```
import pandas as pd
d=pd.read_csv('https://github.com/YBI-Foundation/Dataset/raw/main/Movies%20Recommendation.csv
d.head()
```

	Movie_ID	Movie_Title	Movie_Genre	Movie_Language	Movie_Budget	Movie_Popularity
0	1	Four Rooms	Crime Comedy	en	4000000	22.876230
1	2	Star Wars	Adventure Action Science Fiction	en	11000000	126.393695
2	3	Finding Nemo	Animation Family	en	94000000	85.688789
3	4	Forrest Gump	Comedy Drama Romance	en	55000000	138.133331
4	5	American Beauty	Drama	en	15000000	80.878605

5 rows × 21 columns

```
d.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4760 entries, 0 to 4759
Data columns (total 21 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Movie_ID                   4760 non-null   int64
```

```
 1   Movie_Title               4760 non-null   object
 2   Movie_Genre               4760 non-null   object
 3   Movie_Language            4760 non-null   object
 4   Movie_Budget              4760 non-null   int64
 5   Movie_Popularity          4760 non-null   float64
 6   Movie_Release_Date        4760 non-null   object
 7   Movie_Revenue             4760 non-null   int64
 8   Movie_Runtime             4758 non-null   float64
 9   Movie_Vote                4760 non-null   float64
 10  Movie_Vote_Count          4760 non-null   int64
 11  Movie_Homepage            1699 non-null   object
 12  Movie_Keywords            4373 non-null   object
 13  Movie_Overview            4757 non-null   object
 14  Movie_Production_House    4760 non-null   object
 15  Movie_Production_Country  4760 non-null   object
 16  Movie_Spoken_Language     4760 non-null   object
 17  Movie_Tagline             3942 non-null   object
 18  Movie_Cast                4733 non-null   object
 19  Movie_Crew                4760 non-null   object
 20  Movie_Director            4738 non-null   object
dtypes: float64(3), int64(4), object(14)
memory usage: 781.1+ KB
```

```
d.shape
```

```
(4760, 21)
```

```
d.columns
```

```
Index(['Movie_ID', 'Movie_Title', 'Movie_Genre', 'Movie_Language',
       'Movie_Budget', 'Movie_Popularity', 'Movie_Release_Date',
       'Movie_Revenue', 'Movie_Runtime', 'Movie_Vote', 'Movie_Vote_Count',
       'Movie_Homepage', 'Movie_Keywords', 'Movie_Overview',
       'Movie_Production_House', 'Movie_Production_Country',
       'Movie_Spoken_Language', 'Movie_Tagline', 'Movie_Cast', 'Movie_Crew',
       'Movie_Director'],
      dtype='object')
```

Get Feature Selection

```
d_f=d[['Movie_Genre','Movie_Keywords','Movie_Tagline','Movie_Cast','Movie_Director']].fillna(
```

```
d_f.shape
```

```
(4760, 5)
```

```
d_f
```

	Movie_Genre	Movie_Keywords	Movie_Tagline	Movie_Cast	Movie_Director
0	Crime Comedy	hotel new year's eve witch bet hotel room	Twelve outrageous guests. Four scandalous requ...	Tim Roth Antonio Banderas Jennifer Beals Madon...	Allison Anders
1	Adventure Action Science Fiction	android galaxy hermit death star lightsaber	A long time ago in a galaxy far, far away...	Mark Hamill Harrison Ford Carrie Fisher Peter ...	George Lucas
2	Animation Family	father son relationship harbor underwater fish...	There are 3.7 trillion fish in the ocean, they...	Albert Brooks Ellen DeGeneres Alexander Gould ...	Andrew Stanton
3	Comedy Drama Romance	vietnam veteran hippie mentally disabled runni...	The world will never be the same, once you've ...	Tom Hanks Robin Wright Gary Sinise Mykelti Wil...	Robert Zemeckis
4	Drama	male nudity female nudity adultery midlife cri...	Look closer.	Kevin Spacey Annette Bening Thora Birch Wes Be...	Sam Mendes

```
x=d_f['Movie_Genre']+' '+d_f['Movie_Keywords']+' '+d_f['Movie_Tagline']+' '+d_f['Movie_Cast']
```

```
x
```

```
0       Crime Comedy hotel new year's eve witch bet ho...
1       Adventure Action Science Fiction android galax...
2       Animation Family father son relationship harbo...
3       Comedy Drama Romance vietnam veteran hippie me...
4       Drama male nudity female nudity adultery midli...
                              ...
4755    Horror  The hot spot where Satan's waitin'. Li...
4756    Comedy Family Drama  It’s better to stand out ...
4757    Thriller Drama christian film sex trafficking ...
4758                                              Family
4759    Documentary music actors legendary perfomer cl...
Length: 4760, dtype: object
```

```
x.shape
```

```
(4760,)
```

Get Feature Text conversion to Tokens

```
from sklearn.feature_extraction.text import TfidfVectorizer
```

```
t=TfidfVectorizer()
```

```
x=t.fit_transform(x)
```

```
x.shape
```

```
(4760, 20663)
```

```
print(x)
```

```
  (0, 655)      0.16262957145774184
  (0, 18431)    0.1962711409320156
  (0, 11694)    0.14164337533513574
  (0, 11447)    0.165264014462748
  (0, 1580)     0.18076757769738178
  (0, 9423)     0.09798087551420058
  (0, 1359)     0.14164337533513574
  (0, 791)      0.13496700839177703
  (0, 15749)    0.1426376240765807
  (0, 18370)    0.11240630649274984
  (0, 10918)    0.08630813208986426
  (0, 13470)    0.062232628720122286
  (0, 20032)    0.17577654512129037
  (0, 6638)     0.08566211133363479
  (0, 19985)    0.10643704814029137
  (0, 9505)     0.13290090230347504
  (0, 18206)    0.09757953526456405
  (0, 13559)    0.07246213547989389
  (0, 4620)     0.11982714282573949
  (0, 6499)     0.11404364072001853
  (0, 8506)     0.19736418022707264
  (0, 8998)     0.14681661831949364
  (0, 1700)     0.1962711409320156
  (0, 11113)    0.1544047030004985
  (0, 13561)    0.08227116414935932
  :     :
  (4757, 13230) 0.1917059258424139
  (4757, 6453)  0.1278233416956338
  (4757, 3498)  0.17043427365383906
  (4757, 10211) 0.24070268624796623
  (4757, 8337)  0.17486057368807015
  (4757, 10100) 0.18223818201928874
  (4757, 18314) 0.07832581222100125
  (4757, 18398) 0.0998144365796161
  (4757, 9206)  0.117467101278103
  (4757, 13168) 0.1488517539356107
  (4757, 5315)  0.05807442979491625
  (4758, 6194)  1.0
  (4759, 13584) 0.34167984405714247
  (4759, 16871) 0.34167984405714247
  (4759, 14123) 0.34167984405714247
  (4759, 212)   0.3258920132175228
  (4759, 10738) 0.3060016986743309
  (4759, 3603)  0.2831147072916264
```

```
  (4759, 8590)  0.26591080832993275
  (4759, 13011) 0.3146903689708659
  (4759, 6746)  0.2179341128724163
  (4759, 5146)  0.1842532043388515
  (4759, 18451) 0.2133479262794535
  (4759, 1693)  0.2179341128724163
  (4759, 12912) 0.15991356654874977
```

Get Similarity Score Using Cosine Similarity

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

```
ss=cosine_similarity(x)
```

```
ss
```

```
array([[1.        , 0.01337921, 0.03525659, ..., 0.        , 0.        ,
        0.        ],
       [0.01337921, 1.        , 0.0079559 , ..., 0.        , 0.        ,
        0.        ],
       [0.03525659, 0.0079559 , 1.        , ..., 0.        , 0.07948776,
        0.        ],
       ...,
       [0.        , 0.        , 0.        , ..., 1.        , 0.        ,
        0.        ],
       [0.        , 0.        , 0.07948776, ..., 0.        , 1.        ,
        0.        ],
       [0.        , 0.        , 0.        , ..., 0.        , 0.        ,
        1.        ]])
```

```
ss.shape
```

```
(4760, 4760)
```

Get Movie name as InputFrom User and Validate for Closest Spelling

```
F=input('Enter Your Favourite movie name : ')
```

```
Enter Your Favourite movie name : avtaar
```

```
A=d['Movie_Title'].tolist()
```

```
import difflib
```

```
M=difflib.get_close_matches(F,A)
```

Double-click (or enter) to edit

```
close_match=M[0]
print(close_match)
```

```
Avatar
```

```
i=d[d.Movie_Title==close_match]['Movie_ID'].values[0]
print(i)
```

```
2692
```

```
R=list(enumerate(ss[i]))
print(R)
```

```
[(0, 0.009758882373553546), (1, 0.0), (2, 0.0), (3, 0.007895011959041548), (4, 0.002646:
```

```
len(R)
```

```
4760
```

Get All Movie Sort Based On Recommendation Score wrt Favourite Movie

```
s=sorted(R,key=lambda x:x[1],reverse=True)
print(s)
```

```
[(2692, 1.0), (3779, 0.10186325605202706), (2903, 0.09989313625883267), (4614, 0.094094
```

```
print('Top 30 movies suggested for you: \n')
i=1
for movie in s:
  index=movie[0]
  t=d[d.index==index]['Movie_Title'].values[0]
  if(i<31):
      print(i,'.',t)
      i+=1
```

```
Top 30 movies suggested for you:

1 . Niagara
2 . My Week with Marilyn
3 . Harry Brown
4 . The Curse of Downers Grove
5 . The Boy Next Door
```

```
6 . Back to the Future
7 . Welcome to the Sticks
8 . The Juror
9 . Some Like It Hot
10 . The Kentucky Fried Movie
11 . Enough
12 . Eye for an Eye
13 . Alice Through the Looking Glass
14 . Superman III
15 . Duel in the Sun
16 . Premium Rush
17 . The Misfits
18 . Small Soldiers
19 . Camping Sauvage
20 . All That Jazz
21 . Beyond the Black Rainbow
22 . The Raid
23 . Tora! Tora! Tora!
24 . Brokeback Mountain
25 . To Kill a Mockingbird
26 . Edge of Darkness
27 . World Trade Center
28 . The Dark Knight Rises
29 . Out of Time
30 . Source Code
```

✓ 2s completed at 7:50 PM