In [1]:

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

In [3]:

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

In [5]:

```
movies.head()
```

Out[5]:

	budget	genres	homepage	id	keywords	original_language	original_title	over
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Avatar	I cent parap Mar
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	en	Pirates of the Caribbean: At World's End	Ca Barb bel dead
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	en	Spectre	A cı mes B send
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853,...	en	The Dark Knight Rises	Follo the (of D Att Ha
4	260000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id":...	en	John Carter	Carte w fc m

```
In [7]: credits.head()
```

Out[7]:

	movie_id	title	cast	crew
0	19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...
1	285	Pirates of the Caribbean: At World's End	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"credit_id": "52fe4232c3a36847f800b579", "de...
2	206647	Spectre	[{"cast_id": 1, "character": "James Bond", "cr...	[{"credit_id": "54805967c3a36829b5002c41", "de...
3	49026	The Dark Knight Rises	[{"cast_id": 2, "character": "Bruce Wayne / Ba...	[{"credit_id": "52fe4781c3a36847f81398c3", "de...
4	49529	John Carter	[{"cast_id": 5, "character": "John Carter", "c...	[{"credit_id": "52fe479ac3a36847f813eaa3", "de...

```
In [9]: credits.head(1)
```

Out[9]:

	movie_id	title	cast	crew
0	19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...

```
In [11]: movies =movies.merge(credits,on="title")
```

```
In [13]: movies.head(1)
```

Out[13]:

	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577

1 rows × 23 columns

In [15]:
```python
movies = movies[["movie_id","title","overview","genres","keywords","cast","crew"]]
```

In [17]:
```python
movies.head(1)
```

Out[17]:

	movie_id	title	overview	genres	keywords	cast	crew
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	[{"id": 1463, "name": "culture clash"}, {"id":...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...

In [19]:
```python
movies.isnull().sum()
```

Out[19]:
```
movie_id    0
title       0
overview    3
genres      0
keywords    0
cast        0
crew        0
dtype: int64
```

In [21]:
```python
movies.dropna(inplace=True)
```

In [23]:
```python
#check for duplicated rows
movies.duplicated().sum()
```

```
Out[23]: 0
```

```
In [25]: movies.iloc[0].genres
```

```
Out[25]: '[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"}, {"id": 878, "name":
         "Science Fiction"}]'
```

```
In [27]: def convert(obj):
             L=[]
             for i in ast.literal_eval(obj):
                 L.append(i['name'])
             return L
```

```
In [29]: import ast
```

```
In [31]: movies['genres'] =movies['genres'].apply(convert)
```

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

Out[32]:

	movie_id	title	overview	genres	keywords	cast	crew
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...	[Action, Adventure, Fantasy, Science Fiction]	[{"id": 1463, "name": "culture clash"}, {"id":...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...

```
In [35]: movies.iloc[0].keywords
```

```
Out[35]: '[{"id": 1463, "name": "culture clash"}, {"id": 2964, "name": "future"}, {"id": 3386, "name": "space war"}, {"id": 3
         388, "name": "space colony"}, {"id": 3679, "name": "society"}, {"id": 3801, "name": "space travel"}, {"id": 9685, "n
         ame": "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": "cgi"}, {"id": 11399,
         "name": "marine"}, {"id": 13065, "name": "soldier"}, {"id": 14643, "name": "battle"}, {"id": 14720, "name": "love af
         fair"}, {"id": 165431, "name": "anti war"}, {"id": 193554, "name": "power relations"}, {"id": 206690, "name": "mind
         and soul"}, {"id": 209714, "name": "3d"}]'
```

```
In [37]: def convertt(obj):
             L=[]
             for i in ast.literal_eval(obj):
```

```
        L.append(i["name"])
    return L
```

In [39]:

```
movies["keywords"] =movies["keywords"].apply(convertt)
```

In [40]:

```
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 [43]:

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

In [44]:

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

Out[44]:

	movie_id	title	overview	genres	keywords	cast	crew
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[{"credit_id": "52fe48009251416c750aca23", "de...

In [45]:

```
def convertt1(obj):
    L=[]
    for i in ast.literal_eval(obj):
        if i["job"]=="Director":
            L.append(i["name"])
            break
    return L
```

In [46]:

```
movies["crew"]=movies["crew"].apply(convertt1)
```

In [47]:

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

Out[47]:

	movie_id	title	overview	genres	keywords	cast	crew
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[James Cameron]

In [48]:
```
movies['overview']=movies['overview'].apply(lambda x:x.split())
```

In [49]:
```
movies.head(1)
```

Out[49]:

	movie_id	title	overview	genres	keywords	cast	crew
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[James Cameron]

In [50]:
```
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["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 [51]:
```
movies.head(1)
```

Out[51]:

	movie_id	title	overview	genres	keywords	cast	crew
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCameron]

In [52]:
```
movies['tags'] = movies['overview']+movies['genres']+movies['keywords']+movies['crew']
```

In [53]:
```
movies.head(1)
```

Out[53]:

	movie_id	title	overview	genres	keywords	cast	crew	tags
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[JamesCameron]	[In, the, 22nd, century,, a, paraplegic, Marin...

In [54]:

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

In [55]:

```
new_df.head(1)
```

Out[55]:

	movie_id	title	tags
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...

In [56]:

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

```
C:\Users\parth\AppData\Local\Temp\ipykernel_15720\3089450492.py: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
  new_df['tags'] = new_df['tags'].apply(lambda x:" ".join(x))
```

In [57]:

```
new_df.head(1)
```

Out[57]:

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

In [58]:

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

```
C:\Users\parth\AppData\Local\Temp\ipykernel_15720\3214958533.py: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
  new_df['tags'] = new_df['tags'].apply(lambda x:x.lower())
```

In [59]:

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

In [77]:

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

In [79]:

```
 new_df["tags"].apply(stem)
```

Out[79]:

```
0       in the 22nd century, a parapleg marin is dispa...
1       captain barbossa, long believ to be dead, ha c...
2       a cryptic messag from bond’ past send him on a...
3       follow the death of district attorney harvey d...
4       john carter is a war-weary, former militari ca...
                              ...
4804    el mariachi just want to play hi guitar and ca...
4805    a newlyw couple' honeymoon is upend by the arr...
4806    "signed, sealed, delivered" introduc a dedic q...
4807    when ambiti new york attorney sam is sent to s...
4808    ever sinc the second grade when he first saw h...
Name: tags, Length: 4806, dtype: object
```

In [81]:

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

```
C:\Users\parth\AppData\Local\Temp\ipykernel_15720\447794818.py: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
  new_df["tags"]= new_df["tags"].apply(stem)
```

In [83]:

```
new_df.head(1)
```

Out[83]:

	movie_id	title	tags
0	19995	Avatar	in the 22nd century, a parapleg marin is dispa...

In [85]:

```
#converting the text into vectors (text vectorization) using bag of words
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features = 5000 , stop_words = "english")
```

In [87]:

```
new_df.head(1)
```

Out[87]:

	movie_id	title	tags
0	19995	Avatar	in the 22nd century, a parapleg marin is dispa...

In [91]:

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

In [93]:

```
vectors[0]
```

Out[93]:

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

In [95]:

```
features=cv.get_feature_names_out()
```

In [97]:

```
print(features)
```

```
['000' '007' '10' ... 'zombies' 'zone' 'zoo']
```

In [99]:

```
# for the vector we are not using any euclidean distance we are using cosine distance
#the lower the angle between the 2 vector they are more similar to each other
```

In [101...

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

In [109...

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

In [117...

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

Out[117...

```
[(1216, 0.2849014411490948),
 (539, 0.26940795304016235),
 (3730, 0.2676516895156553),
 (507, 0.2649064714130087),
 (582, 0.25182770057259657)]
```

In [177...

```
def recommend(movie):
    movie_index=new_df[new_df["title"]== movie].index[0]
    distances = similarity[movie_index]
    movies_list = sorted(list(enumerate(distances)),reverse=True,key=lambda x:x[1])[1:6]

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

In [179...

```
recommend('Spectre')
```

```
Skyfall
Quantum of Solace
Never Say Never Again
From Russia with Love
Octopussy
```

In [149...

```
import pickle
```

In [157...

```
pickle.dump(new_df,open('movies.pkl','wb'))
```

In [159...

```
new_df['title'].values
```

Out[159...

```
array(['Avatar', "Pirates of the Caribbean: At World's End", 'Spectre',
       ..., 'Signed, Sealed, Delivered', 'Shanghai Calling',
       'My Date with Drew'], dtype=object)
```

In [161...

```
new_df.to_dict
```

Out[161…

```
<bound method DataFrame.to_dict of       movie_id                                     title  \
0        19995                                    Avatar
1          285  Pirates of the Caribbean: At World's End
2       206647                                   Spectre
3        49026                     The Dark Knight Rises
4        49529                               John Carter
...        ...                                       ...
4804      9367                               El Mariachi
4805     72766                                 Newlyweds
4806    231617                 Signed, Sealed, Delivered
4807    126186                          Shanghai Calling
4808     25975                         My Date with Drew

                                                   tags
0     in the 22nd century, a parapleg marin is dispa...
1     captain barbossa, long believ to be dead, ha c...
2     a cryptic messag from bond’ past send him on a...
3     follow the death of district attorney harvey d...
4     john carter is a war-weary, former militari ca...
...                                                 ...
4804  el mariachi just want to play hi guitar and ca...
4805  a newlyw couple' honeymoon is upend by the arr...
4806  "signed, sealed, delivered" introduc a dedic q...
4807  when ambiti new york attorney sam is sent to s...
4808  ever sinc the second grade when he first saw h...

[4806 rows x 3 columns]>
```

In [163…

```
pickle.dump(new_df.to_dict(),open('movie_dict.pkl','wb'))
```

In [165…

```
pickle.dump(similarity,open('similarity.pkl','wb'))
```

In [ ]: