

In [2]:

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

Dataset

In [25]:

```
book=pd.read_csv('C:/Users/Hp/Downloads/book.csv',encoding='unicode_escape')
```

In [6]:

```
book
```

Out[6]:

	Unnamed: 0	User.ID	Book.Title	Book.Rating
0	1	276726	Classical Mythology	5
1	2	276729	Clara Callan	3
2	3	276729	Decision in Normandy	6
3	4	276736	Flu: The Story of the Great Influenza Pandemic...	8
4	5	276737	The Mummies of Urumchi	6
...	...	...	...	...
9995	9996	162121	American Fried: Adventures of a Happy Eater.	7
9996	9997	162121	Cannibal In Manhattan	9
9997	9998	162121	How to Flirt: A Practical Guide	7
9998	9999	162121	Twilight	8
9999	10000	162129	Kids Say the Darndest Things	6

10000 rows × 4 columns

# Performing EDA

In [7]:

```
book.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 4 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Unnamed: 0      10000 non-null  int64
1   User.ID         10000 non-null  int64
2   Book.Title      10000 non-null  object
3   Book.Rating     10000 non-null  int64
dtypes: int64(3), object(1)
memory usage: 312.6+ KB
```

In [9]:

```
book1=book.iloc[:,1:]
```

In [10]:

book1

Out[10]:

	User.ID	Book.Title	Book.Rating
0	276726	Classical Mythology	5
1	276729	Clara Callan	3
2	276729	Decision in Normandy	6
3	276736	Flu: The Story of the Great Influenza Pandemic...	8
4	276737	The Mummies of Urumchi	6
...	...	...	...
9995	162121	American Fried: Adventures of a Happy Eater.	7
9996	162121	Cannibal In Manhattan	9
9997	162121	How to Flirt: A Practical Guide	7
9998	162121	Twilight	8
9999	162129	Kids Say the Darndest Things	6

10000 rows × 3 columns

In [11]:

```
#rename column
book2=book1.rename({'User.ID':'UserId','Book.Title':'BookTitle','Book.Rating':'BookRating'})
```

In [12]:

```
book2
```

Out[12]:

	UserId	BookTitle	BookRating
0	276726	Classical Mythology	5
1	276729	Clara Callan	3
2	276729	Decision in Normandy	6
3	276736	Flu: The Story of the Great Influenza Pandemic...	8
4	276737	The Mummies of Urumchi	6
...	...	...	...
9995	162121	American Fried: Adventures of a Happy Eater.	7
9996	162121	Cannibal In Manhattan	9
9997	162121	How to Flirt: A Practical Guide	7
9998	162121	Twilight	8
9999	162129	Kids Say the Darndest Things	6

10000 rows × 3 columns

In [13]:

```
book3=book2.copy()
```

In [14]:

```
#duplicated rows  
book3[book3.duplicated()].shape
```

Out[14]:

(2, 3)

In [16]:

```
book3
```

Out[16]:

	Userld	BookTitle	BookRating
0	276726	Classical Mythology	5
1	276729	Clara Callan	3
2	276729	Decision in Normandy	6
3	276736	Flu: The Story of the Great Influenza Pandemic...	8
4	276737	The Mummies of Urumchi	6
...	...	...	...
9995	162121	American Fried: Adventures of a Happy Eater.	7
9996	162121	Cannibal In Manhattan	9
9997	162121	How to Flirt: A Practical Guide	7
9998	162121	Twilight	8
9999	162129	Kids Say the Darndest Things	6

10000 rows × 3 columns

In [17]:

```
#print duplicate data
book3[book3.duplicated()]
```

Out[17]:

	Userld	BookTitle	BookRating
5051	2152	Le nouveau soleil de Teur	7
7439	3757	The Magician's Tale	7

In [18]:

```
book3=book3.drop_duplicates()  
book3
```

Out[18]:

	Userld	BookTitle	BookRating
0	276726	Classical Mythology	5
1	276729	Clara Callan	3
2	276729	Decision in Normandy	6
3	276736	Flu: The Story of the Great Influenza Pandemic...	8
4	276737	The Mummies of Urumchi	6
...	...	...	...
9995	162121	American Fried: Adventures of a Happy Eater.	7
9996	162121	Cannibal In Manhattan	9
9997	162121	How to Flirt: A Practical Guide	7
9998	162121	Twilight	8
9999	162129	Kids Say the Darndest Things	6

9998 rows × 3 columns

In [19]:

```
#number of unique users in the dataset  
len(book3.UserId.unique())
```

Out[19]:

2182

In [20]:

```
len(book3.BookTitle.unique())
```

Out[20]:

9659

In [21]:

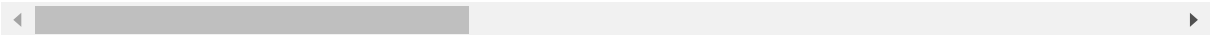
```
user_book=book3.pivot_table(index='UserId',
                             columns='BookTitle',
                             values='BookRating').reset_index(drop=True)

user_book
```

Out[21]:

BookTitle	Jason, Madison &mp	Stories;Merril;1985;McClelland &mp	Other	Repairing PC Drives &mp	'48	'O Au No Keia: Voices from Hawai'i's Mahu and Transgender Communities	...AND THE HORSE HE RODE IN ON : THE PEOPLE V. KENNETH STARR	A  M
0	NaN		NaN	NaN	NaN	NaN	NaN	
1	NaN		NaN	NaN	NaN	NaN	NaN	
2	NaN		NaN	NaN	NaN	NaN	NaN	
3	NaN		NaN	NaN	NaN	NaN	NaN	
4	NaN		NaN	NaN	NaN	NaN	NaN	
...	...		...	...	...	...	...	
2177	NaN		NaN	NaN	NaN	NaN	NaN	
2178	NaN		NaN	NaN	NaN	NaN	NaN	
2179	NaN		NaN	NaN	NaN	NaN	NaN	
2180	NaN		NaN	NaN	NaN	NaN	NaN	
2181	NaN		NaN	NaN	NaN	NaN	NaN	

2182 rows × 9659 columns



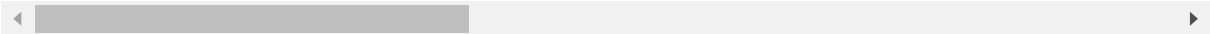
In [22]:

```
user_book.index=book3.UserId.unique()  
user_book
```

Out[22]:

BookTitle	Jason, Madison &amp;	Stories;Merril;1985;McClelland &amp;	Other Repairing PC Drives &amp;	'48	'O Au No Keia: Voices from Hawai'i's Mahu and Transgender Communities	...AND THE HORSE HE RODE IN ON : THE PEOPLE V. KENNETH STARR	A M
276726	NaN		NaN	NaN	NaN	NaN	
276729	NaN		NaN	NaN	NaN	NaN	
276736	NaN		NaN	NaN	NaN	NaN	
276737	NaN		NaN	NaN	NaN	NaN	
276744	NaN		NaN	NaN	NaN	NaN	
...	...		...	...	...	...	
162107	NaN		NaN	NaN	NaN	NaN	
162109	NaN		NaN	NaN	NaN	NaN	
162113	NaN		NaN	NaN	NaN	NaN	
162121	NaN		NaN	NaN	NaN	NaN	
162129	NaN		NaN	NaN	NaN	NaN	

2182 rows × 9659 columns



In [23]:

```
#Impute those NaNs with 0 values
user_book.fillna(0,inplace=True)
user_book
```

Out[23]:

BookTitle	Jason, Madison &mp	Stories;Merril;1985;McClelland &mp	Other Repairing PC Drives &mp	'48	'O Au No Keia: Voices from Hawai'i's Mahu and Transgender Communities	...AND THE HORSE HE RODE IN ON : THE PEOPLE V. KENNETH STARR	0 A I Mill
276726	0.0		0.0	0.0	0.0	0.0	
276729	0.0		0.0	0.0	0.0	0.0	
276736	0.0		0.0	0.0	0.0	0.0	
276737	0.0		0.0	0.0	0.0	0.0	
276744	0.0		0.0	0.0	0.0	0.0	
...	...		...	...	...	...	
162107	0.0		0.0	0.0	0.0	0.0	
162109	0.0		0.0	0.0	0.0	0.0	
162113	0.0		0.0	0.0	0.0	0.0	
162121	0.0		0.0	0.0	0.0	0.0	
162129	0.0		0.0	0.0	0.0	0.0	

2182 rows × 9659 columns

In [24]:

```
#Calculating Cosine Similarity between Users
from sklearn.metrics import pairwise_distances
from scipy.spatial.distance import cosine,correlation
```



In [26]:

```
user_sim= 1-pairwise_distances(user_book.values,metric='cosine')
user_sim
```

Out[26]:

```
array([[1., 0., 0., ..., 0., 0., 0.],
       [0., 1., 0., ..., 0., 0., 0.],
       [0., 0., 1., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 1., 0., 0.],
       [0., 0., 0., ..., 0., 1., 0.],
       [0., 0., 0., ..., 0., 0., 1.]])
```

In [27]:

```
#Store the results in a dataframe
user_sim_df=pd.DataFrame(user_sim)
```

In [28]:

```
#set the index and column names to userids
user_sim_df.index=book3.UserId.unique()
user_sim_df.columns=book3.UserId.unique()
```

In [29]:

```
user_sim_df.iloc[0:5,0:5]
```

Out[29]:

	276726	276729	276736	276737	276744
276726	1.0	0.0	0.0	0.0	0.0
276729	0.0	1.0	0.0	0.0	0.0
276736	0.0	0.0	1.0	0.0	0.0
276737	0.0	0.0	0.0	1.0	0.0
276744	0.0	0.0	0.0	0.0	1.0

In [30]:

```
np.fill_diagonal(user_sim, 0)
user_sim_df.iloc[0:5,0:5]
```

Out[30]:

	276726	276729	276736	276737	276744
276726	0.0	0.0	0.0	0.0	0.0
276729	0.0	0.0	0.0	0.0	0.0
276736	0.0	0.0	0.0	0.0	0.0
276737	0.0	0.0	0.0	0.0	0.0
276744	0.0	0.0	0.0	0.0	0.0

In [31]:

```
#Most similar users  
user_sim_df.idxmax(axis=1)[0:10]
```

Out[31]:

```
276726    276726  
276729    276726  
276736    276726  
276737    276726  
276744    276726  
276745    276726  
276747    276726  
276748    161677  
276751    276726  
276754    276726  
dtype: int64
```

In [ ]: