In [1]:
```
import numpy as np
import pandas as pd
import difflib
```

In [2]:
```
movie = pd.read_csv('tmdb_5000_movies.csv')
crides = pd.read_csv('tmdb_5000_credits.csv')
```

In [3]:
```
movie.head(1)
```

Out[3]:

	budget	genres	homepage	id	keywords	original_language	original_
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Av

In [4]:
```
crides.head(1)['cast'].values
```

Out[4]:
```
array(['[{"cast_id": 242, "character": "Jake Sully", "credit_id": "5602a
8a7c3a3685532001c9a", "gender": 2, "id": 65731, "name": "Sam Worthingto
n", "order": 0}, {"cast_id": 3, "character": "Neytiri", "credit_id": "52
fe48009251416c750ac9cb", "gender": 1, "id": 8691, "name": "Zoe Saldana",
"order": 1}, {"cast_id": 25, "character": "Dr. Grace Augustine", "credit
_id": "52fe48009251416c750aca39", "gender": 1, "id": 10205, "name": "Sig
ourney Weaver", "order": 2}, {"cast_id": 4, "character": "Col. Quaritc
h", "credit_id": "52fe48009251416c750ac9cf", "gender": 2, "id": 32747,
"name": "Stephen Lang", "order": 3}, {"cast_id": 5, "character": "Trudy
Chacon", "credit_id": "52fe48009251416c750ac9d3", "gender": 1, "id": 176
47, "name": "Michelle Rodriguez", "order": 4}, {"cast_id": 8, "characte
r": "Selfridge", "credit_id": "52fe48009251416c750ac9e1", "gender": 2,
"id": 1771, "name": "Giovanni Ribisi", "order": 5}, {"cast_id": 7, "char
acter": "Norm Spellman", "credit_id": "52fe48009251416c750ac9dd", "gende
r": 2, "id": 59231, "name": "Joel David Moore", "order": 6}, {"cast_id":
9, "character": "Moat", "credit_id": "52fe48009251416c750ac9e5", "gende
r": 1, "id": 30485, "name": "CCH Pounder", "order": 7}, {"cast_id": 11,
"character": "Eytukan", "credit_id": "52fe48009251416c750ac9ed", "gende
r": 2, "id": 15853, "name": "Wes Studi", "order": 8}, {"cast_id": 10, "c
```

In [5]:
```
movies = movie.merge(crides, on = 'title' )
```

In [6]:
```
movies.head(1)
```

Out[6]:

	budget	genres	homepage	id	keywords	original_language	original_
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Av

1 rows × 23 columns

columns we have to maintain in the data set

1. genres
2. id
3. keywords
4. orginal_title
5. overview
6. cast
7. crew

In [7]:
```
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 [8]:
```
movies = movies[[ 'id','genres', 'keywords','original_title','overview','ca
```

In [9]:
```
movies.head()
```

Out[9]:

	id	genres	keywords	original_title	overview	cast	
0	19995	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 1463, "name": "culture clash"}, {"id":...	Avatar	In the 22nd century, a paraplegic Marine is di...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"cred "52fe48009251416c750ac
1	285	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"cred "52fe4232c3a36847f800b
2	206647	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 470, "name": "spy"}, {"id": 818, "name...	Spectre	A cryptic message from Bond's past sends him o...	[{"cast_id": 1, "character": "James Bond", "cr...	[{"cred "54805967c3a36829b5002
3	49026	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	[{"id": 849, "name": "dc comics"}, {"id": 853,...	The Dark Knight Rises	Following the death of District Attorney Harve...	[{"cast_id": 2, "character": "Bruce Wayne / Ba...	[{"cred "52fe4781c3a36847f8139
4	49529	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 818, "name": "based on novel"}, {"id":...	John Carter	John Carter is a war-weary, former military ca...	[{"cast_id": 5, "character": "John Carter", "c...	[{"cred "52fe479ac3a36847f813e

In [10]:
```
movies.duplicated().sum()
```

Out[10]:
```
0
```

In [11]:
```
movies.iloc[1].genres
```

Out[11]:
```
'[{"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"}, {"id": 2
8, "name": "Action"}]'
```

In [12]:
```
movies.isnull().sum()
```

Out[12]:
```
id                0
genres            0
keywords          0
original_title    0
overview          3
cast              0
crew              0
dtype: int64
```

In [13]:
```
## '[{"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"}, {"id":
## ['Adventutr','Fantacy','Action' ]
```

In [14]:
```
import ast
ast.literal_eval
```

Out[14]:
```
<function ast.literal_eval(node_or_string)>
```

In [15]:
```
def convert(obj):
    L =[]
    for i in ast. literal_eval(obj):
        L.append(i['name'])
    return L
```

In [16]:
```
# convert('[{"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"},
```

In [17]:
```
import ast
```

In [18]:
```
movies['genres'] = movies['genres'].apply(convert)
```

In [19]:
```
movies.head(2)
```

Out[19]:

	id	genres	keywords	original_title	overview	cast	cre
0	19995	[Action, Adventure, Fantasy, Science Fiction]	[{"id": 1463, "name": "culture clash"}, {"id":...	Avatar	In the 22nd century, a paraplegic Marine is di...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_ic "52fe48009251416c750aca23 "de
1	285	[Adventure, Fantasy, Action]	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_ic "52fe4232c3a36847f800b579 "de

In [20]:
```
movies['keywords'] = movies['keywords'].apply(convert)
```

In [21]:
```
movies.head(1)
```

Out[21]:

	id	genres	keywords	original_title	overview	cast	crev
0	19995	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	Avatar	In the 22nd century, a paraplegic Marine is di...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id "52fe48009251416c750aca23 "de.

In [22]:

```
movies.iloc[1].cast
```

```
Out[22]: '[{"cast_id": 4, "character": "Captain Jack Sparrow", "credit_id": "52fe42
         32c3a36847f800b50d", "gender": 2, "id": 85, "name": "Johnny Depp", "orde
         r": 0}, {"cast_id": 5, "character": "Will Turner", "credit_id": "52fe4232c
         3a36847f800b511", "gender": 2, "id": 114, "name": "Orlando Bloom", "orde
         r": 1}, {"cast_id": 6, "character": "Elizabeth Swann", "credit_id": "52fe4
         232c3a36847f800b515", "gender": 1, "id": 116, "name": "Keira Knightley",
         "order": 2}, {"cast_id": 12, "character": "William \\"Bootstrap Bill\\" Tu
         rner", "credit_id": "52fe4232c3a36847f800b52d", "gender": 2, "id": 1640,
         "name": "Stellan Skarsg\\u00e5rd", "order": 3}, {"cast_id": 10, "characte
         r": "Captain Sao Feng", "credit_id": "52fe4232c3a36847f800b525", "gender":
         2, "id": 1619, "name": "Chow Yun-fat", "order": 4}, {"cast_id": 9, "charac
         ter": "Captain Davy Jones", "credit_id": "52fe4232c3a36847f800b521", "gend
         er": 2, "id": 2440, "name": "Bill Nighy", "order": 5}, {"cast_id": 7, "cha
         racter": "Captain Hector Barbossa", "credit_id": "52fe4232c3a36847f800b51
         9", "gender": 2, "id": 118, "name": "Geoffrey Rush", "order": 6}, {"cast_i
         d": 14, "character": "Admiral James Norrington", "credit_id": "52fe4232c3a
         36847f800b535", "gender": 2, "id": 1709, "name": "Jack Davenport", "orde
         r": 7}, {"cast_id": 13, "character": "Joshamee Gibbs", "credit_id": "52fe4
         232c3a36847f800b531", "gender": 2, "id": 2449, "name": "Kevin McNally", "o
         rder": 8}, {"cast_id": 11, "character": "Lord Cutler Beckett", "credit_i
         d": "52fe4232c3a36847f800b529", "gender": 2, "id": 2441, "name": "Tom Holl
         ander", "order": 9}, {"cast_id": 19, "character": "Tia Dalma", "credit_i
         d": "52fe4232c3a36847f800b549", "gender": 1, "id": 2038, "name": "Naomie H
         arris", "order": 10}, {"cast_id": 8, "character": "Governor Weatherby Swan
         n", "credit_id": "52fe4232c3a36847f800b51d", "gender": 2, "id": 378, "nam
         e": "Jonathan Pryce", "order": 11}, {"cast_id": 37, "character": "Captain
         Teague Sparrow", "credit_id": "52fe4232c3a36847f800b5b3", "gender": 2, "i
         d": 1430, "name": "Keith Richards", "order": 12}, {"cast_id": 16, "charact
         er": "Pintel", "credit_id": "52fe4232c3a36847f800b53d", "gender": 2, "id":
         1710, "name": "Lee Arenberg", "order": 13}, {"cast_id": 15, "character":
         "Ragetti", "credit_id": "52fe4232c3a36847f800b539", "gender": 2, "id": 171
         1, "name": "Mackenzie Crook", "order": 14}, {"cast_id": 18, "character":
         "Lieutenant Theodore Groves", "credit_id": "52fe4232c3a36847f800b545", "ge
         nder": 2, "id": 4031, "name": "Greg Ellis", "order": 15}, {"cast_id": 55,
         "character": "Cotton", "credit_id": "57e28d2ec3a3681a01005b5c", "gender":
         2, "id": 1715, "name": "David Bailie", "order": 16}, {"cast_id": 17, "char
         acter": "Marty", "credit_id": "52fe4232c3a36847f800b541", "gender": 2, "i
         d": 4030, "name": "Martin Klebba", "order": 17}, {"cast_id": 57, "characte
         r": "Ian Mercer", "credit_id": "57e28d78c3a36808b900bf4f", "gender": 0, "i
         d": 939, "name": "David Schofield", "order": 18}, {"cast_id": 62, "charact
         er": "Scarlett", "credit_id": "57e28ec5c3a3681a50005855", "gender": 1, "i
         d": 2450, "name": "Lauren Maher", "order": 19}, {"cast_id": 63, "characte
         r": "Giselle", "credit_id": "57e28ed692514123f5005635", "gender": 1, "id":
         2452, "name": "Vanessa Branch", "order": 20}, {"cast_id": 60, "character":
         "Mullroy", "credit_id": "57e28db2c3a3681a01005bc7", "gender": 2, "id": 171
         4, "name": "Angus Barnett", "order": 21}, {"cast_id": 59, "character": "Mu
         rtogg", "credit_id": "57e28da192514118f7006008", "gender": 0, "id": 1713,
         "name": "Giles New", "order": 22}, {"cast_id": 58, "character": "Tai Huan
         g", "credit_id": "57e28d8ec3a3681a01005bab", "gender": 2, "id": 22075, "na
         me": "Reggie Lee", "order": 23}, {"cast_id": 64, "character": "Henry Turne
         r", "credit_id": "57e29119925141151100a6cc", "gender": 2, "id": 61259, "na
         me": "Dominic Scott Kay", "order": 24}, {"cast_id": 39, "character": "Mist
         ress Ching", "credit_id": "52fe4232c3a36847f800b5bd", "gender": 1, "id": 3
         3500, "name": "Takayo Fischer", "order": 25}, {"cast_id": 40, "character":
         "Lieutenant Greitzer", "credit_id": "52fe4232c3a36847f800b5c1", "gender":
         2, "id": 1224149, "name": "David Meunier", "order": 26}, {"cast_id": 49,
         "character": "Hadras", "credit_id": "56d1871c92514174680010cf", "gender":
```

```
2, "id": 429401, "name": "Ho-Kwan Tse", "order": 27}, {"cast_id": 56, "cha
racter": "Clacker", "credit_id": "57e28d4b92514125710055cb", "gender": 0,
"id": 1123, "name": "Andy Beckwith", "order": 28}, {"cast_id": 51, "charac
ter": "Penrod", "credit_id": "56ec8c14c3a3682260003c53", "gender": 2, "i
d": 1056117, "name": "Peter Donald Badalamenti II", "order": 29}, {"cast_i
d": 61, "character": "Cotton\'s Parrot (voice)", "credit_id": "57e28dcc925
1412463005678", "gender": 2, "id": 21700, "name": "Christopher S. Capp",
"order": 30}, {"cast_id": 65, "character": "Captain Teague", "credit_id":
"58bc2a37c3a368663003740b", "gender": 2, "id": 1430, "name": "Keith Richar
ds", "order": 31}, {"cast_id": 66, "character": "Captain Jocard", "credit_
id": "58bc2a8e925141609e03a179", "gender": 2, "id": 2603, "name": "Hakeem
Kae-Kazim", "order": 32}, {"cast_id": 67, "character": "Captain Ammand",
"credit_id": "58e2a21ac3a36872af00f9c2", "gender": 0, "id": 70577, "name":
"Ghassan Massoud", "order": 33}]'
```

In [23]:

```
def convert3(obj):
    L =[]
    counter = 0
    for i in ast. literal_eval(obj):
        if counter!=3:
            L.append(i['name'])
            counter+= 1
        else:
            break
    return L
```

In [24]:

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

In [25]:

```
movies.head(1)
```

Out[25]:

	id	genres	keywords	original_title	overview	cast	cre
0	19995	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	Avatar	In the 22nd century, a paraplegic Marine is di...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[{"credit_i "52fe48009251416c750aca2 "de

In [26]:
```
movies.iloc[1].crew
```

Out[26]:

```
'[{"credit_id": "52fe4232c3a36847f800b579", "department": "Camera", "gende
r": 2, "id": 120, "job": "Director of Photography", "name": "Dariusz Wolsk
i"}, {"credit_id": "52fe4232c3a36847f800b4fd", "department": "Directing",
"gender": 2, "id": 1704, "job": "Director", "name": "Gore Verbinski"}, {"c
redit_id": "52fe4232c3a36847f800b54f", "department": "Production", "gende
r": 2, "id": 770, "job": "Producer", "name": "Jerry Bruckheimer"}, {"credi
t_id": "52fe4232c3a36847f800b503", "department": "Writing", "gender": 2,
"id": 1705, "job": "Screenplay", "name": "Ted Elliott"}, {"credit_id": "52
fe4232c3a36847f800b509", "department": "Writing", "gender": 2, "id": 1706,
"job": "Screenplay", "name": "Terry Rossio"}, {"credit_id": "52fe4232c3a36
847f800b57f", "department": "Editing", "gender": 0, "id": 1721, "job": "Ed
itor", "name": "Stephen E. Rivkin"}, {"credit_id": "52fe4232c3a36847f800b5
85", "department": "Editing", "gender": 2, "id": 1722, "job": "Editor", "n
ame": "Craig Wood"}, {"credit_id": "52fe4232c3a36847f800b573", "departmen
t": "Sound", "gender": 2, "id": 947, "job": "Original Music Composer", "na
me": "Hans Zimmer"}, {"credit_id": "52fe4232c3a36847f800b555", "departmen
t": "Production", "gender": 2, "id": 2444, "job": "Executive Producer", "n
ame": "Mike Stenson"}, {"credit_id": "52fe4232c3a36847f800b561", "departme
nt": "Production", "gender": 2, "id": 2445, "job": "Producer", "name": "Er
ic McLeod"}, {"credit_id": "52fe4232c3a36847f800b55b", "department": "Prod
uction", "gender": 2, "id": 2446, "job": "Producer", "name": "Chad Oman"},
{"credit_id": "52fe4232c3a36847f800b567", "department": "Production", "gen
der": 0, "id": 2447, "job": "Producer", "name": "Peter Kohn"}, {"credit_i
d": "52fe4232c3a36847f800b56d", "department": "Production", "gender": 0,
"id": 2448, "job": "Producer", "name": "Pat Sandston"}, {"credit_id": "52f
e4232c3a36847f800b58b", "department": "Production", "gender": 1, "id": 221
5, "job": "Casting", "name": "Denise Chamian"}, {"credit_id": "52fe4232c3a
36847f800b597", "department": "Art", "gender": 2, "id": 1226, "job": "Prod
uction Design", "name": "Rick Heinrichs"}, {"credit_id": "52fe4232c3a36847
f800b59d", "department": "Art", "gender": 2, "id": 553, "job": "Art Direct
ion", "name": "John Dexter"}, {"credit_id": "52fe4232c3a36847f800b591", "d
epartment": "Production", "gender": 1, "id": 3311, "job": "Casting", "nam
e": "Priscilla John"}, {"credit_id": "52fe4232c3a36847f800b5a3", "departme
nt": "Art", "gender": 1, "id": 4032, "job": "Set Decoration", "name": "Che
ryl Carasik"}, {"credit_id": "52fe4232c3a36847f800b5a9", "department": "Co
stume & Make-Up", "gender": 0, "id": 4033, "job": "Costume Design", "nam
e": "Liz Dann"}, {"credit_id": "52fe4232c3a36847f800b5af", "department":
"Costume & Make-Up", "gender": 1, "id": 4034, "job": "Costume Design", "na
me": "Penny Rose"}, {"credit_id": "56427ce8c3a3686a53000d8b", "departmen
t": "Sound", "gender": 2, "id": 5132, "job": "Music Supervisor", "name":
"Bob Badami"}, {"credit_id": "55993c15c3a36855db002f33", "department": "Ar
t", "gender": 2, "id": 146439, "job": "Conceptual Design", "name": "James
Ward Byrkit"}, {"credit_id": "52fe4232c3a36847f800b5b9", "department": "Co
stume & Make-Up", "gender": 1, "id": 406204, "job": "Makeup Department Hea
d", "name": "Ve Neill"}, {"credit_id": "56e47f7892514132690017bd", "depart
ment": "Crew", "gender": 2, "id": 1259516, "job": "Stunts", "name": "John
Dixon"}, {"credit_id": "5740be639251416597000849", "department": "Crew",
"gender": 0, "id": 1336716, "job": "CGI Supervisor", "name": "Dottie Starl
ing"}, {"credit_id": "56427c639251412fc8000dc1", "department": "Directin
g", "gender": 1, "id": 1344278, "job": "Script Supervisor", "name": "Pamel
a Alch"}, {"credit_id": "57083101c3a3681d320004e6", "department": "Crew",
"gender": 0, "id": 1368867, "job": "Special Effects Coordinator", "name":
"Allen Hall"}, {"credit_id": "56427d5ec3a3686a62000d4a", "department": "So
und", "gender": 0, "id": 1368884, "job": "Music Editor", "name": "Melissa
Muik"}, {"credit_id": "56427c7b9251412fd4000e07", "department": "Directin
g", "gender": 1, "id": 1395290, "job": "Script Supervisor", "name": "Sharr
on Reynolds"}, {"credit_id": "56427d2bc3a3686a53000d9b", "department": "So
und", "gender": 0, "id": 1399327, "job": "Music Editor", "name": "Barbara
McDermott"}, {"credit_id": "56427cb4c3a3686a53000d87", "department": "Dire
cting", "gender": 1, "id": 1400738, "job": "Script Supervisor", "name": "K
aren Golden"}, {"credit_id": "56427d169251412fd4000e23", "department": "So
und", "gender": 0, "id": 1534197, "job": "Music Editor", "name": "Katie Gr
eathouse"}]'
```

In [27]:

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

In [28]:

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

In [29]:

```
movies.head(2)
```

Out[29]:

	id	genres	keywords	original_title	overview	cast	crew
0	19995	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	Avatar	In the 22nd century, a paraplegic Marine is di...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[James Cameron]
1	285	[Adventure, Fantasy, Action]	[ocean, drug abuse, exotic island, east india ...	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	[Johnny Depp, Orlando Bloom, Keira Knightley]	[Gore Verbinski]

In [30]:

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

Out[30]:

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

In [31]:
```
movies['genres'] = movies['genres'].apply(lambda x: [i.replace(" ", "")for
movies['keywords'] = movies['keywords'].apply(lambda x: [i.replace(" ", "")
movies['cast'] = movies['cast'].apply(lambda x: [i.replace(" ", "")for i in
movies['crew'] = movies['crew'].apply(lambda x: [i.replace(" ", "")for i in
```

In [32]:
```
movies.head(1)
```

Out[32]:

	id	genres	keywords	original_title	overview	cast	cre
0	19995	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	Avatar	In the 22nd century, a paraplegic Marine is di...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCamero

In [33]:
```
movies['tags'] = movies['overview'].map(str) + movies['genres'].map(str) +
```

In [34]:
```
movies.head(1)
```

Out[34]:

	id	genres	keywords	original_title	overview	cast	cre
0	19995	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	Avatar	In the 22nd century, a paraplegic Marine is di...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCamero

In [35]:
```
new_df = movies[['id', 'original_title','tags']]
```

In [36]:
```
new_df.head(1)
```

Out[36]:

	id	original_title	tags
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...

In [37]:
```
new_df.rename(columns={'id': 'movie_id'}, inplace = True)
```

```
C:\Users\windows\AppData\Local\Temp\ipykernel_13528\1939039232.py:1: Setti
ngWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/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.rename(columns={'id': 'movie_id'}, inplace = True)
```

In [38]:
```
new_df.head(1)
```

Out[38]:

	movie_id	original_title	tags
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...

In [39]:
```
new_df.rename(columns={'original_title': 'title'}, inplace = True)
```

```
C:\Users\windows\AppData\Local\Temp\ipykernel_13528\1264711564.py:1: Setti
ngWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/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.rename(columns={'original_title': 'title'}, inplace = True)
```

In [40]:
```
new_df
```

Out[40]:

	movie_id	title	tags
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...
1	285	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...
2	206647	Spectre	A cryptic message from Bond’s past sends him o...
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...
4	49529	John Carter	John Carter is a war-weary, former military ca...
...	...	...	...
4804	9367	El Mariachi	El Mariachi just wants to play his guitar and ...
4805	72766	Newlyweds	A newlywed couple's honeymoon is upended by th...
4806	231617	Signed, Sealed, Delivered	"Signed, Sealed, Delivered" introduces a dedic...
4807	126186	Shanghai Calling	When ambitious New York attorney Sam is sent t...
4808	25975	My Date with Drew	Ever since the second grade when he first saw ...

4809 rows × 3 columns

In [41]:
```
new_df
```

Out[41]:

	movie_id	title	tags
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...
1	285	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...
2	206647	Spectre	A cryptic message from Bond’s past sends him o...
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...
4	49529	John Carter	John Carter is a war-weary, former military ca...
...	...	...	...
4804	9367	El Mariachi	El Mariachi just wants to play his guitar and ...
4805	72766	Newlyweds	A newlywed couple's honeymoon is upended by th...
4806	231617	Signed, Sealed, Delivered	"Signed, Sealed, Delivered" introduces a dedic...
4807	126186	Shanghai Calling	When ambitious New York attorney Sam is sent t...
4808	25975	My Date with Drew	Ever since the second grade when he first saw ...

4809 rows × 3 columns

In [42]:
```
new_df['tags'][0]
```

Out[42]:
```
"In the 22nd century, a paraplegic Marine is dispatched to the moon Pandor
a on a unique mission, but becomes torn between following orders and prote
cting an alien civilization.['Action', 'Adventure', 'Fantasy', 'ScienceFic
tion']['cultureclash', 'future', 'spacewar', 'spacecolony', 'society', 'sp
acetravel', 'futuristic', 'romance', 'space', 'alien', 'tribe', 'alienplan
et', 'cgi', 'marine', 'soldier', 'battle', 'loveaffair', 'antiwar', 'power
relations', 'mindandsoul', '3d']['SamWorthington', 'ZoeSaldana', 'Sigourne
yWeaver']['JamesCameron']"
```

In [43]:
```
new_df['tags'][0]
```

Out[43]:
```
"In the 22nd century, a paraplegic Marine is dispatched to the moon Pandor
a on a unique mission, but becomes torn between following orders and prote
cting an alien civilization.['Action', 'Adventure', 'Fantasy', 'ScienceFic
tion']['cultureclash', 'future', 'spacewar', 'spacecolony', 'society', 'sp
acetravel', 'futuristic', 'romance', 'space', 'alien', 'tribe', 'alienplan
et', 'cgi', 'marine', 'soldier', 'battle', 'loveaffair', 'antiwar', 'power
relations', 'mindandsoul', '3d']['SamWorthington', 'ZoeSaldana', 'Sigourne
yWeaver']['JamesCameron']"
```

In [44]:
```
new_df['tags'] = new_df['tags']. str.replace('[^a-zA-Z0-9 ]',' ')
```

```
C:\Users\windows\AppData\Local\Temp\ipykernel_13528\2567082568.py:1: Futur
eWarning: The default value of regex will change from True to False in a f
uture version.
  new_df['tags'] = new_df['tags']. str.replace('[^a-zA-Z0-9 ]',' ')
C:\Users\windows\AppData\Local\Temp\ipykernel_13528\2567082568.py:1: Setti
ngWithCopyWarning:
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-doc
s/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']. str.replace('[^a-zA-Z0-9 ]',' ')
```

In [45]:
```
new_df['tags'] = new_df['tags'].apply(lambda x:x.lower())
```

```
C:\Users\windows\AppData\Local\Temp\ipykernel_13528\3214958533.py:1: Setti
ngWithCopyWarning:
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-doc
s/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(lambda x:x.lower())
```

In [46]:
```
new_df['tags'][0]
```

Out[46]:
```
'in the 22nd century  a paraplegic marine is dispatched to the moon pandor
a on a unique mission  but becomes torn between following orders and prote
cting an alien civilization  action  adventure  fantasy  sciencefic
tion  cultureclash  future  spacewar  spacecolony  society  sp
acetravel  futuristic  romance  space  alien  tribe  alienplan
et  cgi  marine  soldier  battle  loveaffair  antiwar  power
relations  mindandsoul  3d  samworthington  zoesaldana  sigourne
yweaver  jamescameron  '
```

In [47]:
```
new_df.head()
```

Out[47]:

	movie_id	title	tags
0	19995	Avatar	in the 22nd century a paraplegic marine is di...
1	285	Pirates of the Caribbean: At World's End	captain barbossa long believed to be dead ha...
2	206647	Spectre	a cryptic message from bond s past sends him o...
3	49026	The Dark Knight Rises	following the death of district attorney harve...
4	49529	John Carter	john carter is a war weary former military ca...

In [48]:

```
!pip install nltk
```

```
Requirement already satisfied: nltk in c:\users\windows\anaconda3\lib\site
-packages (3.7)
Requirement already satisfied: regex>=2021.8.3 in c:\users\windows\anacond
a3\lib\site-packages (from nltk) (2022.7.9)
Requirement already satisfied: tqdm in c:\users\windows\anaconda3\lib\site
-packages (from nltk) (4.64.1)
Requirement already satisfied: click in c:\users\windows\anaconda3\lib\sit
e-packages (from nltk) (8.0.4)
Requirement already satisfied: joblib in c:\users\windows\anaconda3\lib\si
te-packages (from nltk) (1.1.1)
Requirement already satisfied: colorama in c:\users\windows\anaconda3\lib
\site-packages (from click->nltk) (0.4.6)
```

In [49]:

```
import nltk
```

In [50]:

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

In [51]:

```
def stem(text):
    y = []

    for i in text.split():
        y.append(ps.stem(i))

    return " ".join(y)
```

In [52]:

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

```
C:\Users\windows\AppData\Local\Temp\ipykernel_13528\3213734980.py:1: Setti
ngWithCopyWarning:
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-doc
s/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 [53]:

```
new_df['tags'][0]
```

Out[53]:

```
'in the 22nd centuri a parapleg marin is dispatch to the moon pandora on a
uniqu mission but becom torn between follow order and protect an alien civ
il action adventur fantasi sciencefict cultureclash futur spacewar spaceco
loni societi spacetravel futurist romanc space alien tribe alienplanet cgi
marin soldier battl loveaffair antiwar powerrel mindandsoul 3d samworthing
ton zoesaldana sigourneyweav jamescameron'
```

In [54]:
```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000, stop_words='english')
```

In [55]:
```
vectors = cv.fit_transform(new_df['tags']).toarray()
```

In [56]:
```
vectors
```

Out[56]:
```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]], dtype=int64)
```

In [57]:
```
vectors[0]
```

Out[57]:
```
array([0, 0, 0, ..., 0, 0, 0], dtype=int64)
```

In [58]:
```
cv.get_feature_names_out()
```

Out[58]:
```
array(['000', '007', '10', ..., 'zone', 'zoo', 'zooeydeschanel'],
      dtype=object)
```

In [59]:
```
from sklearn.metrics.pairwise import cosine_similarity
```

In [60]:
```
similarity = cosine_similarity(vectors)
```

In [61]:
```
similarity[1]
```

Out[61]:
```
array([0.08238526, 1.        , 0.06063391, ..., 0.02251887, 0.        ,
       0.02585438])
```

In [62]:
```
sorted(similarity[0],reverse=True)
```

Out[62]:
```
[1.0000000000000004,
 0.2625754538144587,
 0.2506402059138015,
 0.24773936993814405,
 0.2471557663714903,
 0.24403555043462524,
 0.23372319715296228,
 0.2335709179335258,
 0.23128442344214897,
 0.23007892341722033,
 0.22880215766121476,
 0.2264554068289191,
 0.22417941532712204,
 0.21977383072747697,
 0.21618989813247,
 0.2135042050734495,
 0.21182963643408081,
 0.20965696734438366,
 0.20780973338645242,
 0.2046468711716401
```

In [63]:
```
## enumerate will helpp to it never loose the index position after the sort
sorted(list(enumerate(similarity[0])), reverse=True, key = lambda x:x[1])[1
```

Out[63]:
```
[(2409, 0.2625754538144587),
 (3731, 0.2506402059138015),
 (1216, 0.24773936993814405),
 (539, 0.2471557663714903),
 (507, 0.24403555043462524)]
```

In [64]:
```
def recommend(movie):
    movie_index = new_df[new_df['title'] == movie].index[0]
    distances = similarity[movie_index]
    movie_list = sorted(list(enumerate(distances)), reverse=True, key = lam

    for i in movie_list:
        print(new_df.iloc[i[0]].title)
```

In [65]:
```
recommend('Batman')
```

```
Batman
Batman & Robin
Batman Begins
Batman Returns
The R.M.
```

In [66]:
```
print(similarity.shape)
```

```
(4809, 4809)
```

In [ ]:
```
movie_name = input('Enter your favourite movie name : ' )
```

In [ ]:

```
# creating a list with all the movie names givn in the dataset

list_of_all_titels = movie['title'].tolist()
print(list_of_all_titels)
```

In [ ]:

```
# finding the close match for the movie name given by the user
find_close_match = difflib.get_close_matches(movie_name, list_of_all_titels
print(find_close_match)
```

In [ ]: