

```
In [4]: import pandas as pd
import numpy as np
from sklearn.metrics import pairwise_distances
from scipy.spatial.distance import cosine, correlation
```

```
In [5]: book = pd.read_csv('book (2).csv')
book
```

Out[5]:

	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 [8]: del book['Unnamed: 0']
```

In [9]: book

Out[9]:

	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 [10]: book.sort\_values(['User.ID'])

Out[10]:

	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 valkirias	7
2397	278854	A Treasury of Irish Myth, Legend, and Folklore	6

10000 rows × 3 columns

In [12]: len(book['User.ID'].unique())

Out[12]: 2182

In [13]: len(book['Book.Title'].unique())

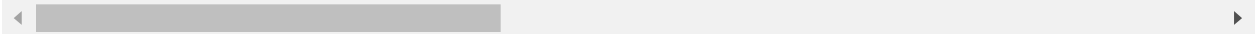
Out[13]: 9659

```
In [14]: # converting Long data into wide data using pivot table
book2=book.pivot_table(index='User.ID',columns='Book.Title', values='Book.Rating')
book2
```

Out[14]:

Book.Title	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	01-4 A No Miller
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 [15]: book2.index=book['User.ID'].unique()  
book2
```

Out[15]:

Other 185;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	01-01-00: A Novel of the Millennium	1,401 More Things That P*Ss Me Off	10 Commandments Of Dating	1 Gre Fanta Shc Shi Stori
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [16]:

```
# impute NaNs with 0 values
book2.fillna(0,inplace=True)
book2
```

Out[16]:

Book.Title	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	01-0' A Nov Millenr
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 [19]:

```
# calculating Cosine Similarity between Users on array data
user_sim = 1-pairwise_distances(book2.values,metric='cosine')
user_sim
```

Out[19]:

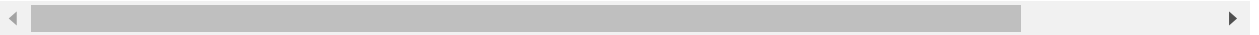
array([[1., 0., 0., ..., 0., 0., 0.],
[0., 1., 0., ..., 0., 0., 0.],
[0., 0., 1., ..., 0., 0., 0.],
...,
[0., 0., 0., ..., 1., 0., 0.],
[0., 0., 0., ..., 0., 1., 0.],
[0., 0., 0., ..., 0., 0., 1.]])

```
In [20]: user_sim2=pd.DataFrame(user_sim)
user_sim2
```

Out[20]:

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

2182 rows × 2182 columns



```
In [21]: # set the index and column names to User ids
user_sim2.index=book['User.ID'].unique()
user_sim2.columns=book['User.ID'].unique()
user_sim2
```

Out[21]:

	276726	276729	276736	276737	276744	276745	276747	276748	276751	276754	...	1
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 [22]: #Nullifying diagonal values
np.fill_diagonal(user_sim,0)
user_sim2
```

Out[22]:

	276726	276729	276736	276737	276744	276745	276747	276748	276751	276754	...	1
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 [24]: # most similar users
user_sim2.idxmax(axis=1)
```

Out[24]:

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 [25]: # extract the books which user id 162107 & 276726 have watched
book[(book['User.ID']==162107)|(book['User.ID']==276726)]
```

Out[25]:

	User.ID	Book.Title	Book.Rating
0	276726	Classical Mythology	5
9987	162107	What's Bred in the Bone	7

```
In [26]: # extract the books which user id 276729 & 276726 have watched
book[(book['User.ID']==276729)|(book['User.ID']==276726)]
```

Out[26]:

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

```
In [27]: user_1= book[(book['User.ID']==276729)]
user_2= book[(book['User.ID']==276726)]
```

```
In [28]: user_1['Book.Title']
```

Out[28]: 1 Clara Callan  
2 Decision in Normandy  
Name: Book.Title, dtype: object

```
In [29]: user_2['Book.Title']
```

Out[29]: 0 Classical Mythology  
Name: Book.Title, dtype: object

```
In [30]: pd.merge(user_1, user_2,on='Book.Title', how='outer')
```

Out[30]:

	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 [ ]: