```python
import pandas as pd


df = pd.read_csv(r"Datasets\customer_shopping_behavior.csv")
df.head()
```

```
   Customer ID  Age Gender Item Purchased  Category  Purchase Amount
(USD)  \
0            1   55   Male         Blouse  Clothing
53
1            2   19   Male        Sweater  Clothing
64
2            3   50   Male          Jeans  Clothing
73
3            4   21   Male        Sandals  Footwear
90
4            5   45   Male         Blouse  Clothing
49

        Location Size      Color  Season  Review Rating Subscription
Status  \
0       Kentucky    L       Gray  Winter            3.1
Yes
1          Maine    L     Maroon  Winter            3.1
Yes
2  Massachusetts    S     Maroon  Spring            3.1
Yes
3   Rhode Island    M     Maroon  Spring            3.5
Yes
4         Oregon    M  Turquoise  Spring            2.7
Yes

   Shipping Type Discount Applied Promo Code Used  Previous Purchases
\
0        Express              Yes             Yes                  14

1        Express              Yes             Yes                   2

2  Free Shipping              Yes             Yes                  23

3   Next Day Air              Yes             Yes                  49

4  Free Shipping              Yes             Yes                  31


  Payment Method Frequency of Purchases
0          Venmo            Fortnightly
1           Cash            Fortnightly
2    Credit Card                 Weekly
```

```
3         PayPal                  Weekly
4         PayPal                Annually
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3900 entries, 0 to 3899
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Customer ID             3900 non-null   int64  
 1   Age                     3900 non-null   int64  
 2   Gender                  3900 non-null   object 
 3   Item Purchased          3900 non-null   object 
 4   Category                3900 non-null   object 
 5   Purchase Amount (USD)   3900 non-null   int64  
 6   Location                3900 non-null   object 
 7   Size                    3900 non-null   object 
 8   Color                   3900 non-null   object 
 9   Season                  3900 non-null   object 
 10  Review Rating           3863 non-null   float64
 11  Subscription Status     3900 non-null   object 
 12  Shipping Type           3900 non-null   object 
 13  Discount Applied        3900 non-null   object 
 14  Promo Code Used         3900 non-null   object 
 15  Previous Purchases      3900 non-null   int64  
 16  Payment Method          3900 non-null   object 
 17  Frequency of Purchases  3900 non-null   object 
dtypes: float64(1), int64(4), object(13)
memory usage: 548.6+ KB
```

```
df.describe()
```

```
       Customer ID          Age  Purchase Amount (USD)  Review Rating
\
count  3900.000000  3900.000000            3900.000000    3863.000000

mean   1950.500000    44.068462              59.764359       3.750065

std    1125.977353    15.207589              23.685392       0.716983

min       1.000000    18.000000              20.000000       2.500000

25%     975.750000    31.000000              39.000000       3.100000

50%    1950.500000    44.000000              60.000000       3.800000
```

```
75%    2925.250000    57.000000              81.000000        4.400000

max    3900.000000    70.000000             100.000000        5.000000
```

```
       Previous Purchases
count         3900.000000
mean            25.351538
std             14.447125
min              1.000000
25%             13.000000
50%             25.000000
75%             38.000000
max             50.000000
```

```
df.columns = df.columns.str.strip().str.lower()
df.columns = df.columns.str.replace(' ','_')
df = df.rename(columns = {'purchase_amount_(usd)' :
'purchase_amount'})
df.columns
```

```
Index(['customer_id', 'age', 'gender', 'item_purchased', 'category',
       'purchase_amount', 'location', 'size', 'color', 'season',
       'review_rating', 'subscription_status', 'shipping_type',
       'discount_applied', 'promo_code_used', 'previous_purchases',
       'payment_method', 'frequency_of_purchases'],
      dtype='object')
```

```
df.isnull().sum()
```

```
customer_id                0
age                        0
gender                     0
item_purchased             0
category                   0
purchase_amount            0
location                   0
size                       0
color                      0
season                     0
review_rating             37
subscription_status        0
shipping_type              0
discount_applied           0
promo_code_used            0
previous_purchases         0
payment_method             0
```

```
frequency_of_purchases     0
dtype: int64
```

```
df['review_rating'] = df.groupby('category')
['review_rating'].transform(lambda x: x.fillna(x.median()))
```

```
df.isnull().sum()
```

```
customer_id                0
age                        0
gender                     0
item_purchased             0
category                   0
purchase_amount            0
location                   0
size                       0
color                      0
season                     0
review_rating              0
subscription_status        0
shipping_type              0
discount_applied           0
promo_code_used            0
previous_purchases         0
payment_method             0
frequency_of_purchases     0
dtype: int64
```

create new columns

```
# create a column 'age_group'

df['age'].min()
df['age'].max()

labels = ['young_adults', 'adults', 'middle-aged', 'senior']
df['age_group'] = pd.qcut(df['age'], q=4, labels = labels)
```

```
df[['age','age_group']].head(10)
```

```
   age     age_group
0   55   middle-aged
```

```
1   19  young_adults
2   50   middle-aged
3   21  young_adults
4   45   middle-aged
5   46   middle-aged
6   63        senior
7   27  young_adults
8   26  young_adults
9   57   middle-aged
```

```
# create column purchase_frequency_days

frequency_mapping = {
    'Fortnightly' : 14,
    'Weekly' : 7,
    'Bi-Weekly' : 14,
    'Monthly' : 30,
    'Quarterly' : 90,
    'Annually' : 365,
    'Every 3 Months' : 90 }

df['purchase_frequency_days'] =
df['frequency_of_purchases'].map(frequency_mapping)
```

```
df[['frequency_of_purchases','purchase_frequency_days']].head(10)
```

```
  frequency_of_purchases  purchase_frequency_days
0            Fortnightly                       14
1            Fortnightly                       14
2                 Weekly                        7
3                 Weekly                        7
4               Annually                      365
5                 Weekly                        7
6              Quarterly                       90
7                 Weekly                        7
8               Annually                      365
9              Quarterly                       90
```

```
df[['discount_applied','promo_code_used']].head(10)
```

```
  discount_applied promo_code_used
0              Yes             Yes
1              Yes             Yes
2              Yes             Yes
3              Yes             Yes
4              Yes             Yes
```

```
5               Yes              Yes
6               Yes              Yes
7               Yes              Yes
8               Yes              Yes
9               Yes              Yes
```

```
(df['discount_applied'] == df['promo_code_used']).all()    # check
that is both columns carry exactly same value or not
```

```
np.True_
```

```
# remove 'promo_code_used' column

df = df.drop('promo_code_used', axis=1)

df.columns
```

```
Index(['customer_id', 'age', 'gender', 'item_purchased', 'category',
       'purchase_amount', 'location', 'size', 'color', 'season',
       'review_rating', 'subscription_status', 'shipping_type',
       'discount_applied', 'previous_purchases', 'payment_method',
       'frequency_of_purchases', 'age_group',
'purchase_frequency_days'],
      dtype='object')
```

lets connect this database with mysql

```
from sqlalchemy import create_engine


engine =
create_engine("mysql+pymysql://root:8520147@localhost/customer_behavio
ur")

df.to_sql('customers',con=engine, if_exists='replace', index = False)
```

```
3900
```