In [1]:

```
1 import pandas as pd
2 import numpy as np
3 from sklearn.feature_extraction.text import TfidfVectorizer #for each
4 from sklearn.metrics.pairwise import cosine_similarity
5
```

In [2]:

```
df=pd.read_csv("movies1.csv")
df
```

Out[2]:

	index	budget	genres	homepage	id
0	0	237000000	Action Adventure Fantasy Science Fiction	http://www.avatarmovie.com/	19995
1	1	300000000	Adventure Fantasy Action	http://disney.go.com/disneypictures/pirates/	285
2	2	245000000	Action Adventure Crime	http://www.sonypictures.com/movies/spectre/	206647
3	3	250000000	Action Crime Drama Thriller	http://www.thedarkknightrises.com/	49026
4	4	260000000	Action Adventure Science Fiction	http://movies.disney.com/john-carter	49529
...	...	...	...	...	...
4798	4798	220000	Action Crime Thriller	NaN	9367
4799	4799	9000	Comedy Romance	NaN	72766
4800	4800	0	Comedy Drama Romance TV Movie	http://www.hallmarkchannel.com/signedsealeddel...	231617
4801	4801	0	NaN	http://shanghaicalling.com/	126186

	index	budget	genres	homepage	id
4802	4802	0	Documentary	NaN	25975

4803 rows × 24 columns

In [3]:
```
1 df.columns
```

Out[3]:
```
Index(['index', 'budget', 'genres', 'homepage', 'id', 'keywords',
       'original_language', 'original_title', 'overview', 'popularity',
       'production_companies', 'production_countries', 'release_date',
       'revenue', 'runtime', 'spoken_languages', 'status', 'tagline', 't
itle',
       'vote_average', 'vote_count', 'cast', 'crew', 'director'],
      dtype='object')
```

In [4]:
```
1 df.shape
```

Out[4]:
```
(4803, 24)
```

In [5]:
```
1 features=['genres','keywords','original_language','title','cast','dir
2 for feature in features:
3     df[feature]=df[feature].fillna('') #filling missing values wuth s
4
```

In [6]:
```
1 def combined_features(row):
2     return row['title']+","+row['genres']+","+row['keywords']+",
3 "+row['original_language']+","+row['cast']+","+row['director']
4 df['combined_features'] = df.apply(combined_features,axis=1)
5 #column wise axis=1 dataframe same columns merge
6 df['combined_features'] #one more column with all those combined featu
7
```

Out[6]:
```
0       Avatar,Action Adventure Fantasy Science Fictio...
1       Pirates of the Caribbean: At World's End,Adven...
2       Spectre,Action Adventure Crime,spy based on no...
3       The Dark Knight Rises,Action Crime Drama Thril...
4       John Carter,Action Adventure Science Fiction,b...
                              ...
4798    El Mariachi,Action Crime Thriller,united state...
4799    Newlyweds,Comedy Romance,,en,Edward Burns Kerr...
4800    Signed, Sealed, Delivered,Comedy Drama Romance...
4801    Shanghai Calling,,,en,Daniel Henney Eliza Coup...
4802    My Date with Drew,Documentary,obsession camcor...
Name: combined_features, Length: 4803, dtype: object
```

In [7]:
```
tfid=TfidfVectorizer()
tfidv=tfid.fit_transform(df['combined_features']) #text word vector
tfidv.toarray()

```

Out[7]:
```
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.]])
```

In [8]:
```
tfidv.shape
```

Out[8]:
```
(4803, 17502)
```

In [9]:
```
cosine_sim=cosine_similarity(tfidv)
cosine_sim #compare the vectors

```

Out[9]:
```
array([[1.00000000e+00, 2.48272960e-02, 4.22228416e-02, ...,
        1.12200797e-03, 1.34747833e-03, 1.11917020e-03],
       [2.48272960e-02, 1.00000000e+00, 1.42630716e-02, ...,
        4.07129084e-02, 1.11032424e-03, 9.22197982e-04],
       [4.22228416e-02, 1.42630716e-02, 1.00000000e+00, ...,
        1.15436886e-03, 5.90455012e-02, 1.15144924e-03],
       ...,
       [1.12200797e-03, 4.07129084e-02, 1.15436886e-03, ...,
        1.00000000e+00, 1.17574481e-03, 5.83911241e-02],
       [1.34747833e-03, 1.11032424e-03, 5.90455012e-02, ...,
        1.17574481e-03, 1.00000000e+00, 1.17277112e-03],
       [1.11917020e-03, 9.22197982e-04, 1.15144924e-03, ...,
        5.83911241e-02, 1.17277112e-03, 1.00000000e+00]])
```

In [10]:
```
cosine_sim.shape
```

Out[10]:
```
(4803, 4803)
```

In [11]:
```
movie=input("Enter movie name: ")
def get_index(mn):
    return df[df.title==mn].index[0]
mi=get_index(movie)
mi

```

```
Enter movie name: Avatar
```

Out[11]:
```
0
```

In [12]:
```
sm=list(enumerate(cosine_sim[mi])) #0 se aage badhenge
print(sm)

```

```
[(0, 1.0000000000000002), (1, 0.024827295986620684), (2, 0.042222841569151265), (3, 0.008160607303690774), (4, 0.10625865464875475), (5, 0.06782484284827672), (6, 0.0013395252359036881), (7, 0.03695123127145235), (8, 0.022342422412007637), (9, 0.027496115325422923), (10, 0.06727359460847546), (11, 0.015687532413989805), (12, 0.0268928002342309), (13, 0.026225825804090122), (14, 0.05361238394260148), (15, 0.020132122146436357), (16, 0.03874126084538814), (17, 0.027494314158529494), (18, 0.06237344854351263), (19, 0.027289742758032448), (20, 0.028575460084002833), (21, 0.015171695489496225), (22, 0.02126049131775769), (23, 0.04444890691031617), (24, 0.015793538029016164), (25, 0.04646807786430685), (26, 0.08523979691334363), (27, 0.03667482299340698), (28, 0.03904258280160722), (29, 0.042527070873668735), (30, 0.06768700996200248), (31, 0.05857713702365093), (32, 0.03158621013232506), (33, 0.03771431221443131), (34, 0.0013210362060361365), (35, 0.03829621557186802), (36, 0.03440671321126578), (37, 0.09007853975155278), (38, 0.02876247070167063), (39, 0.03543766832556243), (40, 0.010817383893253206), (41, 0.04043087219409347), (42, 0.001323843755365448), (43, 0.11020562994807107), (44, 0.09848714935864478), (45, 0.056294011657772404), (46, 0.09876911686079831), (47, 0.19866721509827104), (48, 0.020442220239308525
```

In [13]:
```
sorted_sm=sorted(sm,key=lambda x:x[1],reverse=True)
#sort ACCORDING TO COSINE SIMILARITY AND NOT INDEX (for similar movies
print(sorted_sm)

```

```
[(0, 1.0000000000000002), (94, 0.260404789255364), (2403, 0.24376564738125260), (3158, 0.23803624917955862), (1053, 0.20194749885992924), (47, 0.19866721509827104), (56, 0.19712445427856712), (2696, 0.1897795878727181), (1951, 0.18881613734762134), (838, 0.18663143698859147), (239, 0.18496148339313892), (461, 0.18368439032772652), (661, 0.1814497357860887), (3730, 0.17715289892247293), (4593, 0.1768467770260661), (1354, 0.17541667725979684), (812, 0.17150770465155143), (2198, 0.16706774901811014), (643, 0.16288809609689386), (2229, 0.15723227093502493), (278, 0.15319368015627702), (206, 0.15022386032411839), (4332, 0.145583808400141), (1922, 0.14458714692336305), (1531, 0.14290969941766685), (1473, 0.14198913043777192), (1650, 0.13517185371456344), (3105, 0.13307216352776288), (1275, 0.12281233405428948), (3208, 0.12082272144275255), (2060, 0.1207450086089577), (740, 0.12010058516647538), (855, 0.11751743353400285), (1245, 0.11692189572497795), (541, 0.11656281952883685), (770, 0.11468259294644012), (539, 0.11437988050633119), (184, 0.1138893640839234), (1804, 0.113673027588991), (3202, 0.11339374463845599), (1099, 0.11242295583034376), (4108, 0.11206775434844227), (3184, 0.11067379999544863), (43, 0.11020562994807107), (1873, 0.10953786850819162), (1759, 0.1089342346486678), (305, 0.10867681827914595), (3534,
```

In [14]:

```
def get_info(index):
    return df[df.index==index]['title'].values[0]+":
"+ df[df.index==index]['cast'].values[0]
i=0
for movie in sorted_sm:
    print(get_info(movie[0]))
    i=i+1 #FOR NEXT RECOMMENDATION
    if i>10:
        break

```

```
Avatar: Sam Worthington Zoe Saldana Sigourney Weaver Stephen Lang Michel
le Rodriguez
Guardians of the Galaxy: Chris Pratt Zoe Saldana Dave Bautista Vin Diese
l Bradley Cooper
Aliens: Sigourney Weaver Michael Biehn James Remar Paul Reiser Lance Hen
riksen
Alien: Tom Skerritt Sigourney Weaver Veronica Cartwright Harry Dean Stan
ton John Hurt
Galaxy Quest: Tim Allen Sigourney Weaver Alan Rickman Tony Shalhoub Sam
Rockwell
Star Trek Into Darkness: Chris Pine Zachary Quinto Zoe Saldana Karl Urba
n Simon Pegg
Star Trek Beyond: Chris Pine Zachary Quinto Karl Urban Simon Pegg Zoe Sa
ldana
Jason X: Kane Hodder Lexa Doig Chuck Campbell Lisa Ryder David Cronenber
g
Space Dogs: Anna Bolshova Evgeny Mironov Sergey Garmash Aleksandr Bashir
ov Elena Yakovleva
Alien³: Sigourney Weaver Charles S. Dutton Charles Dance Pete Postlethwa
ite Ralph Brown
Gravity: Sandra Bullock George Clooney Ed Harris Orto Ignatiussen Phaldu
t Sharma
```