

In [ ]:

```
import numpy as np
import pandas as pd
```

In [ ]:

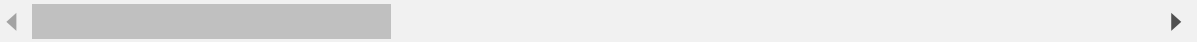
```
movies = pd.read_csv('tmdb_5000_movies.csv')
credits = pd.read_csv('tmdb_5000_credits.csv')
```

In [3]:

```
movies.head()
```

Out[3]:

	budget	genres	homepage	id	keywords	original
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id": 1463, "name": "culture clash"}]	
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"}]	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "name": "nautilus"}]	
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name": "based on novel"}]	
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "name": "Thriller"}]	http://www.thedarkknighttrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853, "name": "batman"}]	
4	260000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id": 1463, "name": "culture clash"}]	



In [4]:

```
credits.head(1).values
```

Out[4]:

```
array([[19995, 'Avatar',  
       '({"cast_id": 242, "character": "Jake Sully", "credit_id": "5602a8  
a7c3a3685532001c9a", "gender": 2, "id": 65731, "name": "Sam Worthington",  
"order": 0}, {"cast_id": 3, "character": "Neytiri", "credit_id": "52fe4800  
9251416c750ac9cb", "gender": 1, "id": 8691, "name": "Zoe Saldana", "orde  
r": 1}, {"cast_id": 25, "character": "Dr. Grace Augustine", "credit_id":  
"52fe48009251416c750aca39", "gender": 1, "id": 10205, "name": "Sigourney W  
eaver", "order": 2}, {"cast_id": 4, "character": "Col. Quaritch", "credit_  
id": "52fe48009251416c750ac9cf", "gender": 2, "id": 32747, "name": "Stephe  
n Lang", "order": 3}, {"cast_id": 5, "character": "Trudy Chacon", "credit_  
id": "52fe48009251416c750ac9d3", "gender": 1, "id": 17647, "name": "Michel  
le Rodriguez", "order": 4}, {"cast_id": 8, "character": "Selfridge", "cred  
it_id": "52fe48009251416c750ac9e1", "gender": 2, "id": 1771, "name": "Giov  
anni Ribisi", "order": 5}, {"cast_id": 7, "character": "Norm Spellman", "c  
redit_id": "52fe48009251416c750ac9dd", "gender": 2, "id": 59231, "name":  
"Joel David Moore", "order": 6}, {"cast_id": 9, "character": "Moat", "cred  
it_id": "52fe48009251416c750ac9e5", "gender": 1, "id": 30485, "name": "CCH  
Pounder", "order": 7}, {"cast_id": 11, "character": "Fvtukan", "credit_i
```

In [5]:

```
movies = movies.merge(credits,on='title') # merge two datasets
```

In [6]:

```
movies.shape
```

Out[6]:

```
(4809, 23)
```

In [14]:

```
credits.shape
```

Out[14]:

```
(4803, 4)
```

In [15]:

```
movies.head(3)
```

Out[15]:

	budget	genres	homepage	id	keywords	original
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id": ...	
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	

3 rows × 23 columns



In [ ]:

In [16]:

```
# now removing columns that are not required at using only required columns
```

In [17]:

```
movies.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4809 entries, 0 to 4808
Data columns (total 23 columns):
#   Column                Non-Null Count  Dtype
---  -
0   budget                4809 non-null  int64
1   genres                4809 non-null  object
2   homepage              1713 non-null  object
3   id                    4809 non-null  int64
4   keywords              4809 non-null  object
5   original_language     4809 non-null  object
6   original_title        4809 non-null  object
7   overview              4806 non-null  object
8   popularity            4809 non-null  float64
9   production_companies  4809 non-null  object
10  production_countries  4809 non-null  object
11  release_date          4808 non-null  object
12  revenue               4809 non-null  int64
13  runtime               4807 non-null  float64
14  spoken_languages      4809 non-null  object
15  status                4809 non-null  object
16  tagline               3965 non-null  object
17  title                 4809 non-null  object
18  vote_average          4809 non-null  float64
19  vote_count            4809 non-null  int64
20  movie_id              4809 non-null  int64
21  cast                  4809 non-null  object
22  crew                  4809 non-null  object
dtypes: float64(3), int64(5), object(15)
memory usage: 901.7+ KB
```

In [18]:

```
movies = movies[['id', 'title', 'genres', 'keywords', 'overview', 'cast', 'crew']]
```

In [19]:

```
movies.head(3)
```

Out[19]:

	id	title	genres	keywords	overview	cast	cr
0	19995	Avatar	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 1463, "name": "culture clash"}, {"id": "...	In the 22nd century, a paraplegic Marine is di...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_i
1	285	Pirates of the Caribbean: At World's End	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	Captain Barbossa, long believed to be dead, ha...	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_i
2	206647	Spectre	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 470, "name": "spy"}, {"id": 818, "name...	A cryptic message from Bond's past sends him o...	[{"cast_id": 1, "character": "James Bond", "cr...	[{"credit_i

In [20]:

```
movies.isnull().sum()
```

Out[20]:

```
id      0
title   0
genres  0
keywords 0
overview 3
cast     0
crew     0
dtype: int64
```

In [21]:

```
movies.dropna(inplace=True) # removed rows with null values
```

In [22]:

```
movies.isnull().sum() # now there are no rows with null values
```

Out[22]:

```
id          0
title       0
genres      0
keywords    0
overview    0
cast        0
crew        0
dtype: int64
```

In [ ]:

In [23]:

```
movies.duplicated().sum() # there are no duplicate values preset in dataset
```

Out[23]:

```
0
```

In [24]:

```
movies.iloc[0].genres
```

Out[24]:

```
'[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"}, {"id": 878, "name": "Science Fiction"}]'
```

In [25]:

```
import ast
# basically we need ast.literal_eval which convert string dictionary format to dictionary a
# genres dictionary is in string format it should be converted to have access of name
```

In [26]:

```
# '[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fant
# this is in the form of list of dictionaries which we need to remove
```

In [27]:

```
def convert(object):
    L=[]
    for i in ast.literal_eval(object):
        L.append(i['name'])
    return L
```

In [28]:

```
movies['genres'] = movies['genres'].apply(convert)
```

In [29]:

```
movies.head(4)
```

Out[29]:

	id	title	genres	keywords	overview	cast	crew
0	19995	Avatar	[Action, Adventure, Fantasy, Science Fiction]	[{"id": 1463, "name": "culture clash"}, {"id": ...	In the 22nd century, a paraplegic Marine is di...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de.
1	285	Pirates of the Caribbean: At World's End	[Adventure, Fantasy, Action]	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	Captain Barbossa, long believed to be dead, ha...	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_id": "52fe4232c3a36847f800b579", "de.
2	206647	Spectre	[Action, Adventure, Crime]	[{"id": 470, "name": "spy"}, {"id": 818, "name...	A cryptic message from Bond's past sends him o...	[{"cast_id": 1, "character": "James Bond", "cr...	[{"credit_id": "54805967c3a36829b5002c41", "de.
3	49026	The Dark Knight Rises	[Action, Crime, Drama, Thriller]	[{"id": 849, "name": "dc comics"}, {"id": 853, ...	Following the death of District Attorney Harve...	[{"cast_id": 2, "character": "Bruce Wayne / Ba...	[{"credit_id": "52fe4781c3a36847f81398c3", "de.

In [ ]:

In [30]:

```
# now same thing with movies keywords
```

In [31]:

```
movies.head(1).keywords.values
```

Out[31]:

```
array([{'id': 1463, 'name': 'culture clash'}, {'id': 2964, 'name': 'futur
e'}, {'id': 3386, 'name': 'space war'}, {'id': 3388, 'name': 'space colon
y'}, {'id': 3679, 'name': 'society'}, {'id': 3801, 'name': 'space travel'},
{'id': 9685, 'name': 'futuristic'}, {'id': 9840, 'name': 'romance'}, {'id':
9882, 'name': 'space'}, {'id': 9951, 'name': 'alien'}, {'id': 10148, 'name':
'tribe'}, {'id': 10158, 'name': 'alien planet'}, {'id': 10987, 'name': 'cg
i'}, {'id': 11399, 'name': 'marine'}, {'id': 13065, 'name': 'soldier'}, {'i
d': 14643, 'name': 'battle'}, {'id': 14720, 'name': 'love affair'}, {'id': 1
65431, 'name': 'anti war'}, {'id': 193554, 'name': 'power relations'}, {'i
d': 206690, 'name': 'mind and soul'}, {'id': 209714, 'name': '3d'}],
      dtype=object)
```

In [32]:

```
movies['keywords'] = movies['keywords'].apply(convert)
```

In [33]:

```
movies.head(4)
```

Out[33]:

	id	title	genres	keywords	overview	cast	crea
0	19995	Avatar	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	In the 22nd century, a paraplegic Marine is di...	{{"cast_id": 242, "character": "Jake Sully", "...	{{"credit_id" "52fe48009251416c750aca23" "de.
1	285	Pirates of the Caribbean: At World's End	[Adventure, Fantasy, Action]	[ocean, drug abuse, exotic island, east india ...	Captain Barbossa, long believed to be dead, ha...	{{"cast_id": 4, "character": "Captain Jack Spa...	{{"credit_id" "52fe4232c3a36847f800b579" "de.
2	206647	Spectre	[Action, Adventure, Crime]	[spy, based on novel, secret agent, sequel, mi...	A cryptic message from Bond's past sends him o...	{{"cast_id": 1, "character": "James Bond", "cr...	{{"credit_id" "54805967c3a36829b5002c41" "de.
3	49026	The Dark Knight Rises	[Action, Crime, Drama, Thriller]	[dc comics, crime fighter, terrorist, secret i...	Following the death of District Attorney Harve...	{{"cast_id": 2, "character": "Bruce Wayne / Ba...	{{"credit_id" "52fe4781c3a36847f81398c3" "de.

In [ ]:



In [34]:

```
movies.head(1).cast.values
```

Out[34]:

```
array([[{"cast_id": 242, "character": "Jake Sully", "credit_id": "5602a8a7c3a3685532001c9a", "gender": 2, "id": 65731, "name": "Sam Worthington", "order": 0}, {"cast_id": 3, "character": "Neytiri", "credit_id": "52fe48009251416c750ac9cb", "gender": 1, "id": 8691, "name": "Zoe Saldana", "order": 1}, {"cast_id": 25, "character": "Dr. Grace Augustine", "credit_id": "52fe48009251416c750aca39", "gender": 1, "id": 10205, "name": "Sigourney Weaver", "order": 2}, {"cast_id": 4, "character": "Col. Quaritch", "credit_id": "52fe48009251416c750ac9cf", "gender": 2, "id": 32747, "name": "Stephen Lang", "order": 3}, {"cast_id": 5, "character": "Trudy Chacon", "credit_id": "52fe48009251416c750ac9d3", "gender": 1, "id": 17647, "name": "Michelle Rodriguez", "order": 4}, {"cast_id": 8, "character": "Selfridge", "credit_id": "52fe48009251416c750ac9e1", "gender": 2, "id": 1771, "name": "Giovanni Ribisi", "order": 5}, {"cast_id": 7, "character": "Norm Spellman", "credit_id": "52fe48009251416c750ac9dd", "gender": 2, "id": 59231, "name": "Joel David Moore", "order": 6}, {"cast_id": 9, "character": "Moat", "credit_id": "52fe48009251416c750ac9e5", "gender": 1, "id": 30485, "name": "CCH Pounder", "order": 7}, {"cast_id": 11, "character": "Eytukan", "credit_id": "52fe48009251416c750ac9ed", "gender": 2, "id": 15853, "name": "Wes Studi", "order": 8}, {"cast_id": 10, "character": "Tsu'Tey", "credit_id": "52fe48009251416c750ac9e9", "gender": 2, "id": 10964, "name": "Laz Alonso", "order": 9}, {"cast_id": 12, "character": "Dr. Max Patel", "credit_id": "52fe48009251416c750ac9f1", "gender": 2, "id": 95697, "name": "Dileep Rao", "order": 10}, {"cast_id": 13, "character": "Lyle Wainfleet", "credit_id": "52fe48009251416c750ac9f5", "gender": 2, "id": 98215, "name": "Matt Gerald", "order": 11}, {"cast_id": 32, "character": "Private Fike", "credit_id": "52fe48009251416c750aca5b", "gender": 2, "id": 154153, "name": "Sean Anthony Moran", "order": 12}, {"cast_id": 33, "character": "Cryo Vault Med Tech", "credit_id": "52fe48009251416c750aca5f", "gender": 2, "id": 397312, "name": "Jason Whyte", "order": 13}, {"cast_id": 34, "character": "Venture Star Crew Chief", "credit_id": "52fe48009251416c750aca63", "gender": 2, "id": 42317, "name": "Scott Lawrence", "order": 14}, {"cast_id": 35, "character": "Lock Up Trooper", "credit_id": "52fe48009251416c750aca67", "gender": 2, "id": 986734, "name": "Kelly Kilgour", "order": 15}, {"cast_id": 36, "character": "Shuttle Pilot", "credit_id": "52fe48009251416c750aca6b", "gender": 0, "id": 1207227, "name": "James Patrick Pitt", "order": 16}, {"cast_id": 37, "character": "Shuttle Co-Pilot", "credit_id": "52fe48009251416c750aca6f", "gender": 0, "id": 1180936, "name": "Sean Patrick Murphy", "order": 17}, {"cast_id": 38, "character": "Shuttle Crew Chief", "credit_id": "52fe48009251416c750aca73", "gender": 2, "id": 1019578, "name": "Peter Dillion", "order": 18}, {"cast_id": 39, "character": "Tractor Operator / Troupe", "credit_id": "52fe48009251416c750aca77", "gender": 0, "id": 91443, "name": "Kevin Dorman", "order": 19}, {"cast_id": 40, "character": "Dragon Gunship Pilot", "credit_id": "52fe48009251416c750aca7b", "gender": 2, "id": 173391, "name": "Kelson Henderson", "order": 20}, {"cast_id": 41, "character": "Dragon Gunship Gunner", "credit_id": "52fe48009251416c750aca7f", "gender": 0, "id": 1207236, "name": "David Van Horn", "order": 21}, {"cast_id": 42, "character": "Dragon Gunship Navigator", "credit_id": "52fe48009251416c750aca83", "gender": 0, "id": 215913, "name": "Jacob Tomuri", "order": 22}, {"cast_id": 43, "character": "Suit #1", "credit_id": "52fe48009251416c750aca87", "gender": 0, "id": 143206, "name": "Michael Blain-Rozgay", "order": 23}, {"cast_id": 44, "character": "Suit #2", "credit_id": "52fe48009251416c750aca8b", "gender": 2, "id": 169676, "name": "Jon Curry", "order": 24}, {"cast_id": 46, "character": "Ambient Room Tech", "credit_id": "52fe48009251416c750aca8f", "gender": 0, "id": 1048610, "name": "Luke Hawker", "order": 25}, {"cast_id": 47, "character": "Ambient Room Tech / Troupe", "credit_id": "52fe48009251416c750aca93", "gender": 0, "id": 42288, "name": "Woody Schultz", "order": 26},
```

```
{
  "cast_id": 48, "character": "Horse Clan Leader", "credit_id": "52fe48009251416c750aca97", "gender": 2, "id": 68278, "name": "Peter Mensah", "order": 27},
  {"cast_id": 49, "character": "Link Room Tech", "credit_id": "52fe48009251416c750aca9b", "gender": 0, "id": 1207247, "name": "Sonia Yee", "order": 28},
  {"cast_id": 50, "character": "Basketball Avatar / Troupe", "credit_id": "52fe48009251416c750aca9f", "gender": 1, "id": 1207248, "name": "Jahnel Curfman", "order": 29},
  {"cast_id": 51, "character": "Basketball Avatar", "credit_id": "52fe48009251416c750acaa3", "gender": 0, "id": 89714, "name": "Ilram Choi", "order": 30},
  {"cast_id": 52, "character": "Na\\'vi Child", "credit_id": "52fe48009251416c750acaa7", "gender": 0, "id": 1207249, "name": "Kyla Warren", "order": 31},
  {"cast_id": 53, "character": "Troupe", "credit_id": "52fe48009251416c750acaab", "gender": 0, "id": 1207250, "name": "Lisa Roumain", "order": 32},
  {"cast_id": 54, "character": "Troupe", "credit_id": "52fe48009251416c750acaaf", "gender": 1, "id": 83105, "name": "Debra Wilson", "order": 33},
  {"cast_id": 57, "character": "Troupe", "credit_id": "52fe48009251416c750acabb", "gender": 0, "id": 1207253, "name": "Chris Mala", "order": 34},
  {"cast_id": 55, "character": "Troupe", "credit_id": "52fe48009251416c750acab3", "gender": 0, "id": 1207251, "name": "Taylor Kibby", "order": 35},
  {"cast_id": 56, "character": "Troupe", "credit_id": "52fe48009251416c750acab7", "gender": 0, "id": 1207252, "name": "Jodie Landau", "order": 36},
  {"cast_id": 58, "character": "Troupe", "credit_id": "52fe48009251416c750acabf", "gender": 0, "id": 1207254, "name": "Julie Lamm", "order": 37},
  {"cast_id": 59, "character": "Troupe", "credit_id": "52fe48009251416c750acac3", "gender": 0, "id": 1207257, "name": "Cullen B. Madden", "order": 38},
  {"cast_id": 60, "character": "Troupe", "credit_id": "52fe48009251416c750acac7", "gender": 0, "id": 1207259, "name": "Joseph Brady Madden", "order": 39},
  {"cast_id": 61, "character": "Troupe", "credit_id": "52fe48009251416c750acacb", "gender": 0, "id": 1207262, "name": "Frankie Torres", "order": 40},
  {"cast_id": 62, "character": "Troupe", "credit_id": "52fe48009251416c750acacf", "gender": 1, "id": 115860, "name": "Austin Wilson", "order": 41},
  {"cast_id": 63, "character": "Troupe", "credit_id": "52fe48019251416c750acad3", "gender": 1, "id": 983705, "name": "Sara Wilson", "order": 42},
  {"cast_id": 64, "character": "Troupe", "credit_id": "52fe48019251416c750acad7", "gender": 0, "id": 1207263, "name": "Tamica Washington-Miller", "order": 43},
  {"cast_id": 65, "character": "Op Center Staff", "credit_id": "52fe48019251416c750acadb", "gender": 1, "id": 1145098, "name": "Lucy Briant", "order": 44},
  {"cast_id": 66, "character": "Op Center Staff", "credit_id": "52fe48019251416c750acadf", "gender": 2, "id": 33305, "name": "Nathan Meister", "order": 45},
  {"cast_id": 67, "character": "Op Center Staff", "credit_id": "52fe48019251416c750acae3", "gender": 0, "id": 1207264, "name": "Gerry Blair", "order": 46},
  {"cast_id": 68, "character": "Op Center Staff", "credit_id": "52fe48019251416c750acae7", "gender": 2, "id": 33311, "name": "Matthew Chamberlain", "order": 47},
  {"cast_id": 69, "character": "Op Center Staff", "credit_id": "52fe48019251416c750acaeb", "gender": 0, "id": 1207265, "name": "Paul Yates", "order": 48},
  {"cast_id": 70, "character": "Op Center Duty Officer", "credit_id": "52fe48019251416c750acae f", "gender": 0, "id": 1207266, "name": "Wray Wilson", "order": 49},
  {"cast_id": 71, "character": "Op Center Staff", "credit_id": "52fe48019251416c750acaf3", "gender": 2, "id": 54492, "name": "James Gaylyn", "order": 50},
  {"cast_id": 72, "character": "Dancer", "credit_id": "52fe48019251416c750acaf7", "gender": 0, "id": 1207267, "name": "Melvin Leno Clark III", "order": 51},
  {"cast_id": 73, "character": "Dancer", "credit_id": "52fe48019251416c750acafb", "gender": 0, "id": 1207268, "name": "Carvon Futrell", "order": 52},
  {"cast_id": 74, "character": "Dancer", "credit_id": "52fe48019251416c750acaff", "gender": 0, "id": 1207269, "name": "Brandon Jelkes", "order": 53},
  {"cast_id": 75, "character": "Dancer", "credit_id": "52fe48019251416c750acb03", "gender": 0, "id": 1207270, "name": "Micah Moch", "order": 54},
  {"cast_id": 76, "character": "Dancer", "credit_id": "52fe48019251416c750acb07", "gender": 0, "id": 1207271, "name": "Hanniyah Muhammad", "order": 55},
  {"cast_id": 77, "character": "Dancer", "credit_id": "52fe48019251416c750acb0b", "gender": 0, "id": 1207272, "name": "Christopher Nolen", "order": 56},
  {"cast_id": 78, "character": "Dancer", "credit_id": "52fe48019251416c750acb0f", "gender": 0,
```

```

{id": 1207273, "name": "Christa Oliver", "order": 57}, {"cast_id": 79, "character": "Dancer", "credit_id": "52fe48019251416c750acb13", "gender": 0, "id": 1207274, "name": "April Marie Thomas", "order": 58}, {"cast_id": 80, "character": "Dancer", "credit_id": "52fe48019251416c750acb17", "gender": 0, "id": 1207275, "name": "Bravita A. Threatt", "order": 59}, {"cast_id": 81, "character": "Mining Chief (uncredited)", "credit_id": "52fe48019251416c750acb1b", "gender": 0, "id": 1207276, "name": "Colin Bleasdale", "order": 60}, {"cast_id": 82, "character": "Veteran Miner (uncredited)", "credit_id": "52fe48019251416c750acb1f", "gender": 0, "id": 107969, "name": "Mike Bodnar", "order": 61}, {"cast_id": 83, "character": "Richard (uncredited)", "credit_id": "52fe48019251416c750acb23", "gender": 0, "id": 1207278, "name": "Matt Clayton", "order": 62}, {"cast_id": 84, "character": "Nav'i (uncredited)", "credit_id": "52fe48019251416c750acb27", "gender": 1, "id": 147898, "name": "Nicole Dionne", "order": 63}, {"cast_id": 85, "character": "Trooper (uncredited)", "credit_id": "52fe48019251416c750acb2b", "gender": 0, "id": 1207280, "name": "Jamie Harrison", "order": 64}, {"cast_id": 86, "character": "Trooper (uncredited)", "credit_id": "52fe48019251416c750acb2f", "gender": 0, "id": 1207281, "name": "Allan Henry", "order": 65}, {"cast_id": 87, "character": "Ground Technician (uncredited)", "credit_id": "52fe48019251416c750acb33", "gender": 2, "id": 1207282, "name": "Anthony Ingruber", "order": 66}, {"cast_id": 88, "character": "Flight Crew Mechanic (uncredited)", "credit_id": "52fe48019251416c750acb37", "gender": 0, "id": 1207283, "name": "Ashley Jeffery", "order": 67}, {"cast_id": 14, "character": "Samson Pilot", "credit_id": "52fe48009251416c750ac9f9", "gender": 0, "id": 98216, "name": "Dean Knowsley", "order": 68}, {"cast_id": 89, "character": "Trooper (uncredited)", "credit_id": "52fe48019251416c750acb3b", "gender": 0, "id": 1201399, "name": "Joseph Mika-Hunt", "order": 69}, {"cast_id": 90, "character": "Banshee (uncredited)", "credit_id": "52fe48019251416c750acb3f", "gender": 0, "id": 236696, "name": "Terry Notary", "order": 70}, {"cast_id": 91, "character": "Soldier (uncredited)", "credit_id": "52fe48019251416c750acb43", "gender": 0, "id": 1207287, "name": "Kai Pantano", "order": 71}, {"cast_id": 92, "character": "Blast Technician (uncredited)", "credit_id": "52fe48019251416c750acb47", "gender": 0, "id": 1207288, "name": "Logan Pithyou", "order": 72}, {"cast_id": 93, "character": "Vindum Raah (uncredited)", "credit_id": "52fe48019251416c750acb4b", "gender": 0, "id": 1207289, "name": "Stuart Pollock", "order": 73}, {"cast_id": 94, "character": "Hero (uncredited)", "credit_id": "52fe48019251416c750acb4f", "gender": 0, "id": 584868, "name": "Raja", "order": 74}, {"cast_id": 95, "character": "Ops Centreworker (uncredited)", "credit_id": "52fe48019251416c750acb53", "gender": 0, "id": 1207290, "name": "Gareth Ruck", "order": 75}, {"cast_id": 96, "character": "Engineer (uncredited)", "credit_id": "52fe48019251416c750acb57", "gender": 0, "id": 1062463, "name": "Rhian Sheehan", "order": 76}, {"cast_id": 97, "character": "Col. Quaritch's Mech Suit (uncredited)", "credit_id": "52fe48019251416c750acb5b", "gender": 0, "id": 60656, "name": "T. J. Storm", "order": 77}, {"cast_id": 98, "character": "Female Marine (uncredited)", "credit_id": "52fe48019251416c750acb5f", "gender": 0, "id": 1207291, "name": "Jodie Taylor", "order": 78}, {"cast_id": 99, "character": "Ikran Clan Leader (uncredited)", "credit_id": "52fe48019251416c750acb63", "gender": 1, "id": 1186027, "name": "Alicia Vela-Bailey", "order": 79}, {"cast_id": 100, "character": "Geologist (uncredited)", "credit_id": "52fe48019251416c750acb67", "gender": 0, "id": 1207292, "name": "Richard Whiteside", "order": 80}, {"cast_id": 101, "character": "Na'vi (uncredited)", "credit_id": "52fe48019251416c750acb6b", "gender": 0, "id": 103259, "name": "Nikie Zambo", "order": 81}, {"cast_id": 102, "character": "Ambient Room Tech / Troupe", "credit_id": "52fe48019251416c750acb6f", "gender": 1, "id": 42286, "name": "Julene Renee", "order": 82}]]],
dtype=object)

```

In [ ]:

In [35]:

```
# now we will convert cast  
# here we will only use top 3 (first 3) actors  
# convert function will be modified as follows:
```

In [36]:

```
def convert3(object):  
    cnt=0  
    L=[]  
    for i in ast.literal_eval(object):  
        L.append(i['name'])  
        cnt+=1;  
        if cnt==3:  
            break;  
    return L
```

In [37]:

```
movies['cast'] = movies['cast'].apply(convert3)
```

In [38]:

```
movies['cast'].head(3)
```

Out[38]:

```
0    [Sam Worthington, Zoe Saldana, Sigourney Weaver]  
1    [Johnny Depp, Orlando Bloom, Keira Knightley]  
2    [Daniel Craig, Christoph Waltz, Léa Seydoux]  
Name: cast, dtype: object
```

In [ ]:

In [39]:

```
movies['crew'].head(1).values
```

Out[39]:

```
array([{'credit_id': '52fe48009251416c750aca23', 'department': 'Editing', 'gender': 0, 'id': 1721, 'job': 'Editor', 'name': 'Stephen E. Rivkin'}, {'credit_id': '539c47ecc3a36810e3001f87', 'department': 'Art', 'gender': 2, 'id': 496, 'job': 'Production Design', 'name': 'Rick Carter'}, {'credit_id': '54491c89c3a3680fb4001cf7', 'department': 'Sound', 'gender': 0, 'id': 900, 'job': 'Sound Designer', 'name': 'Christopher Boyes'}, {'credit_id': '54491cb70e0a267480001bd0', 'department': 'Sound', 'gender': 0, 'id': 900, 'job': 'Supervising Sound Editor', 'name': 'Christopher Boyes'}, {'credit_id': '539c4a4cc3a36810c9002101', 'department': 'Production', 'gender': 1, 'id': 1262, 'job': 'Casting', 'name': 'Mali Finn'}, {'credit_id': '5544ee3b925141499f0008fc', 'department': 'Sound', 'gender': 2, 'id': 1729, 'job': 'Original Music Composer', 'name': 'James Horner'}, {'credit_id': '52fe48009251416c750ac9c3', 'department': 'Directing', 'gender': 2, 'id': 2710, 'job': 'Director', 'name': 'James Cameron'}, {'credit_id': '52fe48009251416c750ac9d9', 'department': 'Writing', 'gender': 2, 'id': 2710, 'job': 'Writer', 'name': 'James Cameron'}, {'credit_id': '52fe48009251416c750aca17', 'department': 'Editing', 'gender': 2, 'id': 2710, 'job': 'Editor', 'name': 'James Cameron'}, {'credit_id': '52fe48009251416c750aca29', 'd
```

In [40]:

```
def fetch_director(obj):
    L = []
    for i in ast.literal_eval(obj):
        if(i['job'] == 'Director'):
            L.append(i['name'])
            break;
    return L;
```

In [41]:

```
movies['crew'] = movies['crew'].apply(fetch_director)
```

In [42]:

```
movies.head(4)
```

Out[42]:

	id	title	genres	keywords	overview	cast	crew
0	19995	Avatar	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	In the 22nd century, a paraplegic Marine is di...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[James Cameron]
1	285	Pirates of the Caribbean: At World's End	[Adventure, Fantasy, Action]	[ocean, drug abuse, exotic island, east india ...	Captain Barbossa, long believed to be dead, ha...	[Johnny Depp, Orlando Bloom, Keira Knightley]	[Gore Verbinski]
2	206647	Spectre	[Action, Adventure, Crime]	[spy, based on novel, secret agent, sequel, mi...	A cryptic message from Bond's past sends him o...	[Daniel Craig, Christoph Waltz, Léa Seydoux]	[Sam Mendes]
3	49026	The Dark Knight Rises	[Action, Crime, Drama, Thriller]	[dc comics, crime fighter, terrorist, secret i...	Following the death of District Attorney Harve...	[Christian Bale, Michael Caine, Gary Oldman]	[Christopher Nolan]

In [ ]:

In [43]:

```
movies.head(1).overview.values
```

Out[43]:

```
array(['In the 22nd century, a paraplegic Marine is dispatched to the moon P  
andora on a unique mission, but becomes torn between following orders and pr  
otecting an alien civilization.'],  
      dtype=object)
```

In [ ]:

In [44]:

```
movies['overview'] = movies['overview'].apply(lambda x:x.split())
```

In [45]:

```
movies.head(3)
```

Out[45]:

	id	title	genres	keywords	overview	cast	crew
0	19995	Avatar	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	[In, the, 22nd, century,, a, paraplegic, Marin...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[James Cameron]
1	285	Pirates of the Caribbean: At World's End	[Adventure, Fantasy, Action]	[ocean, drug abuse, exotic island, east india ...	[Captain, Barbossa,, long, believed, to, be, d...	[Johnny Depp, Orlando Bloom, Keira Knightley]	[Gore Verbinski]
2	206647	Spectre	[Action, Adventure, Crime]	[spy, based on novel, secret agent, sequel, mi...	[A, cryptic, message, from, Bond's, past, send...	[Daniel Craig, Christoph Waltz, Léa Seydoux]	[Sam Mendes]

In [46]:

```
movies['cast'].apply(lambda x:[i.replace(" ", "") for i in x])
```

Out[46]:

```
0      [SamWorthington, ZoeSaldana, SigourneyWeaver]
1      [JohnnyDepp, OrlandoBloom, KeiraKnightley]
2      [DanielCraig, ChristophWaltz, LéaSeydoux]
3      [ChristianBale, MichaelCaine, GaryOldman]
4      [TaylorKitsch, LynnCollins, SamanthaMorton]
...
4804   [CarlosGallardo, JaimeHoyos, PeterMarquardt]
4805   [EdwardBurns, KerryBishé, MarshaDietlein]
4806   [EricMabius, KristinBooth, CrystalLowe]
4807   [DanielHenney, ElizaCoupe, BillPaxton]
4808   [DrewBarrymore, BrianHerzlinger, CoreyFeldman]
Name: cast, Length: 4806, dtype: object
```

In [47]:

```
movies['genres'] = movies['genres'].apply(lambda x:[i.replace(" ", "") for i in x])
movies['keywords'] = movies['keywords'].apply(lambda x:[i.replace(" ", "") for i in x])
movies['overview'] = movies['overview'].apply(lambda x:[i.replace(" ", "") for i in x])
movies['cast'] = movies['cast'].apply(lambda x:[i.replace(" ", "") for i in x])
movies['crew'] = movies['crew'].apply(lambda x:[i.replace(" ", "") for i in x])
```

In [48]:

```
movies.head()
```

Out[48]:

	id	title	genres	keywords	overview	cast	c
0	19995	Avatar	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...]	[In, the, 22nd, century,, a, paraplegic, Marin...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCame
1	285	Pirates of the Caribbean: At World's End	[Adventure, Fantasy, Action]	[ocean, drugabuse, exoticisland, eastindiatrad...	[Captain, Barbossa,, long, believed, to, be, d...	[JohnnyDepp, OrlandoBloom, KeiraKnightley]	[GoreVerbir
2	206647	Spectre	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...]	[A, cryptic, message, from, Bond's, past, send...	[DanielCraig, ChristophWaltz, LéaSeydoux]	[SamMen
3	49026	The Dark Knight Rises	[Action, Crime, Drama, Thriller]	[dcomics, crimefighter, terrorist, secretiden...	[Following, the, death, of, District, Attorney...	[ChristianBale, MichaelCaine, GaryOldman]	[ChristopherNc
4	49529	John Carter	[Action, Adventure, ScienceFiction]	[basedonnovel, mars, medallion, spacetravel, p...	[John, Carter, is, a, war-weary,, former, mili...	[TaylorKitsch, LynnCollins, SamanthaMorton]	[AndrewStar

In [ ]:

In [49]:

```
movies['tags'] = movies['genres'] + movies['keywords'] + movies['overview'] + movies['cast']
```

In [50]:

```
movies.head(1).tags.values
```

Out[50]:

```
array([list(['Action', 'Adventure', 'Fantasy', 'ScienceFiction', 'cultureclash', 'future', 'spacewar', 'spacecolony', 'society', 'spacetravel', 'futuristic', 'romance', 'space', 'alien', 'tribe', 'alienplanet', 'cgi', 'marine', 'soldier', 'battle', 'loveaffair', 'antiwar', 'powerrelations', 'mindandsoul', '3d', 'In', 'the', '22nd', 'century,', 'a', 'paraplegic', 'Marine', 'is', 'dispatched', 'to', 'the', 'moon', 'Pandora', 'on', 'a', 'unique', 'mission,', 'but', 'becomes', 'torn', 'between', 'following', 'orders', 'and', 'protecting', 'an', 'alien', 'civilization.', 'SamWorthington', 'ZoeSaldana', 'SigourneyWeaver', 'JamesCameron'])],
      dtype=object)
```



In [51]:

```
movies['tags'] = movies['tags'].apply(lambda x: " ".join(x))
```

In [52]:

```
movies.head(3)
```

Out[52]:

	id	title	genres	keywords	overview	cast	cre
0	19995	Avatar	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...]	[In, the, 22nd, century,, a, paraplegic, Marin...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCamero
1	285	Pirates of the Caribbean: At World's End	[Adventure, Fantasy, Action]	[ocean, drugabuse, exoticisland, eastindiatrad...	[Captain, Barbossa,, long, believed, to, be, d...	[JohnnyDepp, OrlandoBloom, KeiraKnightley]	[GoreVerbinsl
2	206647	Spectre	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...]	[A, cryptic, message, from, Bond's, past, send...	[DanielCraig, ChristophWaltz, LéaSeydoux]	[SamMende

In [ ]:

In [53]:

```
movies['tags'] = movies['tags'].apply(lambda x:x.lower())
```

In [54]:

```
movies.head()
```

Out[54]:

	id	title	genres	keywords	overview	cast	c
0	19995	Avatar	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	[In, the, 22nd, century,, a, paraplegic, Marin...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCame
1	285	Pirates of the Caribbean: At World's End	[Adventure, Fantasy, Action]	[ocean, drugabuse, exoticisland, eastindiatrad...	[Captain, Barbossa,, long, believed, to, be, d...	[JohnnyDepp, OrlandoBloom, KeiraKnightley]	[GoreVerbir
2	206647	Spectre	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...	[A, cryptic, message, from, Bond's, past, send...	[DanielCraig, ChristophWaltz, LéaSeydoux]	[SamMen
3	49026	The Dark Knight Rises	[Action, Crime, Drama, Thriller]	[dccomics, crimefighter, terrorist, secretiden...	[Following, the, death, of, District, Attorney...	[ChristianBale, MichaelCaine, GaryOldman]	[ChristopherNc
4	49529	John Carter	[Action, Adventure, ScienceFiction]	[basedonnovel, mars, medallion, spacetravel, p...	[John, Carter, is, a, war-weary,, former, mili...	[TaylorKitsch, LynnCollins, SamanthaMorton]	[AndrewStar

In [55]:

```
new_df = movies[['id','title','tags']]
```

In [56]:

```
new_df.head()
```

Out[56]:

	id	title	tags
0	19995	Avatar	action adventure fantasy sciencefiction cultur...
1	285	Pirates of the Caribbean: At World's End	adventure fantasy action ocean drugabuse exoti...
2	206647	Spectre	action adventure crime spy basedonnovel secret...
3	49026	The Dark Knight Rises	action crime drama thriller dccomics crimefigh...
4	49529	John Carter	action adventure sciencefiction basedonnovel m...

In [ ]:

In [57]:

```
import nltk
```

In [58]:

```
from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()
```

In [59]:

```
# creating vector
```

In [289]:

```
def stem(text):
    L = []
    for i in text.split():
        L.append(ps.stem(i))
    return " ".join(L)    # return string form

'''
role of this function is to stem words like
loving,love,loved  all will become love
'''
```

Out[289]:

```
'\nrole of this function is to stem words like\nloving,love,loved  all will
become love\n'
```

In [290]:

```
from sklearn.feature_extraction.text import CountVectorizer
```

In [291]:

```
cv = CountVectorizer(max_features=5000, stop_words='english')  
# stop_words = 'english' removes the english stop words such as a, the, and, an.....  
# max_features = 5000 means most frequent 5000 words from join of all the tags
```

In [292]:

```
vectors = cv.fit_transform(new_df['tags']).toarray()
```

In [293]:

```
vectors[0]
```

Out[293]:

```
array([0, 0, 0, ..., 0, 0, 0], dtype=int64)
```

In [ ]:

In [294]:

```
cv.get_feature_names()
```

```
'andrew',  
'android',  
'andy',  
'andygarcía',  
'angel',  
'angela',  
'angelabassett',  
'angeles',  
'angelinajoli',  
'anger',  
'angl',  
'angri',  
'ani',  
'anim',  
'animalattack',  
'animalhorror',  
'animals',  
'anjelicahuston',  
'ann',  
'anna',
```

In [295]:

```
new_df['tags'] = new_df['tags'].apply(stem)
```

<ipython-input-295-be18a4346d89>:1: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy) ([https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy))

```
new_df['tags'] = new_df['tags'].apply(stem)
```

In [296]:

```
new_df.head()
```

Out[296]:

	id	title	tags
0	19995	Avatar	action adventur fantasi sciencefict culturecla...
1	285	Pirates of the Caribbean: At World's End	adventur fantasi action ocean drugabu exoticis...
2	206647	Spectre	action adventur crime spi basedonnovel secreta...
3	49026	The Dark Knight Rises	action crime drama thriller dccomic crimefight...
4	49529	John Carter	action adventur sciencefict basedonnovel mar m...

In [ ]:

In [297]:

```
from sklearn.metrics.pairwise import cosine_similarity
```

In [298]:

```
similarity = cosine_similarity(vectors)
```

In [299]:

```
similarity[0]
```

Out[299]:

```
array([1.          , 0.08458258, 0.08718573, ..., 0.04559608, 0.          ,
       0.          ])
```

In [300]:

```
sorted(list(enumerate(similarity[0])),reverse=True,key=lambda x:x[1])[1:6]
```

Out[300]:

```
[(1216, 0.29061909685954823),  
(3730, 0.26401000024165),  
(507, 0.25903973506580724),  
(539, 0.2537477434955704),  
(2409, 0.2507061052819501)]
```

In [ ]:

In [301]:

```
def recommend(movie):  
    movie_index = new_df[new_df['title'] == movie].index[0]  
    distance= similarity[movie_index]  
    movie_list = sorted(list(enumerate(distance)),reverse=True,key=lambda x:x[1])[1:6]  
    for i in movie_list:  
        print(new_df.iloc[i[0]]['title'])
```

In [308]:

```
try:  
    recommend('Avtar')  
except:  
    print("Not a registerd movie")
```

Not a registerd movie

In [310]:

```
recommend('Avatar')
```

Aliens vs Predator: Requiem  
Falcon Rising  
Independence Day  
Titan A.E.  
Aliens

In [315]:

```
def suggest_movie(movie):  
    try:  
        recommend(movie)  
    except:  
        print("Not a registerd movie")
```

In [ ]:

In [318]:

```
suggest_movie('Avtar') # spelling mistake Avatar not avatar it can be handled using exc
```

Not a registerd movie

In [319]:

```
suggest_movie('Avatar')
```

Aliens vs Predator: Requiem

Falcon Rising

Independence Day

Titan A.E.

Aliens

In [320]:

```
suggest_movie('John Carter')
```

Riddick

Krrish

The Other Side of Heaven

The Legend of Hercules

Get Carter

In [ ]: