5. Using the "Online Retail" dataset, perform product recommendations as described below:

Exclude entries where "Quantity" or "UnitPrice" have negative or zero values, and remove observations with missing CustomerID. Create a Customer-Item Matrix using the pivot table function, replacing NaN values with 0 and non-NaN values with 1. Compute the User-to-User Similarity Matrix. Recommend products to the user who has the highest similarity to customer 17173. Additionally, apply item-based collaborative filtering to identify products similar to the item with stock code 90103

```python
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from sklearn.metrics.pairwise import cosine_similarity
import os

df = pd.read_excel("D:/PYTHON/DATA SCIENCE/DATA/Online Retail.xlsx")

df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   InvoiceNo    541909 non-null  object
 1   StockCode    541909 non-null  object
 2   Description  540455 non-null  object
 3   Quantity     541909 non-null  int64
 4   InvoiceDate  541909 non-null  datetime64[ns]
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object
dtypes: datetime64[ns](1), float64(2), int64(1), object(4)
memory usage: 33.1+ MB

# Filter data
df = df[(df['Quantity'] > 0) & (df['UnitPrice'] > 0)]
df = df.dropna(subset=['CustomerID'])
df['CustomerID'].isna().sum()
df.shape

(397884, 8)
```

Customer-Item Matrix

```
customer_item_matrix = df.pivot_table(index='CustomerID',
columns='StockCode', values='Quantity', aggfunc='sum')
customer_item_matrix
```

```
StockCode    10002   10080   10120   10125   10133   10135   11001   15030
15034  \
CustomerID

12346.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
12347.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
12348.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
12349.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
12350.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
...           ...     ...     ...     ...     ...     ...     ...     ...
...
18280.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
18281.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
18282.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
18283.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN
18287.0       NaN     NaN     NaN     NaN     NaN     NaN     NaN     NaN
NaN

StockCode    15036   ...   90214V  90214W  90214Y  90214Z   BANK CHARGES
C2  DOT  \
CustomerID           ...

12346.0       NaN    ...     NaN     NaN     NaN     NaN            NaN
NaN  NaN
12347.0       NaN    ...     NaN     NaN     NaN     NaN            NaN
NaN  NaN
12348.0       NaN    ...     NaN     NaN     NaN     NaN            NaN
NaN  NaN
12349.0       NaN    ...     NaN     NaN     NaN     NaN            NaN
NaN  NaN
12350.0       NaN    ...     NaN     NaN     NaN     NaN            NaN
NaN  NaN
...           ...    ...     ...     ...     ...     ...            ...
..   ...
18280.0       NaN    ...     NaN     NaN     NaN     NaN            NaN
```

```
NaN   NaN
18281.0        NaN  ...     NaN    NaN    NaN    NaN          NaN
NaN   NaN
18282.0        NaN  ...     NaN    NaN    NaN    NaN          NaN
NaN   NaN
18283.0        NaN  ...     NaN    NaN    NaN    NaN          NaN
NaN   NaN
18287.0        NaN  ...     NaN    NaN    NaN    NaN          NaN
NaN   NaN

StockCode      M   PADS   POST
CustomerID
12346.0       NaN   NaN   NaN
12347.0       NaN   NaN   NaN
12348.0       NaN   NaN   9.0
12349.0       NaN   NaN   1.0
12350.0       NaN   NaN   1.0
...           ...   ...   ...
18280.0       NaN   NaN   NaN
18281.0       NaN   NaN   NaN
18282.0       NaN   NaN   NaN
18283.0       2.0   NaN   NaN
18287.0       NaN   NaN   NaN

[4338 rows x 3665 columns]
```

```python
customer_item_matrix = customer_item_matrix.applymap(lambda x: 1 if x
> 0 else 0)
customer_item_matrix
```

```
C:\Users\AJITH N\AppData\Local\Temp\ipykernel_4404\3723262966.py:1:
FutureWarning: DataFrame.applymap has been deprecated. Use
DataFrame.map instead.
  customer_item_matrix = customer_item_matrix.applymap(lambda x: 1 if
x > 0 else 0)

StockCode   10002  10080  10120  10125  10133  10135  11001  15030
15034  \
CustomerID

12346.0         0      0      0      0      0      0      0      0
0
12347.0         0      0      0      0      0      0      0      0
0
12348.0         0      0      0      0      0      0      0      0
0
12349.0         0      0      0      0      0      0      0      0
0
12350.0         0      0      0      0      0      0      0      0
0
```

...	...	...	...	...	...	...	...	...
18280.0	0	0	0	0	0	0	0	0
18281.0	0	0	0	0	0	0	0	0
18282.0	0	0	0	0	0	0	0	0
18283.0	0	0	0	0	0	0	0	0
18287.0	0	0	0	0	0	0	0	0

StockCode	15036	...	90214V	90214W	90214Y	90214Z	BANK CHARGES	C2	DOT \
CustomerID		...							
12346.0	0	...	0	0	0	0	0	0	0
12347.0	0	...	0	0	0	0	0	0	0
12348.0	0	...	0	0	0	0	0	0	0
12349.0	0	...	0	0	0	0	0	0	0
12350.0	0	...	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	..	...
18280.0	0	...	0	0	0	0	0	0	0
18281.0	0	...	0	0	0	0	0	0	0
18282.0	0	...	0	0	0	0	0	0	0
18283.0	0	...	0	0	0	0	0	0	0
18287.0	0	...	0	0	0	0	0	0	0

StockCode	M	PADS	POST
CustomerID			
12346.0	0	0	0
12347.0	0	0	0
12348.0	0	0	1
12349.0	0	0	1
12350.0	0	0	1
...	..	...	...
18280.0	0	0	0
18281.0	0	0	0

```
18282.0      0      0      0
18283.0      1      0      0
18287.0      0      0      0

[4338 rows x 3665 columns]

user_to_user_sim_matrix =
pd.DataFrame(cosine_similarity(customer_item_matrix),
                              index=customer_item_matrix.index,
                              columns=customer_item_matrix.index)

user_to_user_sim_matrix.head()
```

CustomerID	12346.0	12347.0	12348.0	12349.0	12350.0	12352.0
CustomerID						
12346.0	1.0	0.000000	0.000000	0.000000	0.000000	0.000000
12347.0	0.0	1.000000	0.063022	0.046130	0.047795	0.038484
12348.0	0.0	0.063022	1.000000	0.024953	0.051709	0.027756
12349.0	0.0	0.046130	0.024953	1.000000	0.056773	0.137137
12350.0	0.0	0.047795	0.051709	0.056773	1.000000	0.031575

CustomerID	12353.0	12354.0	12355.0	12356.0	...	18273.0	18274.0
CustomerID					...		
12346.0	0.0	0.000000	0.000000	0.000000	...	0.0	0.000000
12347.0	0.0	0.025876	0.136641	0.094742	...	0.0	0.029709
12348.0	0.0	0.027995	0.118262	0.146427	...	0.0	0.064282
12349.0	0.0	0.030737	0.032461	0.144692	...	0.0	0.105868
12350.0	0.0	0.000000	0.000000	0.033315	...	0.0	0.000000

CustomerID	18276.0	18277.0	18278.0	18280.0	18281.0	18282.0
CustomerID						
12346.0	0.000000	0.0	0.000000	0.000000	0.0	0.000000
12347.0	0.052668	0.0	0.032844	0.062318	0.0	0.113776

```
12348.0       0.113961      0.0   0.000000   0.000000      0.0   0.000000

12349.0       0.000000      0.0   0.039014   0.000000      0.0   0.067574

12350.0       0.000000      0.0   0.000000   0.000000      0.0   0.000000


CustomerID    18283.0    18287.0
CustomerID
12346.0       0.000000   0.000000
12347.0       0.109364   0.012828
12348.0       0.170905   0.083269
12349.0       0.137124   0.030475
12350.0       0.044866   0.000000

[5 rows x 4338 columns]
```

user_to_user_sim_matrix.columns = customer_item_matrix.index
user_to_user_sim_matrix.index = customer_item_matrix.index
user_to_user_sim_matrix

```
CustomerID  12346.0    12347.0    12348.0    12349.0    12350.0    12352.0
\
CustomerID

12346.0        1.0   0.000000   0.000000   0.000000   0.000000   0.000000

12347.0        0.0   1.000000   0.063022   0.046130   0.047795   0.038484

12348.0        0.0   0.063022   1.000000   0.024953   0.051709   0.027756

12349.0        0.0   0.046130   0.024953   1.000000   0.056773   0.137137

12350.0        0.0   0.047795   0.051709   0.056773   1.000000   0.031575

...            ...        ...        ...        ...        ...        ...

18280.0        0.0   0.062318   0.000000   0.000000   0.000000   0.000000

18281.0        0.0   0.000000   0.000000   0.000000   0.000000   0.000000

18282.0        0.0   0.113776   0.000000   0.067574   0.000000   0.037582

18283.0        0.0   0.109364   0.170905   0.137124   0.044866   0.080278

18287.0        0.0   0.012828   0.083269   0.030475   0.000000   0.033898


CustomerID  12353.0    12354.0    12355.0    12356.0  ...   18273.0
18274.0  \
CustomerID                                            ...
```

```
12346.0          0.0   0.000000   0.000000   0.000000   ...        0.0
0.000000
12347.0          0.0   0.025876   0.136641   0.094742   ...        0.0
0.029709
12348.0          0.0   0.027995   0.118262   0.146427   ...        0.0
0.064282
12349.0          0.0   0.030737   0.032461   0.144692   ...        0.0
0.105868
12350.0          0.0   0.000000   0.000000   0.033315   ...        0.0
0.000000
...              ...        ...        ...        ...    ...        ...
...
18280.0          0.0   0.041523   0.000000   0.000000   ...        0.0
0.000000
18281.0          0.0   0.049629   0.000000   0.000000   ...        0.0
0.000000
18282.0          0.0   0.000000   0.160128   0.079305   ...        0.0
0.174078
18283.0          0.0   0.113354   0.034204   0.093170   ...        0.0
0.037184
18287.0          0.0   0.000000   0.108324   0.000000   ...        0.0
0.000000

CustomerID   18276.0    18277.0    18278.0    18280.0    18281.0    18282.0
\
CustomerID

12346.0     0.000000   0.000000   0.000000   0.000000   0.000000   0.000000

12347.0     0.052668   0.000000   0.032844   0.062318   0.000000   0.113776

12348.0     0.113961   0.000000   0.000000   0.000000   0.000000   0.000000

12349.0     0.000000   0.000000   0.039014   0.000000   0.000000   0.067574

12350.0     0.000000   0.000000   0.000000   0.000000   0.000000   0.000000

...              ...        ...        ...        ...        ...        ...

18280.0     0.000000   0.000000   0.105409   1.000000   0.119523   0.000000

18281.0     0.000000   0.000000   0.000000   0.119523   1.000000   0.000000

18282.0     0.000000   0.000000   0.000000   0.000000   0.000000   1.000000

18283.0     0.016480   0.043602   0.000000   0.000000   0.046613   0.017800

18287.0     0.104383   0.000000   0.043396   0.000000   0.000000   0.000000
```

```
CustomerID    18283.0    18287.0
CustomerID
12346.0      0.000000   0.000000
12347.0      0.109364   0.012828
12348.0      0.170905   0.083269
12349.0      0.137124   0.030475
12350.0      0.044866   0.000000
...               ...        ...
18280.0      0.000000   0.000000
18281.0      0.046613   0.000000
18282.0      0.017800   0.000000
18283.0      1.000000   0.096334
18287.0      0.096334   1.000000

[4338 rows x 4338 columns]
```

The person 17173 similar with others

```
user_to_user_sim_matrix.loc[17173.0].sort_values(ascending=False)

CustomerID
17173.0    1.000000
14567.0    0.899401
14937.0    0.899401
18167.0    0.816497
17053.0    0.600925
             ...
14946.0    0.000000
14947.0    0.000000
12399.0    0.000000
12402.0    0.000000
17630.0    0.000000
Name: 17173.0, Length: 4338, dtype: float64

item_bought_A =
set(customer_item_matrix.columns[customer_item_matrix.loc[17173.0].to_
numpy().nonzero()])
item_bought_A

{20914,
 20969,
 20970,
 20971,
 20972,
 21106,
 21107,
 21108,
 21109,
 21110,
 21216,
```

```
 21218,
 21531,
 21533,
 21539,
 21843,
 21844,
 22070,
 22141,
 22142,
 22144,
 22147,
 22150,
 22271,
 22273,
 22274,
 22637,
 22749,
 22750,
 22751,
 22795,
 22940,
 23126,
 23127,
 85066,
 '85123A'}
```

```python
item_bought_by_B =
set(customer_item_matrix.columns[customer_item_matrix.loc[14567.0].to_
numpy().nonzero()])
item_bought_by_B
```

```
{20914,
 20969,
 20970,
 20971,
 20972,
 21216,
 21218,
 21531,
 21533,
 21539,
 21843,
 21844,
 22070,
 22141,
 22142,
 22144,
 22147,
 22150,
 22271,
```

```
 22273,
 22274,
 22568,
 22637,
 22749,
 22750,
 22751,
 22795,
 22940,
 23126,
 23127,
 23128,
 85066,
 '85123A'}

recommendation = item_bought_A-item_bought_by_B
recommendation

{21106, 21107, 21108, 21109, 21110}
```

Interpretation:

the system recommends these products to Customer b(14567)

Item based collobarative

```
item_to_item_sim_matrix =
pd.DataFrame(cosine_similarity(customer_item_matrix.T))
item_to_item_sim_matrix

            0           1           2           3           4           5
6      \
0      1.000000   0.000000   0.094868   0.090351   0.062932   0.098907
0.095346
1      0.000000   1.000000   0.000000   0.032774   0.045655   0.047836
0.000000
2      0.094868   0.000000   1.000000   0.057143   0.059702   0.041703
0.060302
3      0.090351   0.032774   0.057143   1.000000   0.042644   0.044682
0.043073
4      0.062932   0.045655   0.059702   0.042644   1.000000   0.280097
0.045002
...         ...         ...         ...         ...         ...         ...
...
3660   0.029361   0.000000   0.000000   0.000000   0.036955   0.019360
0.055989
3661   0.000000   0.000000   0.000000   0.000000   0.000000   0.104257
0.150756
3662   0.067591   0.016345   0.071247   0.071247   0.070893   0.066852
0.107409
```

```
3663   0.000000   0.000000   0.000000   0.000000   0.000000   0.000000
0.000000
3664   0.078217   0.000000   0.010993   0.070669   0.021877   0.034383
0.058004

              7          8          9     ...      3655   3656       3657
3658   \
0      0.047673   0.075593   0.090815   ...   0.000000    0.0   0.000000
0.0
1      0.000000   0.082261   0.049413   ...   0.000000    0.0   0.000000
0.0
2      0.060302   0.095618   0.028718   ...   0.000000    0.0   0.000000
0.0
3      0.000000   0.051224   0.030770   ...   0.000000    0.0   0.000000
0.0
4      0.060003   0.071358   0.057152   ...   0.000000    0.0   0.000000
0.0
...         ...        ...        ...   ...        ...    ...        ...   ..
.
3660   0.000000   0.000000   0.039996   ...   0.000000    0.0   0.000000
0.0
3661   0.000000   0.000000   0.000000   ...   0.000000    0.0   0.000000
0.0
3662   0.064445   0.059610   0.086959   ...   0.050379    0.0   0.041135
0.0
3663   0.000000   0.000000   0.000000   ...   0.000000    0.0   0.000000
0.0
3664   0.016573   0.026278   0.051301   ...   0.038866    0.0   0.031734
0.0

           3659       3660  3661       3662  3663       3664
0      0.000000   0.029361   0.0   0.067591   0.0   0.078217
1      0.000000   0.000000   0.0   0.016345   0.0   0.000000
2      0.000000   0.000000   0.0   0.071247   0.0   0.010993
3      0.000000   0.000000   0.0   0.071247   0.0   0.070669
4      0.000000   0.036955   0.0   0.070893   0.0   0.021877
...         ...        ...   ...        ...   ...        ...
3660   0.000000   1.000000   0.0   0.026460   0.0   0.020413
3661   0.000000   0.000000   1.0   0.000000   0.0   0.000000
3662   0.090121   0.026460   0.0   1.000000   0.0   0.070490
3663   0.000000   0.000000   0.0   0.000000   1.0   0.000000
3664   0.017381   0.020413   0.0   0.070490   0.0   1.000000

[3665 rows x 3665 columns]

item_to_item_sim_matrix.columns = customer_item_matrix.T.index
item_to_item_sim_matrix.index = customer_item_matrix.T.index
item_to_item_sim_matrix
```

```
StockCode      10002      10080      10120      10125      10133      10135
\
StockCode

10002       1.000000   0.000000   0.094868   0.090351   0.062932   0.098907

10080       0.000000   1.000000   0.000000   0.032774   0.045655   0.047836

10120       0.094868   0.000000   1.000000   0.057143   0.059702   0.041703

10125       0.090351   0.032774   0.057143   1.000000   0.042644   0.044682

10133       0.062932   0.045655   0.059702   0.042644   1.000000   0.280097

...              ...        ...        ...        ...        ...        ...

C2          0.029361   0.000000   0.000000   0.000000   0.036955   0.019360

DOT         0.000000   0.000000   0.000000   0.000000   0.000000   0.104257

M           0.067591   0.016345   0.071247   0.071247   0.070893   0.066852

PADS        0.000000   0.000000   0.000000   0.000000   0.000000   0.000000

POST        0.078217   0.000000   0.010993   0.070669   0.021877   0.034383


StockCode      11001      15030      15034      15036  ...     90214V
90214W  \
StockCode                                             ...

10002       0.095346   0.047673   0.075593   0.090815  ...   0.000000
0.0
10080       0.000000   0.000000   0.082261   0.049413  ...   0.000000
0.0
10120       0.060302   0.060302   0.095618   0.028718  ...   0.000000
0.0
10125       0.043073   0.000000   0.051224   0.030770  ...   0.000000
0.0
10133       0.045002   0.060003   0.071358   0.057152  ...   0.000000
0.0
...              ...        ...        ...        ...  ...        ...   .
..
C2          0.055989   0.000000   0.000000   0.039996  ...   0.000000
0.0
DOT         0.150756   0.000000   0.000000   0.000000  ...   0.000000
0.0
M           0.107409   0.064445   0.059610   0.086959  ...   0.050379
0.0
PADS        0.000000   0.000000   0.000000   0.000000  ...   0.000000
0.0
```

```
POST        0.058004  0.016573  0.026278  0.051301  ...  0.038866
0.0

StockCode     90214Y  90214Z  BANK CHARGES        C2  DOT         M
PADS   \
StockCode

10002      0.000000     0.0      0.000000  0.029361  0.0  0.067591
0.0
10080      0.000000     0.0      0.000000  0.000000  0.0  0.016345
0.0
10120      0.000000     0.0      0.000000  0.000000  0.0  0.071247
0.0
10125      0.000000     0.0      0.000000  0.000000  0.0  0.071247
0.0
10133      0.000000     0.0      0.000000  0.036955  0.0  0.070893
0.0
...            ...     ...           ...       ...  ...       ...   .
..
C2         0.000000     0.0      0.000000  1.000000  0.0  0.026460
0.0
DOT        0.000000     0.0      0.000000  0.000000  1.0  0.000000
0.0
M          0.041135     0.0      0.090121  0.026460  0.0  1.000000
0.0
PADS       0.000000     0.0      0.000000  0.000000  0.0  0.000000
1.0
POST       0.031734     0.0      0.017381  0.020413  0.0  0.070490
0.0

StockCode        POST
StockCode
10002       0.078217
10080       0.000000
10120       0.010993
10125       0.070669
10133       0.021877

...              ...
C2          0.020413
DOT         0.000000
M           0.070490
PADS        0.000000
POST        1.000000

[3665 rows x 3665 columns]
```

Making Recommendations similar to the item with stock code 90103

```
item_to_item_sim_matrix.loc[90103.0].sort_values(ascending=False)
```

```
StockCode
90103     1.000000
90101     0.816497
90059F    0.816497
90059C    0.816497
90059B    0.816497
            ...
90181A    0.000000
90187B    0.000000
22259     0.000000
22263     0.000000
90210A    0.000000
Name: 90103, Length: 3665, dtype: float64
```

```python
top_10_items =
item_to_item_sim_matrix.loc[90103.0].sort_values(ascending=False).head
(10)
top_10_items
```

```
StockCode
90103     1.000000
90101     0.816497
90059F    0.816497
90059C    0.816497
90059B    0.816497
90059E    0.816497
90169     0.707107
90208     0.707107
20678     0.707107
90059D    0.707107
Name: 90103, dtype: float64
```

```python
df.loc[
    df['StockCode'].isin(top_10_items.index),
    ['StockCode','Description']
].drop_duplicates().set_index('StockCode')
```

```
                                Description
StockCode
90059B     DIAMANTE HAIR GRIP PACK/2 BLACK DIA
90059E            DIAMANTE HAIR GRIP PACK/2 RUBY
90059C        DIAMANTE HAIR GRIP PACK/2 MONTANA
90059F        DIAMANTE HAIR GRIP PACK/2 LT ROSE
90208         PAIR OF PINK FLOWER CLUSTER SLIDE
90101                   WHITE FRANGIPANI NECKLACE
90103                  PURPLE FRANGIPANI NECKLACE
20678             LARGE BLACK DIAMANTE HAIRSLIDE
90059D        DIAMANTE HAIR GRIP PACK/2 PERIDOT
90169                           DAISY HAIR COMB
```

Interpretation"

The code lists the top 10 items by StockCode and Description, removing duplicates and setting the StockCode as the index, highlighting key products in the dataset.