

```
from google.colab import files
uploaded = files.upload()

<IPython.core.display.HTML object>

Saving customer_shopping_behavior.xlsx to customer_shopping_behavior
(1).xlsx

import pandas as pd

df = pd.read_excel('customer_shopping_behavior.xlsx')

df.head()

{"type": "dataframe", "variable_name": "df"}
```

df.shape
df.columns

Index(['Customer ID', 'Age', 'Gender', 'Item Purchased', 'Category',
 'Purchase Amount (USD)', 'Location', 'Size', 'Color', 'Season',
 'Review Rating', 'Subscription Status', 'Payment Method',
 'Shipping Type', 'Discount Applied', 'Promo Code Used',
 'Previous Purchases', 'Preferred Payment Method',
 'Frequency of Purchases', 'Date', 'Year', 'Month'],
 dtype='object')

df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10745 entries, 0 to 10744
Data columns (total 22 columns):
 #   Column           Non-Null Count  Dtype  
 --- 
 0   Customer ID     10745 non-null   int64  
 1   Age              10745 non-null   int64  
 2   Gender            10745 non-null   object  
 3   Item Purchased   10745 non-null   object  
 4   Category          10745 non-null   object  
 5   Purchase Amount (USD) 10745 non-null   int64  
 6   Location           10745 non-null   object  
 7   Size               10745 non-null   object  
 8   Color               10745 non-null   object  
 9   Season              10745 non-null   object  
 10  Review Rating     10745 non-null   float64 
 11  Subscription Status 10745 non-null   object  
 12  Payment Method    10745 non-null   object  
 13  Shipping Type     10745 non-null   object  
 14  Discount Applied   10745 non-null   object  
 15  Promo Code Used   10745 non-null   object  
 16  Previous Purchases 10745 non-null   int64
```

```
17 Preferred Payment Method    10745 non-null  object
18 Frequency of Purchases    10745 non-null  object
19 Date                      10745 non-null  int64
20 Year                      10745 non-null  int64
21 Month                     10745 non-null  object
dtypes: float64(1), int64(6), object(15)
memory usage: 1.8+ MB

df.describe(include='all')

{"type": "dataframe"}
```

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

	customer_id	age	gender	item_purchased	category	purchase_amount_usd	color	location	size	season	review_rating	subscription_status	payment_method	shipping_type	discount_applied	promo_code_used	previous_purchases	preferred_payment_method	frequency_of_purchases	date	year	month
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

```
dtype: int64
```

```
type(df)
```

```
pandas.core.frame.DataFrame
```

```
df.columns
```

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

```

len(df.columns)

22

df.columns = [
    'customer_id',
    'age',
    'gender',
    'item_purchased',
    'category',
    'purchase_amount_usd',
    'color',
    'location',
    'size',
    'season',
    'review_rating',
    'subscription_status',
    'payment_method',
    'shipping_type',
    'discount_applied',
    'promo_code_used',
    'previous_purchases',
    'preferred_payment_method',
    'frequency_of_purchases',
    'date',
    'year',
    'month'
]

print(df.columns.tolist())

['customer_id', 'age', 'gender', 'item_purchased', 'category',
'purchase_amount_usd', 'color', 'location', 'size', 'season',
'review_rating', 'subscription_status', 'payment_method',
'shipping_type', 'discount_applied', 'promo_code_used',
'previous_purchases', 'preferred_payment_method',
'frequency_of_purchases', 'date', 'year', 'month']

#create a column age_group
labels = ['Young Adult', 'Adult', 'Middle-aged', 'Senior']
df['age_group'] = pd.qcut(df['age'], q=4, labels = labels)

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

{
  "summary": {
    "name": "df[['age', 'age_group']]",
    "rows": 10,
    "fields": [
      {
        "column": "age",
        "properties": {
          "dtype": "number",
          "std": 16,
          "min": 19,
          "max": 63,
          "num_unique_values": 10,
          "samples": [26, 26, 46],
          "semantic_type": "\",
          "description": "\n"
        }
      },
      {
        "column": "age_group"
      }
    ]
  }
}

```

```

\"age_group\", \n      \"properties\": {\n          \"dtype\":\n          \"category\", \n          \"num_unique_values\": 3, \n          \"samples\": [\n              \"Middle-aged\", \n              \"Young Adult\", \n              \"Senior\" \n          ], \n          \"semantic_type\": \"\", \n          \"description\": \"\" \n      } \n  } \n}, \"type\": \"dataframe\"}

#create column purchase_frequency_days

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

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

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

{"summary": "{\n  \"name\":\n    \"df[['purchase_frequency_days', 'frequency_of_purchases']]\"\n  ,\n  \"rows\": 10,\n  \"fields\": [\n    {\n      \"column\":\n        \"purchase_frequency_days\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 164.6700163530516,\n        \"min\": 7.0,\n        \"max\": 365.0,\n        \"num_unique_values\": 3,\n        \"samples\": [\n          14.0,\n          7.0,\n          365.0\n        ],\n        \"semantic_type\": \"\", \n        \"description\": \"\" \n      }\n    },\n    {\n      \"column\":\n        \"frequency_of_purchases\",\n      \"properties\": {\n        \"dtype\": \"category\",\n        \"num_unique_values\": 4,\n        \"samples\": [\n          \"Weekly\",\n          \"Quarterly\",\n          \"Fortnightly\"\n        ],\n        \"semantic_type\": \"\", \n        \"description\": \"\" \n      }\n    }\n  ]\n}, \"type\": \"dataframe\"}

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

{"summary": "{\n  \"name\":\n    \"df[['discount_applied', 'promo_code_used']]\"\n  ,\n  \"rows\": 10,\n  \"fields\": [\n    {\n      \"column\":\n        \"discount_applied\",\n      \"properties\": {\n        \"dtype\": \"category\",\n        \"num_unique_values\": 1,\n        \"samples\": [\n          \"Yes\"\n        ],\n        \"semantic_type\": \"\", \n        \"description\": \"\" \n      }\n    },\n    {\n      \"column\":\n        \"promo_code_used\",\n      \"properties\": {\n        \"dtype\": \"category\",\n        \"num_unique_values\": 1,\n        \"samples\": [\n          \"Yes\"\n        ],\n        \"semantic_type\": \"\", \n        \"description\": \"\" \n      }\n    }\n  ]\n}, \"type\": \"dataframe\"}

```

```
(df['discount_applied'] == df['promo_code_used']).all()

np.True_

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

df.columns

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

!pip install mysql-connector-python sqlalchemy

Collecting mysql-connector-python
  Downloading mysql_connector_python-9.5.0-cp312-cp312-
manylinux_2_28_x86_64.whl.metadata (7.5 kB)
Requirement already satisfied: sqlalchemy in
/usr/local/lib/python3.12/dist-packages (2.0.45)
Requirement already satisfied: greenlet>=1 in
/usr/local/lib/python3.12/dist-packages (from sqlalchemy) (3.3.0)
Requirement already satisfied: typing-extensions>=4.6.0 in
/usr/local/lib/python3.12/dist-packages (from sqlalchemy) (4.15.0)
  Downloading mysql_connector_python-9.5.0-cp312-cp312-
manylinux_2_28_x86_64.whl (34.1 MB)
----- 34.1/34.1 MB 60.9 MB/s eta
0:00:00
mysql-connector-python
Successfully installed mysql-connector-python-9.5.0

import pandas as pd
from sqlalchemy import create_engine

username = "root"
password = "mehweensmooky10"
host = "127.0.0.1"
port = "3306"
database = "customer_shopping_behavior"

engine = create_engine(
    f"mysql+mysqlconnector://{username}:{password}@{host}:
{port}/{database}"
)

df.to_csv("customer_shopping_clean.csv", index=False)
```

```
from google.colab import files
files.download("customer_shopping_clean.csv")
<IPython.core.display.Javascript object>
<IPython.core.display.Javascript object>
```