```
movie_index = get_index_from_title(movie_user_likes)
similar_movies = list(enumerate(cosine_sim[movie_index]))

sorted_similar_movies = sorted(similar_movies,key= lambda x:[1],reverse=True)
```

```
<ipython-input-27-6dda2dc5f249>:6: DtypeWarning: Columns (0,1,4,9,13,14,19,26
  df = pd.read_csv("/content/movie_dataset.csv")
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-27-6dda2dc5f249> in <cell line: 39>()
     37
     38 movie_index = get_index_from_title(movie_user_likes)
---> 39 similar_movies = list(enumerate(cosine_sim[movie_index]))
     40
     41 sorted_similar_movies = sorted(similar_movies,key= lambda x:
[1],reverse=True)

IndexError: only integers, slices (`:`), ellipsis (`...`), numpy.newaxis
(`None`) and integer or boolean arrays are valid indices
```

Next steps: Explain error

```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity

df = pd.read_csv("/content/movie_dataset.csv")


def get_title_from_index(index):
    return df[df.index == index]["title"].values[0]


def get_index_from_title(title):
    return df[df.title == title]["index"].values[0]
```

7. kamaravichow/movie-recommendation-system-python subject to Apache License 2.0

Rate this answer

```
features = ["keywords", "cast", "genres", "director"]

for feature in features:
    df[feature] = df[feature].fillna("")


def combine_features(row):
    try:
        return row["keywords"] + " " + row["cast"] + " " + row["genres"] + " " + r
    except:
        print("Error", row)


df["combined_features"] = df.apply(combine_features, axis=1)

cv = CountVectorizer()
count_matrix = cv.fit_transform(df["combined_features"])

cosine_sim = cosine_similarity(count_matrix)
movie_user_likes = "Spectre"

movie_index = int(get_index_from_title(movie_user_likes))
similar_movies = list(enumerate(cosine_sim[movie_index]))

sorted_similar_movies = sorted(similar_movies,key= lambda x:[1],reverse=True)

i=0
for movie in sorted_similar_movies:
  print(get_title_from_index(movie[0]))
  i=i+1
  if i>50:
    break

     <ipython-input-29-1d2ec9e01146>:6: DtypeWarning: Columns (0,1,4,9,13,14,19,2
       df = pd.read_csv("/content/movie_dataset.csv")
     Avatar
```

```
Pirates of the Caribbean: At World's End
Spectre
The Dark Knight Rises
John Carter
Spider-Man 3
Tangled
Avengers: Age of Ultron
Harry Potter and the Half-Blood Prince
Batman v Superman: Dawn of Justice
Superman Returns
Quantum of Solace
Pirates of the Caribbean: Dead Man's Chest
The Lone Ranger
Man of Steel
The Chronicles of Narnia: Prince Caspian
The Avengers
Pirates of the Caribbean: On Stranger Tides
Men in Black 3
The Hobbit: The Battle of the Five Armies
The Amazing Spider-Man
Robin Hood
The Hobbit: The Desolation of Smaug
The Golden Compass
King Kong
Titanic
Captain America: Civil War
Battleship
Jurassic World
 'department': 'Sound'
Skyfall
Spider-Man 2
Iron Man 3
Alice in Wonderland
X-Men: The Last Stand
Monsters University
Transformers: Revenge of the Fallen
Transformers: Age of Extinction
Oz: The Great and Powerful
The Amazing Spider-Man 2
TRON: Legacy
Cars 2
```

```
Green Lantern
Toy Story 3
Terminator Salvation
Furious 7
World War Z
X-Men: Days of Future Past
Star Trek Into Darkness
Jack the Giant Slayer
The Great Gatsby
```