In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
from matplotlib import pyplot as plt
from sklearn.metrics import pairwise_distances
from scipy.spatial.distance import cosine,correlation
```

In [3]:

```
dt = pd.read_csv('book (1).csv')
dt
```

Out[3]:

	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

In [4]:

```
dt2 = dt.iloc[:,1:]
dt2
```

Out[4]:

	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 [5]:

```
dt2.sort_values(['User.ID'])
```

Out[5]:

	User.ID	Book.Title	Book.Rating
2401	8	Wings	5
2400	8	The Western way: A practical guide to the West...	5
2399	8	Ancient Celtic Romances	5
2402	8	Truckers	5
2405	8	The Art Of Celtia	7
...	...	...	...
2395	278854	La crónica del Perú (Crónicas de América)	7
2398	278854	Celtic Mythology (Library of the World's Myths...	8
2393	278854	A corrente de Trewis Scott	7
2394	278854	As valkírias	7
2397	278854	A Treasury of Irish Myth, Legend, and Folklore	6

10000 rows × 3 columns

In [7]:

```
len(dt2['User.ID'].unique())
```

Out[7]:

2182

In [8]:

```
len(dt2['Book.Title'].unique())
```

Out[8]:

9659

In [9]:

```
dt3 = dt2.pivot_table(index='User.ID',columns='Book.Title',values='Book.Rating').reset_inde
dt3
```

Out[9]:

Book.Title	Jason, Madison &amp	Other Stories;Merril;1985;McClelland &amp	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
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 [10]:

```
dt3.index = dt2['User.ID'].unique()
dt3
```

Out[10]:

Book.Title	Jason, Madison &amp	Other Stories;Merril;1985;McClelland &amp	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	NaN	
276729	NaN	NaN	NaN	NaN	NaN	NaN	
276736	NaN	NaN	NaN	NaN	NaN	NaN	
276737	NaN	NaN	NaN	NaN	NaN	NaN	
276744	NaN	NaN	NaN	NaN	NaN	NaN	
...	...	...	...	...	...	...	
162107	NaN	NaN	NaN	NaN	NaN	NaN	
162109	NaN	NaN	NaN	NaN	NaN	NaN	
162113	NaN	NaN	NaN	NaN	NaN	NaN	
162121	NaN	NaN	NaN	NaN	NaN	NaN	
162129	NaN	NaN	NaN	NaN	NaN	NaN	

2182 rows × 9659 columns

In [11]:

```
#Impute those NaNs with 0 values.
```

In [12]:

```
dt3.fillna(0, inplace=True)
dt3
```

Out[12]:

Book.Title	Jason, Madison &amp	Other Stories;Merril;1985;McClelland &amp	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	0 A Mil
276726	0.0	0.0	0.0	0.0	0.0	0.0	
276729	0.0	0.0	0.0	0.0	0.0	0.0	
276736	0.0	0.0	0.0	0.0	0.0	0.0	
276737	0.0	0.0	0.0	0.0	0.0	0.0	
276744	0.0	0.0	0.0	0.0	0.0	0.0	
...	...	...	...	...	...	...	
162107	0.0	0.0	0.0	0.0	0.0	0.0	
162109	0.0	0.0	0.0	0.0	0.0	0.0	
162113	0.0	0.0	0.0	0.0	0.0	0.0	
162121	0.0	0.0	0.0	0.0	0.0	0.0	
162129	0.0	0.0	0.0	0.0	0.0	0.0	

2182 rows × 9659 columns

In [13]:

```
#Calculating cosine similarity between users and array
```

In [14]:

```
user = 1-pairwise_distances(dt3.values,metric='cosine')
user
```

Out[14]:

```
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 [15]:

```
# In Dataframe
```

In [17]:

```
user2 = pd.DataFrame(user)
user2
```

Out[17]:

	0	1	2	3	4	5	6	7	8	9	...	2172	2173	2174	2175	2176	2177	2
0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
4	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
2177	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	1.0	
2178	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
2179	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
2180	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	
2181	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	

2182 rows × 2182 columns

In [18]:

```
user2.index = dt2['User.ID'].unique()
user2.columns = dt2['User.ID'].unique()
user2
```

Out[18]:

	276726	276729	276736	276737	276744	276745	276747	276748	276751	276754	..
276726	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
276729	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
276736	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
276737	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	..
276744	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	..
...	...	...	...	...	...	...	...	...	...	...	..
162107	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
162109	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
162113	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
162121	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
162129	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..

2182 rows × 2182 columns

In [19]:

```
# Nullifying diagonal values
```

In [20]:

```
np.fill_diagonal(user,0)
user2
```

Out[20]:

	276726	276729	276736	276737	276744	276745	276747	276748	276751	276754	..
276726	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
276729	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
276736	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
276737	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
276744	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
...	...	...	...	...	...	...	...	...	...	...	..
162107	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
162109	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
162113	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
162121	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..
162129	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	..

2182 rows × 2182 columns

In [23]:

```
user2.idxmax(axis=1)
```

Out[23]:

```
276726    276726
276729    276726
276736    276726
276737    276726
276744    276726
           ...
162107    276726
162109    276726
162113    161453
162121    276726
162129    276726
Length: 2182, dtype: int64
```

In [22]:

```
# extract the books which userId 276729 & 276726 have watched
```

In [24]:

```
dt2[(dt2['User.ID']==276729) | (dt2['User.ID']==276726)]
```

Out[24]:

	User.ID	Book.Title	Book.Rating
0	276726	Classical Mythology	5
1	276729	Clara Callan	3
2	276729	Decision in Normandy	6

In [25]:

```
user_1=dt2[(dt2['User.ID']==276729)]
user_2=dt2[(dt2['User.ID']==276726)]
```

In [26]:

```
user_1['Book.Title']
```

Out[26]:

```
1            Clara Callan
2    Decision in Normandy
Name: Book.Title, dtype: object
```

In [27]:

```
user_2['Book.Title']
```

Out[27]:

```
0    Classical Mythology
Name: Book.Title, dtype: object
```

In [28]:

```
pd.merge(user_1,user_2,on='Book.Title',how='outer')
```

Out[28]:

	User.ID_x	Book.Title	Book.Rating_x	User.ID_y	Book.Rating_y
0	276729.0	Clara Callan	3.0	NaN	NaN
1	276729.0	Decision in Normandy	6.0	NaN	NaN
2	NaN	Classical Mythology	NaN	276726.0	5.0

In [ ]:

```

```