

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**  
**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO MÔN HỌC**  
**ĐỀ TÀI: PHÂN TÍCH HÀNH VI MUA SẴM**  
**TRỰC TUYẾN CỦA KHÁCH HÀNG THEO**  
**THỜI GIAN**

**Môn học: Phân Tích Dữ Liệu**  
**GVHD: ThS. Hồ Hường Thiên**  
**Lớp: DH22IM01**

**Sinh viên phụ trách:**

- Bùi Huỳnh Ngọc Linh - 2254052036**
- Đỗ Quỳnh Giang - 2254052020**
- Trương Thị Bích Thủy - 2254052080**

# MỤC LỤC

<b>I. Khái quát</b> .....	1
1. Giới thiệu đề tài.....	1
2. Xác định bài toán .....	1
<b>II. Nội dung</b> .....	2
1. Chuẩn bị dữ liệu.....	2
2. Xử lý dữ liệu.....	4
2.1 Các công cụ sử dụng để phân tích .....	4
2.2 Tóm tắt dữ liệu cơ bản và thống kê.....	4
2.3 Làm sạch và thao tác dữ liệu .....	6
3. Phân tích dữ liệu .....	10
3.1 Tìm hiểu hành vi của khách hàng theo độ tuổi và giới tính để hiểu thói quen mua sắm trong các nhóm ?.....	10
3.1.1 Phân tích và trực quan hóa theo xu hướng giới tính (Trends about Gender).....	10
3.1.2 Phân tích và trực quan hóa theo xu hướng tuổi (Trends about Age) .....	17
3.1.3 Phân tích và trực quan hóa xu hướng về thời gian giao dịch (Purchase Date) .....	26
3.1.4 Phân tích (kết hợp ba thuộc tính) tổng chi tiêu (Total Spending) theo nhóm tuổi (Age) và giới tính (Gender) .....	30
3.2 Tần suất mua sắm của các nhóm khách hàng theo thời gian? .....	31
3.2.1 Tần suất mua sắm của khách hàng qua 4 năm .....	32
3.2.2 Tần suất giao dịch từng tháng (theo năm).....	36
3.2.3 Tần suất giao dịch của từng khách hàng từng tháng (theo năm) .....	39
3.2.4 Tần suất mua sắm của khách hàng theo giới tính .....	46
3.3 Xu hướng mua sắm theo mùa thành các tháng và quý để xem liệu có sự gia tăng doanh số trong các mùa lễ hội ? .....	54
3.3.1 Thống kê doanh thu qua từng năm.....	54
3.3.2 Xu hướng mua sắm theo mùa (tháng và quý) .....	56
3.4 Mối tương quan giữa các biến .....	69
3.4.1 Tính toán hệ số tương quan ma trận.....	69
3.4.2 Phân tích các biến liên tục.....	70
<b>III. Kết luận &amp; Khuyến nghị</b> .....	71
1. Kết luận .....	71
2. Khuyến nghị kinh doanh & tiếp thị .....	71

# Phân tích hành vi mua sắm trực tuyến của khách hàng theo thời gian

## I. Khái quát

### 1. Giới thiệu đề tài

Trong bối cảnh thương mại điện tử phát triển mạnh mẽ, hành vi mua sắm trực tuyến ngày càng trở thành một chủ đề nghiên cứu hấp dẫn và thiết thực. Đề tài “Phân tích hành vi mua sắm của khách hàng theo thời gian” tập trung làm sáng tỏ sự biến đổi trong thói quen tiêu dùng, đặc biệt là trong các giai đoạn như mùa vụ, dịp lễ hội, và sự kiện đặc biệt. Việc hiểu rõ các xu hướng này không chỉ giúp doanh nghiệp tối ưu hóa chiến lược kinh doanh và cải thiện trải nghiệm khách hàng. Ngoài ra, nghiên cứu này còn cung cấp dữ liệu thực tiễn, góp phần làm rõ mối quan hệ giữa các yếu tố văn hóa, xã hội, và kinh tế đối với hành vi mua sắm.

**Ecommerce Customer Data Analysis** là bộ dữ liệu được thu thập, xử lý bởi Kaggle, bộ dữ liệu được thu thập từ khảo sát hành vi tiêu dùng, bao gồm các thông tin chi tiết về nhân khẩu học, sở thích mua sắm, tần suất mua hàng, và các yếu tố ảnh hưởng đến quyết định mua sắm như khuyến mãi, giá cả, và chất lượng sản phẩm. Dữ liệu này không chỉ giúp khám phá thói quen tiêu dùng theo thời gian mà còn mang lại cái nhìn tổng quan về cách khách hàng phản ứng với các yếu tố thị trường trong từng thời điểm cụ thể.

Trong quá trình thực hiện, các công cụ phân tích dữ liệu như Pandas và Numpy sẽ được sử dụng để xử lý dữ liệu, ngoài ra Matplotlib và Seaborn sẽ hỗ trợ trực quan hóa các xu hướng. Những phân tích này sẽ trả lời các câu hỏi nghiên cứu chính và cung cấp các đề xuất dựa trên dữ liệu thực tế.

Thông qua việc phân tích bộ dữ liệu này trọng tâm thực hiện **Phân tích dữ liệu khám phá** (EDA), các phương pháp khoa học, nghiên cứu từ nhiều góc độ khác nhau. Ngoài ra, dựa trên những phát hiện này, sẽ cung cấp các đề xuất chiến lược cho doanh nghiệp và cung cấp tài liệu tham khảo có giá trị cho các nhà nghiên cứu trong lĩnh vực thương mại điện tử.

### 2. Xác định bài toán

Một trong điều tối quan trọng của quá trình phân tích dữ liệu, trước tiên, cần phải xác định đúng vấn đề và bài toán nhằm đảm bảo kết quả phân tích không chỉ phục vụ mục tiêu chung mà còn hỗ trợ tối ưu các quyết định kinh doanh của các bên liên quan (Giám đốc điều hành (CEO) của công ty, nhân viên kinh doanh hoặc đội ngũ marketing và sản phẩm, v.v.). Cách tiếp cận này cho phép chúng ta nắm bắt rõ hơn các vấn đề hoặc câu hỏi mà họ đang đối mặt, đồng thời giúp chúng ta điều chỉnh và tập trung vào các lĩnh vực và chủ đề phù hợp hơn trong quá trình phân tích.

Dựa trên quá trình nghiên cứu và tham khảo, nhóm em đã xây dựng một số câu hỏi trọng tâm, nhằm hiểu sâu hơn về hành vi mua sắm của các nhóm khách hàng. Các câu hỏi được định hướng để khám phá những khía cạnh quan trọng trong hành vi tiêu dùng theo thời gian, bao gồm:

- Tìm hiểu hành vi của khách hàng theo độ tuổi và giới tính để hiểu thói quen mua sắm trong các nhóm ?
- Tần suất mua sắm của các nhóm khách hàng theo thời gian?
- Xu hướng mua sắm theo mùa thành các tháng và quý để xem liệu có sự gia tăng doanh số trong các mùa lễ hội ?

## II. Nội dung

### 1. Chuẩn bị dữ liệu

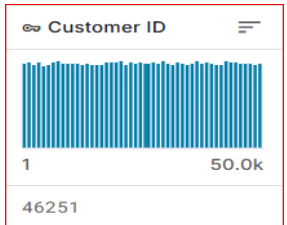
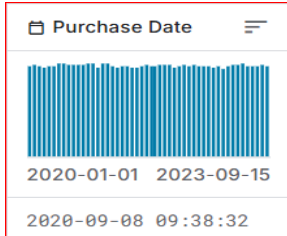
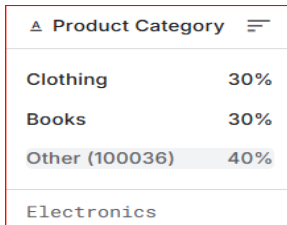
- Phân trình bày chi tiết các thông tin của của bộ dữ liệu :

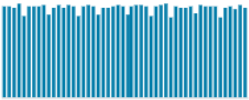

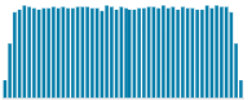
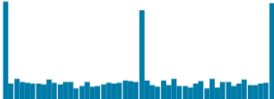

Tên bộ dữ liệu: **E-commerce Customer Data For Behavior Analysis.**

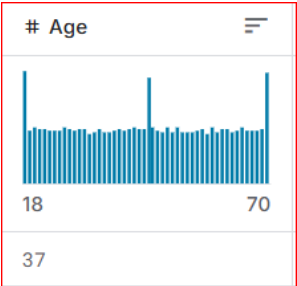
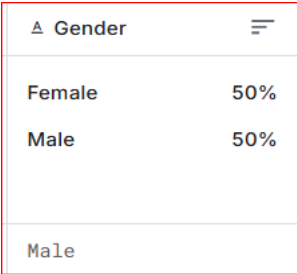
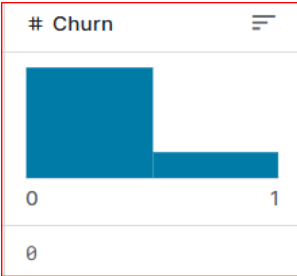
[!\[\]\(c694a3ff3b077d76910920a6a1593ab4\_img.jpg\) E-commerce Customer Data For Behavior Analysis](#)

Nguồn dữ liệu: Kaggle.

Cấu trúc gồm: 250.000 dòng tương ứng với 13 cột phù hợp để đưa ra kết luận tổng quan cho quá trình phân tích dữ liệu về hành vi mua sắm trực tuyến của khách hàng theo thời gian.

Thuộc tính	Mô tả	Minh họa
<b>Customer ID</b>	Mã định danh khách hàng	
<b>Purchase Date</b>	Ngày và giờ giao dịch	
<b>Product Category</b>	Danh mục sản phẩm đã mua ( <b>Electronics, Home, Clothing</b> )	

Product Price	Giá bán của sản phẩm	<div><div># Product Price</div><div></div><div>10500</div><div>12</div></div>						
Quantity	Số lượng sản phẩm được mua	<div><div># Quantity</div><div></div><div>15</div><div>3</div></div>						
Total Purchase Amount	Tổng giá trị giao dịch	<div><div># Total Purchase A...</div><div></div><div>1005350</div><div>740</div><div>2739</div></div>						
Payment Method	Phương thức thanh toán (Credit Card, PayPal, v.v.).	<div><div>▲ Payment Method</div><div><table><tr><td>Credit Card</td><td>40%</td></tr><tr><td>PayPal</td><td>30%</td></tr><tr><td>Other (74677)</td><td>30%</td></tr></table></div><div>Credit Card</div><div>PayPal</div></div>	Credit Card	40%	PayPal	30%	Other (74677)	30%
Credit Card	40%							
PayPal	30%							
Other (74677)	30%							
Customer Age	Độ tuổi của khách hàng.	<div><div># Customer Age</div><div></div><div>1870</div></div>						
Returns	Số lần trả hàng	<div><div># Returns</div><div></div><div>01</div><div>0.0</div></div>						
Customer Name	Tên khách hàng	<div><div>▲ Customer Name</div><div><div>39920 unique values</div><div>Christine Hernandez</div><div>Christine Hernandez</div></div></div>						

<b>Age</b>	Tuổi (trùng với thuộc tính <i>Customer Age</i> ).	
<b>Gender</b>	Giới tính khách hàng	
<b>Churn</b>	Tình trạng khách hàng rời bỏ (0: không rời, 1: đã rời)	

## 2. Xử lý dữ liệu

### 2.1 Các công cụ sử dụng để phân tích

Với đề tài này, nhóm em sử dụng Python, cùng với các gói Pandas, Numpy và Plotly, để thực hiện phân tích của mình. Pandas giúp đơn giản hóa việc xử lý các dataframe 2 chiều, cấu trúc dữ liệu tương tự như các bảng tính Excel, làm cho quá trình làm sạch và thao tác dữ liệu trở nên trực quan hơn. Trong khi đó, bộ công cụ đa dạng của Plotly để tạo các biểu đồ hiện đại, tùy chỉnh và tương tác đảm bảo trải nghiệm trực quan hóa dữ liệu mượt mà và dễ dàng.

### 2.2 Tóm tắt dữ liệu cơ bản và thống kê

```
#Tính toán thống kê tổng quan
data.describe(include='all')
```

**Output:**

	Customer ID	Purchase Date	Product Price	Quantity	Customer Age	Gender
<b>count</b>	250000.00000	250000	250000.000000	250000.000000	250000.000000	250000
<b>unique</b>	NaN	234633	NaN	NaN	NaN	2
<b>top</b>	NaN	9/10/2020 9:00	NaN	NaN	NaN	Female
<b>freq</b>	NaN	5	NaN	NaN	NaN	125560
<b>mean</b>	25004.03624	NaN	254.659512	2.998896	43.940528	NaN
<b>std</b>	14428.27959	NaN	141.568577	1.414694	15.350246	NaN
<b>min</b>	1.00000	NaN	10.000000	1.000000	18.000000	NaN
<b>25%</b>	12497.75000	NaN	132.000000	2.000000	31.000000	NaN
<b>50%</b>	25018.00000	NaN	255.000000	3.000000	44.000000	NaN
<b>75%</b>	37506.00000	NaN	377.000000	4.000000	57.000000	NaN
<b>max</b>	50000.00000	NaN	500.000000	5.000000	70.000000	NaN

```
#Thống kê cho các cột kiểu dữ liệu chuỗi (object)
data.describe(include=['object'])
```

**Output:**

	Purchase Date	Gender
<b>count</b>	250000	250000
<b>unique</b>	234633	2
<b>top</b>	9/10/2020 9:00	Female
<b>freq</b>	5	125560

**Tổng quan về bộ dữ liệu:**

- Có tổng cộng 250,000 giao dịch được ghi nhận, trong đó có 234,633 giá trị ngày giờ khác nhau, cho thấy một số thời điểm mua sắm có thể bị trùng lặp.
- Cột Gender ghi nhận 250,000 khách hàng với 2 nhóm giới tính duy nhất (nam và nữ). Trong đó, khách hàng nữ chiếm ưu thế, với 125,560 giao dịch, cho thấy phụ nữ có xu hướng mua sắm nhiều hơn trong dữ liệu này.

## 2.3 Làm sạch và thao tác dữ liệu

```
import numpy as np
import pandas as pd
import os

# read file
data = pd.read_csv("ecommerce_customer_data_custom_ratios.csv")

# Hiển thị các cột ban đầu
print("Các cột trong dataset:\n", data.columns.tolist())

# Ktra ttin
print(" Kiểm tra thông tin dataset:\n")
data.info() # Lấy thông tin về số lượng và kiểu dữ liệu của các cột trong DataFrame
data.shape # Trả về một tuple chứa tổng số hàng và cột của DataFrame
data.head() # Hiển thị trước 4 hàng đầu tiên của tập dữ liệu
```

### Output:

```
Các cột trong dataset:
['Customer ID', 'Purchase Date', 'Product Category', 'Product Price', 'Quantity', 'Total Purchase Amount', 'Payment Method', 'Customer Age', 'Returns',
'Customer Name', 'Age', 'Gender', 'Churn']
Kiểm tra thông tin dataset:

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 250000 entries, 0 to 249999
Data columns (total 13 columns):
 #   Column                Non-Null Count  Dtype
---  -
 0   Customer ID           250000 non-null  int64
 1   Purchase Date         250000 non-null  object
 2   Product Category     250000 non-null  object
 3   Product Price        250000 non-null  int64
 4   Quantity             250000 non-null  int64
 5   Total Purchase Amount 250000 non-null  int64
 6   Payment Method       250000 non-null  object
 7   Customer Age         250000 non-null  int64
 8   Returns              202404 non-null  float64
 9   Customer Name        250000 non-null  object
10   Age                  250000 non-null  int64
11   Gender               250000 non-null  object
12   Churn                250000 non-null  int64
dtypes: float64(1), int64(7), object(5)
memory usage: 24.8+ MB
```

	Customer ID	Purchase Date	Product Category	Product Price	Quantity	Total Purchase Amount	Payment Method	Customer Age	Returns	Customer Name	Age	Gender	Churn
0	46251	8/9/2020 9:38	Electronics	12	3	740	Credit Card	37	0.0	Christine Hernandez	37	Male	0
1	46251	5/3/2022 12:56	Home	468	4	2739	PayPal	37	0.0	Christine Hernandez	37	Male	0
2	46251	23/5/2022 18:18	Home	288	2	3196	PayPal	37	0.0	Christine Hernandez	37	Male	0
3	46251	12/11/2020 13:13	Clothing	196	1	3509	PayPal	37	0.0	Christine Hernandez	37	Male	0
4	13593	27/11/2020 17:55	Home	449	1	3452	Credit Card	49	0.0	James Grant	49	Female	1

```
# Đếm số lượng giá trị duy nhất trong mỗi cột
print ("Đếm số lượng giá trị duy nhất trong mỗi cột: ")
data.nunique()
```

### Output:



Đếm số lượng giá trị duy nhất trong mỗi cột:

Customer ID	49673
Purchase Date	234633
Product Category	4
Product Price	491
Quantity	5
Total Purchase Amount	5247
Payment Method	4
Customer Age	53
Returns	2
Customer Name	39920
Age	53
Gender	2
Churn	2

dtype: int64

```
# Chuyển cột 'Purchase Date' sang datetime với định dạng DD/MM/YYYY HH:MM
print("\nChuyển định dạng Datetime:")
data['Purchase Date'] = pd.to_datetime(data['Purchase Date'], format='%d/%m/%Y %H:%M')
# Kiểm tra kết quả
print(data['Purchase Date'].head())
```

**Output:**

```
Chuyển định dạng Datetime:
0    2020-09-08 09:38:00
1    2022-03-05 12:56:00
2    2022-05-23 18:18:00
3    2020-11-12 13:13:00
4    2020-11-27 17:55:00
Name: Purchase Date, dtype: datetime64[ns]
```

```
# Danh sách các cột cần xóa
columns_to_remove = ["Customer Name", "Total Purchase Amount", "Payment Method", "Returns", "Age", "Churn"]

# Kiểm tra và xóa cột
existing_columns_to_remove = [col for col in columns_to_remove if col in data.columns]
data = data.drop(columns=existing_columns_to_remove)

# Loại cột bị xóa
print(f"Các cột đã xóa: {existing_columns_to_remove}")
if len(existing_columns_to_remove) < len(columns_to_remove):
    missing_columns = set(columns_to_remove) - set(existing_columns_to_remove)
    print(f"Các cột không tồn tại trong dataset: {missing_columns}")

# Kiểm tra và hiển thị thông tin dataset sau khi xóa cột
print("Dataset sau khi xóa cột:")
print(data.head())
```

**Output:**

Các cột đã xóa: []						
Các cột không tồn tại trong dataset: {'Total Purchase Amount', 'Customer Name', 'Churn', 'Age', 'Returns', 'Payment Method'}						
Dataset sau khi xóa cột:						
	Customer ID	Purchase Date	Product Category	Product Price	Quantity	\
0	46251	2020-09-08 09:38:00	Electronics	12	3	
1	46251	2022-03-05 12:56:00	Home	468	4	
2	46251	2022-05-23 18:18:00	Home	288	2	
3	46251	2020-11-12 13:13:00	Clothing	196	1	
4	13593	2020-11-27 17:55:00	Home	449	1	

	Customer Age	Gender
0	37	Male
1	37	Male
2	37	Male
3	37	Male
4	49	Female

**Loại bỏ các thuộc tính dư thừa:** Sau khi tìm hiểu, phân tích và xây dựng một số câu hỏi trọng tâm, nhóm em quyết định loại bỏ những cột không cần thiết trong bài phân tích này và giữ lại 7 cột sau:

**Customer ID, Purchase Date, Product Category, Product Price, Quantity, Customer Age và Gender.**

### Lý do loại bỏ:

- Total Purchase Amount:

- + Tổng số tiền có thể được tính lại dễ dàng từ Product Price \* Quantity nếu cần.
- + Không phải yếu tố trọng tâm để phân tích hành vi mua sắm theo thời gian.

- Payment Method:

- + Phương thức thanh toán không liên quan trực tiếp đến mục tiêu phân tích.
- + Nó có thể quan trọng trong bài phân tích khác (như hành vi thanh toán), nhưng không cần thiết ở đây.

- Returns:

- + Thông tin về hàng trả lại không phải trọng tâm trong việc phân tích hành vi theo thời gian.
- + Nếu bài phân tích muốn đánh giá chất lượng sản phẩm, thì cột này mới cần thiết.

- Customer Name:

- + Không mang ý nghĩa phân tích vì Customer ID đã đại diện duy nhất cho từng khách hàng.
- + Lưu giữ thông tin này chỉ làm tăng kích thước dữ liệu mà không đóng góp thêm giá trị phân tích.

- Churn:

- + Mặc dù có thể liên quan đến hành vi mua sắm, nhưng nó là một chỉ số dài hạn.
- + Không cần thiết nếu mục tiêu là tập trung vào hành vi theo thời gian.

- Age: Bị trùng với cột Customer Age.

### Lý do giữ lại các cột:

- Customer ID: Giúp xác định ai là người mua sắm, phân nhóm khách hàng để phân tích hành vi.

- Product Price: Giúp phân tích số lượng và hiểu rõ hơn về tổng giá trị giao dịch theo mùa

- Product Category: Đo lường mức độ phổ biến và tổng sức mua của từng loại sản phẩm trong các nhóm tuổi/nhóm giới tính
- Purchase Date: Là cột trọng tâm để phân tích hành vi theo thời gian, giúp tìm hiểu tần suất, thời gian mua hàng.
- Quantity: Giúp đánh giá khối lượng hàng hóa được mua, phản ánh xu hướng mua sắm.
- Customer Age: Cung cấp thông tin nhân khẩu học, hỗ trợ phân tích sự khác biệt về hành vi giữa các nhóm tuổi.
- Gender: Thêm góc nhìn về sự khác biệt giới tính trong hành vi mua hàng.

```
# Lưu file mới
updated_file_path = 'updated_dataset.csv'
try:
    data.to_csv(updated_file_path, index=False)
    print(f"Dataset đã được lưu tại: {updated_file_path}")
except Exception as e:
    raise Exception(f"Lỗi khi lưu dataset: {e}")
```

#### ❖ Kiểm tra giá trị null

```
# Tìm tất cả các dòng có ít nhất một giá trị null
rows_with_null = data[data.isnull().any(axis=1)]
#Kiểm tra nếu bất kỳ giá trị nào trong hàng (axis=1) là null.

print(rows_with_null)
```

#### Output:

```
Empty DataFrame
Columns: [Customer ID, Purchase Date, Product Category, Product Price, Quantity, Customer Age, Gender]
Index: []
```

→ Qua output này ta nhận thấy bộ dataset không có dòng nào có giá trị null.

#### ❖ Kiểm tra giá trị Duplicates

```
#Kiểm tra duplicates, trả về tất cả cột duplicates
data[data.duplicated(keep=False)]
```

#### Output:

Customer ID	Purchase Date	Product Category	Product Price	Quantity	Customer Age	Gender
-------------	---------------	------------------	---------------	----------	--------------	--------

→ Qua output này ta nhận thấy bộ dataset không có dòng nào có giá trị duplicates.

#### ❖ Kiểm tra giá trị ngoại lai

```
#Phát hiện giá trị ngoại lai
# Tính IQR cho cột Age
Q1_age = data['Customer Age'].quantile(0.25) #quý 1 - 25%
Q3_age = data['Customer Age'].quantile(0.75) #quý 3 - 75%
IQR_age = Q3_age - Q1_age

lower_bound_age = Q1_age - 1.5 * IQR_age
upper_bound_age = Q3_age + 1.5 * IQR_age

# Phát hiện giá trị ngoại lai trong cột Age
outliers_age = data[(data['Customer Age'] < lower_bound_age) | (data['Customer Age'] > upper_bound_age)]
print("Outliers in Age:")
print(outliers_age)
```

## Output:

```
Outliers in Age:
Empty DataFrame
Columns: [Customer ID, Purchase Date, Product Category, Product Price, Quantity, Customer Age, Gender]
Index: []
```

→ Tìm giá trị ngoại lai của thuộc tính 'Customer Age' ngoài khoảng quy định [18, 70]. Qua Output, ta thấy không có giá trị nào ngoại lai ngoài khoảng quy định trên.

## 3. Phân tích dữ liệu

### 3.1 Tìm hiểu hành vi của khách hàng theo độ tuổi và giới tính để hiểu thói quen mua sắm trong các nhóm ?

#### 3.1.1 Phân tích và trực quan hóa theo xu hướng giới tính (Trends about Gender)

##### a. Kiểm tra định dạng Dataframe

```
import pandas as pd
import os

#đọc file
df = pd.read_csv('C:/Users/BICH THUY/DATA_A/updateData_CategoryProduct.csv')

#hiển thị 5 hàng đầu tiên của DataFrame df (check định dạng/nội dung)
df.head()
```

## Output:

	Customer ID	Purchase Date	Product Category	Product Price	Quantity	Customer Age	Gender
0	46251	08/09/2020 09:38:00	Electronics	12	3	37	Male
1	46251	05/03/2022 12:56:00	Home	468	4	37	Male
2	46251	23/05/2022 18:18:00	Home	288	2	37	Male
3	46251	12/11/2020 13:13:00	Clothing	196	1	37	Male
4	13593	27/11/2020 17:55:00	Home	449	1	49	Female

→ Hiển thị 5 hàng đầu tiên của Dataframe, ta thấy dữ liệu đúng định dạng và nội dung theo mong đợi.

b. Số liệu thống kê tổng quan theo từng nhóm Giới tính và Tổng chi tiêu

**Mục tiêu:**

- Tóm tắt dữ liệu: Cung cấp cái nhìn tổng quan về hành vi chi tiêu của khách hàng theo giới tính.
- So sánh: Dễ dàng so sánh các chỉ số như trung bình chi tiêu hoặc tổng số khách hàng giữa hai nhóm Male (Nam giới) và Female (Nữ giới)
- Phân tích: Dùng để phát hiện xu hướng, hành vi khác biệt giữa các nhóm giới tính nhằm hỗ trợ chiến lược kinh doanh hoặc tiếp thị.

```
# Thống kê tổng quan theo từng nhóm Gender và Tổng chi tiêu
#groupby('Gender'): chia dataset và thực hiện các phép tính thành 2 nhóm Gtính
#agg(): thực hiện các phép toán thống kê của nhiều cột dữ liệu khác nhau (đã được GROUPBY)
#Bang chua cac GTri thong ke theo dong tuong ung

gender_summary = df.groupby('Gender').agg(total_count=('Customer ID','size'), #tong so KH theo gender
min_amount=('Product Price', 'min'), #GTNN(Total Price) theo Gender
max_amount=('Product Price', 'max'), #GTLN(Total Price) theo Gender
median_amount=('Product Price','median'), #GT trung vi tong chi tien theo Gender
total_amount=('Product Price','sum'), #tong tien chi tieu tat ca KH theo Gender
average_amount=('Product Price','mean'), #trung binh tien chi tieu theo Gender
)

print("\nThống kê tổng quan theo từng nhóm Gender và Tổng chi tiêu:")
print(gender_summary)
```

**Output:**

```
Thống kê tổng quan theo từng nhóm Gender và Tổng chi tiêu:
      total_count  min_amount  max_amount  median_amount  total_amount
Gender
Female         125560         10         500          254.0      31959573
Male          124440         10         500          255.0      31705305

      average_amount
Gender
Female         254.536262
Male          254.783872
```

→ **Nhận xét kết quả thống kê:**

- Số lượng khách hàng nữ và nam khá gần nhau, cho thấy sự phân bố giới tính trong nhóm khách hàng
  - Có thể thấy không có sự khác biệt quá lớn về hành vi mua sắm giữa nam và nữ trong nhóm khách hàng này. Cả hai giới đều có xu hướng chi tiêu tương đương nhau và có sự đa dạng trong mức chi tiêu.
- Để có cái nhìn toàn diện hơn, cần tiến hành phân tích sâu hơn bằng cách chia nhỏ dữ liệu theo các yếu tố khác như độ tuổi, sản phẩm, thời gian mua hàng, v.v.. Điều này sẽ giúp tìm ra những đặc điểm tiêu dùng cụ thể hơn của từng nhóm khách hàng.

c. Biểu đồ phân phối theo nhóm giới tính (Gender Distribution)

Biểu đồ phân phối theo nhóm giới tính (nam nữ) là một trong các phần thiết yếu để **tóm tắt bức tranh tổng thể về cơ cấu Khách hàng** thông qua việc hiểu rõ đối tượng khách hàng (phân khúc khách hàng nào chiếm ưu thế hơn), phân tích và so sánh hành vi (sở thích, nhu cầu, mua sắm) giữa nam - nữ. Từ đó, nhằm **tối ưu hóa**

**chiến lược kinh doanh, tiếp thị và sản phẩm**, đồng thời **hiểu rõ thói quen và sở thích của khách hàng** để phục vụ họ tốt hơn.

```
##1. Gender distribution (Phân tích và trực quan hóa phân phối giới tính)

# cài đặt thư viện
import plotly.express as px
gender_count = df['Gender'].value_counts() # Tính toán Số lượng Khách theo Giới Tính (Age)

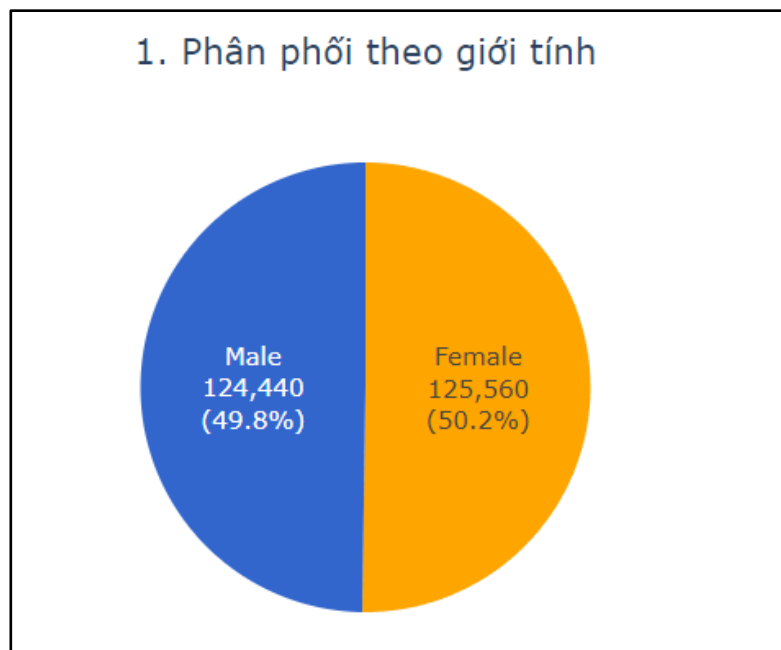
# Tạo biểu đồ Pie Chart
# (px) là thư viện plotly.express dùng vẽ Pie Chart
fig1 = px.pie(values=gender_count,
              names=gender_count.index,
              color=gender_count.index,
              color_discrete_map={
                  'Female': '#FF6692',
                  'Male': '#3366CC'
              },
              title = '1. Phân phối theo giới tính')

# điều chỉnh văn bản hiển thị trong biểu đồ tròn
fig1.update_traces(textposition='inside', textinfo='text',
                  texttemplate='%{label}<br>%{value}<br>(%{percent})')

# điều chỉnh kích thước và ẩn chú giải
fig1.update_layout(title={'x': 0.5, 'y': 0.9},
                  width=400, height=400, showlegend=False)

# hiển thị biểu đồ
fig1.show()
# lưu biểu đồ dưới dạng hình ảnh
fig1.write_image("Gender Distribution.png")
```

**Output:**



Biểu đồ phân phối này cho thấy:

→ Số lượng Khách hàng theo nhóm Giới tính có sự cân bằng tương đối giữa hai đối tượng khách hàng nam và nữ (tỷ lệ phần trăm).

→ Với sự cân bằng này, cả hai nhóm giới tính đều đóng vai trò quan trọng và có sức ảnh hưởng ngang nhau đến doanh thu hoặc hành vi mua sắm. Vì vậy, **chiến lược kinh doanh và tiếp thị cần được thiết kế linh hoạt để đáp ứng cả hai nhóm.**

→ Tuy nhiên, sự cân bằng về số lượng khách hàng không đồng nghĩa với việc hành vi mua sắm của họ là giống nhau.

- Cần so sánh thêm **tổng chi tiêu (Total Spending)** theo từng nhóm giới tính/độ tuổi để hiểu rõ nhóm khách hàng nào chi tiêu nhiều hơn hoặc có giá trị mua sắm lớn hơn.
- Hiểu rõ hành vi mua sắm theo từng nhóm đối tượng khách hàng (theo giới tính) nhằm tạo các chiến dịch tiếp thị riêng biệt, tập trung vào sở thích và nhu cầu đặc thù của nam và nữ để tối ưu hóa hiệu quả tiếp cận.

#### d. Phân tích tổng chi tiêu theo giới tính (Total Spending by Gender)

##### **Mục tiêu:**

- Nhận biết/Phản ánh mức độ tiêu dùng thực tế, hành vi mua sắm của từng giới tính, giúp trả lời các câu hỏi cụ thể như:
  - + Nhóm khách hàng nào (nam hay nữ) có mức chi tiêu cao hơn?
  - + Mức độ ưu tiên của từng giới tính khi chi tiêu vào các sản phẩm/dịch vụ?
- Phân tích tổng chi tiêu giúp doanh nghiệp **phân khúc khách hàng hiệu quả hơn**, dựa trên khả năng và xu hướng chi tiêu của từng giới tính.
- Kiểm tra xem liệu các chiến lược tiếp thị hiện tại có hiệu quả hay không
- Dự đoán xu hướng và định hướng kinh doanh dài hạn: Thói quen chi tiêu thường thay đổi theo thời gian, ngành hàng, và bối cảnh kinh tế. Việc theo dõi và phân tích tổng chi tiêu giữa các giới tính giúp doanh nghiệp dự đoán các xu hướng trong tương lai.

#### **Biểu đồ phân tích tổng chi tiêu theo giới tính (Total Spending by Gender)**

```

# Import thư viện
import pandas as pd
import plotly.express as px # Mã vẽ biểu đồ không thay đổi, chỉ update data từ gender_summa

# Tính toán tổng chi tiêu (total_spending) cho từng dòng dữ liệu
df['Total Spending'] = df['Product Price'] * df['Quantity']

# Tính tổng chi tiêu theo giới tính
gender_summary = df.groupby('Gender', as_index=False).agg({'Total Spending': 'sum'})

# Tính phần trăm tổng chi tiêu theo giới tính
sum_spending = gender_summary['Total Spending'].sum()
gender_summary['Total Spending Percentage'] = (gender_summary['Total Spending'] / sum_spending)

# Vẽ biểu đồ bar
fig2 = px.bar(
    gender_summary,
    x='Total Spending',
    y='Gender',
    orientation='h',
    color='Gender',
    color_discrete_map={'Female': '#FF6692', 'Male': '#3366CC'},
    title='2. Tổng chi tiêu theo giới tính',
    text=gender_summary.apply(
        lambda x: f"${x['Total Spending']:,}<br>({x['Total Spending Percentage']:.2f}%)",
        axis=1
    ),
    labels={'Total Spending': 'Total Spending ($)'})

```

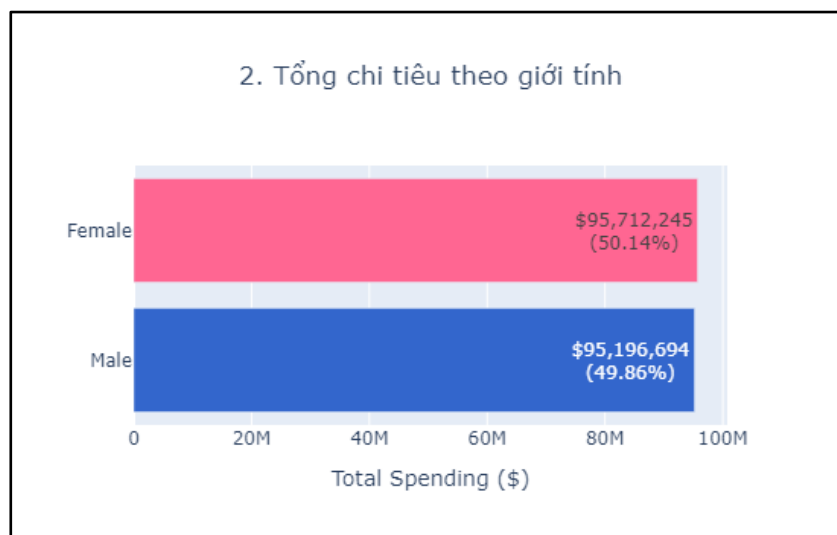
```

# Tùy chỉnh layout
fig2.update_layout(
    title={'x': 0.5, 'y': 0.9},
    yaxis_title=None,
    width=550,
    height=350,
    showlegend=False
)

# Hiển thị biểu đồ
fig2.show()

```

**Output:**





→ Từ biểu đồ tổng chi tiêu theo giới tính, ta thấy rằng tổng chi tiêu của 2 nhóm khách hàng nam (\$95,196,694) và nữ (\$95,712,245) khá cân bằng, với sự chênh lệch không quá lớn. Nói cách khác, hai nhóm Khách hàng giới tính nam và nữ đều đóng góp đáng kể vào tổng doanh thu.

→ Tuy nhiên, để phân tích mức độ ưu tiên chi tiêu vào các sản phẩm/dịch vụ cụ thể của từng giới cần bổ sung thêm thông tin như '**Product Category**' để phân chia sản phẩm thành các nhóm (ví dụ: thời trang, điện tử, mỹ phẩm) để so sánh tỷ lệ chi tiêu của nam và nữ vào từng nhóm.

#### e. Phân tích tổng chi tiêu theo mặt hàng và giới tính (Total Spending by Product Category and Gender)

##### **Mục tiêu:**

Cung cấp cái nhìn chi tiết về thói quen tiêu dùng của từng nhóm khách hàng trong hành vi mua sắm giữa nam - nữ nhằm tạo sự khác biệt trong trải nghiệm khách hàng thông qua một số câu hỏi như:

- **Mức độ ưu tiên của từng giới tính khi chi tiêu vào các sản phẩm/dịch vụ?**
- **Giới tính nào chi tiêu nhiều hơn vào từng loại sản phẩm?**
  - + Biết được sản phẩm nào thu hút sự chú ý của mỗi giới tính, từ đó có thể tối ưu hóa chiến lược marketing cho từng nhóm khách hàng.
- **Loại sản phẩm nào được nam/nữ chi tiêu nhiều nhất?**
  - + Xác định rõ ràng các mặt hàng yêu thích của mỗi giới tính để có thể điều chỉnh danh mục sản phẩm, chiến lược quảng cáo, hoặc các chương trình khuyến mãi phù hợp.
- **Có sự khác biệt lớn về chi tiêu giữa các sản phẩm trong mỗi giới tính không?**
- **Tỷ lệ chi tiêu của từng giới tính trong các mặt hàng có sự chênh lệch lớn không?**

```
#PTICH TỔNG CHI TIÊU THEO MẶT HÀNG VÀ GIỚI TÍNH
import plotly.express as px

# Tính toán tổng chi tiêu (total_spending) cho từng dòng dữ liệu
df['Total Spending'] = df['Product Price'] * df['Quantity']
# Tính tổng chi tiêu cho từng sản phẩm theo giới tính
gender_category_summary = df.groupby(['Gender', 'Product Category'], as_index=False).agg({'Total Spending': 'sum'})

# Tính tổng chi tiêu của từng giới tính
gender_spending_summary = gender_category_summary.groupby('Gender', as_index=False).agg({'Total Spending': 'sum'})

# Vẽ biểu đồ so sánh chi tiêu theo sản phẩm và giới tính
fig3 = px.bar(
    gender_category_summary,
    x='Product Category',
    y='Total Spending',
    color='Gender',
    title="3. Tổng Chi tiêu theo mặt hàng và giới tính",
    labels={'Total Spending': 'Total Spending ($)'},
    barmode='group', # Hiển thị các cột cho từng giới tính
    text='Total Spending' # Hiển thị giá trị chi tiêu trên từng cột
)

# Cập nhật layout cho biểu đồ
fig3.update_layout(
    xaxis_title=None,
    yaxis_title='Total Spending ($)',
    title={'x': 0.5, 'y': 0.9},
    height=400,
    showlegend=True
)

# Hiển thị biểu đồ
fig3.show()
```

## Output:



→ Sự khác biệt thói quen mua sắm/Hành vi tiêu dùng giữa nhóm khách hàng nam và nữ, thông qua biểu đồ này:

- **Mức độ ưu tiên của từng giới tính khi chi tiêu vào các sản phẩm/dịch vụ?**

### + Nữ giới:

- Chi tiêu cao nhất cho Quần áo (Clothing) (\$29,005,863)
- Top 2: Sách (Books) (\$28,491,117)
- Top 3: Điện tử (Electronics) (\$19,164,484)
- Top 4: Nhà cửa (Home) (\$19,132,781)

### + Nam giới:

- Chi tiêu cao nhất cho Sách (Books) (\$28,509,860)
- Top 2: Quần áo (Clothing) (\$28,275,322)

- Top 3: Điện tử (Electronics) (\$19,285,454)
- Top 4: Nhà cửa (Home) (\$19,045,038)

**- Giới tính nào chi tiêu nhiều hơn vào từng loại sản phẩm?**

- + **Sách (Books):** Nam giới (\$28,509,860) chi tiêu nhiều hơn nữ giới (\$28,491,117)
- + **Quần áo (Clothing):** Nữ giới (\$29,005,863) chi tiêu nhiều hơn nam giới (\$28,275,322)
- + **Điện tử (Electronics):** Nam giới (\$19,285,454) chi tiêu nhiều hơn nữ giới (\$19,164,484)
- + **Nhà cửa (Home):** Nữ giới (\$19,132,781) chi tiêu nhiều hơn nam giới (\$19,045,038)

→ **Nữ giới và nam giới:** Quần áo và Sách là hai sản phẩm thu hút nhiều sự chú ý nhất.

→ Tập trung vào Quần áo và Sách cho danh mục sản phẩm, chiến lược quảng cáo, và các chương trình khuyến mãi dành cho cả 2 nhóm Khách hàng này.

**- Loại sản phẩm nào được nam/nữ chi tiêu nhiều nhất? (Xác định rõ ràng các mặt hàng yêu thích của mỗi giới tính)**

- + **Nữ giới:** Quần áo (\$29,005,863)
- + **Nam giới:** Sách (\$28,509,860)

**- Có sự khác biệt lớn về chi tiêu giữa các sản phẩm trong mỗi giới tính không? (Phân tích mức độ đa dạng trong việc chi tiêu của mỗi giới tính nhằm đúng vào nhu cầu cụ thể của khách hàng)**

- + **Nữ giới:** Có sự khác biệt rõ rệt, với Quần áo có chi tiêu cao nhất và Nhà cửa có chi tiêu thấp nhất.
- + **Nam giới:** Sự khác biệt ít rõ ràng hơn, nhưng Sách có chi tiêu cao nhất và Nhà cửa có chi tiêu thấp nhất.

**- Tỷ lệ chi tiêu giữa các giới tính cho từng loại sản phẩm khá cân đối, không có sự chênh lệch lớn. Sự khác biệt lớn nhất là ở mặt hàng Quần áo, số liệu cho thấy nữ giới chi tiêu nhiều hơn nam giới.**

### 3.1.2 Phân tích và trực quan hóa theo xu hướng tuổi (Trends about Age)

a. Định nghĩa/Phân loại khách hàng vào các nhóm tuổi (age groups):

Các nhóm tuổi (Age groups/ Age Range): '0-18', '18-29', '30-39', '40-49', '50-59', '60-70', '70+'

```
# sử dụng np để tạo mảng
import numpy as np

# Định nghĩa các nhóm tuổi
age_bins = [0, 18, 29, 39, 49, 59, 70, np.inf] # Bao gồm 7 gtri từ 0 đến gtri vô cùng
# gán nhãn cho các nhóm tuổi trong age_bins
age_labels = ['0-18', '18-29', '30-39', '40-49', '50-59', '60-70', '70+'] # Đúng 6 nhãn

# Thêm cột "Age Group" vào DataFrame df cho biết nhóm tuổi mỗi KH thuộc về
#pd.cut: hàm pandas chia data theo các nhóm tuổi thuộc bins
df['Age Group'] = pd.cut(df['Customer Age'], bins=age_bins, labels=age_labels)

# ktra và độ tuổi KH xếp theo thứ tự giảm dần
df[['Customer ID', 'Customer Age', 'Age Group']].sort_values('Customer Age', ascending=False)

# Ktr các gtri thiếu missing values trong "Age Group" -> Đã xử lý ở bước tiền xử lý: Ko lỗi
#missing_age_groups = df[df['Age Group'].isnull()]
#print(f"Số giá trị thiếu trong 'Age Group': {len(missing_age_groups)}")
```

**Output:**

	Customer ID	Customer Age	Age Group
<b>109802</b>	38123	70	60-70
<b>8996</b>	28355	70	60-70
<b>115007</b>	20140	70	60-70
<b>115008</b>	20140	70	60-70
<b>42599</b>	34705	70	60-70
...	...	...	...
<b>159476</b>	35947	18	0-18
<b>159475</b>	35947	18	0-18
<b>159474</b>	35947	18	0-18
<b>159473</b>	35947	18	0-18
<b>22907</b>	49598	18	0-18

b. Phân phối độ tuổi theo nhóm tuổi (Age Distribution):

```
## 1. PHÂN PHỐI ĐỘ TUỔI THEO ĐỘ TUỔI

# Đảm bảo cột 'Age Group' trong df được phân loại đúng theo age_labels
df['Age Group'] = pd.Categorical(df['Age Group'], categories=age_labels, ordered=True)

# Lọc bỏ dữ liệu không hợp lệ hoặc các nhóm không liên quan như '<18'
df_filtered = df[df['Age Group'].notnull() & (df['Age Group'] != '0-18')] # Loại bỏ nhóm '<18' rõ ràng

# Đếm tổng số khách hàng sau khi lọc
total_customers = df_filtered['Age Group'].value_counts().sum()

# Đếm số lượng người trong mỗi nhóm độ tuổi
age_group_count = df_filtered['Age Group'].value_counts().sort_index().reset_index(name='Count')
age_group_count.rename(columns={'index': 'Age Group'}, inplace=True) # Đổi tên cột cho dễ hiểu

# Tính tỷ lệ phần trăm số lượng khách hàng trong mỗi nhóm tuổi từ age_group_count
age_group_count['Percentage'] = (age_group_count['Count'] / total_customers) * 100

# Thêm cột hiển thị tỷ lệ phần trăm dạng chuỗi (thêm ký hiệu %)
age_group_count['Percentage'] = age_group_count['Percentage'].round(2).astype(str) + '%'

# In kết quả
#print(age_group_count)
```

```
# Vẽ Line chart + bars
import plotly.express as px

fig1 = px.line(age_group_count,
               x='Age Group',
               y='Count',
               title='1. Phân phối theo độ tuổi',
               markers=True,
               text='Percentage',
               labels={'Count': 'Số lượng Khách Hàng'})

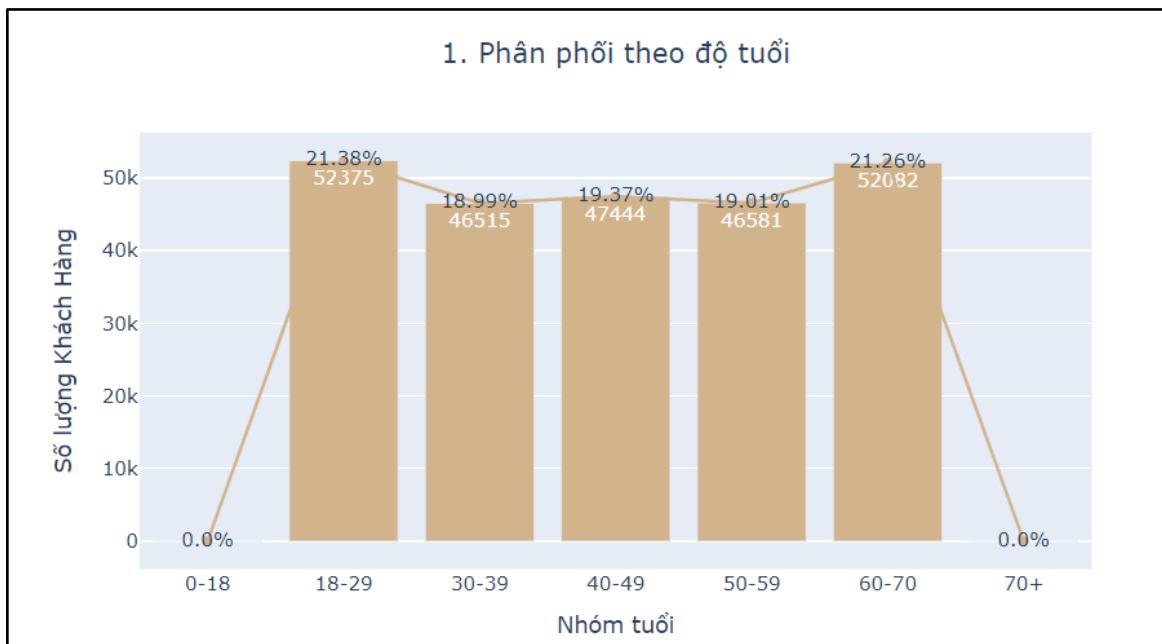
# Layout design
fig1.update_traces(line=dict(color='#D2B48C')) # Đổi màu đường line thành nâu nhạt
fig1.update_traces(marker=dict(color='#D2B48C'), selector=dict(type='scatter')) # Đổi màu marker

fig1.add_bar(x=age_group_count['Age Group'],
             y=age_group_count['Count'],
             text=age_group_count['Count'],
             textposition='inside',
             name='Count',
             marker=dict(color='#D2B48C'))

fig1.update_traces(textfont_color='white', selector=dict(type='bar')) # Đổi màu chữ hiển thị trên bar
fig1.update_layout(title={'x': 0.5, 'y': 0.9}, # Căn giữa tiêu đề
                  xaxis_title='Nhóm tuổi',
                  yaxis_title='Số lượng Khách Hàng',
                  width=750, height=450, # Giảm kích thước biểu đồ xuống 85%
                  showlegend=False)

fig1.show()
```

**Output:**



→ **Tổng quan:** Biểu đồ thể hiện sự tập trung khách hàng cao ở các độ tuổi trưởng thành (18-70), với sự giảm mạnh ở hai đầu (0-18 và 70+).

→ **Nhóm tuổi có số lượng khách hàng cao nhất:**

- Nhóm **18-29** và **60-70** có số lượng khách hàng cao nhất, chiếm hơn 21% mỗi nhóm. Điều này cho thấy hai nhóm này có xu hướng tham gia mua sắm nhiều nhất, có thể vì lý do về nhu cầu, thói quen, hoặc hành vi đặc thù.

→ **Nhóm tuổi có số lượng khách hàng thấp:** Nhóm **0-18** và **70+** không có khách hàng (0%)

- Nhóm **0-18:** Đây có thể là do khách hàng trong độ tuổi này chưa có sức mua lớn hoặc phụ thuộc vào cha mẹ.
- Nhóm **70+:** Khả năng cao đây là do sự hạn chế về sức khỏe, ít tiếp cận công nghệ hoặc nhu cầu tiêu dùng giảm.

→ Các nhóm tuổi từ **30-39**, **40-49**, và **50-59** có tỷ lệ khách hàng khá đồng đều (dao động từ 18.99% đến 19.37%). Điều này cho thấy các nhóm tuổi này cũng đóng góp đáng kể vào số lượng khách hàng tổng thể.

c. Phân tích tổng chi tiêu theo mặt hàng và nhóm tuổi (Total Spending by Product Category and Age Group):

**Mục tiêu:**

- Xác định nhóm tuổi nào tiêu thụ mạnh nhất/đóng góp nhiều nhất vào doanh thu tương ứng với từng loại sản phẩm
- Đo lường mức độ phổ biến và tổng sức mua của từng loại sản phẩm trong các nhóm tuổi.

- Xác định nhóm khách hàng **tiêu thụ mạnh nhất** để tối ưu hóa chiến lược marketing hoặc phân bổ sản phẩm.

```
import pandas as pd
import plotly.express as px

# Giả sử df là DataFrame của bạn
# Kiểm tra và tạo cột 'Total Spending' nếu chưa tồn tại
if 'Total Spending' not in df.columns:
    # Kiểm tra xem các cột 'Product Price' và 'Quantity' có giá trị hợp lệ không
    if 'Product Price' in df.columns and 'Quantity' in df.columns:
        df['Total Spending'] = df['Product Price'].fillna(0) * df['Quantity'].fillna(0)
    else:
        raise ValueError("Cột 'Product Price' hoặc 'Quantity' không tồn tại trong DataFrame.")
# Xử lý dữ liệu trước khi nhóm
# Thêm giá trị "Unknown" vào các cột category nếu chưa có
if 'Age Group' in df.columns and df['Age Group'].dtype.name == 'category':
    df['Age Group'] = df['Age Group'].cat.add_categories(['Unknown'])
    df['Age Group'] = df['Age Group'].fillna("Unknown") # Gán lại giá trị cho cột

if 'Product Category' in df.columns and df['Product Category'].dtype.name == 'category':
    df['Product Category'] = df['Product Category'].cat.add_categories(['Unknown'])
    df['Product Category'] = df['Product Category'].fillna("Unknown") # Gán lại giá trị cho cột

# Chuyển đổi các cột thành kiểu chuỗi nếu cần -> Tránh lỗi khi nhóm
df['Age Group'] = df['Age Group'].astype(str)
df['Product Category'] = df['Product Category'].astype(str)
```

```
# Tính tổng chi tiêu cho từng sản phẩm theo độ tuổi
age_category_summary = df.groupby(['Age Group', 'Product Category'], as_index=False).agg({'Total Spending': 'sum'})
# Kiểm tra sau khi nhóm dữ liệu
#print(len(age_category_summary)) # Kiểm tra số dòng sau khi nhóm theo giới tính và mặt hàng

# Tính tổng chi tiêu của từng độ tuổi
age_spending_summary = age_category_summary.groupby('Age Group', as_index=False).agg({'Total Spending': 'sum'})

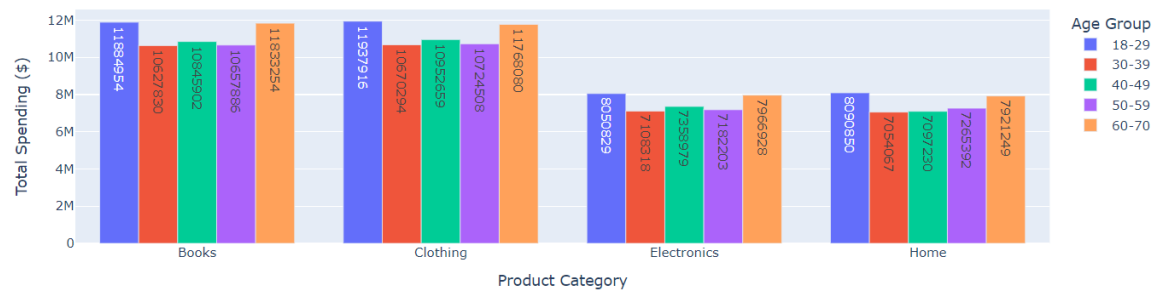
# Vẽ biểu đồ so sánh chi tiêu theo sản phẩm và độ tuổi
fig4 = px.bar(
    age_category_summary,
    x='Product Category',
    y='Total Spending',
    color='Age Group', # Nhóm màu theo độ tuổi
    title="2. Chi tiêu theo mặt hàng và độ tuổi",
    labels={'Total Spending': 'Total Spending ($)'},
    barmode='group', # Hiển thị các cột cho từng độ tuổi
    text='Total Spending' # Hiển thị giá trị chi tiêu trên từng cột
)

# Cập nhật layout cho biểu đồ
fig4.update_layout(
    xaxis_title='Product Category',
    yaxis_title='Total Spending ($)',
    title={'x': 0.5, 'y': 0.9},
    height=400,
    showlegend=True
)

# Hiển thị biểu đồ
fig4.show()
```

**Output:**

2. Tổng Chi tiêu theo mặt hàng và độ tuổi



→ Biểu đồ cho thấy:

- Loại sản phẩm nào thu hút sự chú ý, loại sản phẩm nào được nhóm tuổi chi tiêu nhiều nhất:

- + **Sách (Books):** Nhóm tuổi 18-29 chi tiêu cho sách nhiều nhất, có thể do nhu cầu học tập, làm việc hoặc sở thích đọc sách cao hơn.
- + **Quần áo (Clothing):** Mức chi tiêu cho quần áo khá đồng đều giữa các nhóm tuổi, cho thấy nhu cầu về quần áo là tương đối ổn định.
- + **Đồ điện tử (Electronics):** Tương tự như quần áo, mức chi tiêu cho đồ điện tử cũng khá đồng đều, cho thấy nhu cầu về các sản phẩm công nghệ là phổ biến ở nhiều nhóm tuổi.
- + **Gia dụng (Home):** Nhóm tuổi 18-29 và 30-39 chi tiêu nhiều nhất cho đồ dùng gia đình, có thể do nhu cầu thiết lập cuộc sống riêng hoặc nâng cấp không gian sống.

→ **Quần áo (Clothing)** và **Sách (Books)** là hai loại sản phẩm được chi tiêu nhiều nhất bởi hầu hết các nhóm tuổi.

- **Chi tiêu cao nhất:** Nhóm tuổi **18-29** có xu hướng **chi tiêu cao nhất cho hầu hết các mặt hàng, đặc biệt là sách và đồ dùng gia đình**. Điều này cho thấy nhóm tuổi trẻ hơn có xu hướng tiêu dùng nhiều hơn và có nhu cầu đa dạng về các sản phẩm.

- **Mức chi tiêu** cho các mặt hàng **quần áo và đồ điện tử khá đồng đều** giữa các nhóm tuổi, trong khi chi tiêu cho sách và đồ dùng gia đình có sự phân hóa rõ rệt hơn.

- Tỷ lệ chi tiêu khác biệt giữa các nhóm tuổi trong các mặt hàng: Có sự khác biệt đáng kể về mức chi tiêu giữa các nhóm tuổi. Nhóm tuổi 60-70 có mức chi tiêu thấp nhất cho tất cả các mặt hàng, trong khi nhóm tuổi 18-29 và 30-39 có mức chi tiêu cao nhất.

d. Phân tích chi tiêu trung bình theo độ tuổi/nhóm tuổi (Average Spending by Age/Age Group):

**Mục tiêu:**

- Xác định nhóm tuổi nào có xu hướng **chi tiêu lớn nhất trên mỗi giao dịch cá nhân**, hiểu được sự khác biệt trong hành vi tiêu dùng của các nhóm tuổi.



- Tìm hiểu thói quen mua sắm cá nhân và mức độ ưu tiên chi tiêu của từng nhóm.

```
#3. PTICH CHI TIÊU TRUNG BÌNH THEO NHÓM TUỔI (Average spending by age group)

# Tính toán tổng chi tiêu (total_spending) cho từng dòng dữ liệu
df['Total Spending'] = df['Product Price'] * df['Quantity']
# calculate overall and average of total spending for each age group
overall_avg_spending = df['Total Spending'].mean()
#print(overall_avg_spending)

#tính trung bình chi tiêu theo nhóm tuổi
avg_spending_age_group = df.groupby(['Age Group'])['Total Spending'].mean().reset_index(name='Avg Spending')

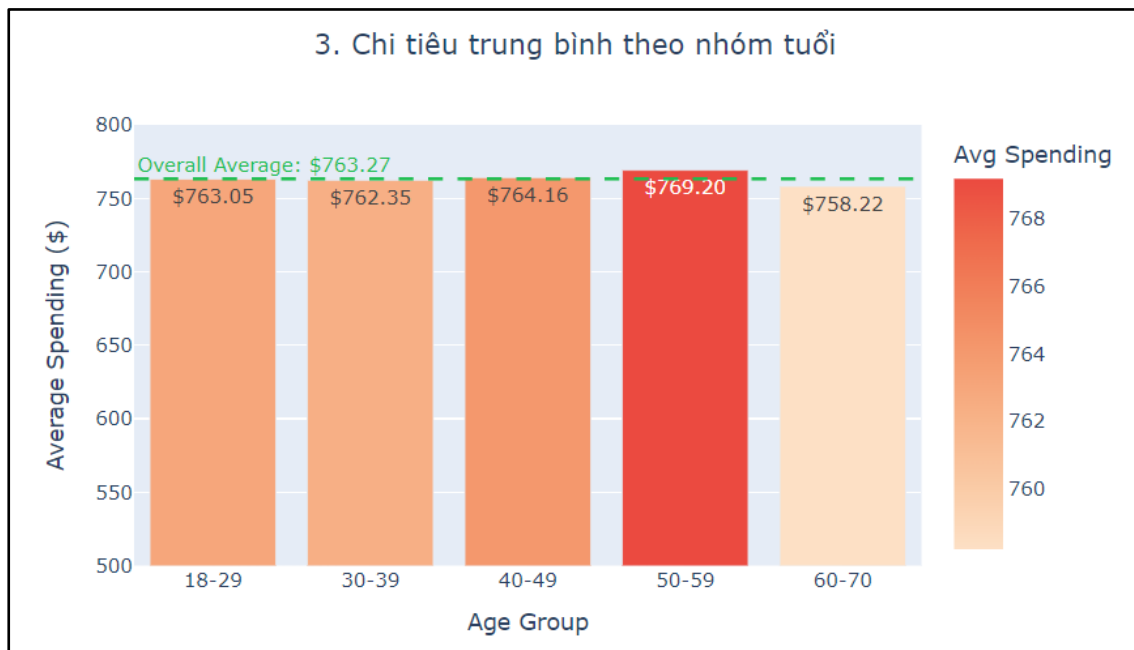
#vẽ biểu đồ
fig6 = px.bar(avg_spending_age_group, x= 'Age Group', y= 'Avg Spending',
              color = 'Avg Spending',
              color_continuous_scale= "peach",
              category_orders={'Age Group': age_labels},
              title='3. Chi tiêu trung bình theo nhóm tuổi',
              text= [f"${value:.2f}" for value in avg_spending_age_group['Avg Spending']],
              hover_data={'Age Group':True, 'Avg Spending':"${value:.2f}"})

fig6.update_traces(hovertemplate="Age Group: %{x}<br>"
                  "Average Spending: %{y:$.2f}")
fig6.add_hline(y=overall_avg_spending, line_dash="dash", line_color="#1CBE4F",
              annotation_text=f'Overall Average: ${overall_avg_spending:.2f}',
              annotation_font_color="#1CBE4F",
              annotation_position="top left")

fig6.update_layout(title={'x': 0.5, 'y': 0.9},
                  width=700, height=450,
                  xaxis_title= 'Age Group',
                  yaxis_title= 'Average Spending ($)',
                  # Set range from 400 to 800
                  # Set starting tick at 400
                  # Set interval between ticks
                  yaxis=dict(range=[500, 800],
                             tick0=500,
                             dtick=50),
                  showlegend=False)

# Hiển thị biểu đồ
fig6.show()
```

**Output:**



→ Biểu đồ cho thấy:

- Nhìn chung, mức chi tiêu trung bình của các nhóm tuổi khá gần nhau, dao động trong khoảng từ 750 USD đến 770 USD. Điều này cho thấy sức mua của các nhóm tuổi này tương đối đồng đều.

- **Xác định nhóm tuổi nào có xu hướng chi tiêu lớn nhất trên mỗi giao dịch cá nhân:**

- + **Nhóm tuổi 50-59** có xu hướng chi tiêu trung bình cao nhất **\$769.20** trên mỗi giao dịch cá nhân so với các nhóm tuổi khác. Điều này cho thấy người tiêu dùng trong độ tuổi này có khả năng chi tiêu cao hơn cho mỗi lần mua sắm.

- **Tìm hiểu thói quen mua sắm cá nhân và mức độ ưu tiên chi tiêu của từng nhóm:**

- + **Nhóm tuổi 18-29, 30-39 và 40-49:** Các nhóm tuổi này có mức chi tiêu tương đối gần nhau, cho thấy xu hướng tiêu dùng của nhóm người trẻ tuổi và trung niên khá ổn định.

- + **Nhóm tuổi 50-59:** Như đã đề cập, nhóm tuổi này có mức chi tiêu cao nhất. Điều này có thể do nhiều yếu tố như: **thu nhập cao hơn, chi tiêu các nhu cầu của gia đình, nhu cầu chăm sóc sức khỏe, v.v..** Cần thêm một vài dữ liệu cụ thể hơn như 'Product Category' để phân tích sâu.

- + **Nhóm tuổi 60-70:** Mức chi tiêu của nhóm tuổi này thấp hơn một chút so với nhóm tuổi 50-59. Điều này có thể do thu nhập giảm sau khi nghỉ hưu hoặc xu hướng tiết kiệm cao hơn.

e. Phân tích chi tiêu trung bình theo mặt hàng và nhóm tuổi (Average Spending by Product Category and Age/Age Group):

**Mục tiêu:** Tìm hiểu sâu hơn và kiểm tra dự đoán về thói quen mua sắm theo từng độ tuổi đối với từng loại sản phẩm/từng mặt hàng, hỗ trợ tối ưu hóa hành vi tiêu dùng của khách hàng

```
#PHÂN TÍCH CHI TIÊU TRUNG BÌNH THEO MẶT HÀNG VÀ NHÓM TUỔI (Average Spending by PCategory and Age group)
import pandas as pd
import plotly.express as px

# Tính trung bình chi tiêu theo nhóm tuổi và nhóm sản phẩm
avg_spending_age_group_category = df.groupby(['Age Group', 'Product Category'])['Total Spending'].mean().reset_index(name='Avg Spending')

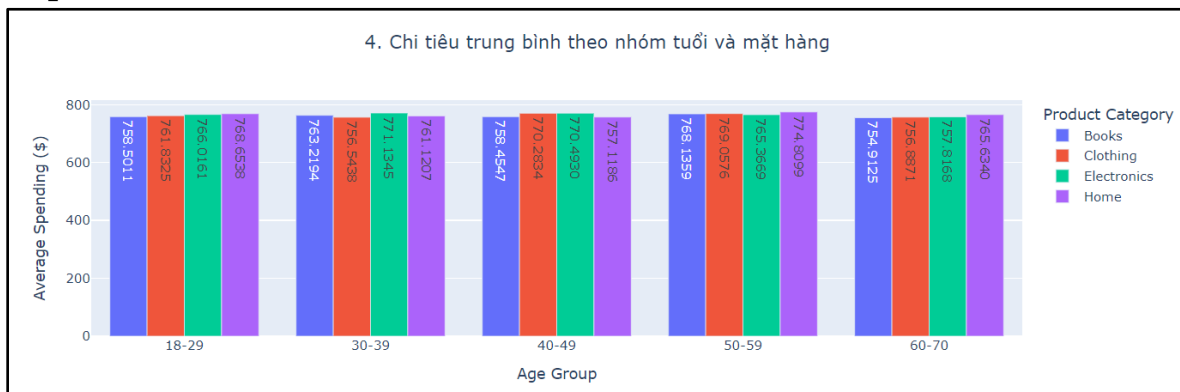
# Tính trung bình chi tiêu tổng thể
overall_avg_spending = df['Total Spending'].mean()

# Vẽ biểu đồ so sánh chi tiêu trung bình theo nhóm tuổi và sản phẩm
fig7 = px.bar(
    avg_spending_age_group_category,
    x='Age Group', # Trục X là nhóm tuổi
    y='Avg Spending', # Trục Y là chi tiêu trung bình
    color='Product Category', # Nhóm màu theo loại sản phẩm
    title="4. Chi tiêu trung bình theo nhóm tuổi và mặt hàng",
    labels={'Avg Spending': 'Average Spending ($)', 'Age Group': 'Age Group'},
    barmode='group', # Hiển thị các cột cho từng loại sản phẩm
    text_auto='.4f' # Định dạng hiển thị giá trị với 2 chữ số thập phân
)

# Cập nhật layout cho biểu đồ
fig7.update_layout(
    xaxis_title='Age Group', # Tiêu đề trục X là 'Age Group'
    yaxis_title='Average Spending ($)', # Tiêu đề trục Y là 'Average Spending'
    title={'x': 0.5, 'y': 0.9},
    height=400,
    showlegend=True
)

# Hiển thị biểu đồ
fig7.show()
```

## Output:



→ Có sự khác biệt nhỏ giữa các nhóm tuổi và mặt hàng, nhưng nhìn chung, mức chi tiêu trung bình không có sự chênh lệch quá lớn.

→ Biểu đồ này cho phép tìm hiểu chi tiết hơn về thói quen mua sắm/ mức độ chi tiêu từng nhóm tuổi đối với từng mặt hàng, chẳng hạn như:

### - Nhóm tuổi 18-29:

+ Có xu hướng chi tiêu cao hơn cho các mặt hàng như "**Books**" và "**Electronics**". Điều này có thể cho thấy nhóm tuổi này có nhu cầu tìm hiểu/học tập, học hỏi và sử dụng công nghệ cao hơn.

+ Mức chi tiêu cho các mặt hàng "Clothing" và "Home" cũng khá cao, cho thấy sự quan tâm đến thời trang và việc trang trí không gian sống.

### - Nhóm tuổi 30-39:

+ Có xu hướng chi tiêu nhiều nhất vào mặt hàng "**Electronics**"

+ Mức chi tiêu tương đối ổn định và không có sự chênh lệch quá lớn giữa các mặt hàng.

+ Điều này cho thấy nhóm tuổi này có nhu cầu mua sắm đa dạng và cân bằng hơn.

### - Nhóm tuổi 60-70:

+ Mức chi tiêu thấp nhất so với các nhóm tuổi khác.

+ Điều này có thể do thu nhập giảm hoặc nhu cầu tiêu dùng giảm đi khi về già.

### 3.1.3 Phân tích và trực quan hóa xu hướng về thời gian giao dịch (Purchase Date)

a. Phân tích Chi tiêu trung bình (Average Spending) theo thời gian mua hàng (buổi sáng/chiều/ tối) và theo giới tính:

Lưu ý về đặc điểm dữ liệu 'Average Spending' trong trường hợp này, có sự phân bố cường độ rõ ràng nên ta có thể áp dụng biểu đồ Heatmap

#### Mục tiêu:

- Xác định thời điểm trong ngày mà khách hàng chi tiêu trung bình nhiều nhất (buổi sáng, buổi chiều, hay buổi tối). Tìm hiểu sự khác biệt về hành vi mua sắm trong các khung giờ khác nhau đối với từng đối tượng khách hàng (nam - nữ)
- So sánh hành vi chi tiêu giữa giới tính: **Phân tích nhóm Khách hàng nam giới hay nữ giới sẽ chi tiêu trung bình cao hơn** trong mỗi khoảng thời gian trong ngày.
- Nhận biết mẫu hành vi đặc thù nhằm tối ưu hóa trải nghiệm từng nhóm đối tượng Khách hàng:
  - + Nhận biết các khung giờ cụ thể mà chi tiêu trung bình tăng đột biến đối với từng giới tính.
  - + Đánh giá liệu có mối quan hệ đặc thù giữa thời gian mua sắm và giới tính không.

```
#phân tích CHI TIÊU TRUNG BÌNH theo thời gian mua hàng vào buổi sáng/chiều tối từ cột 'Purchase Date' và theo giới tính 'Gender'
import plotly.express as px
import pandas as pd

# Chuyển đổi 'Purchase Date' thành kiểu datetime nếu chưa phải
df['Purchase Date'] = pd.to_datetime(df['Purchase Date'], format='%d/%m/%Y %H:%M:%S')

# Tạo cột 'Time of Day' phân loại buổi sáng, chiều, tối
def classify_time_of_day(hour):
    if 6 <= hour < 12:
        return 'Morning' # Buổi sáng
    elif 12 <= hour < 18:
        return 'Afternoon' # Buổi chiều
    else:
        return 'Evening' # Buổi tối

# Áp dụng hàm để phân loại thời gian từ cột 'Purchase Date'
df['Time of Day'] = df['Purchase Date'].dt.hour.apply(classify_time_of_day)

# Nhóm dữ liệu theo giới tính và phân loại thời gian, tính chi tiêu trung bình
time_gender_avg_spending = df.groupby(['Gender', 'Time of Day'])['Total Spending'].mean() #tính chi tiêu trung bình

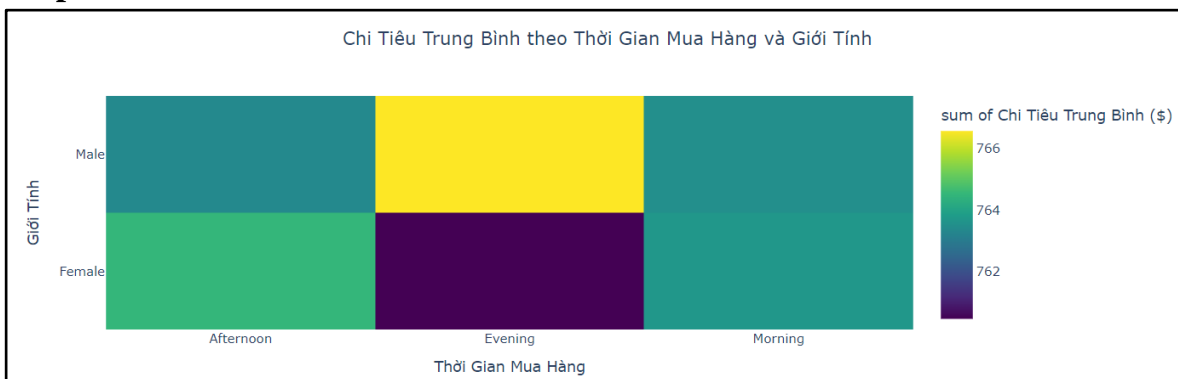
# Chuyển kết quả thành DataFrame với các cột thông thường
time_gender_avg_spending = time_gender_avg_spending.reset_index() # Đưa index trở lại thành cột
time_gender_avg_spending = time_gender_avg_spending.rename(columns={'Total Spending': 'Average Spending'}) # Đổi tên cột
```

```
# Vẽ biểu đồ heatmap
fig_heatmap = px.density_heatmap(time_gender_avg_spending,
                                x='Time of Day',
                                y='Gender',
                                z='Average Spending', # Sử dụng 'Average Spending' làm giá trị cường độ
                                color_continuous_scale='Viridis', # Màu sắc của heatmap
                                title='Chỉ Tiêu Trung Bình theo Thời Gian Mua Hàng và Giới Tính',
                                labels={'Average Spending': 'Chỉ Tiêu Trung Bình ($)', 'Time of Day': 'Thời Gian Mua Hàng', 'Gender': 'Giới Tính'})

# Cập nhật layout cho biểu đồ
fig_heatmap.update_layout(
    title=['x': 0.5, 'y': 0.9],
    xaxis_title='Thời Gian Mua Hàng',
    yaxis_title='Giới Tính',
    height=400,
    showlegend=False
)

# Hiển thị biểu đồ
fig_heatmap.show()
```

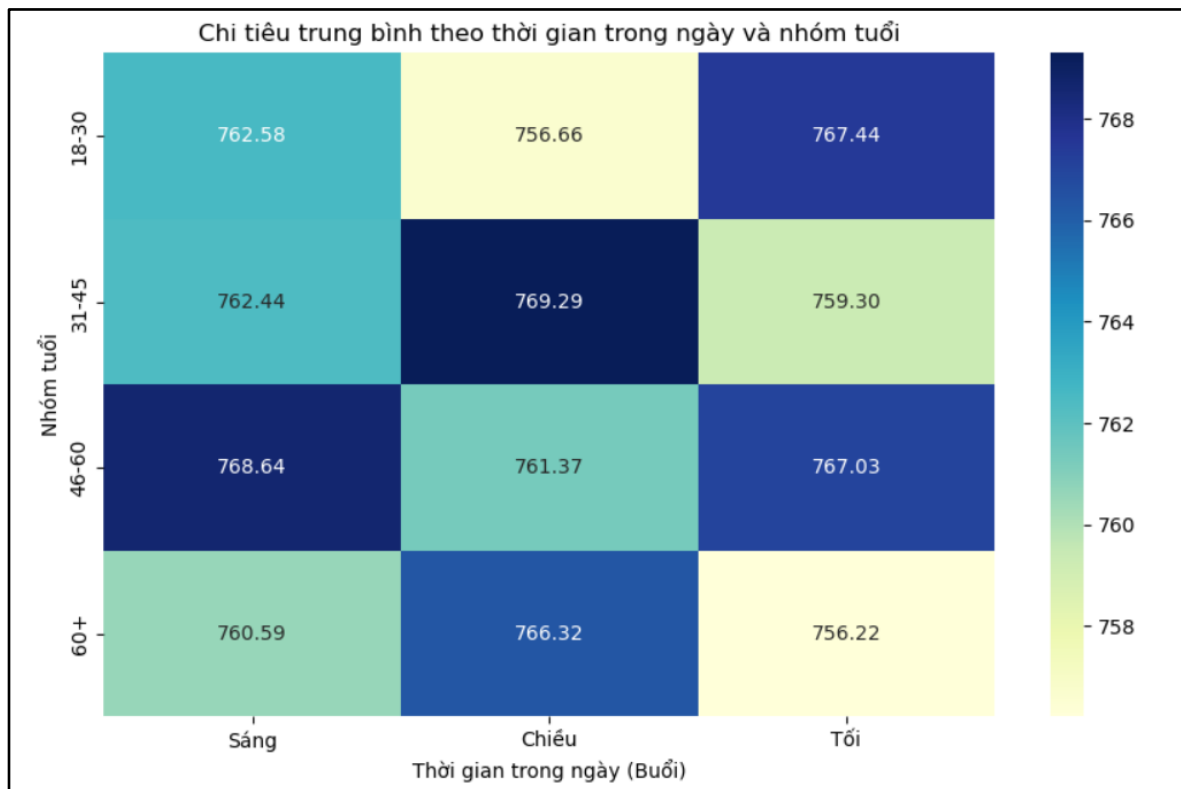
## Output:



→ Biểu đồ cho thấy:

- **Nam giới** chi tiêu trung bình nhiều nhất vào **buổi tối (Evening)**, cho thấy họ có thể có xu hướng mua sắm nhiều hơn vào thời điểm này trong ngày và có xu hướng mua sắm ít hơn vào buổi sáng.
- **Nữ giới** chi tiêu trung bình nhiều nhất vào **buổi sáng (Morning)** và có xu hướng chi tiêu thấp hơn vào buổi tối, có thể cho thấy họ mua sắm ít hơn hoặc ưu tiên thời gian khác trong ngày.
- Mức chi tiêu trung bình cả hai giới tương đối cân bằng vào buổi chiều (Afternoon)
- **Các hành vi đặc thù:**
  - + **Nam giới:** Chi tiêu trung bình tăng đột biến vào buổi tối.
  - + **Nữ giới:** Chi tiêu trung bình tăng đột biến vào buổi sáng.

**Áp dụng tương tự biểu đồ Heatmap, khi phân tích chi tiêu trung bình (Average Spending) theo thời gian mua hàng (Purchase Date) và nhóm tuổi, thu được kết quả:**



→ Biểu đồ cho thấy rõ về hành vi tiêu dùng của từng nhóm tuổi vào các thời điểm cụ thể trong ngày:

- **Nhóm tuổi 18-30:** Chi tiêu nhiều nhất vào buổi tối, có thể là do thời điểm này họ có nhiều thời gian rảnh rỗi để mua sắm và giải trí.
- **Nhóm tuổi 31-45:** Chi tiêu nhiều nhất vào buổi chiều, có thể liên quan đến lịch làm việc và nghỉ ngơi, khi họ có thời gian mua sắm sau giờ làm.
- **Nhóm tuổi 46-60:** Chi tiêu nhiều nhất vào buổi sáng, có thể do lịch trình ổn định và thói quen mua sắm sớm trong ngày.
- **Nhóm tuổi 60+:** Chi tiêu nhiều nhất vào buổi chiều, có thể do thời gian rảnh rỗi và thói quen mua sắm vào thời điểm này.

b. Phân tích Tổng chi tiêu mua hàng (Total Spending) theo thời gian mua hàng (đơn vị giờ) và theo giới tính:

**Mục tiêu:**

- Xác định khung giờ "vàng" hoặc giờ cao điểm cho các chiến lược kinh doanh (theo giới tính)
- Tìm hiểu sự khác biệt về tổng chi tiêu giữa nam và nữ theo từng khung giờ, đánh giá giới tính nào chi tiêu nhiều hơn tại các thời điểm cụ thể.
- Nắm bắt hành vi tiêu dùng đặc biệt của từng giới tính vào các thời điểm cụ thể (ví dụ: nam chi tiêu nhiều hơn vào buổi tối, nữ chi tiêu nhiều hơn vào sáng sớm).

```
#phân tích tổng chi tiêu mua hàng theo thời gian mua hàng theo giờ(HOUR) từ cột 'Purchase Date' và theo giới tính 'Gender'
#LINECHART
import pandas as pd
import plotly.express as px

# Chuyển đổi 'Purchase Date' thành kiểu datetime nếu chưa phải
df['Purchase Date'] = pd.to_datetime(df['Purchase Date'], format='%d/%m/%Y %H:%M:%S')

# Trích xuất giờ từ cột 'Purchase Date'
df['Hour'] = df['Purchase Date'].dt.hour

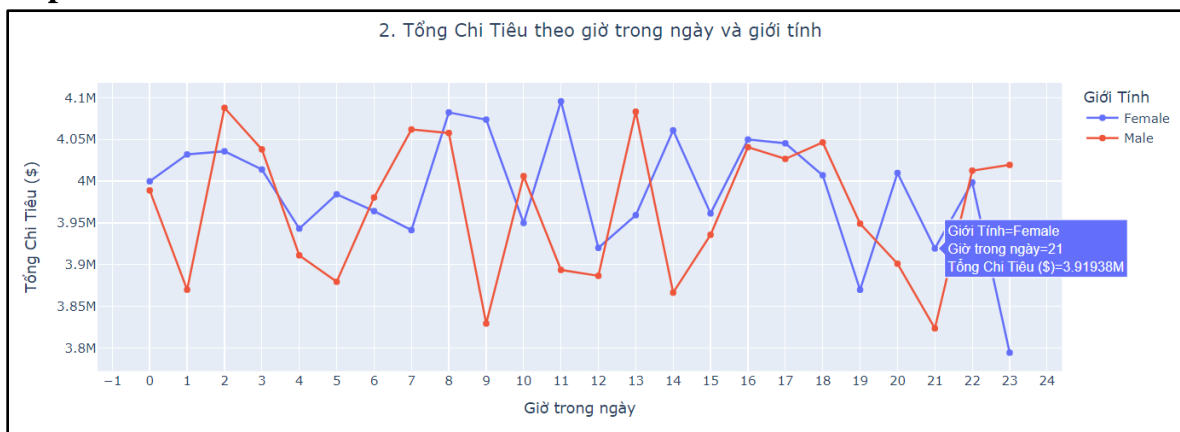
# Nhóm dữ liệu theo giới tính và giờ
hour_gender_spending = df.groupby(['Gender', 'Hour']).agg({'Total Spending': 'sum', 'Purchase Date': 'count'}).reset_index()

# Vẽ biểu đồ Line chart
fig_line = px.line(hour_gender_spending,
                    x='Hour',
                    y='Total Spending', # Hoặc 'Transaction Count' nếu phân tích số lượng giao dịch
                    color='Gender',
                    title='Tổng Chi Tiêu theo giờ trong ngày và giới tính',
                    labels={'Hour': 'Giờ trong ngày', 'Total Spending': 'Tổng Chi Tiêu ($)', 'Gender': 'Giới Tính'},
                    markers=True) # Thêm điểm đánh dấu trên đường tại mỗi giờ
```

```
# Cập nhật layout của biểu đồ
fig_line.update_layout(
    xaxis_title='Giờ trong ngày',
    yaxis_title='Tổng Chi Tiêu ($)',
    title={'x': 0.5, 'y': 0.9},
    height=450,
    showlegend=True,
    xaxis=dict(tickmode='linear', tick0=0, dtick=1) # Hiển thị tất cả các giờ từ 0-23
)

# Hiển thị biểu đồ
fig_line.show()
#Biểu đồ line chart sẽ hiển thị xu hướng chi tiêu theo giờ trong ngày,
#cho phép bạn dễ dàng so sánh giữa các giới tính để tìm ra giờ cao điểm
#hoặc sự khác biệt giữa thói quen mua sắm của nam và nữ.
```

## Output:



→ Biểu đồ này cho thấy:

### 1. Xác định khung giờ "vàng" hoặc giờ cao điểm cho các chiến lược kinh doanh (theo giới tính):

- **Nam giới:** Khung giờ cao điểm cho nam giới có thể là từ 1 giờ đến 2 giờ sáng và từ 13 giờ đến 14 giờ chiều, khi tổng chi tiêu đạt đỉnh, hiển thị bằng màu đỏ đậm.

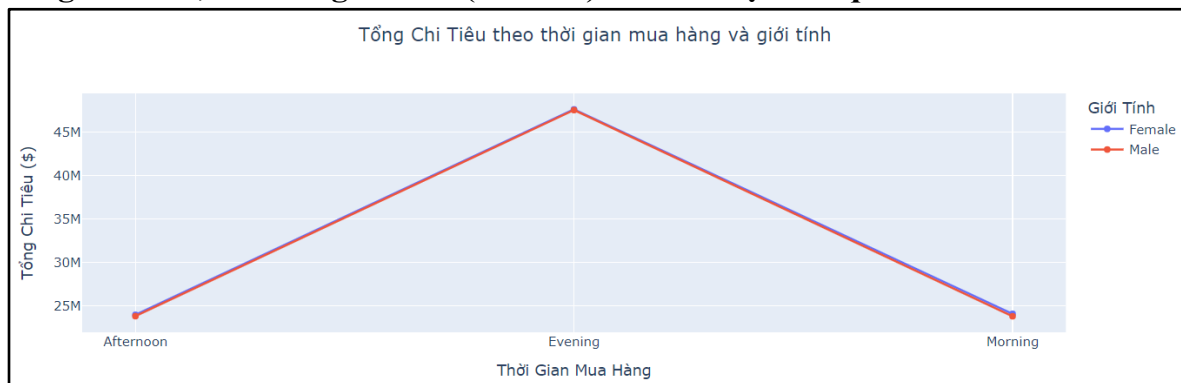
- **Nữ giới:** Khung giờ cao điểm cho nữ giới có thể là từ **10 giờ đến 12 giờ trưa** và từ **14 giờ đến 16 giờ chiều**, khi tổng chi tiêu đạt đỉnh, hiển thị bằng màu xanh đậm.

**2. Tìm hiểu sự khác biệt về tổng chi tiêu giữa nam và nữ theo từng khung giờ, đánh giá giới tính nào chi tiêu nhiều hơn tại các thời điểm cụ thể nhằm nắm bắt hành vi tiêu dùng:**

- **Từ 0 giờ - 1 giờ sáng, 13 giờ - 14 giờ chiều:** Nam giới chi tiêu nhiều hơn nữ giới. Đây có thể là những thời điểm nam giới có nhiều nhu cầu mua sắm hoặc giải trí.

- **Từ 10 giờ - 12 giờ trưa, 14 giờ - 16 giờ chiều:** Nữ giới chi tiêu nhiều hơn nam giới. Điều này cho thấy nữ giới có thể ưu tiên mua sắm vào những thời điểm này, có thể là do lịch trình công việc hoặc hoạt động hàng ngày.

**Áp dụng biểu đồ Line Chart tương tự, khi phân tích tổng chi tiêu (Total Spending) theo thời gian mua hàng (Purchase Date) theo mốc buổi sáng/trưa/tối, và theo giới tính (Gender) ta thu được kết quả:**



**3.1.4 Phân tích (kết hợp ba thuộc tính) tổng chi tiêu (Total Spending) theo nhóm tuổi (Age) và giới tính (Gender)**

**Mục tiêu:** Hỗ trợ phân tích và làm rõ một số câu hỏi như:

- Có sự khác biệt nào trong chi tiêu giữa các nhóm tuổi và giới tính không?
- Nhóm tuổi nào và giới tính nào có tổng chi tiêu cao nhất khi kết hợp cả hai yếu tố này?
- Sự kết hợp giữa độ tuổi và giới tính có ảnh hưởng đến mức chi tiêu theo từng mặt hàng hoặc dịch vụ không?



```
#PTICH TỔNG CHI TIÊU THEO NHÓM TUỔI VÀ GIỚI TÍNH (kết hợp 2 thuộc tính)
import pandas as pd
import plotly.express as px

# Giả sử df đã có các cột 'Total Spending', 'Age Group' và 'Gender'

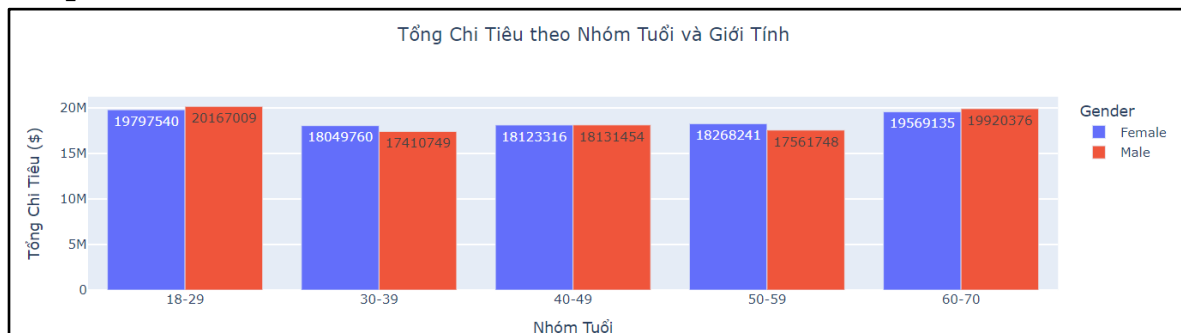
# Nhóm dữ liệu theo 'Age Group' và 'Gender', sau đó tính tổng chi tiêu
total_spending_by_age_gender = df.groupby(['Age Group', 'Gender'], as_index=False)['Total Spending'].sum()

# Vẽ biểu đồ tổng chi tiêu theo nhóm tuổi và giới tính
fig10 = px.bar(
    total_spending_by_age_gender,
    x='Age Group',
    y='Total Spending',
    color='Gender', # Nhóm màu theo giới tính
    barmode='group', # Hiển thị các cột cho từng giới tính
    title='Tổng Chi Tiêu theo Nhóm Tuổi và Giới Tính',
    labels={'Total Spending': 'Tổng Chi Tiêu ($)', 'Age Group': 'Nhóm Tuổi'},
    text='Total Spending' # Hiển thị giá trị chi tiêu trên từng cột
)

# Cập nhật layout cho biểu đồ
fig10.update_layout(
    xaxis_title='Nhóm Tuổi',
    yaxis_title='Tổng Chi Tiêu ($)',
    title={'x': 0.5, 'y': 0.9},
    height=400,
    showlegend=True
)

# Hiển thị biểu đồ
fig10.show()
```

## Output:



→ Biểu đồ cho thấy:

- Có sự khác biệt rõ ràng trong mức chi tiêu giữa các nhóm tuổi và giới tính. Ví dụ, trong nhóm tuổi 18-29, nam giới có tổng chi tiêu cao hơn nữ giới (\$20,167,009 so với \$19,797,540). Tương tự, trong nhóm tuổi 30-39, nữ giới chi tiêu nhiều hơn nam giới (\$18,049,760 so với \$17,410,749).

→ Điều này cho thấy có sự khác biệt trong nhu cầu và ưu tiên chi tiêu giữa các giới tính trong cùng một nhóm tuổi.

- Nhóm tuổi 18-29 và giới tính nam có tổng chi tiêu cao nhất khi kết hợp cả hai yếu tố này, với tổng chi tiêu là \$20,167,009. Điều này cho thấy nam giới trong độ tuổi này có xu hướng chi tiêu nhiều hơn so với các nhóm tuổi và giới tính khác.

## 3.2 Tần suất mua sắm của các nhóm khách hàng theo thời gian?

### 3.2.1 Tần suất mua sắm của khách hàng qua 4 năm

#### a. Tần suất mua sắm của khách hàng:

Việc phân tích tần suất mua sắm của Khách hàng nhằm giúp hiểu rõ hành vi mua sắm của Khách hàng:

- **Lý do:** Phân tích tần suất giúp xác định thói quen mua sắm của từng khách hàng, bao gồm số lần mua sắm trong một khoảng thời gian cụ thể.

- **Mục tiêu:**

- + Xác định nhóm khách hàng trung thành với tần suất mua sắm cao.
- + Nhận biết khách hàng mua sắm không thường xuyên (mua ít) để thực hiện các chiến lược khuyến khích họ mua hàng nhiều hơn.

```
#Tính tần suất mua sắm của từng khách hàng
customer_frequency = data.groupby('Customer ID')['Purchase Date'].count().reset_index() #Tính toán tần suất mua hàng
customer_frequency.columns = ['Customer ID', 'Purchase Frequency'] #gán tên các cột lại cho dễ hiểu

# Hiển thị tần suất mua sắm
print("\nTần suất mua sắm của khách hàng:")
print(customer_frequency.describe())
```

Output:

	Customer ID	Purchase Frequency
count	49673.000000	49673.000000
mean	24991.166489	5.032915
std	14432.765110	2.206427
min	1.000000	1.000000
25%	12491.000000	3.000000
50%	24984.000000	5.000000
75%	37489.000000	6.000000
max	50000.000000	17.000000

→ **Điều này cho thấy:**

- Đa số khách hàng có tần suất mua sắm từ 3 đến 6 lần (giữa các giá trị 25%, 50% và 75%).
- Tần suất mua hàng trung bình của một khách hàng là 5 lần.
- Có sự chênh lệch lớn giữa khách hàng mua ít (1 lần) và khách hàng mua nhiều nhất (17 lần), nhưng số lượng khách hàng mua nhiều rất nhỏ (dựa trên độ lệch chuẩn).

#### b. Phân nhóm khách hàng theo tần suất mua sắm:

Dựa trên các thông số trong dữ liệu đã phân tích ở trên, nhóm em sẽ phân nhóm khách hàng theo tần suất mua sắm như sau để hiểu rõ hơn về hành vi của họ:

- **Nhóm 1: Khách hàng mua ít (1 - 4 lần):** Tập trung các khách hàng ít mua, có thể là người mua thử hoặc không quay lại thường xuyên.

- **Nhóm 2: Khách hàng mua trung bình (5 - 8 lần):** Khách hàng có tần suất mua từ trung bình đến khá thường xuyên.

- **Nhóm 3: Khách hàng mua nhiều (9 - 17 lần):** Những khách hàng có tần suất mua vượt qua mức trung bình cao.

```
# Ép kiểu cột 'Customer ID' thành kiểu string
data['Customer ID'] = data['Customer ID'].astype(str)

# Kiểm tra kiểu dữ liệu của cột 'Customer ID'
print("kiểu dữ liệu: ", data['Customer ID'].dtype)

# In cột 'Customer ID' đầu tiên dưới dạng string
print(data['Customer ID'].apply(str).head(10))
```

```
# Giả sử 'data' là DataFrame của bạn
if 'Customer ID' in data.columns and 'Purchase Date' in data.columns: #Kiểm tra cột có trong dữ liệu không
    # Tính tần suất mua sắm
    #data.groupby('Customer ID')['Purchase Date'].count().reset_index()
    purchase_frequency = data.groupby('Customer ID').size().reset_index(name='Purchase Frequency') #Tính toán tần suất mua hàng
    purchase_frequency.columns = ['Customer ID', 'Purchase Frequency'] #gán tên các cột lại cho dễ hiểu
    # Ép kiểu cột 'Customer ID' trong kết quả thành kiểu string
    # purchase_frequency['Customer ID'] = purchase_frequency['Customer ID'].astype(str)
    # Kiểm tra kết quả
    print(purchase_frequency.describe())
else:
    print("Cột 'Customer ID' hoặc 'Purchase Date' không tồn tại trong dữ liệu.")

# Phân Loại khách hàng theo tần suất mua sắm
def categorize_customers(frequency):
    if frequency <= 4:
        return 'Nhóm 1: Mua ít'
    elif 5 <= frequency <= 8:
        return 'Nhóm 2: Mua trung bình'
    else:
        return 'Nhóm 3: Mua nhiều'

purchase_frequency['Customer Group'] = purchase_frequency['Purchase Frequency'].apply(categorize_customers)

# In kết quả
print(purchase_frequency.head(15))
print(purchase_frequency.to_string(index=False))
```

**Output:**

Customer ID	Purchase Frequency	Customer Group
1	1	Nhóm 1: Mua ít
10	8	Nhóm 2: Mua trung bình
100	3	Nhóm 1: Mua ít
1000	6	Nhóm 2: Mua trung bình
10000	5	Nhóm 2: Mua trung bình
10001	2	Nhóm 1: Mua ít
10002	3	Nhóm 1: Mua ít
10003	8	Nhóm 2: Mua trung bình
10004	6	Nhóm 2: Mua trung bình
10005	3	Nhóm 1: Mua ít
10006	7	Nhóm 2: Mua trung bình
10007	6	Nhóm 2: Mua trung bình
10008	3	Nhóm 1: Mua ít
10009	6	Nhóm 2: Mua trung bình
1001	5	Nhóm 2: Mua trung bình
10010	4	Nhóm 1: Mua ít
10011	3	Nhóm 1: Mua ít

10012	5	Nhóm 2: Mua trung bình
10013	7	Nhóm 2: Mua trung bình
10014	4	Nhóm 1: Mua ít
10015	5	Nhóm 2: Mua trung bình
10016	5	Nhóm 2: Mua trung bình
10017	6	Nhóm 2: Mua trung bình
10018	6	Nhóm 2: Mua trung bình
10019	6	Nhóm 2: Mua trung bình
1002	4	Nhóm 1: Mua ít
10020	6	Nhóm 2: Mua trung bình
10021	4	Nhóm 1: Mua ít
10022	5	Nhóm 2: Mua trung bình
10023	9	Nhóm 3: Mua nhiều
10024	6	Nhóm 2: Mua trung bình
10025	6	Nhóm 2: Mua trung bình
10026	5	Nhóm 2: Mua trung bình
10027	4	Nhóm 1: Mua ít
10028	3	Nhóm 1: Mua ít

→ Trên đây là một vài kết quả Output minh họa cho việc thống kê tần suất mua sắm của khách hàng. Output này được sắp xếp theo thứ tự tăng dần của Customer ID.

### c. Biểu đồ phân phối tần suất mua sắm của khách hàng:

```
# Tính tỷ lệ phần trăm cho mỗi nhóm
group_counts = purchase_frequency['Customer Group'].value_counts() # Đếm số lượng khách hàng trong từng nhóm
total_customers = len(purchase_frequency) # Tổng số khách hàng
group_percentage = (group_counts / total_customers * 100).round(2) # Tính tỷ lệ phần trăm

# Tạo biểu đồ tròn
fig = px.pie(values=group_percentage, names=group_percentage.index,
             color=group_percentage.index,
             color_discrete_map={
                 'Nhóm 1: Mua ít': '#FF6692',
                 'Nhóm 2: Mua trung bình': '#3366CC',
                 'Nhóm 3: Mua nhiều': '#99CC33'
             },
             title='Phân phối tần suất mua sắm của khách hàng')

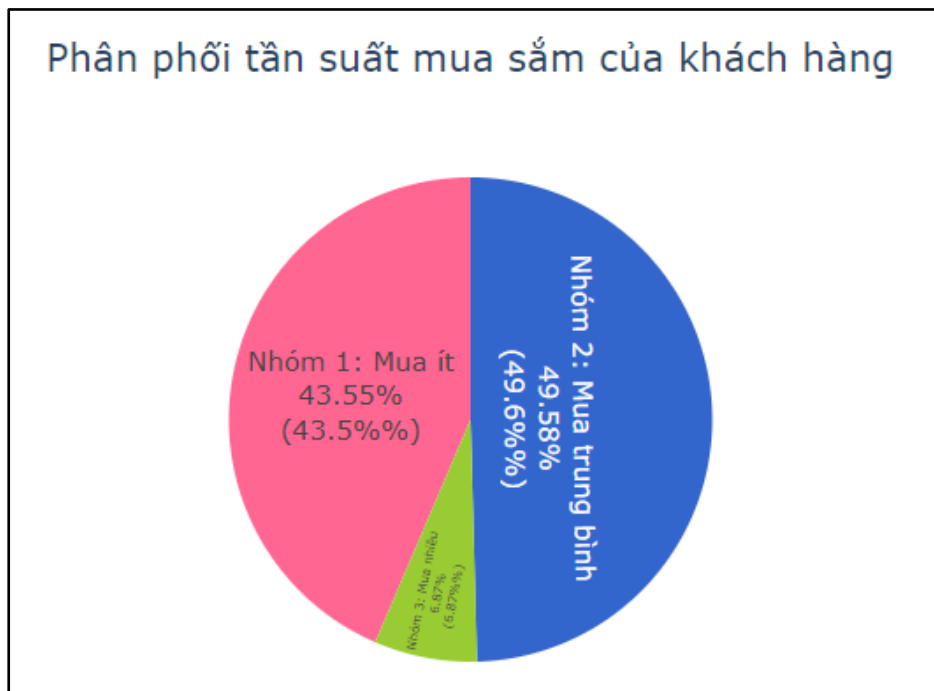
# Điều chỉnh văn bản hiển thị trong biểu đồ tròn
fig.update_traces(textposition='inside', textinfo='text',
                  texttemplate='%{label}<br>{%value}%<br>({percent}%)')

# Điều chỉnh kích thước và ẩn chú giải
fig.update_layout(title={'x': 0.5, 'y': 0.9}, width=400, height=400, showlegend=False)

# Hiển thị biểu đồ
fig.show()

# Lưu biểu đồ dưới dạng hình ảnh
fig.write_image("purchase_frequency_distribution.png")
```

**Output:**



→ Qua biểu đồ ta thấy:

- Nhóm 2 (Mua trung bình) là nhóm chiếm tỷ lệ lớn nhất, gần 50% tổng số khách hàng. Điều này cho thấy phần lớn khách hàng có tần suất mua hàng đều đặn.
- Nhóm 1 (Mua ít) chiếm hơn 43%, điều này có thể chỉ ra rằng có một lượng lớn khách hàng thỉnh thoảng mới mua sắm, đây có thể là nhóm tiềm năng cần chiến lược tiếp thị để kích thích tăng tần suất mua hàng.
- Nhóm 3 (Mua nhiều) là nhóm nhỏ nhất. Đây có thể là nhóm khách hàng trung thành và thường mang lại giá trị doanh thu cao cho doanh nghiệp.

**Từ đó đưa ra một vài chiến lược:**

1. **Đối với nhóm "mua ít":** Kích thích nhóm này tăng tần suất mua hàng.

- **Chiến lược:**

- + Gửi email hoặc tin nhắn cá nhân hóa với các chương trình khuyến mãi hấp dẫn.
- + Triển khai các chương trình ưu đãi định kỳ như: "*Mua lần đầu được giảm giá*", "*Giảm giá cho lần mua thứ 2 trong vòng 1 tháng*".
- + Tìm hiểu lý do nhóm này ít mua: liệu có vấn đề về sản phẩm, giá cả, hoặc trải nghiệm mua sắm không.

2. **Đối với nhóm "mua trung bình":** Chuyển đổi một phần nhóm này thành nhóm "mua nhiều".

- **Chiến lược:**

- + Xây dựng các chương trình khách hàng thân thiết, tích điểm đổi quà để khuyến khích mua thêm.
- + Đưa ra các gói combo hoặc giảm giá theo mức độ mua sắm (*mua nhiều giảm nhiều*).

+ Tăng sự gắn kết qua chiến dịch cá nhân hóa, đề xuất sản phẩm phù hợp dựa trên lịch sử mua hàng.

3. **Đối với nhóm "mua nhiều":** Giữ chân nhóm khách hàng trung thành và tăng giá trị giao dịch trung bình.

**- Chiến lược:**

- + Tạo các gói ưu đãi độc quyền cho nhóm này (*VIP-exclusive discounts*, miễn phí giao hàng,...).
- + Khuyến khích họ giới thiệu thêm bạn bè (chương trình *refer-a-friend*).
- + Tặng quà hoặc ưu đãi cho khách hàng vào các dịp đặc biệt (sinh nhật, lễ tết).

### 3.2.2 Tần suất giao dịch từng tháng (theo năm):

a. Phân tích tần suất giao dịch của khách hàng:

Việc phân tích tần suất giao dịch của khách hàng:

- Giúp thống kê và đánh giá được thời điểm cao điểm và thấp điểm về giao dịch.
- Hiểu xu hướng giao dịch giúp dự báo doanh số và lên kế hoạch kinh doanh tốt hơn.

```
# Chuyển đổi cột Purchase Date thành định dạng datetime (nếu chưa ở định dạng datetime)
data['Purchase Date'] = pd.to_datetime(data['Purchase Date'], errors='coerce')

# Tạo cột Purchase Month từ cột Purchase Date
data['Purchase Month'] = data['Purchase Date'].dt.month_name()

# Tạo cột Purchase Year từ cột Purchase Date
data['Purchase Year'] = data['Purchase Date'].dt.year

# Tạo ra một danh sách các tên tháng trong năm theo thứ tự từ tháng 1 đến tháng 12.
month_order = ['January', 'February', 'March', 'April', 'May', 'June',
               'July', 'August', 'September', 'October', 'November', 'December']

# Tạo cột để lưu trữ thông tin tháng của mỗi giao dịch mua hàng.
data['Purchase Month'] = pd.Categorical(data['Purchase Month'], #chuyển cột Purchase Month -> một categorical, giúp các tháng được sắp xếp theo thứ tự đã
                                     categories = month_order, #Đặt thứ tự các tháng theo danh sách month_order
                                     ordered=True) #cột Purchase Month sẽ được coi là có thứ tự

# Nhóm dữ liệu theo năm mua hàng (Purchase Year) và tháng mua hàng (Purchase Month).
monthly_sales = data.groupby(['Purchase Year', 'Purchase Month']).agg(Count=('Customer ID', 'size')).reset_index()

# Hiển thị kết quả
print(monthly_sales)
```

**Output:**

	Purchase Year	Purchase Month	Count
0	2020	January	5687
1	2020	February	5394
2	2020	March	5683
3	2020	April	5499
4	2020	May	5734
5	2020	June	5539
6	2020	July	5851
7	2020	August	5769
8	2020	September	5591
9	2020	October	5778
10	2020	November	5628
11	2020	December	5892
12	2021	January	5689
13	2021	February	5151
14	2021	March	5914
15	2021	April	5516
16	2021	May	5667
17	2021	June	5536
18	2021	July	5612
19	2021	August	5674
20	2021	September	5445
21	2021	October	5728
22	2021	November	5550
23	2021	December	5617

→ Qua đoạn Output trên là phân thống kê số lượng giao dịch từng tháng ( theo năm) của khách hàng.

**Ví dụ:** Năm 2020 với Tháng 1 có tổng số giao dịch là 5687.

b. Biểu đồ phân phối tần suất giao dịch của khách hàng:

```
#Chuyển cột Purchase Year thành chuỗi để sử dụng trong biểu đồ.
monthly_sales['Purchase Year'] = monthly_sales['Purchase Year'].astype(str)

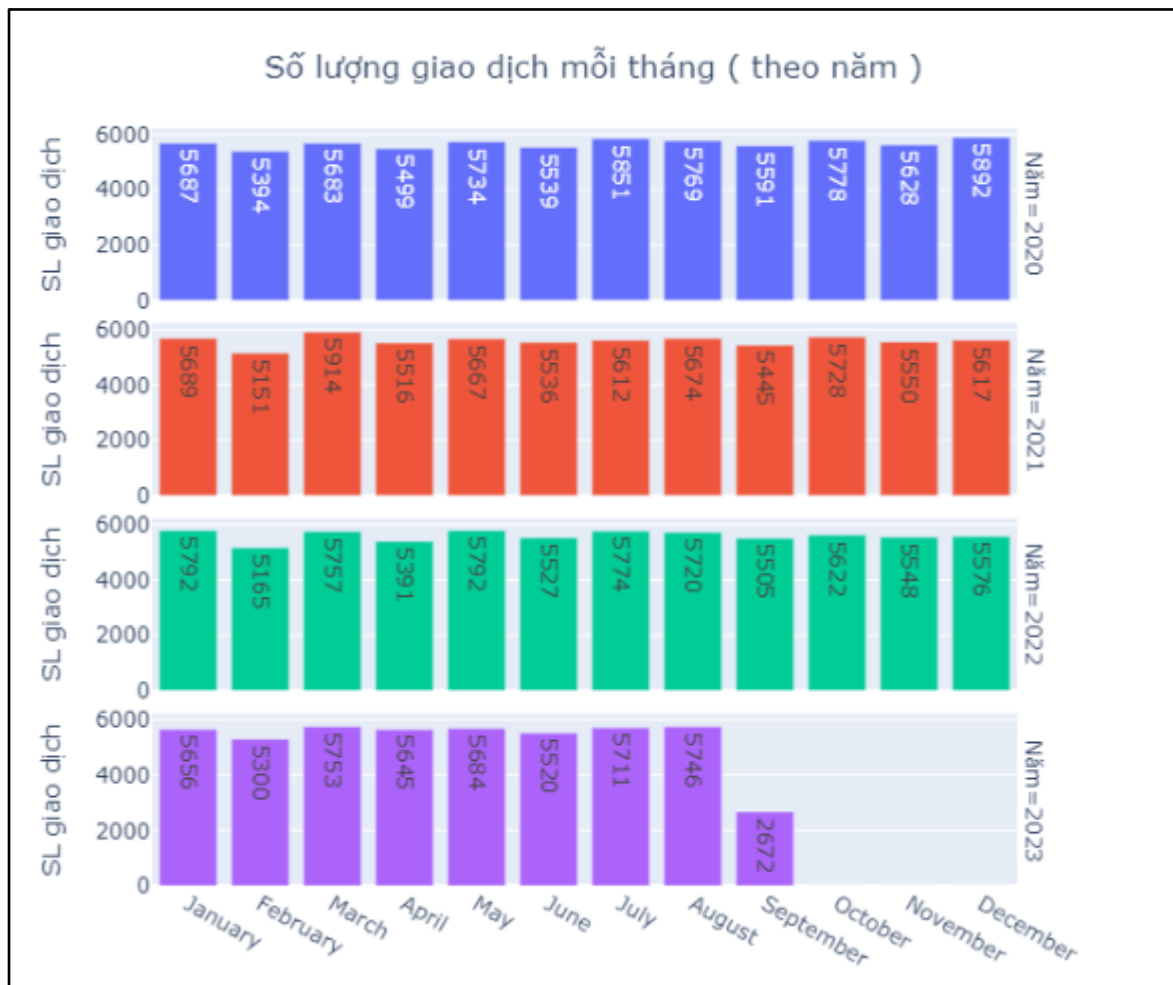
#Tạo biểu đồ cột thể hiện số Lượng đơn hàng (Count) theo tháng (Purchase Month), phân biệt theo năm (Purchase Year).
fig1 = px.bar(monthly_sales, x='Purchase Month', y='Count',
              color='Purchase Year', facet_row='Purchase Year', #Tạo các hàng nhỏ trong biểu đồ, mỗi hàng đại diện cho một năm.
              title='Số lượng giao dịch mỗi tháng ( theo năm )',
              labels={'Count': 'SL giao dịch', 'Purchase Year': 'Năm'},
              hover_data={'Count': ':,.0f'},
              text_auto=True)

fig1.update_layout(title={'x': 0.5, 'y': 0.9}, #Cân giữa tiêu đề biểu đồ.
                  xaxis_title=None,
                  width=650, height=600,
                  showlegend=False) #Tắt hiển thị chú thích.

# Hiển thị biểu đồ
fig1.show()

# Lưu biểu đồ dưới dạng hình ảnh
fig1.write_image("purchase_quantity.png")
```

**Output:**



→ **Nhận xét biểu đồ:**

1. **Tăng trưởng qua các năm:**

- Số lượng giao dịch trong năm 2020 tương đối ổn định theo từng tháng. Tuy nhiên, từ năm 2021 đến 2022, số lượng giao dịch có sự cải thiện nhẹ, cho thấy sự tăng trưởng hoặc mở rộng hoạt động kinh doanh.
- Năm 2023 chỉ có dữ liệu đến tháng 9 và tổng số giao dịch trong năm này có xu hướng giảm, có thể do chưa hoàn tất năm hoặc có sự thay đổi trong thị trường.

2. **Xu hướng giao dịch theo tháng:**

- Số lượng giao dịch trong các tháng duy trì sự ổn định tương đối trong các năm. Tuy nhiên, **một số tháng nổi bật hơn**, chẳng hạn như:
  - + **Tháng 12** có lượng giao dịch cao ở các năm (2020, 2021, 2022), có thể liên quan đến các dịp lễ hội cuối năm (*Christmas, New Year Sale*).
  - + **Tháng 1** (đầu năm) cũng có lượng giao dịch khá cao, có thể do dịp *Tết Nguyên Đán* hoặc các chiến dịch bán hàng sau lễ.
- Các tháng như **tháng 9 và tháng 10** thường ghi nhận lượng giao dịch thấp hơn, có thể liên quan đến giai đoạn không có nhiều dịp mua sắm đặc biệt.

3. **Biến động trong năm 2023:**



- Năm 2023, lượng giao dịch từ tháng 8 trở đi giảm mạnh, đặc biệt tháng 9 giảm đáng kể. Điều này có thể do thiếu chiến dịch khuyến mãi, suy giảm sức mua hoặc các yếu tố bên ngoài (kinh tế, thị trường).

→ Từ đó đưa ra một vài chiến lược:

**1. Tăng cường chiến lược marketing trong các tháng thấp điểm (tháng 9, tháng 10):**

- **Mục tiêu:** Giảm sự biến động theo mùa, tăng số lượng giao dịch trong các tháng ít nổi bật.

- **Chiến lược:**

- + Triển khai các chương trình khuyến mãi hoặc chiến dịch giảm giá như "Back to School" (tháng 9) hoặc "Early Holiday Sale" (tháng 10).
- + Kết hợp các sự kiện hoặc chủ đề đặc biệt (giảm giá giao dịch vào các ngày cuối tuần, giảm giá độc quyền cho khách hàng thân thiết).

**2. Tận dụng mùa cao điểm (tháng 12, tháng 1):**

- **Mục tiêu:** Tối đa hóa doanh thu từ các tháng cao điểm.

- **Chiến lược:**

- + Chuẩn bị sớm các chiến dịch khuyến mãi, đặc biệt trong dịp cuối năm như *Black Friday*, *Christmas Sale*, *New Year Sale*.
- + Đưa ra các gói ưu đãi hấp dẫn cho khách hàng, như giảm giá khi mua combo hoặc tặng quà kèm theo đơn hàng.
- + Cải thiện dịch vụ giao hàng và trải nghiệm khách hàng để đáp ứng nhu cầu cao trong mùa này.

**3. Phát triển nhóm khách hàng trung thành:**

- **Mục tiêu:** Khuyến khích khách hàng mua sắm thường xuyên hơn, đặc biệt trong các tháng thấp điểm.

- **Chiến lược:**

- + Triển khai chương trình khách hàng thân thiết, tích điểm đổi quà hoặc nhận ưu đãi khi đạt ngưỡng chi tiêu.
- + Cung cấp ưu đãi đặc biệt cho khách hàng trung thành vào các tháng thấp điểm, như miễn phí giao hàng hoặc giảm giá khi mua lặp lại.

**3.2.3 Tần suất giao dịch của từng khách hàng từng tháng (theo năm)**

a. Phân tích tần suất giao dịch của từng khách hàng từng tháng (theo năm):

Phân tích tần suất giao dịch của mỗi khách hàng từng tháng (theo năm) giúp hiểu rõ xu hướng hành vi khách hàng:

- **Tìm hiểu thói quen mua sắm:** Phân tích theo tháng giúp doanh nghiệp hiểu rõ tần suất mua sắm của khách hàng, chẳng hạn họ mua thường xuyên vào đầu hoặc cuối tháng.
- **Phân tích sự thay đổi hành vi theo thời gian:** Quan sát cách hành vi mua sắm thay đổi theo các giai đoạn, như mùa lễ hội, khuyến mãi, hoặc các sự kiện kinh tế - xã hội.

```
# Tính tần suất giao dịch của từng khách hàng theo từng tháng và năm
transaction_frequency = (
    data.groupby(['Customer ID', 'Purchase Year', 'Purchase Month'], observed=True)
    .size()
    .reset_index(name='Transaction Count') # Tạo cột 'Transaction Count' hiển thị số lần mua sắm
)

# Lọc chỉ giữ lại những tháng có ít nhất 1 giao dịch
transaction_frequency = transaction_frequency[transaction_frequency['Transaction Count'] > 0]

# Sắp xếp dữ liệu theo Customer ID, Purchase Year, và Purchase Month
transaction_frequency = transaction_frequency.sort_values(by=['Customer ID', 'Purchase Year', 'Purchase Month'])

# Hiển thị toàn bộ các dòng nếu cần
pd.set_option('display.max_rows', None) # Không giới hạn số lượng dòng

# Lưu dữ liệu ra file CSV
transaction_frequency.to_csv('transaction_frequency.csv', index=False, encoding='utf-8-sig')

# Hiển thị 10 dòng đầu tiên để kiểm tra
print(transaction_frequency.head(25))
```

**Output:**

	Customer ID	Purchase Year	Purchase Month	Transaction Count
0	1	2023	July	1
1	10	2020	November	1
2	10	2021	January	1
3	10	2021	August	1
4	10	2022	April	1
5	10	2022	November	2
6	10	2022	December	1
7	10	2023	January	1
8	100	2020	August	1
9	100	2022	July	1
10	100	2023	August	1
11	1000	2021	January	1
12	1000	2021	February	1
13	1000	2021	March	1
14	1000	2022	May	1
15	1000	2023	March	1
16	1000	2023	April	1
17	10000	2020	May	1
18	10000	2021	January	1
19	10000	2021	April	1
20	10000	2022	November	1
21	10000	2023	February	1
22	10001	2021	May	1
23	10001	2022	July	1
24	10002	2022	May	1

→ Qua Output trên, ta thấy rằng:

**- Tần suất giao dịch của khách hàng thường không đều:**

- + Một số khách hàng có giao dịch ở nhiều năm (ví dụ: khách hàng ID 10).
- + Một số khách hàng chỉ có giao dịch trong một năm hoặc một vài tháng cụ thể (ví dụ: khách hàng ID 10002 chỉ có giao dịch vào tháng 5/2022).

**- Khách hàng trung thành và khách hàng không thường xuyên:**

- + Khách hàng **ID 10** có giao dịch trải dài trong nhiều tháng và năm, cho thấy đây có thể là khách hàng trung thành.
- + Khách hàng **ID 1** chỉ xuất hiện một lần duy nhất, có thể được xếp vào nhóm khách hàng không thường xuyên.

**- Tần suất giao dịch tập trung ở một số thời điểm:**

- + Có thể thấy sự xuất hiện của các giao dịch nhiều hơn ở một số tháng cụ thể, ví dụ như tháng 11 hoặc tháng 7. Điều này có thể liên quan đến các chương trình khuyến mãi, lễ hội hoặc thời gian cao điểm mua sắm.

**b. Thời gian giữa các giao dịch:**

```
# Lưu biểu đồ dưới dạng hình ảnh
#fig1.write_image("purchase_quantity.png")

# Tính thời gian giữa các giao dịch
# Tạo khoảng thời gian giữa các giao dịch cho từng khách hàng
data = data.sort_values(by=['Customer ID', 'Purchase Date'])
data['Time Difference'] = data.groupby('Customer ID')['Purchase Date'].diff().dt.days

# Tính thời gian trung bình giữa các giao dịch
average_time_between = data.groupby('Customer ID')['Time Difference'].mean().reset_index()
average_time_between.columns = ['Customer ID', 'Avg Time Between Purchases']

# Hiển thị kết quả
print("\nThời gian trung bình giữa các giao dịch của từng khách hàng:")
print(average_time_between.head(20))
```

**Output:**

Thời gian trung bình giữa các giao dịch của từng khách hàng:		
	Customer ID	Avg Time Between Purchases
0	1	NaN
1	10	113.000000
2	100	548.000000
3	1000	162.800000
4	10000	255.250000
5	10001	416.000000
6	10002	213.500000
7	10003	169.571429
8	10004	192.600000
9	10005	574.500000
10	10006	207.333333
11	10007	235.400000
12	10008	55.500000
13	10009	195.000000
14	1001	277.000000
15	10010	310.333333
16	10011	40.500000
17	10012	252.500000
18	10013	175.166667
19	10014	293.000000

→ Qua Output này ta có thể thấy rằng:

- Giúp hiểu rõ tần suất mua sắm ở cấp độ cá nhân: điều này thể hiện sau bao nhiêu ngày thì khách hàng đó quay lại mua hàng tiếp:

+ Customer ID = 1: Có giá trị NaN, có nghĩa là khách hàng này chỉ có một lần giao dịch, không có đủ dữ liệu để tính khoảng thời gian giữa các giao dịch.

+ Customer ID = 10: Thời gian trung bình giữa các giao dịch là 113 ngày. Điều này có nghĩa là khách hàng này thường quay lại mua sắm sau khoảng 113 ngày.

c. Kết hợp tần suất và thời gian trung bình giữa các giao dịch:

```

from sklearn.preprocessing import MinMaxScaler

# Kết hợp các dữ liệu: Purchase Frequency và Average Time Between Transactions
combined_data = purchase_frequency.merge(average_time_between_transactions, on='Customer ID')

# Chuẩn hóa các chỉ tiêu cần thiết
scaler = MinMaxScaler()
combined_data[['Average Time Between Transactions']] = scaler.fit_transform(
    combined_data[['Average Time Between Transactions']]
)

# Đảo ngược thang điểm của 'Average Time Between Transactions' (thời gian càng ngắn càng tốt)
combined_data['Average Time Between Transactions'] = 1 - combined_data['Average Time Between Transactions']

# Tính Loyalty Score
combined_data['Loyalty Score'] = combined_data[['
    'Purchase Frequency', 'Average Time Between Transactions'
]].mean(axis=1)

# Sắp xếp theo Loyalty Score giảm dần
combined_data = combined_data.sort_values(by='Loyalty Score', ascending=False)

```

```

# Hiển thị top 10 khách hàng trung thành
print("Top 10 khách hàng trung thành nhất:")
print(combined_data[['Customer ID', 'Loyalty Score', 'Purchase Frequency', 'Average Time Between Transactions']].head(10))

# Loại bỏ cột 'Customer Group' nếu có
if 'Customer Group' in combined_data.columns:
    combined_data = combined_data.drop(columns=['Customer Group'])
# Lưu kết quả ra file CSV
combined_data.to_csv('loyal_customers.csv', index=False, encoding='utf-8-sig')

```

## Output:

	Customer ID	Purchase Frequency	Average Time Between Transactions	Loyalty Score
1	39817	17	0.9487528132033008	8.97437640660165
2	47087	17	0.9421417854463616	8.97107089272318
3	36437	17	0.9381564141035259	8.969078207051762
4	28852	15	0.94770121101704	7.97385060550852
5	12529	15	0.9405744293216162	7.970287214660808
6	6426	15	0.9396099024756189	7.969804951237809
7	14400	15	0.9383774515057336	7.969188725752867
8	5252	15	0.9381631122066231	7.969081556103312
9	35424	15	0.9336619869253028	7.9668309934626516
10	809	15	0.9307684063873111	7.965384203193656
11	28144	14	0.9509492757804836	7.475474637890242
12	45979	14	0.9496220208898378	7.474811010444919
13	48471	14	0.9488718333429511	7.474435916671475
14	3576	14	0.9437936407178718	7.4718968203589355
15	5409	14	0.9360032315771251	7.4680016157885625
16	20497	14	0.9341566160770962	7.467078308038548
17	40531	14	0.9326562409833228	7.466328120491662
18	30838	14	0.9285590628426337	7.464279531421317

→ Qua đây bảng trên ta thấy được:

- **Nhóm khách hàng thường xuyên mua sắm:** Các khách hàng có Purchase Frequency cao (17 lần) và thời gian trung bình giữa các giao dịch thấp (<

0.95) là những khách hàng có giá trị cao đối với doanh nghiệp. Điểm trung thành của họ cũng cao hơn.

- **Nhóm khách hàng ít thường xuyên hơn:** Những khách hàng có Purchase Frequency thấp hơn (14 lần) và thời gian trung bình giữa các giao dịch cao hơn ( $> 0.93$ ) có thể cần được khuyến khích mua sắm thường xuyên hơn thông qua các chiến dịch tiếp thị hoặc ưu đãi.

→ Từ đó đưa ra các chiến lược:

- Tập trung vào nhóm khách hàng trung thành với tần suất cao để duy trì mối quan hệ và tăng giá trị lâu dài.
- Đưa ra các chương trình khuyến mãi hoặc ưu đãi để kích thích nhóm khách hàng ít mua hàng tăng tần suất giao dịch.

d. Biểu đồ thể hiện những khách hàng trung thành:

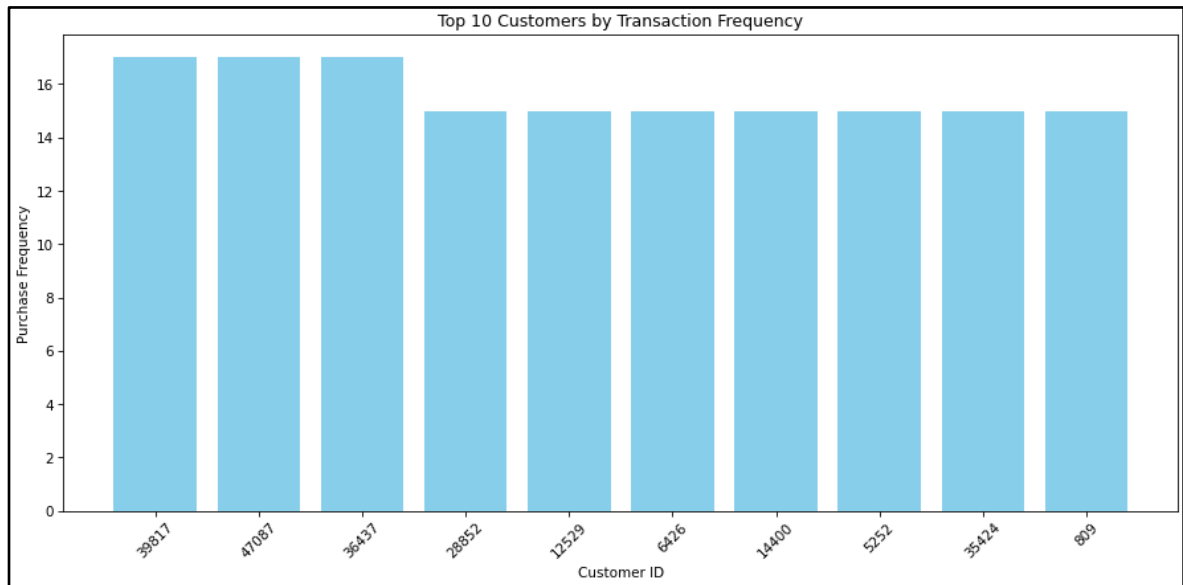
- Để vẽ được biểu đồ thể hiện những khách hàng trung thành cần dựa trên các tiêu chí: Purchase Frequency (Tổng số giao dịch), Average Time Between Transactions (Khoảng thời gian trung bình giữa các lần mua sắm) và Loyalty Score (Điểm trung thành)
- KH có tổng số giao dịch lớn nhất, khoảng thời gian giữa các lần mua sắm ngắn nhất và điểm trung thành cao nhất mới được gọi là khách hàng trung thành.

```
# Sắp xếp khách hàng theo các tiêu chí: Transaction Count, Average Time Between Transactions, và Loyalty Score
top_customers = combined_data.sort_values(
    by=['Purchase Frequency', 'Average Time Between Transactions', 'Loyalty Score'],
    ascending=[False, False, False] # Giảm dần theo tất cả tiêu chí
).head(10)

# Vẽ biểu đồ cột
plt.figure(figsize=(12, 6))
plt.bar(top_customers['Customer ID'], top_customers['Purchase Frequency'], color='skyblue')
plt.xlabel('Customer ID')
plt.ylabel('Purchase Frequency')
plt.title('Top 10 Customers by Transaction Frequency')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

# In thông tin chi tiết top 10 khách hàng
#print("Thông tin chi tiết Top 10 khách hàng:")
#print(top_customers[['Customer ID', 'Transaction Count', 'Average Time Between Transactions', 'Loyalty Score']])
```

**Output:**



→ **Kết luận từ biểu đồ:**

- **Nhóm khách hàng trung thành cao** (tần suất 17 lần) nên được duy trì và chăm sóc kỹ lưỡng, vì họ đóng góp lớn vào doanh thu.
- **Nhóm khách hàng trung thành thấp hơn** (tần suất 14–15 lần) có tiềm năng để nâng cao mức độ gắn bó thông qua các chương trình kích thích.

→ **Từ đó đưa ra các chiến lược đề xuất:**

1. Duy trì nhóm khách hàng trung thành cao (17 lần):

- **Cá nhân hóa trải nghiệm:** Gửi email hoặc tin nhắn cá nhân hóa với nội dung ưu đãi đặc biệt hoặc lời cảm ơn.
- **Chương trình khách hàng VIP:** Cung cấp các quyền lợi đặc biệt như giảm giá độc quyền, quà tặng sinh nhật, hoặc quyền truy cập sớm vào sản phẩm mới.
- **Khảo sát ý kiến:** Hỏi ý kiến nhóm khách hàng này để cải thiện sản phẩm/dịch vụ, nhằm tăng sự hài lòng và giữ chân họ.

2. Kích thích nhóm khách hàng trung thành thấp hơn (14–15 lần):

- **Ưu đãi định kỳ:** Áp dụng các chương trình khuyến mãi như giảm giá khi đạt một ngưỡng mua sắm nhất định (ví dụ: "Mua 3 lần tiếp theo, giảm 20%").
- **Chương trình tích điểm:** Tặng điểm thưởng cho mỗi giao dịch để khuyến khích mua hàng thường xuyên hơn.

3. Phân khúc khách hàng để tối ưu hóa chiến lược:

- Sử dụng dữ liệu như **tần suất mua hàng, thời gian giữa các giao dịch, và điểm trung thành** để phân nhóm khách hàng và đưa ra chiến lược phù hợp với từng nhóm.

### 3.2.4 Tần suất mua sắm của khách hàng theo giới tính

#### a. Tần suất mua sắm của khách hàng theo giới tính:

**Mục tiêu:** Hiểu hành vi mua sắm theo giới tính giúp xác định nhóm khách hàng (nam hoặc nữ) có xu hướng mua sắm thường xuyên hơn. Điều này hỗ trợ trong việc nhận diện các đặc điểm hành vi của từng nhóm.

```
data = pd.read_csv("updated_dataset.csv", sep=',')
# Lấy thông tin giới tính của từng khách hàng (đảm bảo không bị lặp)
customer_gender = data[['Customer ID', 'Gender']].drop_duplicates() # Thêm () để gọi method

# Chuyển cả hai cột thành chuỗi
purchase_frequency['Customer ID'] = purchase_frequency['Customer ID'].astype(str)
customer_gender['Customer ID'] = customer_gender['Customer ID'].astype(str)

# Kết hợp thông tin giới tính vào bảng tần suất mua sắm
purchase_frequency = purchase_frequency.merge(customer_gender, on='Customer ID', how='left')

# Tính toán tần suất mua sắm theo giới tính
frequency_by_gender = purchase_frequency.groupby('Gender')['Purchase Frequency'].agg(['mean', 'sum', 'count']).reset_index()
frequency_by_gender.columns = ['Gender', 'Average Purchase Frequency', 'Total Purchase Frequency', 'Customer Count']

# Hiển thị kết quả
print("Tần suất mua sắm theo giới tính:")
print(frequency_by_gender)

# Lưu kết quả ra file CSV nếu cần
frequency_by_gender.to_csv('purchase_frequency_by_gender.csv', index=False, encoding='utf-8-sig')
```

**Output:**

Tần suất mua sắm theo giới tính:			
	Gender	Average Purchase Frequency	Total Purchase Frequency \
0	Female	5.030852	125560
1	Male	5.034999	124440
	Customer Count		
0	24958		
1	24715		

→ Qua output này ta nhận thấy:

- Tần suất mua sắm trung bình của từng giới tính. Đây là giá trị trung bình số lần mua sắm trên mỗi khách hàng trong từng nhóm giới tính.
  - + Nữ (Female): 5.03 lần.
  - + Nam (Male): 5.03 lần. → Điều này cho thấy tần suất mua sắm trung bình của khách hàng nam và nữ tương đối đồng đều.
- Tổng số lần mua sắm của tất cả khách hàng trong từng nhóm giới tính.
  - + Nữ (Female): 125,560 lần.
  - + Nam (Male): 124,440 lần. → Khách hàng nữ có tổng tần suất mua sắm nhỉnh hơn một chút so với khách hàng nam.



- Số lượng khách hàng trong từng nhóm giới tính.
  - + Nữ (Female): 24,958 khách hàng.
  - + Nam (Male): 24,715 khách hàng.

→ Số lượng khách hàng nữ cũng nhỉnh hơn một chút so với khách hàng nam.

### Nhận xét:

- Dựa trên kết quả này, không có sự khác biệt quá lớn giữa hành vi mua sắm của hai giới tính, tuy nhiên, nhóm nữ có mức độ mua sắm tổng thể cao hơn một chút.

### b. Biểu đồ thể hiện tần suất mua sắm của khách hàng theo giới tính:

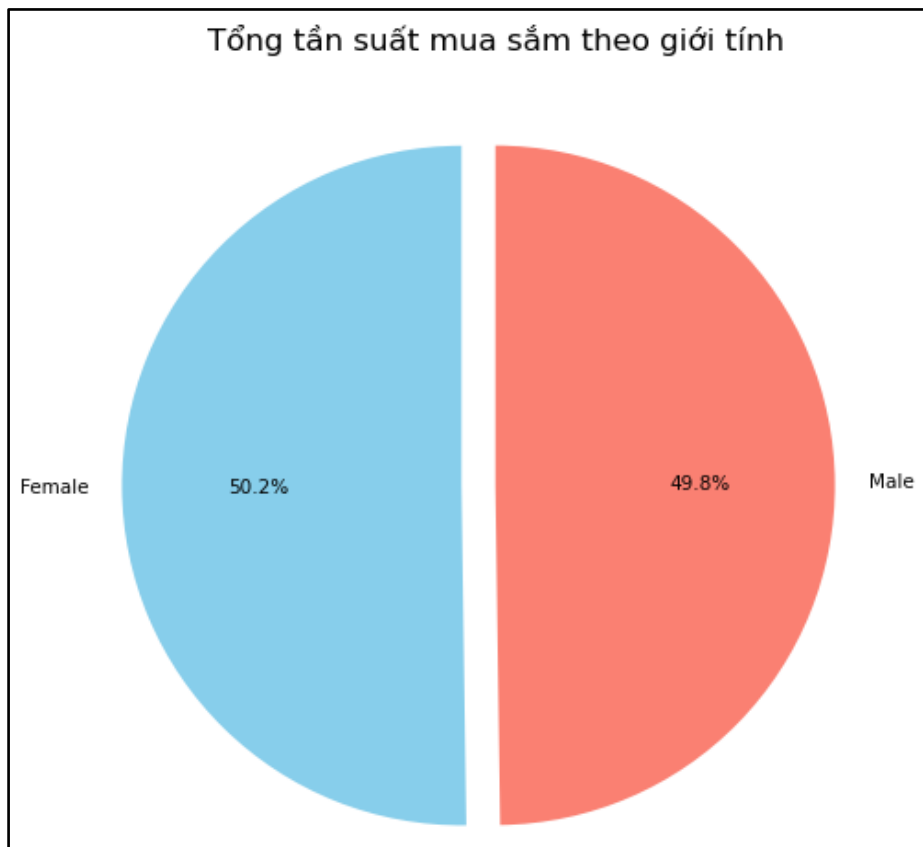
```
# Dữ liệu cần cho biểu đồ tròn (tổng tần suất mua sắm theo giới tính)
labels = frequency_by_gender['Gender'] # Nhãn giới tính
sizes = frequency_by_gender['Total Purchase Frequency'] # Tổng tần suất mua sắm
colors = ['skyblue', 'salmon'] # Màu sắc cho từng giới tính (tùy chọn)

# Vẽ biểu đồ tròn
plt.figure(figsize=(8, 8))
plt.pie(
    sizes,
    labels=labels,
    autopct='%1.1f%%', # Hiển thị phần trăm trên biểu đồ
    startangle=90, # Bắt đầu từ góc 90 độ
    colors=colors, # Áp dụng màu sắc
    explode=[0.1, 0] if len(labels) == 2 else None # Nổi bật nhóm đầu tiên nếu chỉ có 2 giới tính
)

# Thêm tiêu đề
plt.title('Total Purchase Frequency by Gender', fontsize=16)

# Hiển thị biểu đồ
plt.show()
```

### Output:



→ Biểu đồ trên thể hiện rõ tỷ lệ tổng tần suất mua sắm của khách hàng nữ là **50.2%**, trong khi khách hàng nam là **49.8%**. Qua đó thấy rằng cả nam và nữ đều có hành vi mua sắm ổn định, không có sự khác biệt đáng kể. Tuy nhiên, khách hàng nữ nhỉnh hơn về **số lượng khách hàng** và tổng tần suất, điều này có thể gợi ý rằng nhóm nữ có một vai trò nhỏ nhưng đáng chú ý trong tổng doanh thu.

→ **Đưa ra những chiến lược phù hợp:**

- Các chiến lược quảng cáo có thể được phân bổ đồng đều giữa nam và nữ. Tuy nhiên, có thể ưu tiên một chút vào nhóm nữ vì họ chiếm phần lớn trong tổng tần suất mua sắm và số lượng khách hàng.
- Phát triển các chương trình khuyến mãi nhắm đến sở thích **mua sắm của nữ giới**, chẳng hạn như các ưu đãi thời trang, làm đẹp, hoặc các sản phẩm thường được tiêu dùng bởi phụ nữ.
- Phân bổ nguồn lực chăm sóc khách hàng một cách cân bằng giữa nam và nữ.
- Đối với nhóm nữ, có thể nhấn mạnh vào trải nghiệm mua sắm và các dịch vụ hỗ trợ sau bán hàng, vì họ có thể nhạy bén hơn với chất lượng dịch vụ.

c. Tần suất mua sắm của khách hàng theo giới tính từng tháng (theo năm):

**Mục tiêu:**

- Tìm hiểu sự thay đổi tần suất mua sắm theo tháng trong năm cho cả nam và nữ → Có cái nhìn rõ ràng về những thời điểm trong năm khi khách hàng của

từng giới tính có xu hướng mua sắm nhiều hơn. Điều này có thể liên quan đến các dịp lễ hội, khuyến mãi, hay mùa giảm giá.

- Xác định mùa vụ và các sự kiện đặc biệt ảnh hưởng đến hành vi mua sắm để phân tích những tháng có tần suất mua sắm cao hơn đối với từng giới tính và liên kết với các sự kiện (ví dụ: lễ Tết, Black Friday, các chiến dịch khuyến mãi) → Hiểu rõ hơn về cách các sự kiện, mùa vụ có thể tác động đến hành vi mua sắm của từng giới tính, từ đó giúp các doanh nghiệp lên kế hoạch marketing hiệu quả hơn.

- Phân tích sự thay đổi của tần suất mua sắm qua các năm đối với từng giới tính → Điều này sẽ giúp bạn xác định được các xu hướng lâu dài, chẳng hạn như liệu nam và nữ có xu hướng mua sắm nhiều hơn theo năm hay không và tại sao.

```
# Tính tần suất giao dịch của từng khách hàng theo từng tháng và năm
transaction_frequency = (
    data.groupby(['Gender', 'Purchase Year', 'Purchase Month'], observed=True)
    .size()
    .reset_index(name='Transaction Count') # Tạo cột 'Transaction Count'
)

# Sắp xếp dữ liệu theo Gender, Purchase Year và Purchase Month
transaction_frequency = transaction_frequency.sort_values(by=['Gender', 'Purchase Year', 'Purchase Month'])

# Hiển thị dữ liệu
print("Tần suất giao dịch theo giới tính từng tháng (theo năm):")
print(transaction_frequency)
```

**Output:**

Tần suất giao dịch theo giới tính từng tháng (theo năm):

	Gender	Purchase Year	Purchase Month	Transaction Count
0	Female	2020	January	2922
1	Female	2020	February	2690
2	Female	2020	March	2809
3	Female	2020	April	2724
4	Female	2020	May	2919
5	Female	2020	June	2742
6	Female	2020	July	2930
7	Female	2020	August	2908
8	Female	2020	September	2840
9	Female	2020	October	2913
10	Female	2020	November	2791
11	Female	2020	December	2973
12	Female	2021	January	2746
13	Female	2021	February	2597
14	Female	2021	March	2985
15	Female	2021	April	2784
16	Female	2021	May	2874
17	Female	2021	June	2778
18	Female	2021	July	2816
19	Female	2021	August	2806
20	Female	2021	September	2743
21	Female	2021	October	2984
22	Female	2021	November	2767
23	Female	2021	December	2764
24	Female	2022	January	2828

45	Male	2020	January	2765
46	Male	2020	February	2704
47	Male	2020	March	2874
48	Male	2020	April	2775
49	Male	2020	May	2815
50	Male	2020	June	2797
51	Male	2020	July	2921
52	Male	2020	August	2861
53	Male	2020	September	2751
54	Male	2020	October	2865
55	Male	2020	November	2837
56	Male	2020	December	2919
57	Male	2021	January	2943
58	Male	2021	February	2554
59	Male	2021	March	2929
60	Male	2021	April	2732
61	Male	2021	May	2793
62	Male	2021	June	2758
63	Male	2021	July	2796
64	Male	2021	August	2868
65	Male	2021	September	2702
66	Male	2021	October	2744
67	Male	2021	November	2783
68	Male	2021	December	2853
69	Male	2022	January	2964
70	Male	2022	February	2605
71	Male	2022	March	2805

→ Qua đoạn output trên ta thấy được:

**- Sự ổn định theo thời gian:**

- + Cả hai giới (Nam và Nữ) đều có tần suất giao dịch khá ổn định trong suốt các tháng từ năm 2020 đến 2023. Số lượng giao dịch dao động trong khoảng từ 2,500 đến 3,000 giao dịch mỗi tháng, cho thấy thói quen mua sắm duy trì đều đặn bất kể mùa hay giới tính.
- + Tuy nhiên tới Tháng 9 năm 2023 thì có sự sụt giảm đột ngột: Tần suất giao dịch của Nữ là 1325 và Nam là 1347. Điều này xảy ra có khả năng liên quan đến tình trạng kinh tế thế giới.

**- So sánh giữa Nam và Nữ:**

- + Tần suất giao dịch của Nữ nhìn chung cao hơn Nam trong nhiều tháng, nhưng khoảng cách không đáng kể (thường chênh lệch khoảng 50–200 giao dịch mỗi tháng).
- + Điều này cho thấy phụ nữ có xu hướng mua sắm nhiều hơn một chút so với nam giới.

**- Xu hướng theo năm:**

- + Năm 2020 và 2021: Cả hai giới đều có xu hướng tăng nhẹ trong các tháng cuối năm, đặc biệt là tháng 11 và tháng 12, có thể do nhu cầu mua sắm cho các dịp lễ hội cuối năm (như Giáng Sinh, Tết).
- + Năm 2022: Số liệu cũng duy trì xu hướng cao ở các tháng đầu năm, cho thấy thói quen mua sắm liên tục được duy trì.

d. Biểu đồ thể hiện Tần suất mua sắm của khách hàng theo giới tính từng tháng (theo năm):

**Mục tiêu:** Biểu đồ này sẽ cung cấp cái nhìn tổng quan về tần suất mua sắm của khách hàng theo giới tính từng tháng (theo năm). Qua đó hỗ trợ phân tích hành vi khách hàng, so sánh giữa giới tính, và lập kế hoạch chiến lược kinh doanh, tiếp thị hiệu quả hơn.

```
# Danh sách tháng theo thứ tự
month_order = ['January', 'February', 'March', 'April', 'May', 'June',
               'July', 'August', 'September', 'October', 'November', 'December']

# Chuyển cột 'Purchase Month' thành dạng Categorical và sắp xếp theo thứ tự tháng
transaction_frequency['Purchase Month'] = pd.Categorical(
    transaction_frequency['Purchase Month'], categories=month_order, ordered=True
)

# Sắp xếp dữ liệu theo 'Purchase Year' và 'Purchase Month'
transaction_frequency = transaction_frequency.sort_values(by=['Purchase Year', 'Purchase Month'])

# Lấy danh sách các năm duy nhất
years = transaction_frequency['Purchase Year'].unique()

# Tạo biểu đồ cho từng năm
fig, axes = plt.subplots(2, 2, figsize=(20, 12), dpi=120) # Tăng độ phân giải với dpi
axes = axes.flatten() # Chuyển mảng 2D của axes thành 1D để dễ xử lý

# Lặp qua từng năm và vẽ biểu đồ tương ứng
for i, year in enumerate(years):
    # Lọc dữ liệu cho năm hiện tại
    data_year = transaction_frequency[transaction_frequency['Purchase Year'] == year]
```

```
# Vẽ biểu đồ cột nhóm
sns.barplot(
    data=data_year,
    x='Purchase Month', # Trục X là tháng
    y='Transaction Count', # Trục Y là số giao dịch
    hue='Gender', # Phân biệt theo giới tính
    palette='husl', # Thay đổi palette màu sắc
    ax=axes[i], # Vẽ lên trục tương ứng
    edgecolor='black' # Thêm đường viền cho cột
)

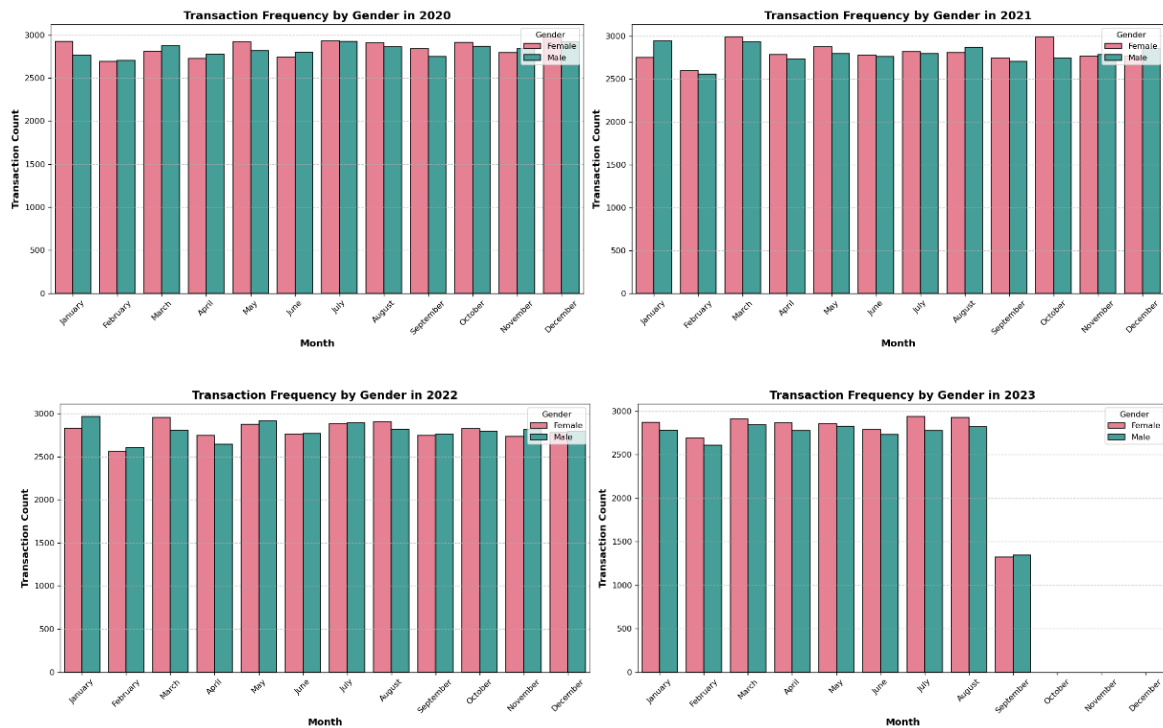
# Cải thiện hiển thị cho từng biểu đồ
axes[i].set_title(f'Transaction Frequency by Gender in {year}', fontsize=14, weight='bold')
axes[i].set_xlabel('Month', fontsize=12, weight='bold')
axes[i].set_ylabel('Transaction Count', fontsize=12, weight='bold')
axes[i].legend(title='Gender', loc='upper right', fontsize=10)
axes[i].tick_params(axis='x', rotation=45) # Xoay nhãn tháng
axes[i].grid(axis='y', linestyle='--', alpha=0.7) # Thêm đường kẻ phụ ngang

# Tinh chỉnh khoảng cách giữa các biểu đồ
plt.tight_layout()

# Lưu biểu đồ vào tệp hình ảnh
plt.savefig('transaction_frequency_by_gender.png', dpi=60, bbox_inches='tight')

plt.show()
```

**Output:**



→ **Kết luận biểu đồ:**

#### - Xu hướng chung:

- + Tần suất mua sắm ổn định qua các tháng và năm từ 2020 đến 2022, điều này cho thấy hành vi mua sắm của khách hàng nước ngoài duy trì mức độ nhất quán.
- + Năm 2023: Sự sụt giảm đáng kể vào tháng 9 và không có dữ liệu vào tháng 10 đến 12, có thể do thiếu dữ liệu hoặc thay đổi xu hướng mua sắm (như sự dịch chuyển sang các kênh mua sắm khác hoặc biến động kinh tế).

#### - Mùa cao điểm mua sắm:

- + Đầu năm (tháng 1-2): Có sự tăng trưởng đáng kể, có thể liên quan đến các sự kiện như Năm mới (New Year) và các đợt giảm giá đầu năm.
- + Giữa năm (tháng 4-8): Thời kỳ mua sắm ổn định, có thể do các sự kiện như Black Friday Summer Sales, Back-to-School Sales.
- + Cuối năm (tháng 10-12): Đây thường là mùa mua sắm cao điểm toàn cầu với các sự kiện như Black Friday, Cyber Monday, Giáng Sinh (Christmas).

#### - Sự khác biệt giới tính:

- + Hành vi mua sắm giữa nam giới và nữ giới tương đối giống nhau, với tần suất gần như cân bằng trong từng tháng.
- + Tuy nhiên, vào một số thời điểm như tháng 4-6, nữ giới có xu hướng mua sắm nhiều hơn, có thể liên quan đến nhu cầu tiêu dùng đặc biệt như mua sắm mùa hè hoặc các dịp lễ liên quan.

→ **Đưa ra một số chiến lược:**

1. Tận dụng mùa mua sắm cao điểm toàn cầu:

- Tháng 1-2 (New Year Sales):
  - + Triển khai các chương trình giảm giá đầu năm như "New Year Clearance" để thu hút khách hàng.
  - + Tập trung quảng bá các mặt hàng nhu yếu phẩm, thời trang và đồ gia dụng.
- Tháng 7-8 (Back-to-School Season):
  - + Khuyến mãi các sản phẩm như đồ dùng học tập, quần áo, đồ điện tử.
  - + Tạo các gói sản phẩm dành cho học sinh và phụ huynh.
- Tháng 11-12 (Holiday Season):
  - + Đẩy mạnh các chiến dịch Black Friday, Cyber Monday và Christmas với các chương trình giảm giá mạnh, voucher và miễn phí vận chuyển.
  - + Khuyến khích mua sắm quà tặng thông qua chiến dịch "Holiday Gifting" nhắm đến cả nam và nữ.

2. Cá nhân hóa theo giới tính và nhu cầu mua sắm:

- Đối với nữ giới: Tập trung quảng bá các sản phẩm như thời trang, làm đẹp vào tháng 4-6.
- Đối với nam giới: Tăng cường khuyến mãi sản phẩm công nghệ, điện tử hoặc thể thao vào các mùa mua sắm lớn như Black Friday.

3. Chiến lược ưu đãi và giữ chân khách hàng:

- Chương trình khách hàng thân thiết (Loyalty Programs): Cung cấp điểm thưởng và ưu đãi cho khách hàng mua sắm thường xuyên.
- Chiết khấu có thời hạn: Sử dụng các ưu đãi giới hạn thời gian để tạo tâm lý mua hàng ngay lập tức.
- Voucher cá nhân hóa: Gửi mã giảm giá hoặc quà tặng sinh nhật thông qua email marketing.

**3.3 Xu hướng mua sắm theo mùa thành các tháng và quý để xem liệu có sự gia tăng doanh số trong các mùa lễ hội ?**

**3.3.1 Thống kê doanh thu qua từng năm**

**Mục tiêu:**

- **Đánh giá tổng quan tình hình kinh doanh:**
  - + Thống kê doanh thu giúp đánh giá hiệu suất kinh doanh của từng năm, xác định xu hướng tăng trưởng hay suy giảm của doanh nghiệp theo thời gian.
- **Phân tích sự biến động doanh thu:**



- + Thông qua các chỉ số như **min**, **max**, **mean** (trung bình), **median** (trung vị), và **sum** (tổng doanh thu), bạn có thể thấy:
  - Sự chênh lệch giữa các mức doanh thu.
  - Độ tập trung của doanh thu quanh một giá trị trung tâm (trung bình hay trung vị).
  - Xác định khoảng dao động doanh thu cao nhất và thấp nhất.
- **Xác định xu hướng tăng trưởng hoặc suy giảm:**
  - + So sánh doanh thu giữa các năm giúp phát hiện:
    - Năm nào doanh thu cao nhất?
    - Năm nào doanh thu giảm mạnh?
    - Xu hướng tăng trưởng ổn định hay có biến động bất thường.

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px

# Đọc dữ liệu
data = pd.read_csv('updated_dataset.csv')

# Kiểm tra kết quả
print("Kiểm tra định dạng dữ liệu ban đầu: ")
print(data['Purchase Date'].head())
data['Purchase Date'] = pd.to_datetime(data['Purchase Date'], format='%Y-%m-%d %H:%M:%S')
print("Dữ liệu sau khi chuyển đổi định dạng: ")
print(data['Purchase Date'].head())

# Tạo cột 'Purchase Year' từ 'Purchase Date'
data['Purchase Year'] = data['Purchase Date'].dt.year
# In kết quả kiểm tra
print("Dữ liệu sau khi thêm cột Purchase Year: ")
print(data[['Purchase Date', 'Purchase Year']].head())

# 1. Tổng quan về doanh thu hàng năm
annual_sales_summary = data.groupby('Purchase Year')['Product Price'].agg(['min', 'max', 'mean', 'median', 'sum',])
print("Thông kê doanh thu qua từng năm: ")
annual_sales_summary
```

## Output:

```
Kiểm tra định dạng dữ liệu ban đầu:
0    2020-09-08 09:38:00
1    2022-03-05 12:56:00
2    2022-05-23 18:18:00
3    2020-11-12 13:13:00
4    2020-11-27 17:55:00
Name: Purchase Date, dtype: object
Dữ liệu sau khi chuyển đổi định dạng:
0    2020-09-08 09:38:00
1    2022-03-05 12:56:00
2    2022-05-23 18:18:00
3    2020-11-12 13:13:00
4    2020-11-27 17:55:00
Name: Purchase Date, dtype: datetime64[ns]
Dữ liệu sau khi thêm cột Purchase Year:
      Purchase Date  Purchase Year
0 2020-09-08 09:38:00           2020
1 2022-03-05 12:56:00           2022
2 2022-05-23 18:18:00           2022
3 2020-11-12 13:13:00           2020
4 2020-11-27 17:55:00           2020
```

Thống kê doanh thu qua từng năm:						
	min	max	mean	median	sum	
Purchase Year						
2020	10	500	254.352061	254.0	17307386	
2021	10	500	255.581857	256.0	17149287	
2022	10	500	253.557534	254.0	17031206	
2023	10	500	255.352591	256.0	12176999	

→ Từ kết quả thống kê trên ta có thể nhận xét như sau:

- Doanh thu ổn định từ 2020 đến 2022 với giá trị tổng doanh thu tương tự nhau (~17 triệu).
- Giảm mạnh vào năm 2023, tổng doanh thu chỉ còn khoảng 12 triệu, cho thấy dấu hiệu suy giảm.
- Trung bình và trung vị doanh thu trong các năm không thay đổi nhiều, điều này cho thấy giá trị các giao dịch khá đồng đều.

### 3.3.2 Xu hướng mua sắm theo mùa (tháng và quý)

**Mục tiêu:** Hướng phân tích này tập trung vào việc xác định sự thay đổi doanh thu trong các tháng/quý, đặc biệt là trong các mùa lễ hội

```
# Tạo cột 'Purchase Month' từ tháng của 'Purchase Date'
data['Purchase Month'] = data['Purchase Date'].dt.month_name()
#Sắp xếp thứ tự tháng sau đó chuyển thành dạng danh mục
month_order = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December']

data['Purchase Month'] = pd.Categorical(data['Purchase Month'],
                                       categories=month_order,
                                       ordered=True)

#Tính tổng và trung bình doanh thu hàng tháng
print("\nTổng và trung bình doanh thu hàng tháng: ")
monthly_sales = data.groupby(['Purchase Year', 'Purchase Month'], observed=True).agg(Total_Revenue=('Product Price', 'sum'),
                                                                                      Count=('Product Price', 'size'),
                                                                                      Avg_Revenue=('Product Price', 'mean')).reset_index()

print(monthly_sales)
```

**Output:**

Tổng và trung bình doanh thu hàng tháng:						
	Purchase Year	Purchase Month	Total_Revenue	Count	Avg_Revenue	
0	2020	January	1425914	5687	250.732196	
1	2020	February	1374063	5394	254.739155	
2	2020	March	1451223	5683	255.362133	
3	2020	April	1392226	5499	253.178032	
4	2020	May	1468358	5734	256.079177	
5	2020	June	1428564	5539	257.910092	
6	2020	July	1492024	5851	255.003247	
7	2020	August	1454314	5769	252.091177	
8	2020	September	1422890	5591	254.496512	
9	2020	October	1467944	5778	254.057459	
10	2020	November	1426961	5628	253.546731	
11	2020	December	1502905	5892	255.075526	
12	2021	January	1463209	5689	257.199684	
13	2021	February	1306953	5151	253.728014	
14	2021	March	1512837	5914	255.806053	
15	2021	April	1419166	5516	257.281726	

**Qua output này ta có thể thấy rằng:**

- Doanh thu tăng mạnh vào cuối năm (tháng 12) do mùa lễ hội, với T12 năm 2020 đạt doanh thu cao nhất (1.502.905)
- Doanh thu ổn định trong các tháng, dao động từ 1,37 triệu đến 1,5 triệu
- Số lượng đơn hàng tương đối đều, từ 5.100 đến 5.900, cho thấy tần suất mua sắm ổn định
- Doanh thu trung bình trên mỗi đơn hàng giữ mức ổn định (252 - 258)
- Xu hướng tăng nhẹ đầu năm 2021 so với cùng kỳ 2020 (T1)

a. Tính doanh thu theo từng giao dịch:

**Mục tiêu:** Việc tính doanh thu theo từng giao dịch giúp chi tiết hóa dữ liệu, xác định chính xác các thời điểm doanh thu tăng cao trong mùa lễ hội, từ đó phát hiện biến động và đánh giá thói quen chi tiêu của khách hàng. Điều này hỗ trợ phân tích vai trò của từng đơn hàng trong tổng doanh thu, cung cấp cơ sở cho chiến lược kinh doanh và tối ưu hóa hoạt động bán hàng vào các thời điểm có doanh thu cao.

```
# 1. Tính doanh thu theo từng giao dịch

data['Total Price'] = data['Quantity'] * data['Product Price']

# Tạo thêm các cột bổ sung để phân tích
data['Purchase Month'] = data['Purchase Date'].dt.month_name()
data['Purchase Year'] = data['Purchase Date'].dt.year
data['Purchase Quarter'] = data['Purchase Date'].dt.quarter
data['Season'] = data['Purchase Date'].dt.month.map({
    12: 'Winter', 1: 'Winter', 2: 'Winter',
    3: 'Spring', 4: 'Spring', 5: 'Spring',
    6: 'Summer', 7: 'Summer', 8: 'Summer',
    9: 'Fall', 10: 'Fall', 11: 'Fall'
})

# Kiểm tra dữ liệu sau khi thêm cột
print("Dữ liệu sau khi thêm cột: ")
print(data.head())
```

## Output:

```
Dữ liệu sau khi thêm cột:
  Customer ID  Purchase Date  Product Price  Quantity  Customer Age \
0      46251  2020-09-08 09:38:00          8748           3          37
1      46251  2022-03-05 12:56:00       1916928           4          37
2      46251  2022-05-23 18:18:00        18432           2          37
3      46251  2020-11-12 13:13:00          196           1          37
4      13593  2020-11-27 17:55:00          449           1          49

  Gender  Purchase Year  Purchase Month  Purchase Quarter  Season \
0   Male           2020      September              3     Fall
1   Male           2022         March              1  Spring
2   Male           2022         May              2  Spring
3   Male           2020      November              4     Fall
4  Female           2020      November              4     Fall

  Holiday Season  Total Price
0  Non-Holiday Season      26244
1  Non-Holiday Season    7667712
2   Holiday Season      36864
3   Holiday Season         196
4   Holiday Season         449
```

→ Nhìn vào output ta có thể dễ dàng xác định được:

- Tổng doanh thu giữa các tháng cao điểm và các tháng thấp điểm, ngoài ra còn có thể biết được các thông tin của khách hàng và chi tiết từng giao dịch của 1 khách hàng cụ thể.
- Từ đó có thể so sánh doanh thu cụ thể của từng giao dịch qua nhiều thời điểm và có thể xác định tháng nào hoặc mùa nào đóng góp doanh thu lớn

nhất một cách dễ dàng, thuận tiện cho việc xác định hành vi mua sắm của khách hàng thông qua mỗi giao dịch của họ từ nhiều thời điểm khác nhau.

→ Từ đó ta có thể biết được:

- Xu hướng theo thời gian: Phân tích sự gia tăng doanh thu vào các tháng và quý cụ thể.
- Phát hiện mùa cao điểm: Nhận diện các mùa lễ hội có doanh số tăng đột biến.
- Đánh giá hiệu quả kinh doanh: So sánh doanh thu giữa các tháng và quý để đo lường hiệu suất.
- Phân tích hành vi khách hàng: Hiểu thói quen mua sắm trong các thời điểm khác nhau.
- Hỗ trợ lập kế hoạch: Làm cơ sở cho chiến lược tiếp thị và quản lý hàng tồn kho.
- Tối ưu hóa doanh thu: Tập trung khai thác các thời điểm mua sắm mạnh mẽ để tăng doanh số.

b. Biểu đồ thể hiện doanh thu theo tháng:

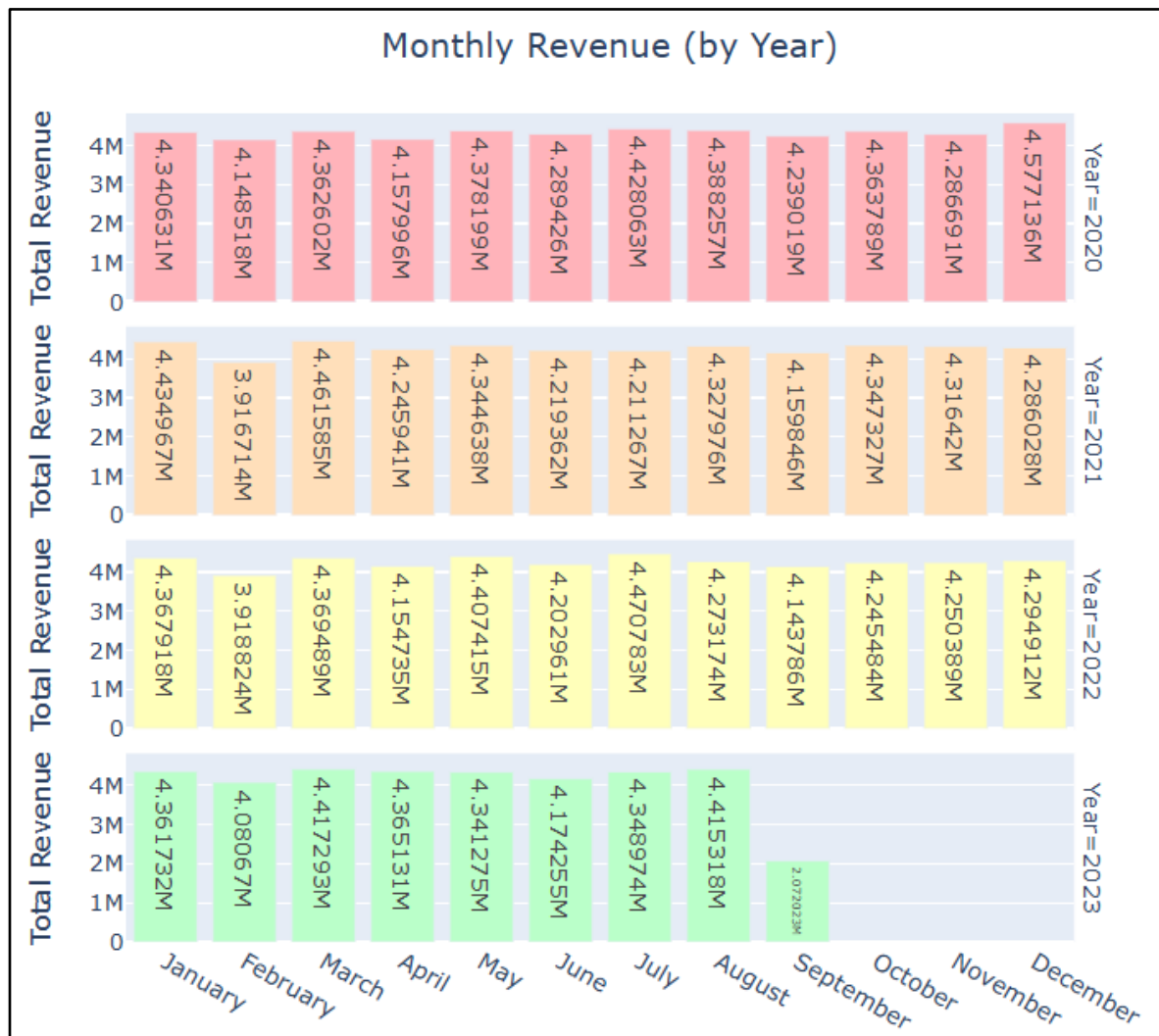
**Mục tiêu:** Biểu đồ doanh thu theo tháng giúp trực quan hóa xu hướng doanh thu theo thời gian, dễ dàng phát hiện các tháng có sự gia tăng doanh số trong mùa lễ hội. Biểu đồ này cũng giúp so sánh hiệu quả doanh thu giữa các tháng và quý, từ đó hỗ trợ việc điều chỉnh chiến lược kinh doanh và marketing, tối ưu hóa doanh thu trong các thời điểm quan trọng.

```

#2. Phân tích doanh thu theo thời gian từng tháng
# Nhóm dữ liệu theo 'Purchase Year' và 'Purchase Month' để tính tổng doanh thu
monthly_sales = data.groupby(['Purchase Year', 'Purchase Month'], observed=True).agg(
    Total_Revenue=('Total Price', 'sum'), # Tổng doanh thu của từng tháng
    Order_Count=('Total Price', 'size'), # Tổng số Lượng đơn hàng của từng tháng
    Avg_Revenue=('Total Price', 'mean') # Doanh thu trung bình của từng giao dịch
).reset_index()
# Sắp xếp thứ tự các tháng trong năm
month_order = ['January', 'February', 'March', 'April', 'May', 'June',
               'July', 'August', 'September', 'October', 'November', 'December']
monthly_sales['Purchase Month'] = pd.Categorical(
    monthly_sales['Purchase Month'], categories=month_order, ordered=True
)
# Sắp xếp dữ liệu theo 'Purchase Year' và 'Purchase Month'
monthly_sales = monthly_sales.sort_values(by=['Purchase Year', 'Purchase Month'])
# Kiểm tra kết quả
print("\nDoanh thu qua từng tháng:")
# Định nghĩa bảng màu pastel
pastel_colors = ['#FFB3BA', '#FFDFBA', '#FFFFBA', '#BAFFC9', '#BAE1FF']
# Vẽ biểu đồ doanh thu theo tháng
fig1 = px.bar(
    monthly_sales,
    x='Purchase Month', # Trục X là các tháng
    y='Revenue', # Trục Y là doanh thu
    color='Purchase Year', # Màu sắc theo từng năm
    facet_row='Purchase Year', # Tách biểu đồ theo từng năm
    title='Monthly Revenue (by Year)',
    labels={'Revenue': 'Total Revenue', 'Purchase Year': 'Year'},
    hover_data={'Revenue': ':.2f'}, # Hiển thị doanh thu chính xác
    text_auto=True,
    color_discrete_sequence=pastel_colors # Sử dụng bảng màu tự định nghĩa
)
# Cập nhật bố cục biểu đồ
fig1.update_layout(
    title={'x': 0.5, 'y': 0.9}, # Căn giữa tiêu đề
    xaxis_title=None,
    width=650,
    height=600,
    showlegend=False
)
fig1.show()

```

**Output:**



#### Nhận xét biểu đồ:

Quan sát biểu đồ ta có thể nhận biết được hành vi mua sắm trực tuyến của khách hàng dựa trên việc phân tích doanh thu theo tháng như sau:

- Có xu hướng mua sắm mạnh vào các dịp lễ hội:
  - + Người tiêu dùng có xu hướng chi tiêu mạnh vào các sự kiện giảm giá lớn, đặc biệt là cuối năm, nơi các sự kiện như Black Friday và Giáng sinh chi phối hành vi mua sắm.
  - + Các giai đoạn cuối quý 4 luôn là thời điểm quan trọng để thúc đẩy doanh thu.
- Thất chặt chi tiêu vào đầu và giữa năm:
  - + Giai đoạn đầu năm (tháng 1, 2) và giữa năm (tháng 6-9) là thời điểm doanh thu thấp. Chứng tỏ khách hàng thường có xu hướng giảm chi tiêu cho việc mua sắm vào các tháng không có sự kiện lớn và tập trung vào các nhu cầu thiết yếu.

#### Kết luận và đưa ra đề xuất:

- Biểu đồ cho thấy xu hướng mua sắm của khách hàng bị chi phối bởi các mùa lễ hội lớn, với đỉnh doanh thu thường rơi vào tháng 11 và 12.
- Các tháng đầu năm và giữa năm thường là giai đoạn doanh thu giảm, do không có nhiều sự kiện mua sắm lớn.
- Doanh nghiệp cần đẩy mạnh các chiến dịch tiếp thị và khuyến mãi vào mùa lễ hội cuối năm, đồng thời có các chương trình kích cầu vào những tháng thấp điểm như tháng 1, 2 và tháng 9.

=> Nhìn chung, hành vi mua sắm trực tuyến của người tiêu dùng chịu ảnh hưởng lớn từ các sự kiện văn hóa và thương mại theo thời gian.

### c. Biểu đồ thể hiện doanh thu theo quý:

#### Mục tiêu:

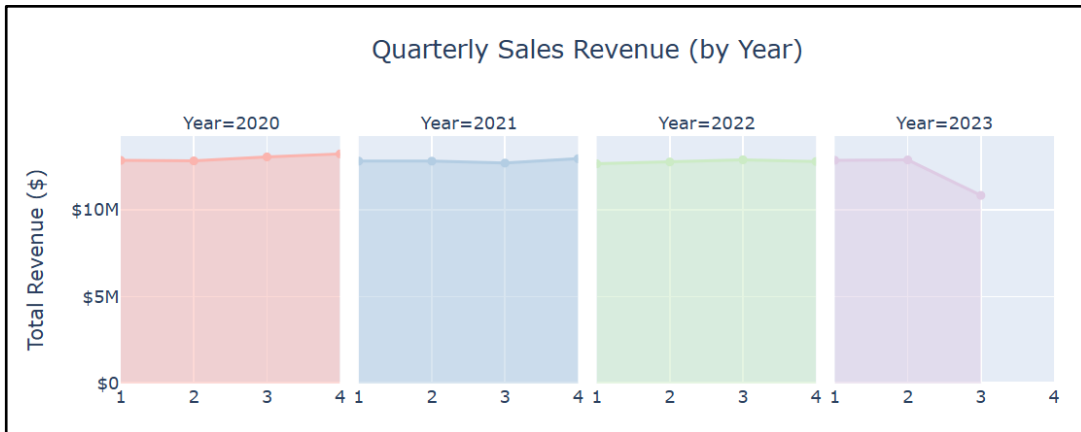
- Tổng hợp và trực quan hóa xu hướng doanh thu trong từng quý để nhận diện các giai đoạn mua sắm cao điểm.
- So sánh doanh thu giữa các quý nhằm xác định sự gia tăng doanh số vào các mùa lễ hội.
- Đưa ra bức tranh tổng quan về sự biến động doanh thu trong năm, từ đó phát hiện các quý có doanh thu đột biến hoặc suy giảm.
- Hỗ trợ doanh nghiệp tối ưu hóa chiến lược kinh doanh, tập trung nguồn lực vào các quý có tiềm năng tăng trưởng mạnh mẽ.

```
#3. Phân tích doanh thu theo quý
quarterly_sales = data.groupby(['Purchase Year', 'Purchase Quarter']).agg(Total_Revenue=('Total Price', 'sum'),
                                                                           Avg_Revenue=('Total Price', 'mean'),
                                                                           Count=('Total Price', 'size')).reset_index()

# Chuyển đổi cột 'Purchase Year' sang dạng chuỗi
quarterly_sales['Purchase Year'] = quarterly_sales['Purchase Year'].astype(str)
# Vẽ biểu đồ
fig2 = px.area(
    quarterly_sales,
    x='Purchase Quarter',
    y='Total_Revenue',
    color='Purchase Year',
    facet_col='Purchase Year',
    color_discrete_sequence=px.colors.qualitative.Pastel1, # Màu Pastel
    title='Quarterly Sales Revenue (by Year)',
    labels={'Total_Revenue': 'Total Revenue ($)', 'Purchase Quarter': 'Quarter', 'Purchase Year': 'Year'},
    hover_data={'Total_Revenue': ':$, .0f'},
    markers=True
)
# Cập nhật trục Y để hiển thị giá trị tiền tệ với tiền tố "$"
fig2.update_yaxes(tickprefix="$")
# Tùy chỉnh bố cục biểu đồ
fig2.update_layout(
    title={'x': 0.5, 'y': 0.9},
    xaxis_title=None,
    xaxis2_title=None,
    xaxis3_title=None,
    xaxis4_title=None,
    width=800,
    height=350,
    showlegend=False
)
# Hiển thị biểu đồ
fig2.show()
```



## Output:



### Nhận xét biểu đồ:

Nhìn vào biểu đồ ta có thể thấy được xu hướng doanh thu theo quý có những đặc trưng như sau:

#### 1. Tổng quan xu hướng doanh thu theo quý:

- Sự thay đổi doanh thu qua các quý trong mỗi năm có thể được nhận thấy rõ ràng nhờ biểu đồ này. Các đỉnh doanh thu thường xuất hiện vào Q4 (Quý 4), trùng với các mùa lễ hội lớn cuối năm như Black Friday, Cyber Monday, và Giáng Sinh.
- Quý 1 (Q1) và Quý 2 (Q2) thường có doanh thu thấp hơn do thời gian này ít các dịp mua sắm lớn.
- Quý 3 (Q3) có dấu hiệu tăng trưởng hơn so với Q1 và Q2, chủ yếu nhờ vào các sự kiện như Back-to-School vào tháng 8 và sự chuẩn bị cho các tháng lễ hội cuối năm.

#### 2. Quý 4 là cao điểm mua sắm trong năm:

- Quý 4 (tháng 10, 11, 12) nổi bật với doanh thu cao nhất trong năm, do trùng với các dịp lễ hội mua sắm lớn như:
  - + Tháng 10: Halloween.
  - + Tháng 11: Black Friday, Cyber Monday.
  - + Tháng 12: Giáng Sinh và mùa mua sắm năm mới.
- Đây là mùa lễ hội lớn nhất và quan trọng nhất, thúc đẩy nhu cầu tiêu dùng mạnh mẽ, tạo ra đỉnh doanh thu trong năm.

### Rút ra nhận xét về Hành vi mua sắm của người tiêu dùng qua việc phân tích doanh thu theo quý:

Hành vi mua sắm của người tiêu dùng có xu hướng:

- Tăng cao vào Quý 4 nhờ lễ hội lớn.
- Giảm ở Quý 1 và Quý 2, thể hiện sự tiết chế chi tiêu sau mùa lễ hội.
- Tăng nhẹ vào Quý 3 nhờ mùa tựu trường và chuẩn bị cho cuối năm.

- Bị ảnh hưởng mạnh mẽ bởi các dịp lễ hội và chiến lược khuyến mãi, cho thấy sự cần thiết phải tập trung vào các giai đoạn này để tối ưu hóa doanh thu.

### Ứng dụng từ biểu đồ để đưa ra đề xuất:

- Doanh thu tăng mạnh vào Quý 4 cho thấy các doanh nghiệp cần tập trung nguồn lực và chiến dịch marketing vào thời điểm này để tối đa hóa lợi nhuận
- Quý 3 cũng là một quý quan trọng nhờ mùa tựu trường và các chương trình khuyến mãi chuẩn bị cho mùa lễ hội cuối năm
- Các quý Q1 và Q2 tuy có doanh thu thấp hơn nhưng vẫn cần các chương trình kích cầu, chẳng hạn như các sự kiện đầu xuân hoặc khuyến mãi giảm giá để duy trì doanh thu

→ Doanh nghiệp cần xây dựng các chiến lược bán hàng và marketing phù hợp theo từng quý, tập trung vào việc tối đa hóa doanh thu trong các mùa lễ hội và lên kế hoạch chuẩn bị cho các giai đoạn nhu cầu thấp hơn.

#### d. Biểu đồ so sánh doanh thu giữa các tháng có lễ hội và tháng không có lễ hội:

##### Mục tiêu:

- Xác định mức độ ảnh hưởng của các mùa lễ hội đến doanh thu bán hàng.
- So sánh sự khác biệt về doanh thu giữa các tháng có lễ hội và không có lễ hội để đánh giá tác động của yếu tố thời gian và sự kiện đặc biệt.
- Phát hiện các xu hướng mua sắm tăng mạnh vào các dịp lễ hội, từ đó khẳng định vai trò quan trọng của các sự kiện này đối với doanh số.
- Hỗ trợ doanh nghiệp lên kế hoạch và tối ưu hóa chiến lược tiếp thị, khuyến mãi trong các tháng lễ hội để tối đa hóa doanh thu.

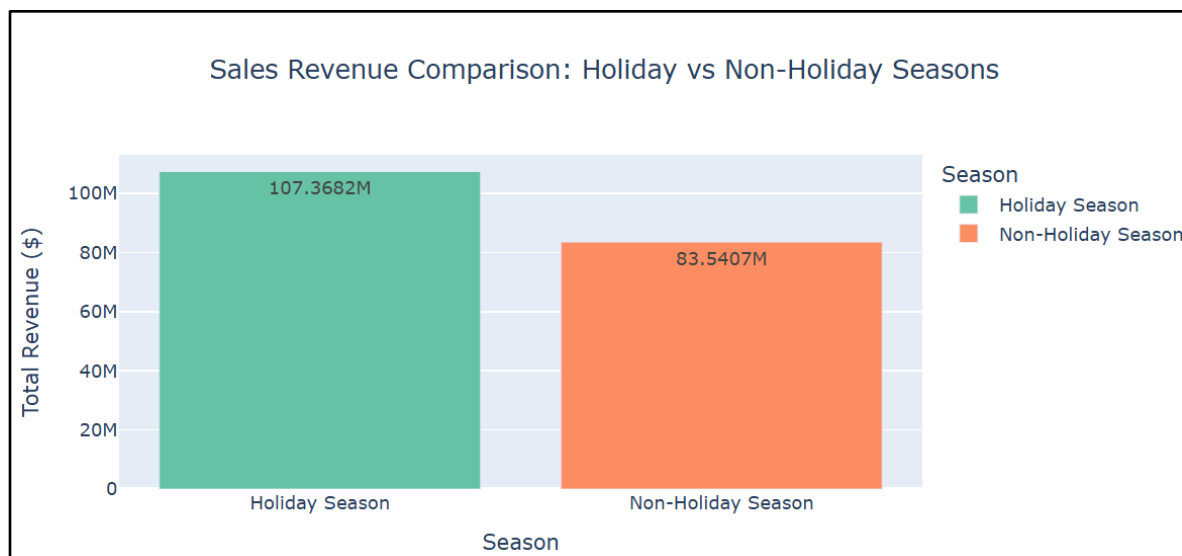
```
# 4. So sánh doanh thu giữa các tháng có lễ hội (tháng: 2, 5, 7, 8, 10, 11, 12) và không lễ hội
# Tạo cột 'Holiday Season' để phân loại
data['Holiday Season'] = data['Purchase Date'].dt.month.isin([2, 5, 7, 8, 10, 11, 12]).map({True: 'Holiday Season', False: 'Non-Holiday Season'})

# Tính tổng doanh thu và số lượng giao dịch cho mỗi mùa
seasonal_sales = data.groupby(['Holiday Season']).agg(
    Total_Revenue=('Total Price', 'sum'),
    Avg_Revenue=('Total Price', 'mean'),
    Count=('Total Price', 'size')
).reset_index()

# Trực quan hóa doanh thu giữa mùa Lễ hội và không Lễ hội
fig = px.bar(seasonal_sales,
             x='Holiday Season',
             y='Total_Revenue',
             color='Holiday Season',
             title='Sales Revenue Comparison: Holiday vs Non-Holiday Seasons',
             labels={'Total_Revenue': 'Total Revenue ($)', 'Holiday Season': 'Season'},
             text_auto=True,
             color_discrete_sequence=px.colors.qualitative.Set2)

fig.update_layout(title={'x': 0.5}, width=800, height=400)
fig.show()
```

## Output:



### → Nhận xét:

1. Doanh thu trong mùa lễ hội cao hơn đáng kể:
  - Các tháng có lễ hội (tháng 2, 5, 7, 8, 10, 11, 12) thường có tổng doanh thu lớn hơn nhiều so với các tháng không có lễ hội
  - Điều này phản ánh rõ xu hướng chi tiêu mạnh mẽ của khách hàng trong các dịp lễ hội, khi nhu cầu mua sắm quà tặng, đồ trang trí và nhu yếu phẩm tăng mạnh
2. Doanh thu trong các tháng không lễ hội thấp hơn:
  - Các tháng không có lễ hội thường có doanh thu thấp hơn rõ rệt, phản ánh hành vi mua sắm tiết chế của người tiêu dùng trong những giai đoạn thấp điểm
  - Người tiêu dùng ít có động lực chi tiêu lớn nếu không có dịp đặc biệt hay khuyến mãi thúc đẩy
3. Kết luận: Sự khác biệt rõ rệt giữa mùa lễ hội và không lễ hội cho thấy rằng:
  - Hành vi mua sắm của khách hàng chịu ảnh hưởng mạnh bởi các dịp lễ hội và các sự kiện đặc biệt trong năm
  - Các nhà bán lẻ và nền tảng thương mại điện tử cần tập trung chiến lược marketing và khuyến mãi vào các mùa lễ hội để tối đa hóa doanh thu

### → Đưa ra đề xuất cho doanh nghiệp:

- Tập trung vào các chiến dịch khuyến mãi trong mùa lễ hội lớn trong năm như: Black Friday, Cyber Monday, Back-to-School và Giáng Sinh

- Chuẩn bị hàng tồn kho và logistics từ trước các mùa cao điểm để đáp ứng nhu cầu tăng vọt của người tiêu dùng
- Xây dựng chiến lược quảng cáo phù hợp cho từng mùa lễ hội, nhấn mạnh vào các mặt hàng như quà tặng, trang trí, thực phẩm và đồ dùng học tập

e. Tính giá trị trung bình và độ lệch chuẩn theo các khoảng thời gian:

**Mục tiêu:**

- Đánh giá xu hướng chung: Xác định mức doanh thu trung bình và số lượng sản phẩm mua trong từng tháng, quý hoặc mùa để nắm bắt xu hướng mua sắm theo thời gian.
- Phân tích mức độ biến động: Sử dụng độ lệch chuẩn để đo lường sự dao động của doanh thu và số lượng sản phẩm, từ đó nhận diện các giai đoạn có biến động lớn.
- Xác định các giai đoạn đặc biệt: Phát hiện các thời điểm có doanh thu hoặc số lượng sản phẩm tăng/giảm bất thường, đặc biệt trong các tháng lễ hội.
- Hỗ trợ ra quyết định chiến lược: Cung cấp cơ sở dữ liệu giúp doanh nghiệp lên kế hoạch tiếp thị và quản lý hàng tồn kho hiệu quả hơn trong các mùa cao điểm hoặc thấp điểm.

```
# 5. Tính giá trị trung bình và độ lệch chuẩn theo các khoảng thời gian:
# Tính theo tháng
monthly_stats = data.groupby('Purchase Month').agg(
    Avg_Product_Price=('Product Price', 'mean'),
    Std_Product_Price=('Product Price', 'std'),
    Avg_Quantity=('Quantity', 'mean'),
    Std_Quantity=('Quantity', 'std')
).reset_index()

# Tính theo quý
quarterly_stats = data.groupby('Purchase Quarter').agg(
    Avg_Product_Price=('Product Price', 'mean'),
    Std_Product_Price=('Product Price', 'std'),
    Avg_Quantity=('Quantity', 'mean'),
    Std_Quantity=('Quantity', 'std')
).reset_index()

# Tính theo mùa
seasonal_stats = data.groupby('Season').agg(
    Avg_Product_Price=('Product Price', 'mean'),
    Std_Product_Price=('Product Price', 'std'),
    Avg_Quantity=('Quantity', 'mean'),
    Std_Quantity=('Quantity', 'std')
).reset_index()

# In kết quả để kiểm tra
print("Kết quả tính theo tháng: ")
print(monthly_stats)
print("\nKết quả tính theo quý: ")
print(quarterly_stats)
print("\nKết quả tính theo mùa: ")
print(seasonal_stats)
```

**Output:**

Kết quả tính theo tháng:					
	Purchase Month	Avg_Product_Price	Std_Product_Price	Avg_Quantity	\
0	April	256.130017	141.084850	2.996508	
1	August	254.216683	141.542512	2.994893	
2	December	255.693123	141.524581	3.008428	
3	February	254.652070	141.547956	2.992765	
4	January	254.307220	142.191424	3.016255	
5	July	254.461914	141.510323	2.998606	
6	June	254.397026	142.405744	2.999503	
7	March	254.047864	141.504954	2.991950	
8	May	254.964506	141.463769	2.998339	
9	November	255.529415	141.672035	3.009387	
10	October	253.321404	140.786241	2.989549	
11	September	254.353615	141.399693	2.991880	
Std_Quantity					
0		1.419730			
1		1.411130			
2		1.414851			
3		1.409644			
4		1.415220			
5		1.420025			
6		1.413974			
7		1.415781			
8		1.412635			
9		1.415598			
10		1.418030			
11		1.409840			
Kết quả tính theo quý:					
	Purchase Quarter	Avg_Product_Price	Std_Product_Price	Avg_Quantity	\
0	1	254.325929	141.750955	3.000493	
1	2	255.160582	141.650722	2.998121	
2	3	254.343599	141.486866	2.995313	
3	4	254.841889	141.326652	3.002395	
Std_Quantity					
0		1.413690			
1		1.415393			
2		1.413874			
3		1.416168			
Kết quả tính theo mùa:					
	Season	Avg_Product_Price	Std_Product_Price	Avg_Quantity	Std_Quantity
0	Fall	254.391053	141.288092	2.996646	1.414303
1	Spring	255.030940	141.355543	2.995576	1.415988
2	Summer	254.358155	141.811113	2.997646	1.415044
3	Winter	254.814836	141.781646	3.005959	1.413208

→ **Nhận xét:** Phân tích giá trị trung bình và độ lệch chuẩn theo các khoảng thời gian cho thấy:

- Hành vi mua sắm tăng mạnh vào các tháng/quý có lễ hội lớn (tháng 11, 12; Quý 4), với giá trung bình và số lượng mua sắm cao hơn.

- Độ lệch chuẩn cao hơn trong các mùa lễ hội, cho thấy sự đa dạng trong hành vi mua sắm, từ khách hàng bình dân đến cao cấp.
- Các giai đoạn thấp điểm (Quý 1, Quý 2, mùa hè) thường có hành vi mua sắm ổn định hơn, nhưng không đạt mức đỉnh điểm như mùa lễ hội.

**Rút ra nhận xét về xu hướng mua sắm trực tuyến của người tiêu dùng qua việc Tính giá trị trung bình và độ lệch chuẩn theo các khoảng thời gian:**

**1. Xu hướng mua sắm thay đổi theo thời gian:**

- Giá trung bình sản phẩm:
  - + Tăng cao trong các mùa lễ hội: Đặc biệt là vào các tháng thuộc Quý 4 (tháng 10-12), khi người tiêu dùng chi tiêu mạnh cho các sản phẩm quà tặng, thời trang, và đồ gia dụng. Điều này phản ánh nhu cầu tiêu thụ các sản phẩm cao cấp và phục vụ lễ hội.
  - + Giảm trong giai đoạn thấp điểm: Giá trung bình thấp hơn trong Quý 1 và Quý 2 (tháng 1-6), khi khách hàng ưu tiên mua sắm các sản phẩm thiết yếu hoặc hàng hóa giá cả phải chăng.

**2. Biến động trong hành vi mua sắm giữa các nhóm khách hàng:**

- Độ lệch chuẩn cao trong mùa lễ hội (Quý 4):
  - + Hành vi mua sắm trở nên phân hóa mạnh mẽ hơn trong các mùa lễ hội. Một số khách hàng tập trung mua sắm các sản phẩm đắt tiền hoặc mua với số lượng lớn (quà tặng, đồ trang trí), trong khi nhóm khác mua ít hơn hoặc chọn các sản phẩm giá trị thấp.
- Độ lệch chuẩn thấp trong các giai đoạn không lễ hội (Quý 1 & Quý 2):
  - + Hành vi mua sắm đồng đều hơn, với ít chênh lệch giữa các nhóm khách hàng. Điều này phản ánh nhu cầu ổn định đối với các sản phẩm thiết yếu hoặc phục vụ sinh hoạt hàng ngày.

**3. Đặc điểm mua sắm theo từng thời điểm cụ thể:**

- Tháng cao điểm:
  - + Các tháng lễ hội (tháng 11-12, tháng 2, tháng 5) có mức chi tiêu cao và sự phân hóa lớn trong giá sản phẩm, cho thấy tầm quan trọng của các dịp lễ lớn (Black Friday, Giáng Sinh, Valentine, Mother's Day).
- Mùa hè (Quý 3):
  - + Xu hướng chi tiêu phục hồi với sự gia tăng nhẹ về giá trung bình và số lượng mua sắm, nhờ các sự kiện như Back-to-School và Father's Day.

**4. Nhận xét từ số lượng mua trung bình và độ lệch chuẩn:**

- Số lượng mua trung bình tăng trong mùa lễ hội:
  - + Người tiêu dùng thường mua nhiều sản phẩm hơn vào các dịp lễ, không chỉ để sử dụng cá nhân mà còn làm quà tặng.
- Độ lệch chuẩn cao trong mùa lễ:
  - + Chênh lệch lớn về số lượng mua sắm giữa các nhóm khách hàng, từ những người mua ít đến những khách hàng mua sắm số lượng lớn cho gia đình hoặc mục đích kinh doanh.

→ **Đưa ra đề xuất:**

- Tối ưu hóa chiến lược kinh doanh:
  - + Tập trung vào sản phẩm cao cấp và đa dạng trong mùa lễ hội để đáp ứng nhu cầu phong phú của khách hàng.
  - + Tạo ra các chương trình khuyến mãi phù hợp vào các giai đoạn thấp điểm nhằm kích thích mua sắm.
- Lên kế hoạch chuỗi cung ứng:
  - + Dự đoán chính xác hơn nhu cầu sản phẩm theo từng giai đoạn thời gian để quản lý tồn kho hiệu quả.
- Phân khúc khách hàng:
  - + Nhắm mục tiêu các nhóm khách hàng với chiến lược phù hợp, ví dụ: các sản phẩm cao cấp trong mùa lễ hội và sản phẩm giá rẻ trong các giai đoạn khác.

### 3.4 Môi trường quan giữa các biến

#### 3.4.1 Tính toán hệ số tương quan ma trận

```
# Tính hệ số tương quan Pearson giữa các biến định lượng:

import seaborn as sns
import matplotlib.pyplot as plt

# Chọn các biến liên tục
continuous_vars = ['Product Price', 'Quantity', 'Customer Age']

# Tính ma trận tương quan
correlation_matrix = data[continuous_vars].corr()

# In kết quả ma trận tương quan
print(correlation_matrix)
```

**Output:**

	Product Price	Quantity	Total Spending	Customer Age
Product Price	1.000000	-0.000308	0.716939	-0.003860
Quantity	-0.000308	1.000000	0.609182	0.000041
Customer Age	-0.003860	0.000041	-0.002005	1.000000

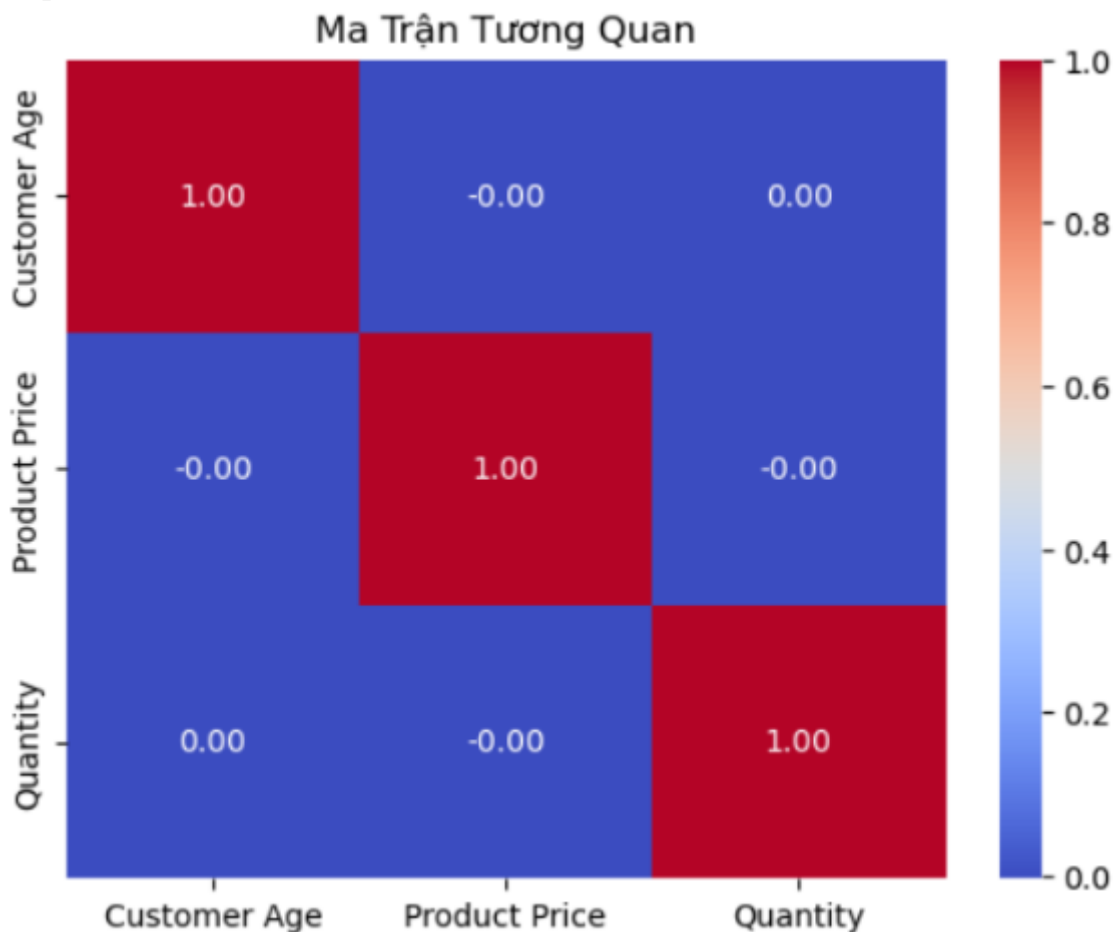
### 3.4.2 Phân tích các biến liên tục

```
#1. Ptich tuong quan PEARSON các biến ltuc
import seaborn as sns
import matplotlib.pyplot as plt

# Chọn các biến liên tục
continuous_cols = ['Customer Age', 'Product Price', 'Quantity']
correlation_matrix = df[continuous_cols].corr()

# Trục quan hóa ma trận tương quan
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f')
plt.title('Ma Trận Tương Quan')
plt.show()
```

Output:



→ Dựa trên giá trị tương quan ( $-0.5 < r < 0.5$ ), biểu đồ cho thấy:

- Tuổi khách hàng (**Customer Age**): Không có mối quan hệ đáng kể với các biến khác (hệ số tương quan gần bằng 0). Điều này có nghĩa là tuổi của khách hàng không ảnh hưởng đến giá sản phẩm họ mua, số lượng họ mua, và cũng không ảnh hưởng đến tổng chi tiêu.  
→ Không đóng vai trò đáng kể trong việc dự đoán tổng chi tiêu.



- Giá sản phẩm (**Product Price**) và Số lượng sản phẩm (**Quantity**): Có tương quan dương khá mạnh với Tổng chi tiêu (**Total Spending**). Điều này có nghĩa là khi giá sản phẩm tăng hoặc số lượng sản phẩm mua tăng, tổng chi tiêu của khách hàng cũng có xu hướng tăng.

### **III. Kết luận & Khuyến nghị**

#### **1. Kết luận**

##### **Hành vi mua sắm theo độ tuổi và giới tính:**

- Các nhóm khách hàng trẻ (18-29 tuổi) chiếm tỷ lệ giao dịch cao nhất, trong đó số lượng giao dịch của nam và nữ gần như tương đương.
- Nữ giới có xu hướng mua nhiều hơn trong các danh mục như quần áo và sách, trong khi nam giới tập trung vào điện tử và đồ gia dụng.
- Độ tuổi từ 30-49 đóng góp lớn vào doanh thu các danh mục giá trị cao, đặc biệt là đồ gia dụng.

##### **Tần suất mua sắm theo thời gian:**

- Doanh thu cao nhất ghi nhận vào tháng 1, tháng 3 và tháng 12, tương ứng với các mùa khuyến mãi lớn (đầu năm và cuối năm).
- Doanh thu giảm trong các tháng ngắn như tháng 2 hoặc các giai đoạn không có sự kiện lớn. Tuy nhiên, mức giảm này không đáng kể vì doanh thu trung bình hàng tháng vẫn ổn định ở mức cao (~5.5M USD).
- Quý 4 (tháng 10-12) ghi nhận sự sụt giảm nhẹ do doanh số giảm trong tháng 10, nhưng được bù đắp bởi Black Friday và mùa lễ hội tháng 12.

##### **Xu hướng mua sắm theo mùa, tháng và quý:**

- Sách là danh mục phổ biến nhất trong cả năm, chiếm doanh thu cao nhất (~19M USD).
- Quần áo dẫn đầu về số lượng giao dịch và doanh thu trong mùa thu đông (tháng 9-12), phù hợp với nhu cầu tăng cao dịp lễ hội.
- Điện tử có doanh thu cao hơn vào mùa hè (quý 2), cho thấy khách hàng có xu hướng đầu tư vào các sản phẩm công nghệ trong các sự kiện như "Back-to-School" hoặc khuyến mãi giữa năm.

#### **2. Khuyến nghị kinh doanh & tiếp thị**

##### **Chiến lược theo độ tuổi và giới tính:**

- Phát triển các chiến dịch tiếp thị nhắm đến từng nhóm khách hàng cụ thể:
  - + Khách hàng trẻ (18-29 tuổi): Quảng bá các sản phẩm công nghệ và thời trang theo xu hướng.
  - + Khách hàng trung niên (30-49 tuổi): Tăng cường các chương trình giảm giá cho đồ gia dụng và quần áo chất lượng cao.
- Thực hiện các chương trình khuyến mãi riêng biệt theo giới tính:
  - + Nữ giới: Tập trung vào các sản phẩm quần áo và sách kèm các gói quà tặng cho lễ hội.

- + Nam giới: Khuyến mãi các sản phẩm điện tử và đồ gia dụng vào dịp hè và ngày lễ công nghệ.

### **Tăng cường tần suất mua sắm:**

- Ra mắt các chương trình khách hàng thân thiết, chẳng hạn tích điểm đổi thưởng hoặc giảm giá cho các giao dịch lặp lại trong một tháng.
- Tận dụng ngày lễ nhỏ như Valentine, Lễ Độc Thân (11/11) để kích cầu tiêu dùng.

### **Tối ưu hóa danh mục sản phẩm theo mùa:**

- Lên kế hoạch chuẩn bị tồn kho sớm cho các danh mục bán chạy:
  - + Sách: Đảm bảo tồn kho cao vào dịp năm học mới hoặc mùa lễ cuối năm.
  - + Quần áo: Triển khai trước các chiến dịch khuyến mãi vào cuối quý 3 để đẩy mạnh doanh số cho quý 4.
  - + Điện tử: Đầu tư vào các sự kiện giảm giá như Black Friday hoặc mùa hè.
- Phát triển các sản phẩm kết hợp (bundles) để tăng giá trị đơn hàng, chẳng hạn:
  - + Giảm giá khi mua bộ quần áo cùng phụ kiện.
  - + Tặng thêm sách khi mua các sản phẩm theo chủ đề nhất định.

### **Theo dõi và cải thiện hiệu suất:**

- Xây dựng các bảng đo lường KPI theo mùa, tập trung vào các chỉ số như doanh thu, tỷ lệ mua lại, và phản hồi khách hàng.
- Thực hiện khảo sát và thu thập phản hồi thường xuyên để cải thiện chất lượng sản phẩm và dịch vụ, đặc biệt là trong các danh mục điện tử và đồ gia dụng.