In [ ]:

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

In [ ]:

```
warnings.filterwarnings('ignore')
```

In [ ]:

```
columns_name=['user_id','item_id','rating','timestamp']
df=pd.read_csv('u.data',sep="\t",names=columns_name)
```

In [ ]:

```
df.head()
```

In [ ]:

```
df.shape
```

In [ ]:

```
df['user_id']
```

In [ ]:

```
df['user_id'].nunique()
```

In [ ]:

```
df['item_id'].nunique()
```

In [ ]:

```
movies_title=pd.read_csv('u.item',encoding = "ISO-8859-1",sep="\|",header=None)
```

In [ ]:

```
movies_title.shape
```

In [ ]:

```
movies_titles=movies_title[[0,1]]
movies_titles.columns=["item_id","title"]
movies_titles.head()
```

In [ ]:

```
df=pd.merge(df,movies_titles,on="item_id")
```

In [ ]:

```
df
```

In [ ]:

```
df.tail()
```

In [ ]:

```
ratings=pd.DataFrame(df.groupby("title").mean()['rating'])
```

In [ ]:

```
ratings.head()
```

In [ ]:

```
ratings['num of ratings']=pd.DataFrame(df.groupby("title").count()['rating'])
```

Create the recommendar System

In [ ]:

```
df.head()
```

In [ ]:

```
moviemat=df.pivot_table(index="user_id",columns="title",values="rating")
```

In [ ]:

```
moviemat.head()
```

In [ ]:

```
Young_Frankenstein_user_ratings=moviemat['Young Frankenstein (1974)']
```

In [ ]:

```
Young_Frankenstein_user_ratings.head(20)
```

In [ ]:

```
similar_to_Young_Frankenstein=moviemat.corrwith(Young_Frankenstein_user_ratings)
```

In [ ]:

```
similar_to_Young_Frankenstein
```

In [ ]:

```
corr_Young_Frankenstein=pd.DataFrame(similar_to_Young_Frankenstein,columns=['correlation'
])
```

In [ ]:

```
corr_Young_Frankenstein.dropna(inplace=True)
```

In [ ]:

```
corr_Young_Frankenstein
```

In [ ]:

```
corr_Young_Frankenstein.head()
```

In [ ]:

```
corr_Young_Frankenstein.sort_values('correlation',ascending=False).head(10)
```

In [ ]:

```
ratings
```

In [ ]:

```
corr_Young_Frankenstein=corr_Young_Frankenstein.join(ratings['num of ratings'])
```

In [ ]:

```
corr_Young_Frankenstein
```

In [ ]:

```
corr_Young_Frankenstein.head()
```

In [ ]:

```
corr_Young_Frankenstein[corr_Young_Frankenstein['num of ratings']>100].sort_values('correlation',ascending=False)
```

In [ ]:

```
def predict_movies(movie_name):
    movie_user_ratings=moviemat[movie_name]
    similar_to_movie=moviemat.corrwith(movie_user_ratings)
    corr_movie=pd.DataFrame(similar_to_movie,columns=['correlation'])
    corr_movie.dropna(inplace=True)
    corr_movie=corr_movie.join(ratings['num of ratings'])

    predictions=corr_movie[corr_movie['num of ratings']>100].sort_values('correlation',ascending=False)

    return predictions
```

In [ ]:

```
predict_my_movie=predict_movies("Young Frankenstein (1974)")
```

In [ ]:

```
predict_my_movie.head()
```

In [ ]:

```

```

In [ ]:

```

```