Thayer)
41	William John Robert (Dorothy Ann Wonnacott)
45	William John
98	John T (Ada Julia Bone)
112	David John
117	William John Robert
160	John Hatfield
162	John Viktor
165	Frank John William "Frankie"
168	John D
188	John
212	John Henry
226	William John
227	John Hall ("Henry")
324	George John Jr
328	Frank John (Emily Alice Brown)
401	John
418	William John
467	John Montgomery
527	John
548	Frank John
549	John Morgan Jr
550	John Borland Jr
563	John
572	John Irwin ("Irving")
574	Alfred George John
581	John Borland (Marian Longstreth Morris)
583	John Hugo
586	John Denzil
594	John Henry
613	John
624	David John "Dai"
657	John (Catherine)
694	John
698	John Borland
700	John Jacob (Madeleine Talmadge Force)
733	William John
760	John

```
765                        John C (Anna Andrews)
818                     John Fredrik Alexander
822                               John George
825                                      John
848                                      John
864                             John William
```

Find lastnames with start and end char is vowel

[50]:
```
df[df['lastname'].str.contains('^[aeiouAEIOU].+[aeiouAEIOU]$')]
```

[50]:
```
     PassengerId  Survived  Pclass  \
30            31         0       1
49            50         0       3
207          208         1       3
210          211         0       3
353          354         0       3
493          494         0       1
518          519         1       2
784          785         0       3
840          841         0       3

                                                  Name     Sex   Age  SibSp  \
30                              Uruchurtu, Don. Manuel E    male  40.0      0
49          Arnold-Franchi, Mrs. Josef (Josefine Franchi)  female  18.0      1
207                           Albimona, Mr. Nassef Cassem    male  26.0      0
210                                     Ali, Mr. Ahmed    male  24.0      0
353                           Arnold-Franchi, Mr. Josef    male  25.0      1
493                             Artagaveytia, Mr. Ramon    male  71.0      0
518  Angle, Mrs. William A (Florence "Mary" Agnes H…  female  36.0      1
784                                     Ali, Mr. William    male  25.0      0
840                           Alhomaki, Mr. Ilmari Rudolf    male  20.0      0

     Parch             Ticket     Fare Cabin Embarked        lastname title  \
30       0          PC 17601  27.7208   NaN        C       Uruchurtu  Don.
49       0            349237  17.8000   NaN        S  Arnold-Franchi  Mrs.
207      0              2699  18.7875   NaN        C        Albimona   Mr.
210      0  SOTON/O.Q. 3101311   7.0500   NaN        S             Ali   Mr.
353      0            349237  17.8000   NaN        S  Arnold-Franchi   Mr.
493      0          PC 17609  49.5042   NaN        C    Artagaveytia   Mr.
518      0            226875  26.0000   NaN        S           Angle  Mrs.
784      0  SOTON/O.Q. 3101312   7.0500   NaN        S             Ali   Mr.
840      0    SOTON/O2 3101287   7.9250   NaN        S        Alhomaki   Mr.

                                 firstname
30                                Manuel E
49                  Josef (Josefine Franchi)
```

```
207                                Nassef Cassem
210                                        Ahmed
353                                        Josef
493                                        Ramon
518  William A (Florence "Mary" Agnes Hughes)
784                                      William
840                                Ilmari Rudolf
```

3 Date and Time in Pandas

Note why separate objects to handle data and time when python already has datetime functionality?

- syntax wise datetime is very convenient
- But the performance takes a hit while working with huge data. List vs Numpy Array
- The weaknesses of Python's datetime format inspired the NumPy team to add a set of native time series data type to NumPy.
- The datetime64 dtype encodes dates as 64-bit integers, and thus allows arrays of dates to be represented very compactly.

[86]:
```
date = np.array('2024-07-05', dtype=np.datetime64)
date
```

[86]:
```
array('2024-07-05', dtype='datetime64[D]')
```

[87]:
```
date + np.arange(12)
```

[87]:
```
array(['2024-07-05', '2024-07-06', '2024-07-07', '2024-07-08',
       '2024-07-09', '2024-07-10', '2024-07-11', '2024-07-12',
       '2024-07-13', '2024-07-14', '2024-07-15', '2024-07-16'],
      dtype='datetime64[D]')
```

- Because of the uniform type in NumPy datetime64 arrays, this type of operation can be accomplished much more quickly than if we were working directly with Python's datetime objects, especially as arrays get large
- Pandas Timestamp object combines the ease-of-use of python datetime with the efficient storage and vectorized interface of numpy.datetime64
- From a group of these Timestamp objects, Pandas can construct a DatetimeIndex that can be used to index data in a Series or DataFrame

3.1 Timestamp Object

Time stamps reference particular moments in time (e.g. Oct 16th, 2003 at 02:00am)

3.1.1 Creating Timestamp objects

[52]:
```
pd.Timestamp('2024/07/05')
```

[52]:
```
Timestamp('2024-07-05 00:00:00')
```

[57]:
```
# Variations

pd.Timestamp('2024, 07, 05')
pd.Timestamp('2024-07-05')
pd.Timestamp('2024, 07, 05')
pd.Timestamp('2024-07-05')
```

[57]:
```
Timestamp('2024-07-05 00:00:00')
```

[69]:
```
pd.Timestamp('5th July 2024')
```

[69]:
```
Timestamp('2024-07-05 00:00:00')
```

[68]:
```
pd.Timestamp('2024')
```

[68]:
```
Timestamp('2024-01-01 00:00:00')
```

[73]:
```
pd.Timestamp('5th July 2024 1:7pm')
```

[73]:
```
Timestamp('2024-07-05 13:07:00')
```

using Python's datetime module

[76]:
```
import datetime as dt

x = pd.Timestamp(dt.datetime(2024, 7, 5, 13, 10))
x
```

[76]:
```
Timestamp('2024-07-05 13:10:00')
```

3.1.2 Attributes of Timestamp

[77]:
```
x.year
```

[77]:
```
2024
```

[78]:
```
x.month
```

[78]:
```
7
```

[80]:
```
x.day
```

[80]:
```
5
```

[81]:
```
x.hour
```

[81]:
```
13
```

[82]:
```
x.minute
```

[82]:
```
10
```

[83]:
```
x.second
```

[83]:
```
0
```

3.2 DatetimeIndex Object

A collection of pandas Timestamps

3.2.1 Creating DatetimeIndex Object

3.2.2 From strings

[90]:
```
pd.DatetimeIndex(['2024/1/1', '2023/1/1', '2022/1/1', '2021/1/1'])
```

[90]:
```
DatetimeIndex(['2024-01-01', '2023-01-01', '2022-01-01', '2021-01-01'],
dtype='datetime64[ns]', freq=None)
```

3.2.3 Using Python datetime object

[92]:
```
pd.DatetimeIndex([dt.datetime(2024,1,1), dt.datetime(2023,1,1), dt.
↪datetime(2023,1,1)])
```

[92]:
```
DatetimeIndex(['2024-01-01', '2023-01-01', '2023-01-01'],
dtype='datetime64[ns]', freq=None)
```

3.2.4 Using pd.timestamps

[95]:
```
dt_index = pd.DatetimeIndex([pd.Timestamp(2024,1,1), pd.Timestamp(2023,1,1), pd.
↪Timestamp(2022,1,1) ])
```

3.2.5 Creating Series from DatetimeIndex as index

[96]:
```
pd.Series([1,2,3], index=dt_index)
```

[96]:
```
2024-01-01    1
2023-01-01    2
2022-01-01    3
```

```
dtype: int64
```

3.3 date_range function

3.3.1 pd.date_range(start='start/date', end='end/date', freq='D')

- Genrate DatetimeIndex from start date to end date based on frequency
- freq

val	description
'h'	Every hour
'nh'	Every n hour
'D'	Daily date
'2D'	Alternate days
'nD'	days with n gap
'W'	week days
'W-MON'	every Mon
'B'	Business days
'MS'	Month start
'ME'	Month end
'AS'	Year start
'YE'	Year end
'YE-DEC'	Every Dec

periods parameter

- four among start, end, freq and periods, three are compulsory
- if end is not specified we can decide num of results by periods
- if freq is not specified it divides the range among n periods.
- all four will throw error

[102]:
```
pd.date_range(start='2024/1/1', end='2025/1/1', freq='D')
```

[102]:
```
DatetimeIndex(['2024-01-01', '2024-01-02', '2024-01-03', '2024-01-04',
               '2024-01-05', '2024-01-06', '2024-01-07', '2024-01-08',
               '2024-01-09', '2024-01-10',
               …
               '2024-12-23', '2024-12-24', '2024-12-25', '2024-12-26',
               '2024-12-27', '2024-12-28', '2024-12-29', '2024-12-30',
               '2024-12-31', '2025-01-01'],
              dtype='datetime64[ns]', length=367, freq='D')
```

[103]:
```
pd.date_range(start='2024/07/05', end='2024/08/05', freq='2D')
```

[103]:
```
DatetimeIndex(['2024-07-05', '2024-07-07', '2024-07-09', '2024-07-11',
               '2024-07-13', '2024-07-15', '2024-07-17', '2024-07-19',
               '2024-07-21', '2024-07-23', '2024-07-25', '2024-07-27',
```

```
               '2024-07-29', '2024-07-31', '2024-08-02', '2024-08-04'],
              dtype='datetime64[ns]', freq='2D')
```

[104]:
```
pd.date_range(start='2024/07/05', end='2024/08/05', freq='3D')
```

[104]:
```
DatetimeIndex(['2024-07-05', '2024-07-08', '2024-07-11', '2024-07-14',
               '2024-07-17', '2024-07-20', '2024-07-23', '2024-07-26',
               '2024-07-29', '2024-08-01', '2024-08-04'],
              dtype='datetime64[ns]', freq='3D')
```

[105]:
```
pd.date_range(start='2024/07/05', end='2024/08/05', freq='W')
```

[105]:
```
DatetimeIndex(['2024-07-07', '2024-07-14', '2024-07-21', '2024-07-28',
               '2024-08-04'],
              dtype='datetime64[ns]', freq='W-SUN')
```

[106]:
```
pd.date_range(start='2024/07/05', end='2024/08/05', freq='W-MON')
```

[106]:
```
DatetimeIndex(['2024-07-08', '2024-07-15', '2024-07-22', '2024-07-29',
               '2024-08-05'],
              dtype='datetime64[ns]', freq='W-MON')
```

[112]:
```
pd.date_range(start='2024/07/05', end='2024/07/15', freq='100h')
```

[112]:
```
DatetimeIndex(['2024-07-05 00:00:00', '2024-07-09 04:00:00',
               '2024-07-13 08:00:00'],
              dtype='datetime64[ns]', freq='100h')
```

[116]:
```
pd.date_range(start='2024/07/05', end='2030/08/05', freq='YE')
```

[116]:
```
DatetimeIndex(['2024-12-31', '2025-12-31', '2026-12-31', '2027-12-31',
               '2028-12-31', '2029-12-31'],
              dtype='datetime64[ns]', freq='YE-DEC')
```

[119]:
```
# Equidistant dates between range
pd.date_range(start='2024/07/05', end='2030/08/05', periods=10)
```

[119]:
```
DatetimeIndex(['2024-07-05 00:00:00', '2025-03-08 21:20:00',
               '2025-11-10 18:40:00', '2026-07-15 16:00:00',
               '2027-03-19 13:20:00', '2027-11-21 10:40:00',
               '2028-07-25 08:00:00', '2029-03-29 05:20:00',
               '2029-12-01 02:40:00', '2030-08-05 00:00:00'],
              dtype='datetime64[ns]', freq=None)
```

3.4 to_datetime function

Converts an existing objects to pandas timestamp/datetimeindex object

[122]:
```
date_series = pd.Series(['2024/1/1', '2023/1/1', '2022/1/1'])
s = pd.to_datetime(date_series)
s
```

[122]:
```
0   2024-01-01
1   2023-01-01
2   2022-01-01
dtype: datetime64[ns]
```

[123]:
```
s.dt.day_name()
```

[123]:
```
0      Monday
1      Sunday
2    Saturday
dtype: object
```

dt accessor Accessor object for datetimelike properties of the Series values

[127]:
```
s = pd.Series(['2024/1/1', '2023/1/1', '2022/120/1'])
pd.to_datetime(s, errors='coerce').dt.month_name()
```

[127]:
```
0    January
1    January
2        NaN
dtype: object
```

[128]:
```
df = pd.read_csv('expense_data.csv')
df.shape
```

[128]:
```
(277, 11)
```

[131]:
```
df.head()
```

[131]:

	Date	Account	Category	Subcategory	Note	INR	Income/Expense	Note.1	Amount	Currency	Account.1
0	3/2/2022 10:11	CUB - online payment	Food	NaN	Brownie	50.0	Expense	NaN	50.0	INR	50.0
1	3/2/2022 10:11	CUB - online payment	Other	NaN	To lended people	300.0	Expense	NaN	300.0	INR	300.0
2	3/1/2022 19:50	CUB - online payment	Food	NaN	Dinner	78.0	Expense	NaN	78.0	INR	78.0
3	3/1/2022 18:56	CUB - online payment	Transportation	NaN	Metro	30.0	Expense	NaN	30.0	INR	30.0
4	3/1/2022 18:22	CUB - online payment	Food	NaN	Snacks	67.0	Expense	NaN	67.0	INR	67.0

[137]:
```
df['Date'] = pd.to_datetime(df['Date'])
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 277 entries, 0 to 276
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Date            277 non-null    datetime64[ns]
 1   Account         277 non-null    object
 2   Category        277 non-null    object
 3   Subcategory     0 non-null      float64
 4   Note            273 non-null    object
 5   INR             277 non-null    float64
 6   Income/Expense  277 non-null    object
 7   Note.1          0 non-null      float64
 8   Amount          277 non-null    float64
 9   Currency        277 non-null    object
 10  Account.1       277 non-null    float64
dtypes: datetime64[ns](1), float64(5), object(5)
memory usage: 23.9+ KB
```

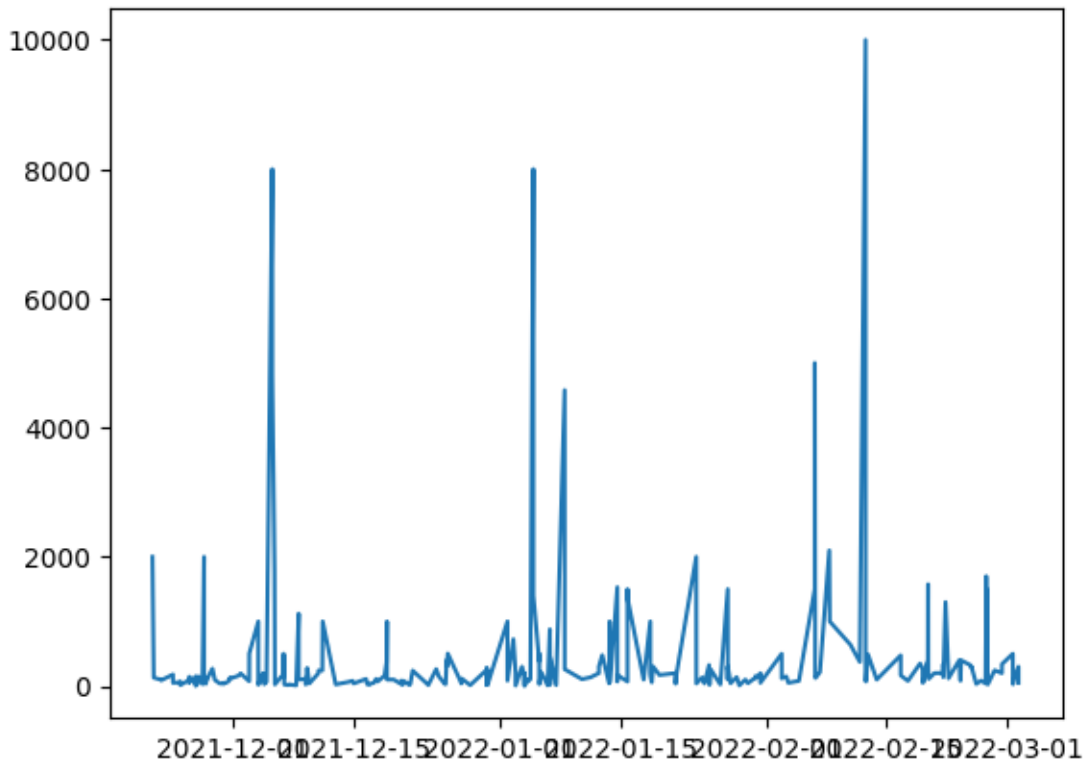
[138]:
```
df['Date'].dt.month_name()
```

[138]:
```
0         March
1         March
2         March
3         March
4         March
          ...
272    November
273    November
274    November
275    November
276    November
Name: Date, Length: 277, dtype: object
```

3.4.1 Plotting Graphs

[139]:
```
import matplotlib.pyplot as plt
plt.plot(df['Date'], df['INR'])
```

[139]:
```
[<matplotlib.lines.Line2D at 0x1cfcc57fb30>]
```



[141]:
```
df['day_name'] = df['Date'].dt.day_name()
```

[142]:
```
df.head()
```

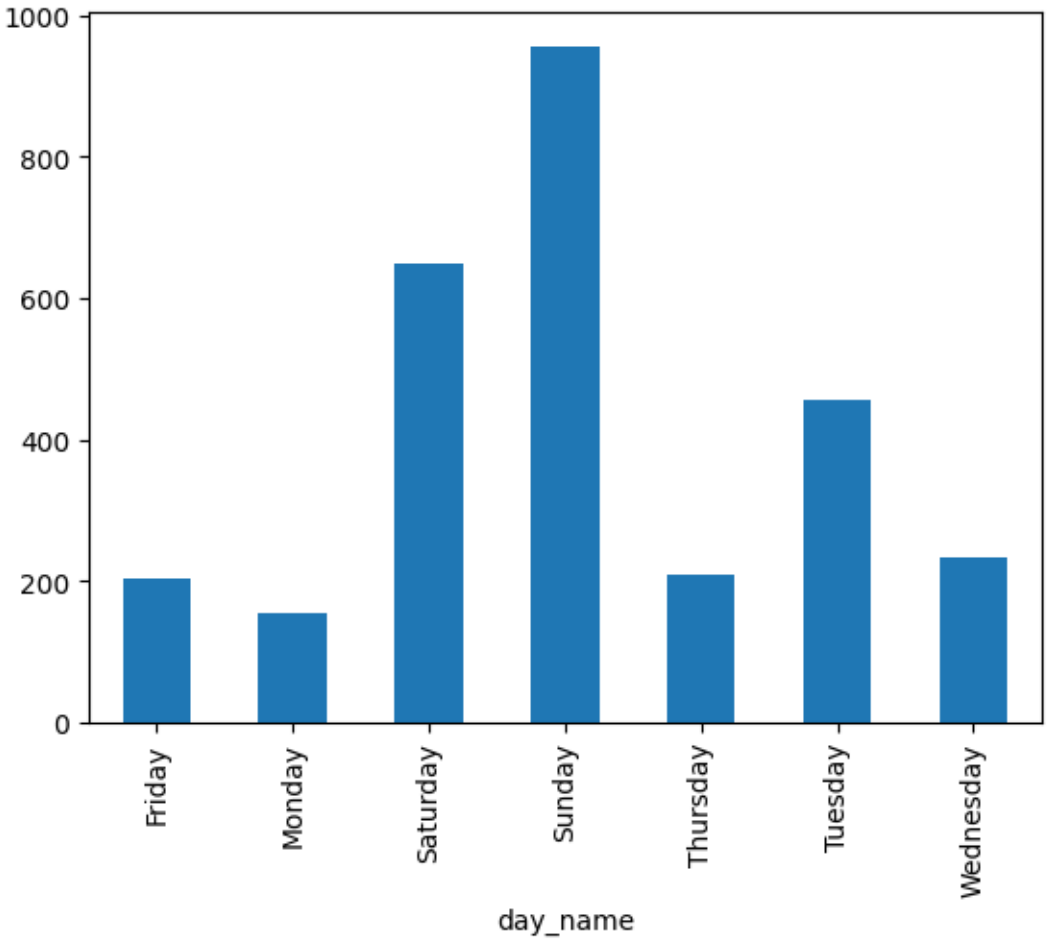
[142]:

	Date	Account	Category	Subcategory	Note	INR	Income/Expense	Note.1	Amount	Currency	Account.1	day_name
0	2022-03-02 10:11:00	CUB - online payment	Food	NaN	Brownie	50.0	Expense	NaN	50.0	INR	50.0	Wednesday
1	2022-03-02 10:11:00	CUB - online payment	Other	NaN	To lended people	300.0	Expense	NaN	300.0	INR	300.0	Wednesday
2	2022-03-01 19:50:00	CUB - online payment	Food	NaN	Dinner	78.0	Expense	NaN	78.0	INR	78.0	Tuesday
3	2022-03-01 18:56:00	CUB - online payment	Transportation	NaN	Metro	30.0	Expense	NaN	30.0	INR	30.0	Tuesday
4	2022-03-01 18:22:00	CUB - online payment	Food	NaN	Snacks	67.0	Expense	NaN	67.0	INR	67.0	Tuesday

```
3    Tuesday
4    Tuesday
```

[143]:
```
df.groupby('day_name')['INR'].mean().plot(kind='bar')
```

[143]:
```
<Axes: xlabel='day_name'>
```



[145]:
```
df['month_name'] = df['Date'].dt.month_name()

df.head()
```

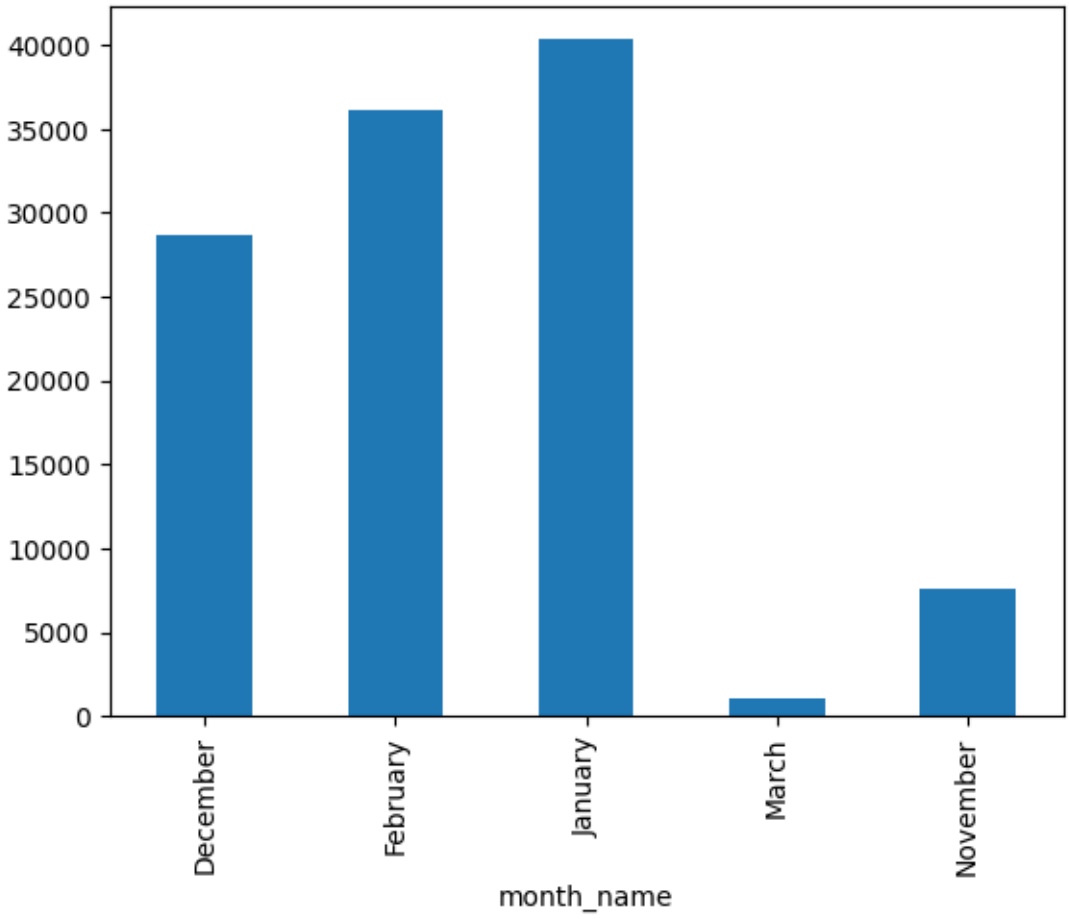
[145]:
```
                 Date               Account        Category  Subcategory  \
0 2022-03-02 10:11:00  CUB - online payment            Food          NaN
1 2022-03-02 10:11:00  CUB - online payment           Other          NaN
2 2022-03-01 19:50:00  CUB - online payment            Food          NaN
3 2022-03-01 18:56:00  CUB - online payment  Transportation          NaN
4 2022-03-01 18:22:00  CUB - online payment            Food          NaN
```

```
                Note    INR Income/Expense  Note.1  Amount Currency  Account.1  \
0           Brownie   50.0        Expense     NaN    50.0      INR       50.0
1  To lended people  300.0        Expense     NaN   300.0      INR      300.0
2            Dinner   78.0        Expense     NaN    78.0      INR       78.0
3             Metro   30.0        Expense     NaN    30.0      INR       30.0
4            Snacks   67.0        Expense     NaN    67.0      INR       67.0

    day_name month_name
0  Wednesday      March
1  Wednesday      March
2    Tuesday      March
3    Tuesday      March
4    Tuesday      March
```

[146]:
```
df.groupby('month_name')['INR'].sum().plot(kind='bar')
```

[146]:
```
<Axes: xlabel='month_name'>
```



[149]:
```
# Expense on month ends

df[df['Date'].dt.is_month_end]
```

[149]:

	Date	Account	Category	Subcategory	Note	INR	Income/Expense	Note.1	Amount	Currency	Account.1	day_name	month_name
7	2022-02-28 11:56:00	CUB - online payment	Food	NaN	Pizza	339.15	Expense	NaN	339.15	INR	339.15	Monday	February
8	2022-02-28 11:45:00	CUB - online payment	Other	NaN	From kumara	200.00	Income	NaN	200.00	INR	200.00	Monday	February
61	2022-01-31 08:44:00	CUB - online payment	Transportation	NaN	Vnr to apk	50.00	Expense	NaN	50.00	INR	50.00	Monday	January
62	2022-01-31 08:27:00	CUB - online payment	Other	NaN	To vicky	200.00	Expense	NaN	200.00	INR	200.00	Monday	January
63	2022-01-31 08:26:00	CUB - online payment	Transportation	NaN	To ksr station	153.00	Expense	NaN	153.00	INR	153.00	Monday	January
242	2021-11-30 14:24:00	CUB - online payment	Gift	NaN	Bharath birthday	115.00	Expense	NaN	115.00	INR	115.00	Tuesday	November
243	2021-11-30 14:17:00	CUB - online payment	Food	NaN	Lunch with company	128.00	Expense	NaN	128.00	INR	128.00	Tuesday	November
244	2021-11-30 10:11:00	CUB - online payment	Food	NaN	Breakfast	70.00	Expense	NaN	70.00	INR	70.00	Tuesday	November