**UNIVERSIDADE PRESBITERIANA MACKENZIE**

**TECNÓLOGO EM CIÊNCIAS DE DADOS**

**PARTICIPANTES DO GRUPO**

Adrieli Machado Zaluski

Caroline Ribeiro Ferreira

Lais César Fonseca

Liliane Gonçalves de Brito Ferraz

Mucio Emanuel Feitosa Ferraz Filho

Otavio Bernardo Scandiuzzi

**TENDÊNCIAS DE COMPRAS EM SHOPPINGS DE ISTAMBUL**

**SÃO PAULO**

**2023**

Sumário

[1. GLOSSÁRIO 3](#_Toc136181220)

[2. INTRODUÇÃO 5](#_Toc136181221)

[3. COMPOSIÇÃO DO GRUPO 6](#_Toc136181222)

[4. CRONOGRAMA DE DESENVOLVIMENTO DO PROJETO 6](#_Toc136181223)

[5. INFORMAÇÕES SOBRE O DIRETÓRIO 8](#_Toc136181224)

[6. APRESENTAÇÃO DO PROJETO 11](#_Toc136181225)

[7. APRESENTAÇÃO DA EMPRESA 12](#_Toc136181226)

[8. PROBLEMA DO ESTUDO 16](#_Toc136181227)

[9. METADADOS 17](#_Toc136181228)

[10. TRATAMENTO DOS DADOS 21](#_Toc136181229)

[11. ANÁLISE EXPLORATÓRIA DE DADOS 23](#_Toc136181230)

[12. CONCLUSÃO 47](#_Toc136181231)

[13. REFERÊNCIAS 49](#_Toc136181232)

[14. LINK VÍDEO 49](#_Toc136181233)

# GLOSSÁRIO

* **Colaboratory:** Conhecido também como “Colab”, é um produto do Google Research, área de pesquisas científicas do Google. O Colab permite que qualquer pessoa escreva e execute código Python arbitrário pelo navegador e é especialmente adequado para aprendizado de máquina, análise de dados e educação.
* **DataFrame:** É uma estrutura de dados bidimensional com os dados alinhados de forma tabular em linhas e colunas.
* **Datasets:** conjuntos de dados organizados em um formato similar ao das tabelas, com linhas e colunas que contém informações sobre determinado tema.
* **GitHub:** GitHub é uma plataforma de hospedagem de código-fonte e arquivos com controle de versão usando o Git. Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma contribuam em projetos privados e/ou Open Source de qualquer lugar do mundo.
* **Kaggle:** É uma plataforma para aprendizado de ciência de dados. É também uma comunidade, a maior da internet, para assuntos relacionados com Data Science.
* **NumPy:** É uma biblioteca para a linguagem de programação Python, que suporta o processamento de grandes, multi-dimensionais arranjos e matrizes, juntamente com uma grande coleção de funções matemáticas de alto nível para operar sobre estas matrizes
* **Matplotlib:** É uma biblioteca de softwares para criação de gráficos e visualizações de dados em geral, feita para e da linguagem de programação Python e sua extensão de matemática NumPy.
* **Pandas:** É uma biblioteca de software criada para a linguagem Python para manipulação e análise de dados. Em particular, oferece estruturas e operações para manipular tabelas numéricas e séries temporais.
* **Python:** É uma linguagem de programação de alto nível, interpretada de script, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte. Foi lançada por Guido van Rossum em 1991.
* **Readme:** É um arquivo com extensão.md, ou seja, ele é escrito em Markdown que é uma linguagem de marcação utilizada para converter o texto em um HTML válido.
* **Seaborn:** É uma biblioteca Python de visualização de dados amplamente popular, comumente usada para tarefas de ciência de dados e aprendizado de máquina.
* **Storytelling:** é a habilidade de contar histórias utilizando enredo elaborado, narrativa envolvente, e recursos audiovisuais. A técnica, cujo caráter é persuasivo, ajuda a promover o seu negócio e a vender seus serviços de forma indireta. Pode ser aplicada na produção de conteúdo, em vendas e em consultorias.
* **String:** Sequências de caracteres alfanuméricos (letras, números e/ou símbolos) amplamente usadas em programação.

# INTRODUÇÃO

O propósito deste projeto é realizar um estudo de caso prático em uma base de dados pública, disponibilizada no site Kaggle¹. Os dados escolhidos para este estudo referem-se a compras em shoppings na cidade de Istambul, na Turquia.

Esses dados serão utilizados para identificar padrões no comportamento de compras e prever tendências futuras de consumo. Para este estudo de ciências de dados será utilizado técnicas adquiridas nos componentes curriculares de introdução a ciência de dados, pensamento computacional e análise exploratória de dados, com o objetivo principal de analisar e interpretar o que esses dados podem mostrar através de aplicação de medidas estatísticas, no qual discutiremos e responderemos algumas perguntas ao longo do projeto, permitindo uma compreensão mais profunda do comportamento dos consumidores em Istambul.

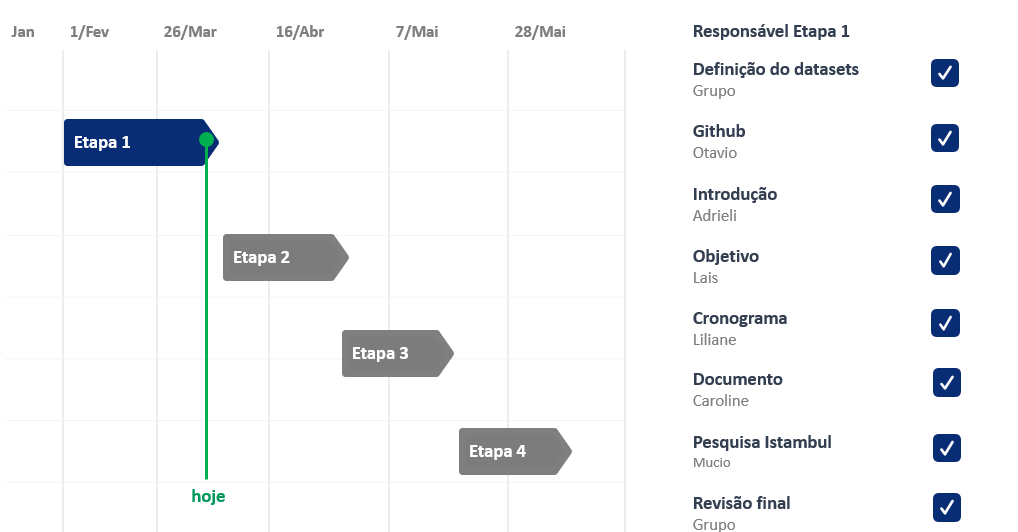
Neste projeto apresentaremos os principais comandos utilizados para realizar as análises dos dados através da aplicação da linguagem Python, com o uso do Colab. Como acessar o metadado, consultar quantidade de atributos, linhas, resumo do dataset, verificar os tipos dos atributos e como importar as bibliotecas necessárias para realizar a exploração de dados de forma mais simplificada.

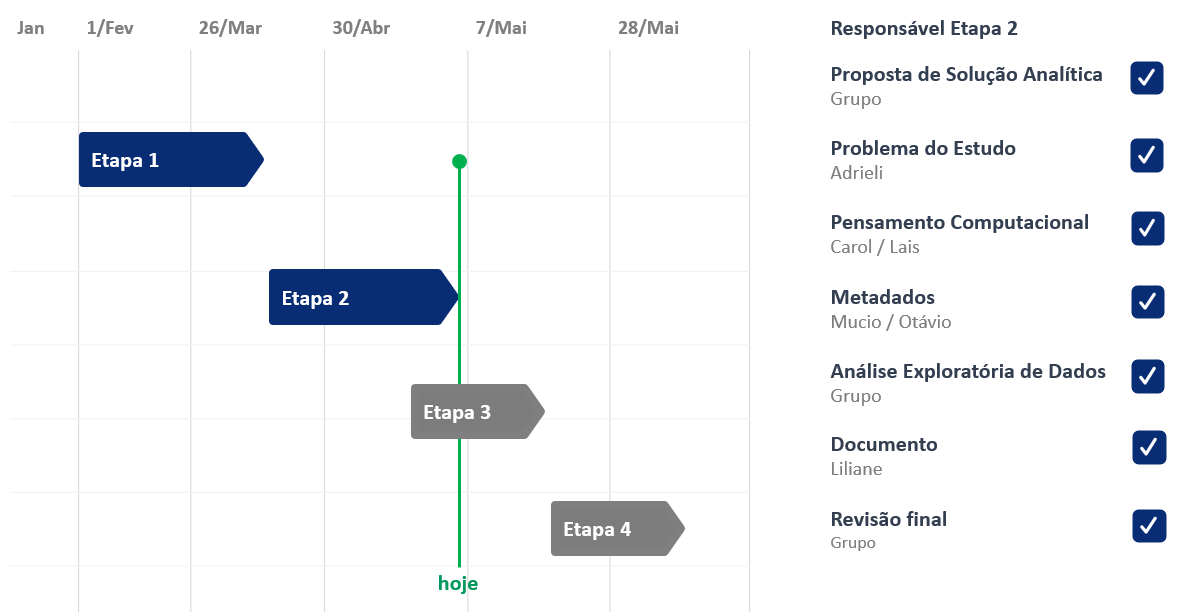
# COMPOSIÇÃO DO GRUPO

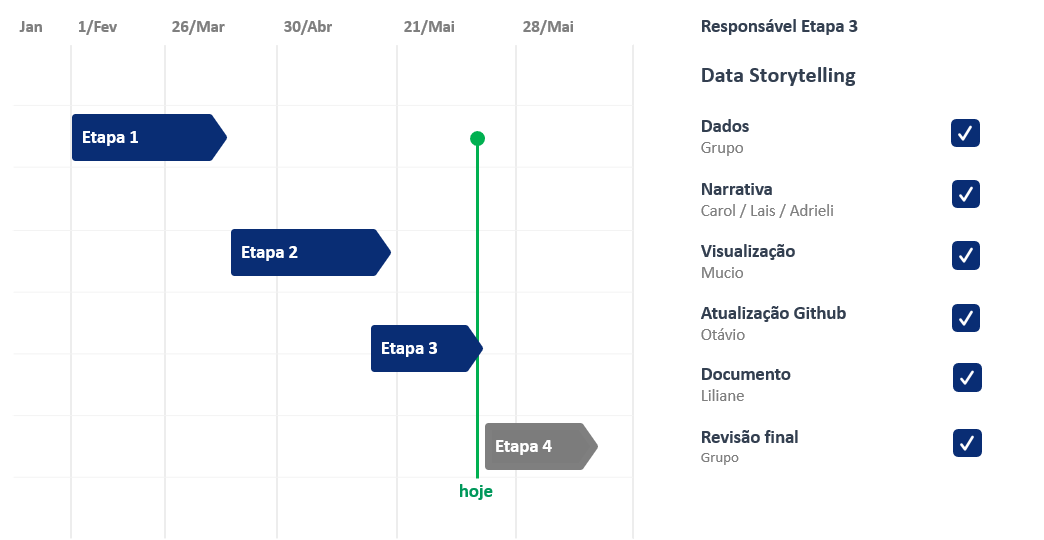
|  |  |
| --- | --- |
| **Integrantes** | **Nº de matrícula** |
| Adrieli Machado Zaluski | 22503668 |
| Caroline Ribeiro Ferreira | 22514635 |
| Lais César Fonseca | 22500790 |
| Liliane Gonçalves de Brito Ferraz | 22501142 |
| Mucio Emanuel Feitosa Ferraz Filho | 22515925 |
| Otavio Bernardo Scandiuzzi | 22511921 |

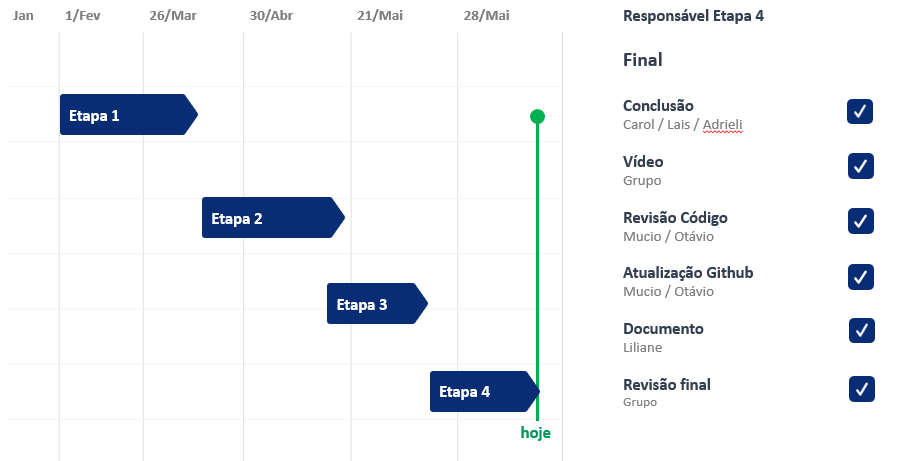
# CRONOGRAMA DE DESENVOLVIMENTO DO PROJETO

Reportar-se o percentual de evolução de entregas referente as ações propostas pelo componente curricular de Projeto Aplicado I do curso de Tecnologia em Ciências de Dados.









# INFORMAÇÕES SOBRE O DIRETÓRIO

Todo o conteúdo do projeto estará disponível no site da GitHub, que poderá ser acessado pelo link:

<https://github.com/OtavioBer/ProjetoAplicadoI>

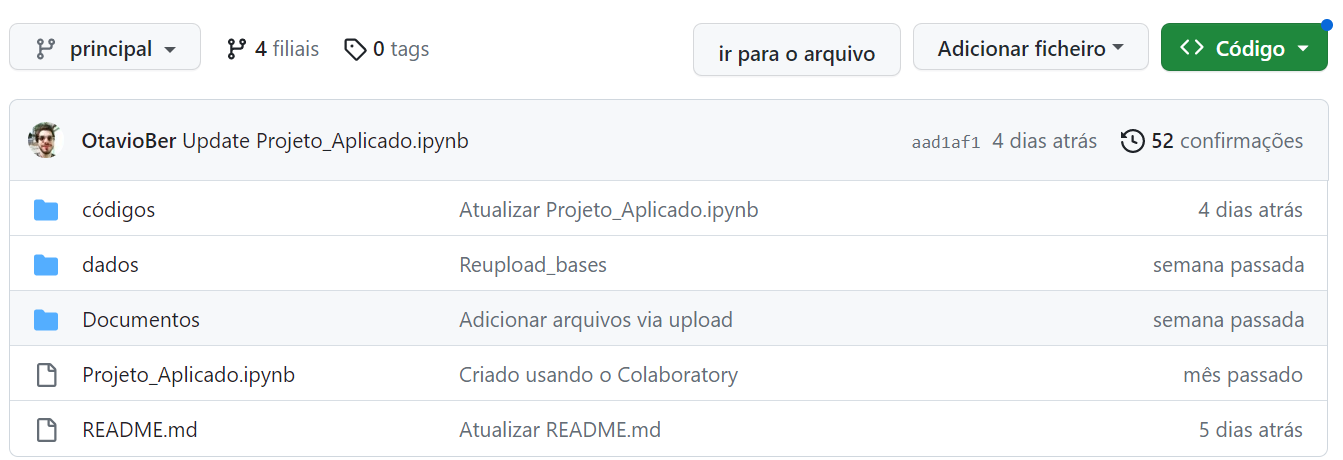
O diretório está organizado por pastas e subpastas.

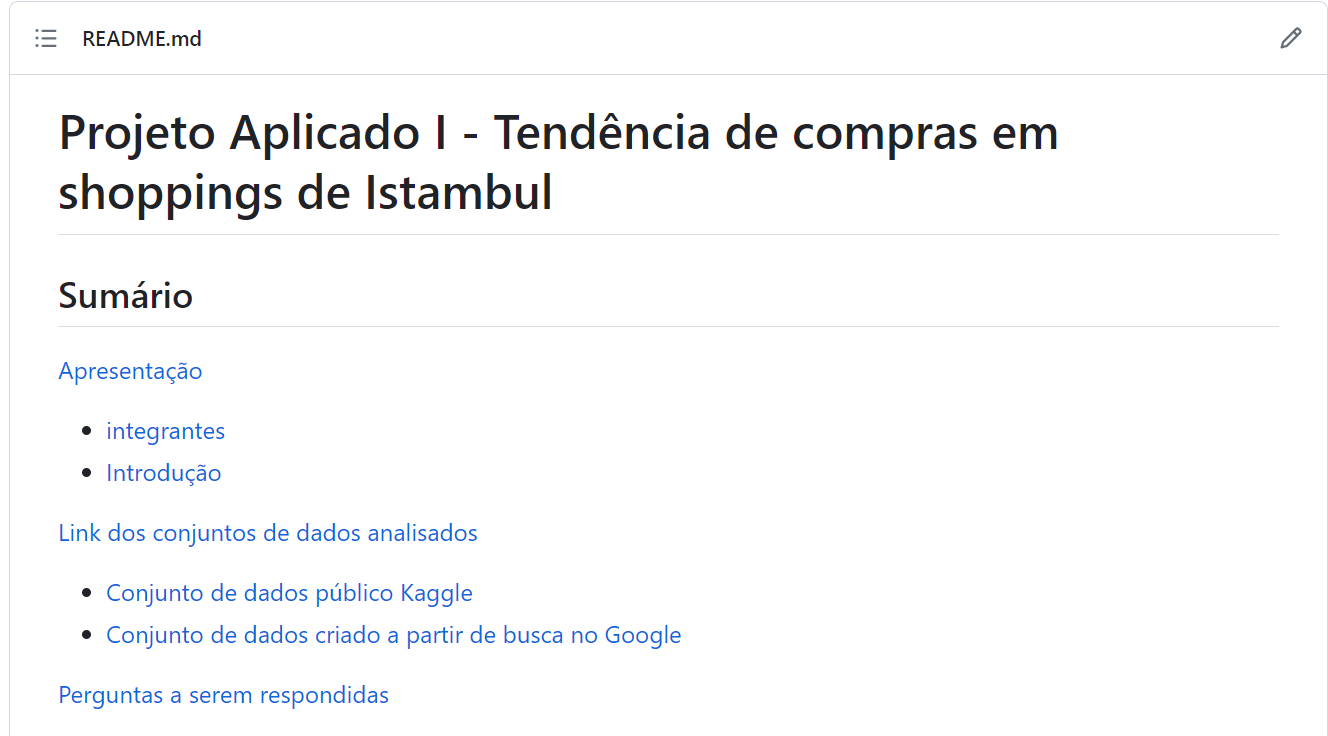
Na pasta “Códigos” será disponibilizado os códigos em Python de cada resposta.

Na pasta “Documentos” temos o cronograma de entregas do projeto, Storytelling e as versões de entrega deste documento.

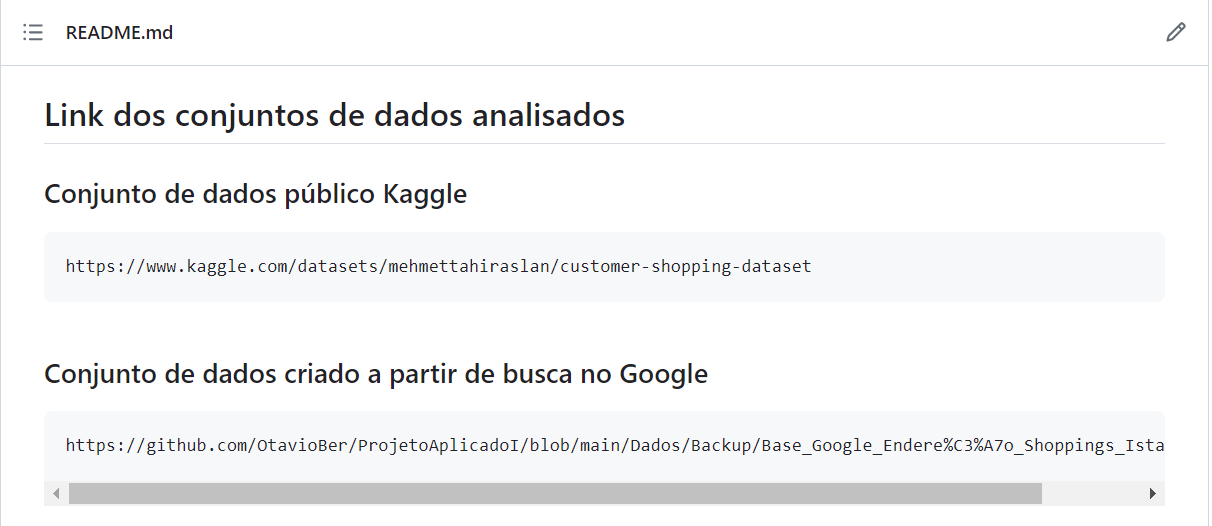
Na pasta “Dados” temos os arquivos utilizados para análise e a pasta de “Backup”, contendo um arquivo original de todos os dados.

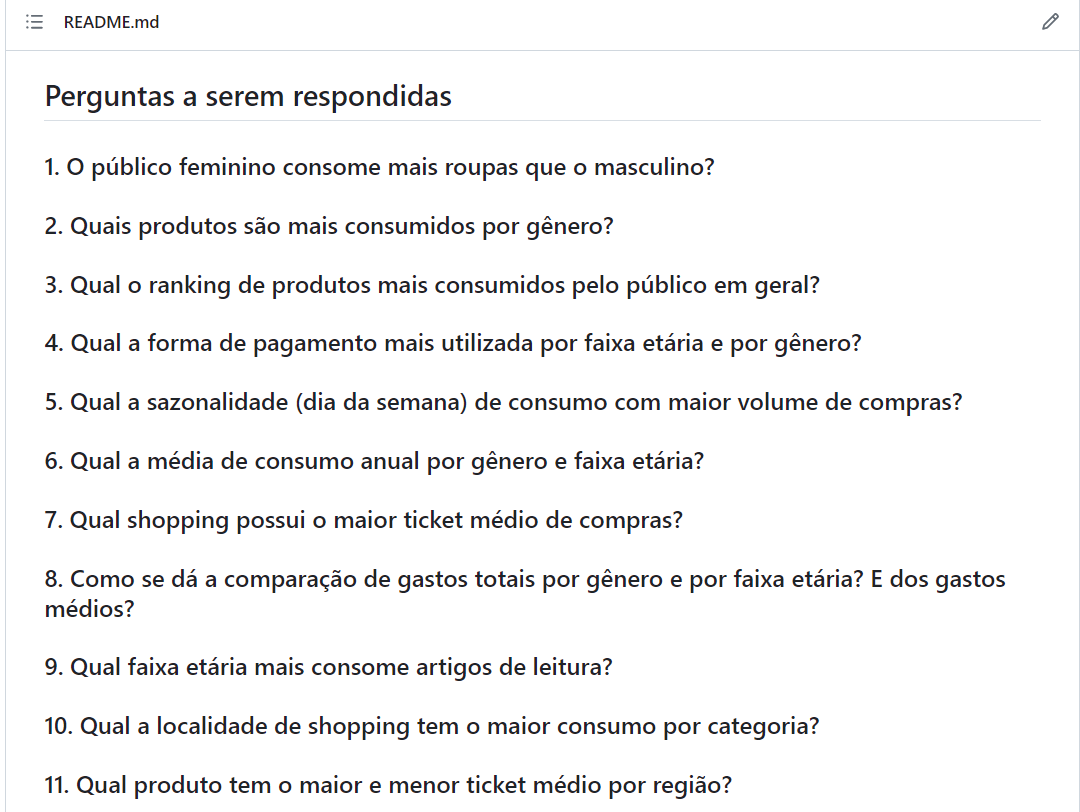
Temos também o arquivo README.md com algumas informações do projeto, que ao decorrer do projeto, poderá ocorrer alterações nas estruturas de pastas, arquivos e no ReadMe.











# APRESENTAÇÃO DO PROJETO

Para o desenvolvimento do projeto teremos como base de estudo os dados públicos disponibilizados na plataforma de datasets Kaggle.

O dataset escolhido nos fornece conjunto de dados que contém informações de compras de 10 shoppings diferentes no período de janeiro/2021 a março/2023 na cidade de Istambul, na Turquia. O conjunto de dados inclui informações essenciais para a realização das análises propostas, como números de faturas, IDs de clientes, faixa etária, gênero, métodos de pagamento, categorias de produtos, quantidade, preço, datas de pedidos e o nome dos shoppings centers.

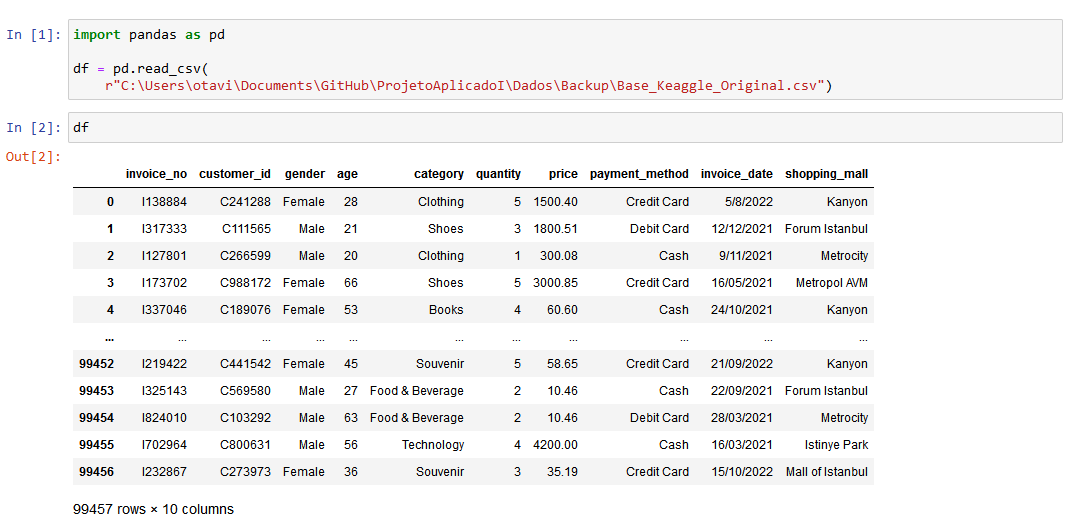
Com posse dos dados, nosso objetivo é, por meio da exploração dos dados, conseguir identificar tendências de compras, padrões, e entender melhor o comportamento de consumo do varejo em Istambul.

Com o objetivo de explorar os dados de forma a identificar a regionalização dos shoppings da cidade de Istambul, realizamos uma busca sobre a localização de cada shopping listado na base de dados e criamos um novo dataset em xls com o endereço, distrito e a região. Com este novo dataset poderemos analisar se a localização do shopping interfere nas demais análises que serão avaliadas.

* 1. **Origem Dataset**

O dataset adquirido no site da Kaggle, contém 99.457 linhas e 10 colunas, no link de acesso:

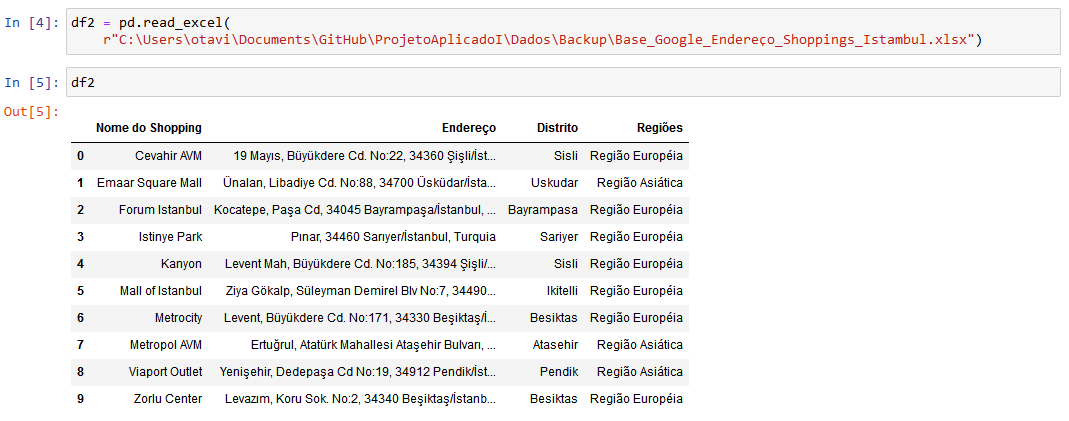
[**https://www.kaggle.com/datasets/mehmettahiraslan/customer-shopping-dataset**](https://www.kaggle.com/datasets/mehmettahiraslan/customer-shopping-dataset)



* 1. **Dados de Apoio**

A base de dados de endereço que foi desenvolvida a partir de uma pesquisa da localização de cada shopping dispõe de 10 linhas e 4 colunas.

<https://github.com/OtavioBer/ProjetoAplicadoI/blob/main/Dados/Backup/Base_Google_Endere%C3%A7o_Shoppings_Istambul.xlsx>



# APRESENTAÇÃO DA EMPRESA

Os dados que serão analisados, são referentes a 10 shopping da cidade em Istambul, na Turquia, onde descreveremos a seguir um breve resumo sobre cada um deles.

Para esta classe de segmento existem semelhanças em seus propósitos, sendo eles:

**Missão:** Proporcionar aos clientes experiências incríveis e momentos mágicos sendo referência em entretenimento, moda, lazer e gastronomia.

**Visão:** Ser referência no mercado de Shoppings Centers, oferecendo marcas de destaque no cenário nacional e internacional, promovendo eventos exclusivos para fomentar a cultura, entretenimento e lazer.

**Valores:** Ética, transparência, compromisso, inovação, eficiência e gestão ambiental.

* 1. **Conhecendo os Shoppings**
* **Kanyon**

Este é um shopping notável dentro de Istambul, isso por que possui uma arquitetura luxuosa, abrigando diversas marcas de grife, além de fitness center, e uma torre de escritórios com 179 residências de luxo. É mais um que se localiza na parte europeia da capital, situando-se no distrito comercial de Levent.

Figura 1 - Cevahir AVM



Fonte: https://www.nenerede.com.tr/ilan/istanbul-cevahir-avm/

* **Cevahir AVM**

O Istanbul Cevahir Shopping foi aberto no ano de 2005 e desde sua abertura até o ano de 2011 ele foi considerado o maior shopping em área bruta da Europa, mas ainda hoje permanece na lista dos maiores do mundo. Este moderno centro comercial turco fica localizado no distrito de Sisli, situado na parte europeia de Istambul.

* **Emaar Square Mall**

O Emaar Square Mall é o maior shopping center da Turquia, possuindo 6 andares em sua estrutura. Foi aberto para o público no ano de 2017 e possui diversas atrações, como um aquário e zoológico subaquático, vista panorâmica e um museu de ilusões. É localizado na parte asiática da cidade, dentro do distrito de Uscudar.

* **Istinye Park**

Este shopping é um dos mais populares de toda a Turquia, sendo conhecido por possuir diversas marcas de luxo ao passo que possui diversas iguarias locais, além de várias opções para lazer e alimentação. Este centro comercial se localiza na parte europeia de Istambul, no distrito de Sariyer.

Figura 2 - Forum Istanbul



Fonte: https://www.tripadvisor.com.br/Attraction\_Review-g293974-d3386223-Reviews-Forum\_Istanbul\_Alisveris\_Merkezi-Istanbul.html

* **Forum Istanbul**

Este é mais um grande shopping da capital turca, empregando mais de 5000 funcionários e abrigando mais de 280 marcas, nacionais e internacionais, algumas das mais conhecidas são: Adidas, Carrefour, Ecco, Lego e Levi’s. Este centro se situa no distrito de Bayrampasa, parte europeia da capital.

* **Mall of Istanbul**

Este é mais um grande e luxuoso shopping da capital turca, possuindo 350 lojas de marcas de luxo, parque de diversões cobertos e cinema. Sua arquitetura é luxuosa, com características que lembram a de um enorme teatro. Sua localização é no distrito de Basaksehir, parte europeia da cidade.

* **Metrocity**

O shopping Metrocity conta com 4 andares em sua estrutura, abrigando uma grande variedade de lanchonetes, restaurantes, cafeterias, além de diversas lojas de marcas nacionais e internacionais. Assim como o Kanyon, fica localizado no distrito comercial de Levent.

* **Zorlu Center**

Este é um centro comercial que apresenta uma ampla variedade de restaurantes e cafés, além de diversas lojas de marcas conhecidas mundialmente. O Zorlu Center possui um ambiente que combina espaço interno e externo, possuindo áreas verdes em sua estrutura. Sua localização é no distrito de Sisli, assim como o Cevahir AVM.

Figura 3 - Metropol AVM



Fonte: https://www.endeksa.com/en/analiz/istanbul/atasehir/ataturk/malls

* **Metropol AVM**

O Metropol Istanbul Shopping Center possui 5000 metros quadrados de área de entretenimento, com playgrounds para as crianças, pátios esportivos e campos abertos, além disso possui cerca de 250 lojas em sua estrutura. Este estabelecimento se localiza no distrito de Atasehir, parte asiática da cidade.

* **Viaport Outlet**

Este é o maior shopping do segmento outlet de toda a Turquia, em sua estrutura estão incluídas áreas verdes, ruas, praças e até um lago, apresentando, assim, uma arquitetura diferente dos shoppings convencionais. Além disso possui cinema, boliche, ampla variedade culinária e diversas lojas. Se situa na parte asiática da capital turca, no distrito de Pendik.

# PROBLEMA DO ESTUDO

O objetivo será gerar alguns insights sobre padrões, tendências de perfil de compras e responder algumas perguntas como:

1. O público feminino consome mais roupas que o masculino?
2. Quais produtos são mais consumidos por gênero?
3. Qual o ranking de produtos mais consumidos pelo público em geral?
4. Qual a forma de pagamento mais utilizada por faixa etária e por gênero?
5. Qual a sazonalidade (dia da semana) de consumo com maior volume de compras?
6. Qual a média de consumo anual por gênero e faixa etária?
7. Qual shopping possui o maior ticket médio de compras?
8. Como se dá a comparação de gastos totais por gênero e por faixa etária? E dos gastos médios?
9. Qual faixa etária mais consome artigos de leitura?
10. Qual a localidade de shopping tem o maior consumo por categoria?
11. Qual produto tem o maior e menor ticket médio por região?
    1. **Pensamento Computacional**

Para iniciar um projeto ou a resolução de um problema, se faz necessário uma avaliação do contexto geral no qual será trabalhado, aplicando métodos importantes para alcançar os objetivos propostos, ter assertividade na entrega, minimizar impactos de erros, realizar a entrega dentro do cronograma definido e na qualidade e custo esperado.

E neste processo existem 4 etapas que subsidiam este momento inicial de um projeto, que são as dimensões do pensamento computacional, sendo elas:

1. **Decomposição:** Separar problemas complexos em problemas menores;
2. **Reconhecimento de padrões:** Analisando problemas menores torna mais simples o processo de identificação de problema semelhantes já resolvidos em outros momentos e/ou projetos;
3. **Abstração:** É um método essencial de ser aplicado, para focar no que é relevante.
4. **Algoritmo:** É o conjunto de instruções bem estruturada dadas ao sistema para que seja alcançando os resultados esperados.

Neste contexto, é importante ressaltar que o pensamento computacional é um processo de resolução de problemas que se faz necessário seguir alguns passos, sendo eles:

1. Formulação de problemas de forma que nos permita usar um computador e outras ferramentas para nos ajudar a resolvê-los.
2. Organização e análise lógica de dados.
3. Representação de dados por meio de abstrações, como modelos e simulações.
4. Automatização de soluções por meio do pensamento algorítmico (uma série de etapas ordenadas).
5. Identificação, análise e implementação de possíveis soluções com o objetivo de alcançar a combinação mais eficiente e efetiva de etapas e recursos.
6. Generalização e transferência deste processo de resolução de problemas para uma grande variedade de problemas.

# METADADOS

* 1. **Tipo de arquivo**

A base de dados adquirida é de extensão csv.

* 1. **Origem dos dados**

Os dados são de domínio público/aberto, do site da Kaggle.

* 1. **Sensibilidade / LGPD**

Ao criar o conjunto de dados, as informações pessoais dos clientes foram anonimizadas para proteger a privacidade, no entanto, esta base de dados não possui dados sensíveis e está de acordo com a Lei Geral de Proteção de Dados Pessoas – LGPD.

* 1. **Validade**

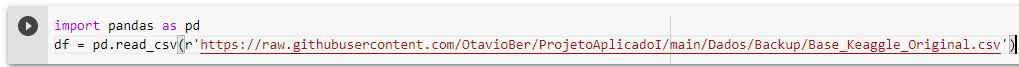
Os dados foram disponibilizados recentemente, março/2023, e está dentro da validade para análise exploratória dos dados e avaliação de comportamento atual de compras.

* 1. **Proprietário do dado**

Mehmet Tahir Aslam, Analista de dados na Crystal System, Istambul, Turquia.

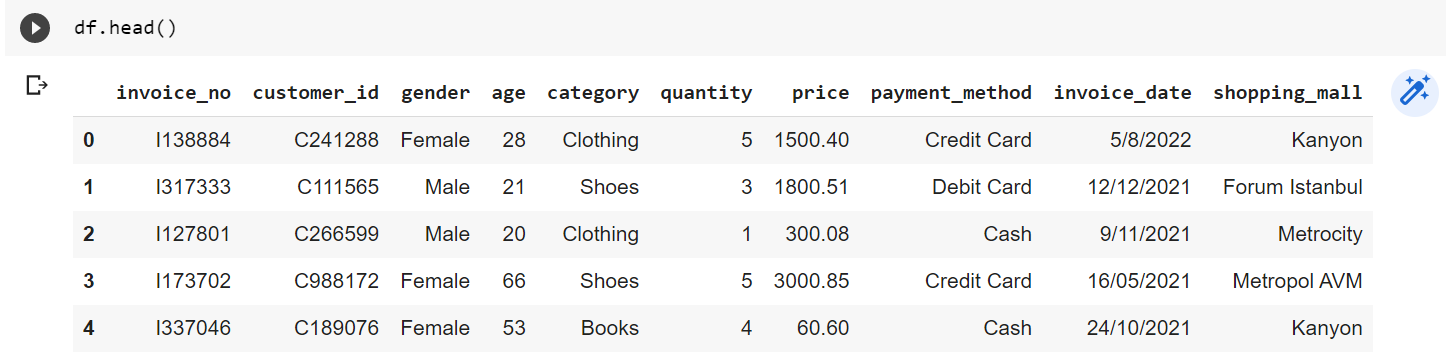
* 1. **Descrição dos atributos dos dados**
* **invoice\_no:** Número da fatura. Nominal. Uma combinação da letra 'I' e um número inteiro de 6 dígitos atribuído exclusivamente a cada operação.
* **customer\_id:** Número do cliente. Nominal. Uma combinação da letra 'C' e um número inteiro de 6 dígitos atribuído exclusivamente a cada operação.
* **gender:** Variável string do sexo do cliente.
* **age:** Variável inteiro positivo da idade do cliente.
* **categoria:** Variável string da categoria do produto adquirido.
* **quantidade:** As quantidades de cada produto (item) por transação. Numérico.
* **preço:** preço unitário. Numérico. Preço do produto por unidade em liras turcas (TL).
* **Payment\_method:** Variável string da forma de pagamento (dinheiro, cartão de crédito ou débito) utilizada na transação.
* **invoice\_date:** Data da fatura. O dia em que uma transação foi gerada.
* **shopping\_mall:** Variável string do nome do shopping onde foi feita a transação.
  1. **Acessando o metadados**

Para realizar a operação de acessar os metadados, se faz necessário a importação da biblioteca “Pandas”, que tem por funcionalidade adquirir dados, selecionar dados de interesse e fazer algumas transformações simples. E para realizar esta operação inserir o comando: “import pandas as pd” e em seguida inserir o comando para buscar o arquivo em seu diretório de origem, “df = pd.read\_csv (r'https://raw.githubusercontent.com/OtavioBer/ProjetoAplicadoI/main/Dados/Backup/Base\_Keaggle\_Original.csv').



* 1. **Conhecendo o metadados**

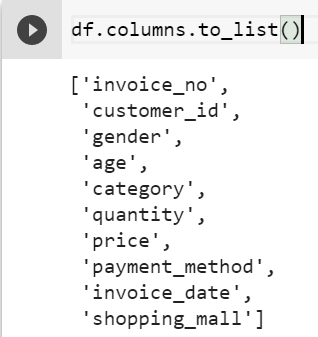
Para se iniciar uma análise com os dados, primeiramente é importante conhecer o metadados que será explorado, para isto é possível utilizar o comando “df.head()”, que é um comando que permite exibir o aspecto inicial dos dados a partir das primeiras linhas, identificando quais os tipos de dados que será analisado e quais são seus atributos.



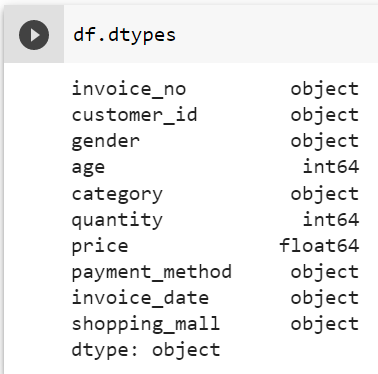
Identificando a quantidades de linhas e colunas do dataset, podemos utilizar o comando “df.shape”:



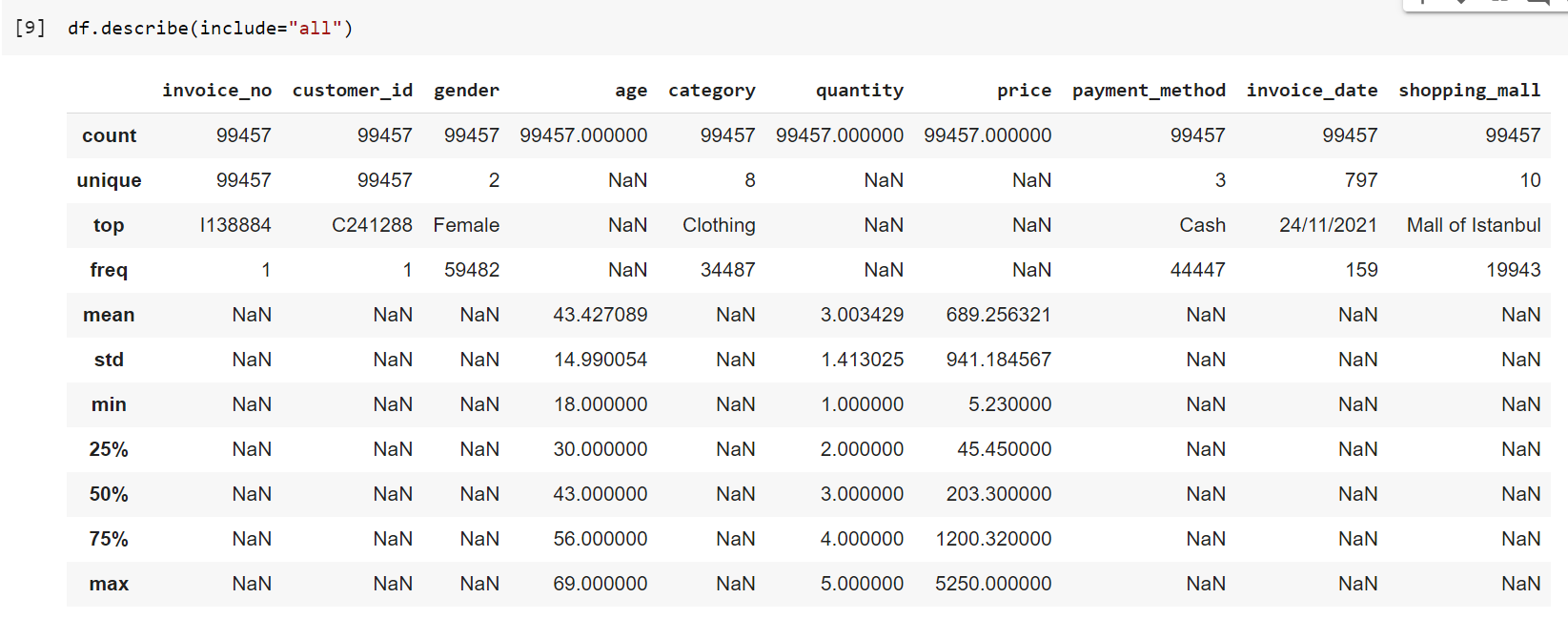
Identificando quais os nomes dos atributos, colunas do dataset, podemos utilizar o comando “df.columns.to\_list()”.



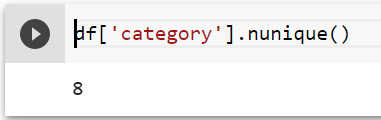
Identificando quais os tipos dos atributos, podemos utilizar o comando “df.dtypes”.



Identificando informações estatísticas sumarizadas dos dados, podemos utilizar o comando “df.describe(include="all")”.



Identificando informações de valores/quantidade únicos em um determinado atributo, para entender com a variedades de informações que será possível desdobrar as análises, podemos utilizar o comando “df['category'].nunique()”.



* 1. **Bibliotecas de exploração de dados**

Na exploração do dataframe é importante identificar quais bibliotecas Python serão necessárias para realizar a exploração dos dados de forma que os códigos aplicados sejam performados adequadamente e de forma eficaz e simplificada. Neste projeto identificamos a necessidade de utilização das seguintes bibliotecas:

* Pandas
* Numpy
* Seborn
* Matplotlib

Utilizando os comandos:

import pandas as pd

import numpy as np

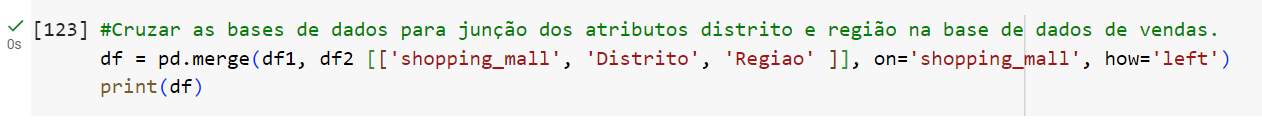
import seaborn as sns

import matplotlib.pyplot as plt

# TRATAMENTO DOS DADOS

Para melhor desempenhar a exploração do dataset escolhido, identificamos a necessidades de criar algumas funções para realizar análises de forma mais resumida e agrupada.

Realizamos o cruzamento de base de dados para obter em um único dataset as informações de distrito e a região de cada shopping que contém na base original do dataset da Kaggle, utilizamos o comando de merge para a junção dos dados.



Também foi desenvolvido função para tratamento de data, separando da data o ano/mês, o ano, o mês e o dia da semana, aplicando o comando:

# converter tipo data da fatura para data

df['invoice\_date'] = pd.to\_datetime(df['invoice\_date'], dayfirst=True)

# criar novas colunas para ano\_mês, ano, mês, dia da semana

df['year\_month'] = df['invoice\_date'].dt.to\_period('M')

df['year'] = df['invoice\_date'].dt.strftime('%Y')

df['month'] = df['invoice\_date'].dt.strftime('%b')

df['day\_of\_week'] = df['invoice\_date'].dt.strftime('%a')

Ordenar os meses e os dias da semana, aplicando o comando:

# definir a ordem do mês

month\_order = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec']

df['month'] = df['month'].astype(pd.CategoricalDtype(categories=month\_order, ordered=True))

# definir a ordem do dia da semana

weekday\_order = ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun']

df['day\_of\_week'] = df['day\_of\_week'].astype(pd.CategoricalDtype(categories=weekday\_order, ordered=True))

Criar cálculo de vendas, onde será realizado a operação de multiplicação entre o atributo de quantity e price, apresentando o resultado em vendas, aplicando o comando:

# criar novas colunas para o cálculo de vendas

df['vendas'] = df['quantity'] \* df['price']

Criar funcionalidade para range de faixa etária a partir do atributo age, aplicando o comando:

# Criar função para classificar a faixa etária

def faixa\_etaria(age):

    if age <= 19:

        age = '18-19'

    elif 20 <= age <= 29:

        age = '20-29'

    elif 30 <= age <= 39:

        age = '30-39'

    elif 40 <= age <= 49:

        age = '40-49'

    elif 50 <= age <= 59:

        age = '50-59'

    else:

        age = '60-69'

    return age

# criar novas colunas para faixa etária

df['faixa\_etaria'] = df['age'].apply(faixa\_etaria)

df['faixa\_etaria'].value\_counts()

# ANÁLISE EXPLORATÓRIA DE DADOS

Realizar as análises dos dados para resolver as questões apresentadas na proposta de análise, e identificar problemas, tendências e comportamentos, aplicando os métodos estatísticos de análise exploratória de dados, através do uso da linguagem Python, utilizando a ferramenta Colaboratory.

* 1. **O público feminino consome mais roupas que o masculino?**

Os códigos abaixo classificam o grupo feminino e masculino que compram na categoria de roupas (clothing) em quantidade de pessoas, neste caso podemos analisar que o público feminino tende a comprar mais roupas que o masculino.

df.loc[(df['gender'] == 'Female') & (df['category'] == 'Clothing'), 'gender'].value\_counts()

**A picture containing text

Description automatically generated**

df.loc[(df['gender'] == 'Male') & (df['category'] == 'Clothing'), 'gender'].value\_counts()

A picture containing graphical user interface

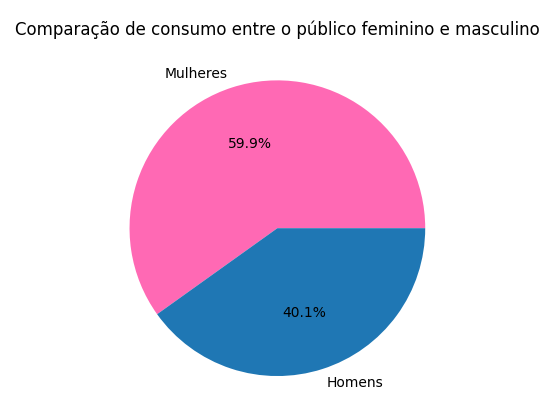
Description automatically generated

Também é possível analisar em quantidade de roupas (clothing), somando a quantidade que cada gênero compra.

df[df['category'] == 'Clothing'].groupby(['gender', 'category'])['quantity'].agg(['sum'])

Graphical user interface, text, application, chat or text message

Description automatically generated



**Destaca-se que o público feminino consome 19,8%🠉 a mais que o público masculino.**

* 1. **Quais produtos são mais consumidos por gênero?**

O seguinte código utilizado para analisar, foi agrupado por gênero (gender) e categoria (category), e no final somando a quantidade, gerando uma lista com a quantidade de produtos vendidos por gênero. Identifica o produto “roupas” como o mais consumido por mulheres e homens.

df.groupby(['gender','category'])['quantity'].agg(['sum'])

**Graphical user interface, text

Description automatically generated**

**Graphical user interface, application

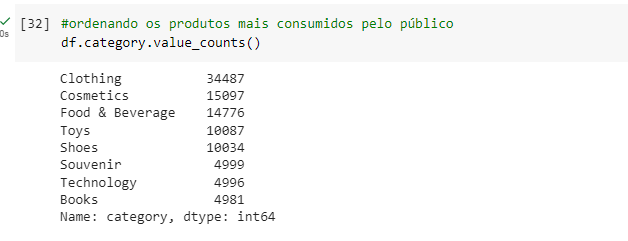
Description automatically generated**

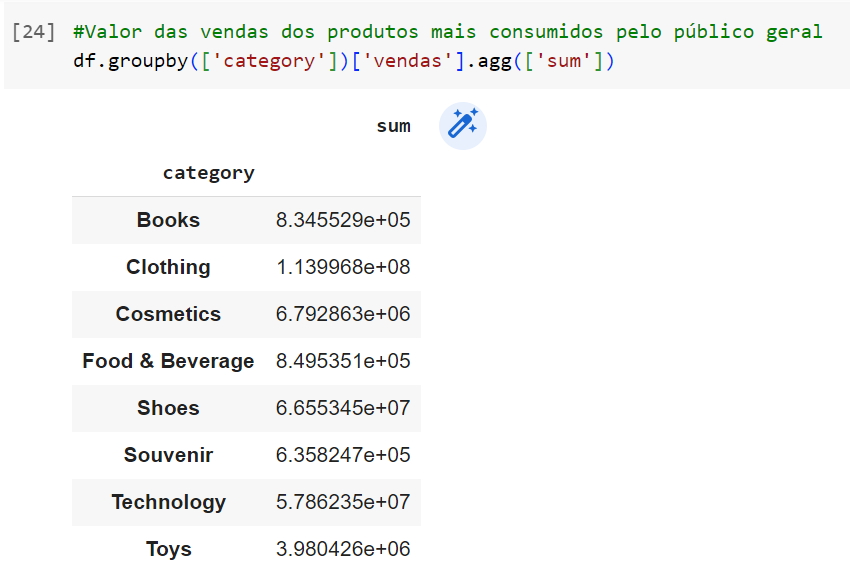
* 1. **Qual** **o ranking de produtos mais consumidos pelo público em geral?**

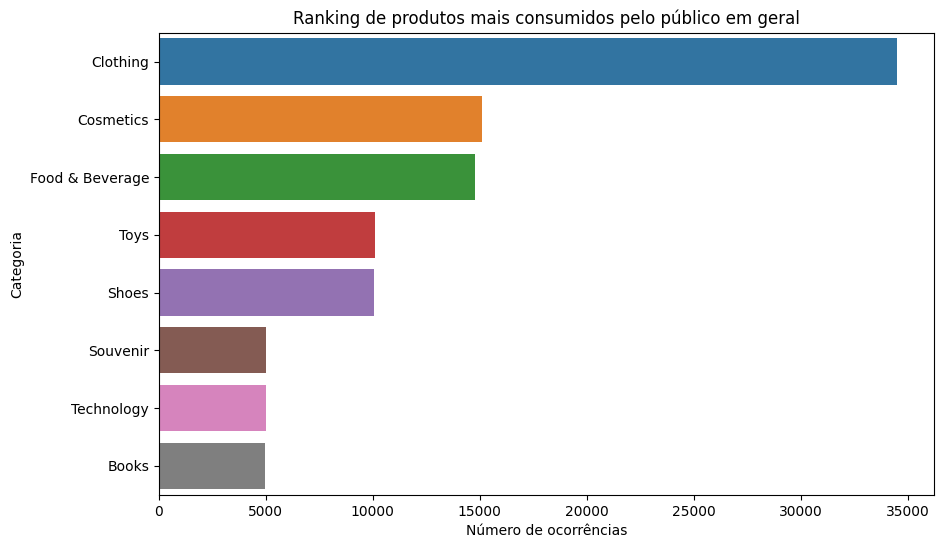
Utilizamos o código abaixo para identificar a quantidade de vezes que cada categoria de produtos mais aparecia no DataFrame, para assim identificar o ranking de produtos mais consumidos pelo público em geral em ordem decrescente.

df.category.value\_counts()

df.groupby(['category'])['vendas'].agg(['sum'])

****



****

**Análise:** As vendas de roupas é o segmento líder, com uma quantidade significativamente maior de vendas em 35%🠉em comparação aos outros produtos listados. Isso indica uma demanda forte por roupas entre os consumidores de shoppings em Istambul. Compondo o top 3 seguem as vendas de Cosméticos e Comidas e Bebidas que possuem uma pequena diferença de 321 de ocorrência entre esses dois setores, ficando quase empatados.

* 1. **Qual a forma de pagamento mais utilizada por faixa etária e por gênero?**

Nessa questão, foi utilizado o método para criar uma faixa etária no DataFrame e um group-by, para que possamos verificar qual a forma de pagamento mais utilizada. No qual foi identificado que o método mais utilizado para homens e mulheres em todas as faixas etárias é o dinheiro (cach).

grouped = df.groupby(['gender', pd.cut(df['age'], bins=[0, 20, 30, 40, 50, 60, 100],

                      labels=['0-20', '21-30', '31-40', '41-50', '51-60', '60+'])])

result = grouped['payment\_method'].agg(pd.Series.mode)

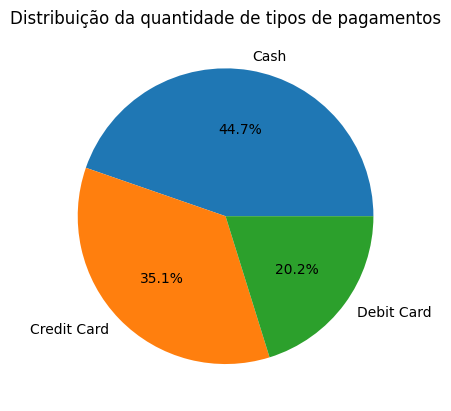
print(result)

Interface gráfica do usuário, Texto

Descrição gerada automaticamente com confiança média



**Destaca-se que a forma de pagamento dinheiro o público masculino tem uma proporção maior em 0,4%🠉 em relação ao feminino.**

****

* 1. **Qual a sazonalidade (dia da semana) de consumo com maior volume de compras?**

Nessa questão, na primeira parte tivemos que incluir uma nova coluna com o nome dos dias da semana, chamada de “day\_of\_week” para ser possível fazer a análise. Na segunda parte, foi feito um group-by com os dias da semana e a soma das quantidades de compras feitas em ordem decrescentes para que possamos concluir a análise.

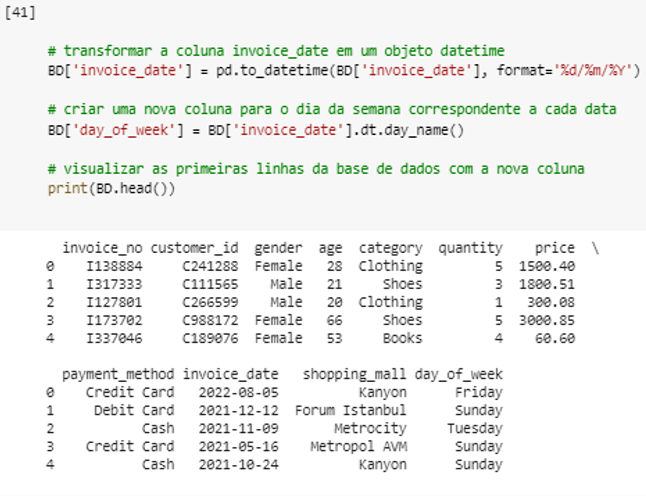
df['invoice\_date'] = pd.to\_datetime(df['invoice\_date'], format='%d/%m/%Y')

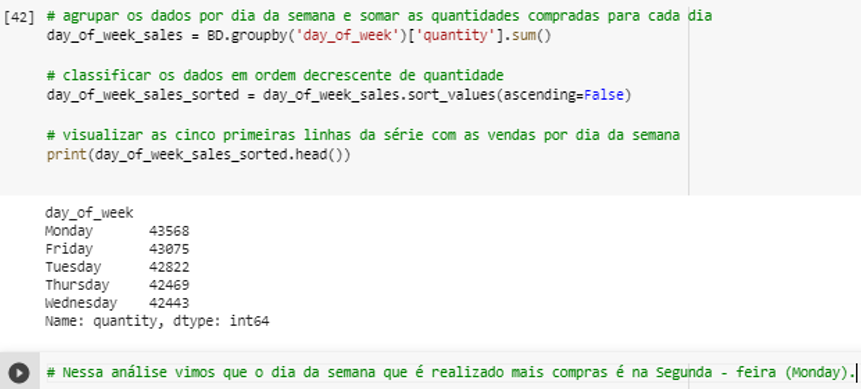
df['day\_of\_week'] = df['invoice\_date'].dt.day\_name()

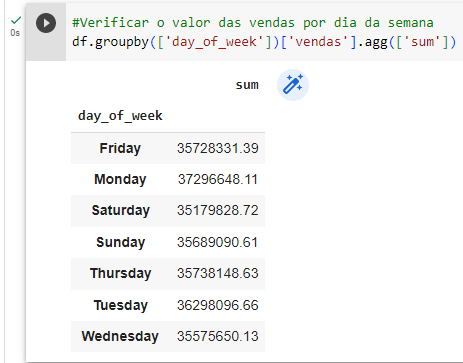
day\_of\_week\_sales = df.groupby('day\_of\_week')['quantity'].sum()

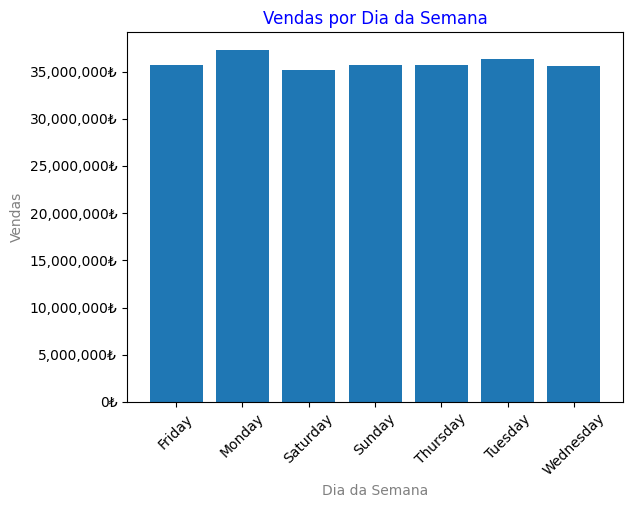
day\_of\_week\_sales\_sorted = day\_of\_week\_sales.sort\_values(ascending=False)

print(day\_of\_week\_sales\_sorted.head())

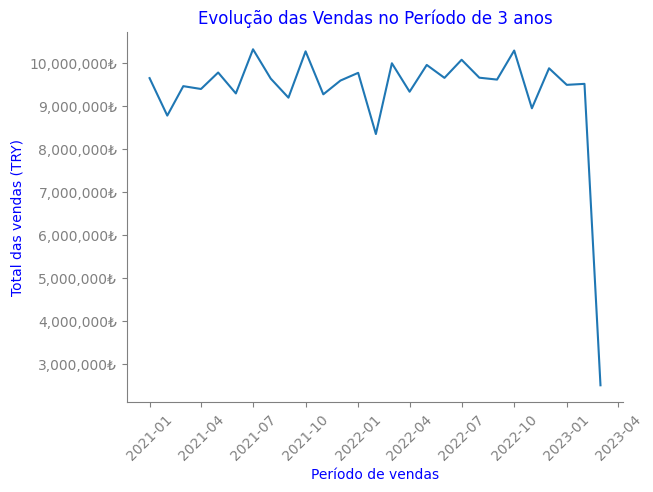








**Análise:** Em relação as tendências de compras em Istambul, na linha de tempo do período analisado, observa-se que os comportamentos de compras tem sido bem linear e até mesmo em relação aos dias da semana o consumo é bem semelhante. Destaca-se na evolução do gráfico mês que houve apenas um desvio abaixo da média e 3 acima da média, sendo esta média em torno de ₺9.579.319 (TRY) de lira turca. E a segunda-feira se destaca com um percentual de 1%🠉 a mais que os demais dias da semana.



* 1. **Qual a média de consumo anual por gênero e faixa etária?**

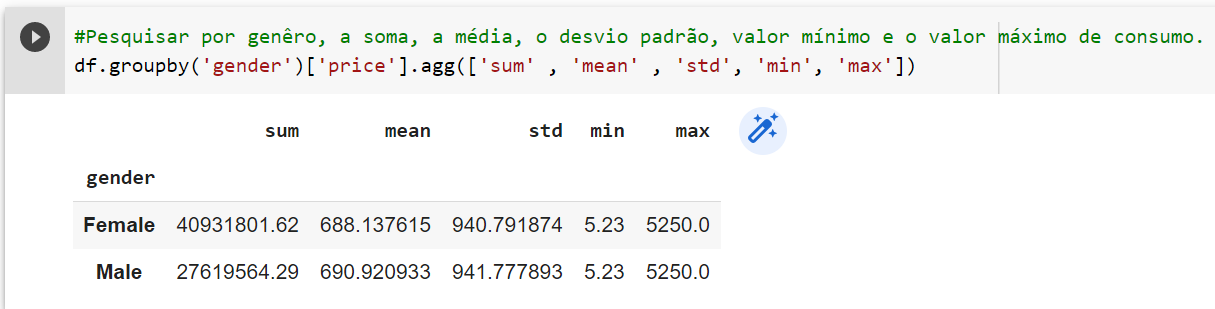
Foi analisado primeiramente os atributos separadamente, por ano, genêro e faixa etária, utilizando a medida de preço e posteriormente os três juntamente. Pesquisar por ano (year) aplicando as medidas de soma, média, desvio padrão, valor mínimo e o valor máximo de consumo anual, aplicou-se o seguinte código:

df.groupby('year')['price'].agg(['sum' , 'mean' , 'std', 'min', 'max'])



Pesquisar por genêro (gender) aplicando as medidas de soma, média, desvio padrão, valor mínimo e o valor máximo de consumo por genêro, aplicou-se o seguinte código:

df.groupby('gender')['price'].agg(['sum' , 'mean' , 'std', 'min', 'max'])



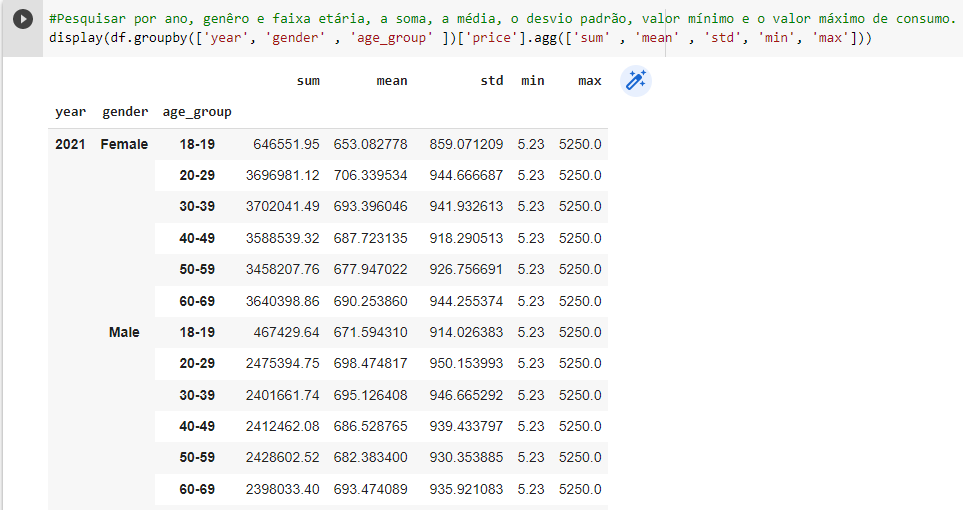
Pesquisar por faixa etária (age\_group) aplicando as medidas de soma, média, desvio padrão, valor mínimo e o valor máximo de consumo por faixa etária, aplicou-se o seguinte código:

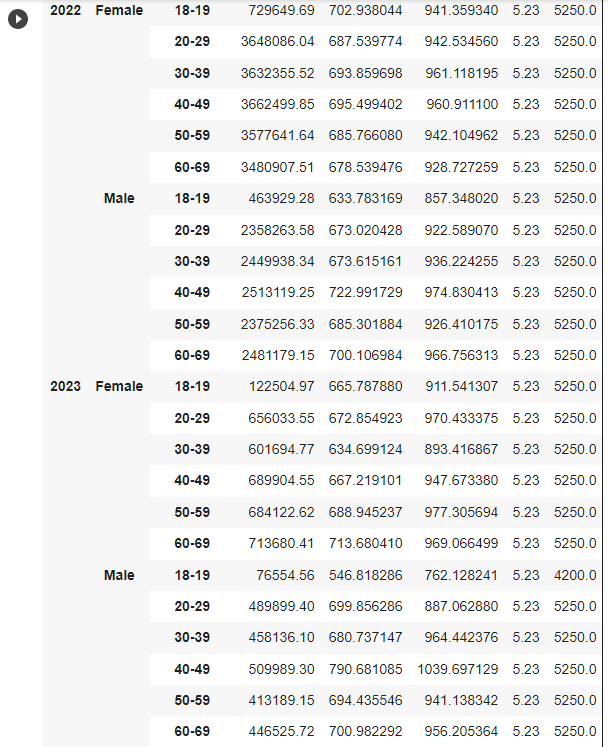
df.groupby('age\_group')['price'].agg(['sum' , 'mean' , 'std', 'min', 'max'])



Pesquisar os três atributos juntos atendendo a questão proposta, sendo ano, genêro e faixa etária aplicando as medidas de soma, média, desvio padrão, valor mínimo e o valor máximo de consumo, aplicou-se o seguinte código:

display(df.groupby(['year', 'gender' , 'age\_group' ])['price'].agg(['sum' , 'mean' , 'std', 'min', 'max']))

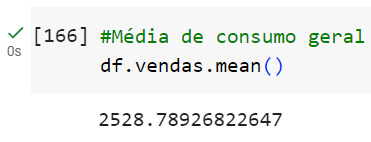


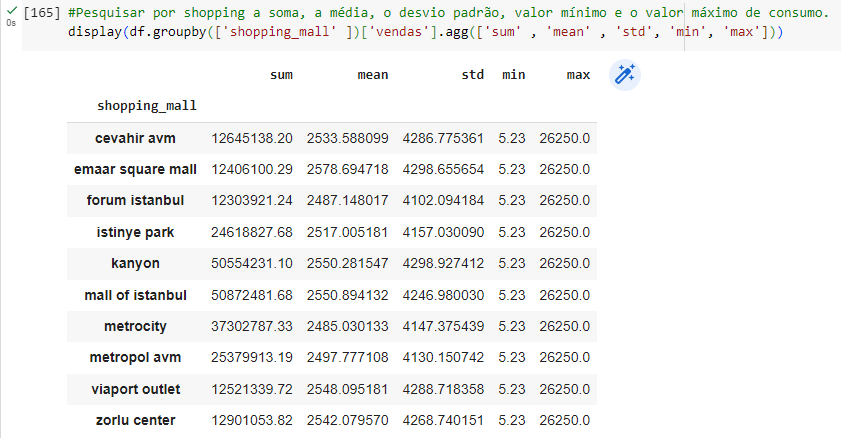


* 1. **Qual shopping possui o maior ticket médio de compras?**

Foi analisado a média geral e a de todos os shoppings, para identificar aquele que obteve o maior ticket médio, aplicando a consulta ao atributo de shopping\_mall, utilizando as medidas de soma, média, desvio padrão, valor mínimo e o valor máximo de consumo, aplicou-se o seguinte código:

display(df.groupby(['shopping\_mall' ])['vendas'].agg(['sum' , 'mean' , 'std', 'min', 'max']))





Para realizar a análise de identificação de qual shopping possui o maior ticket médio de compras, foi necessário realizar o agrupamento e realizar cálculos de comparação, no qual foi identificado o shopping “Emaar Square Mall” atendendo esta análise, aplicou-se o seguinte código:

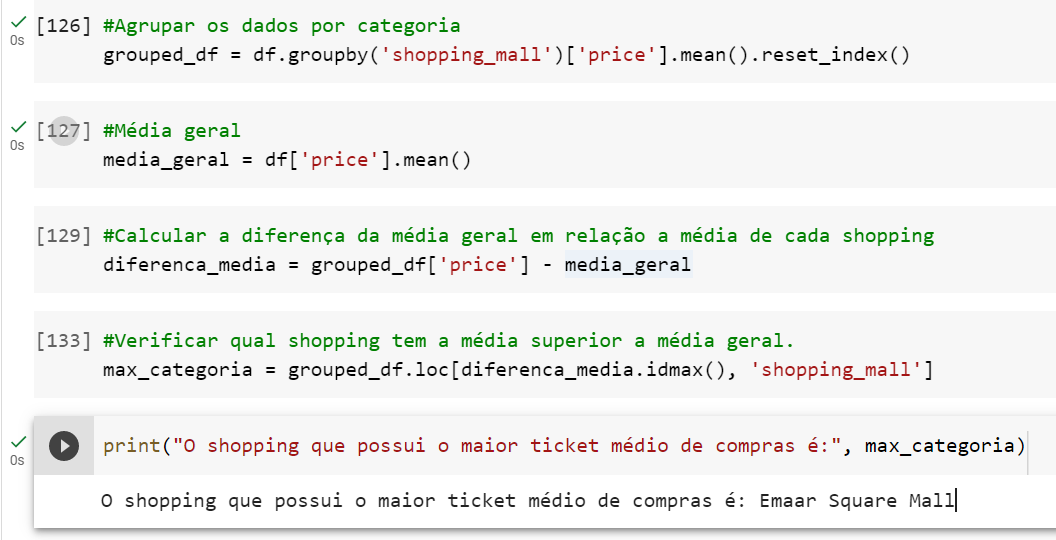
grouped\_df = df.groupby('shopping\_mall')['vendas'].mean().reset\_index()

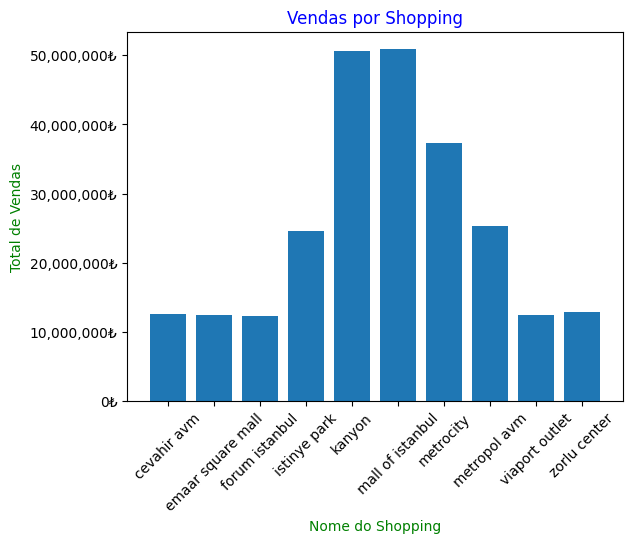
media\_geral = df['vendas'].mean()

diferenca\_media = grouped\_df['vendas'] - media\_geral

max\_categoria = grouped\_df.loc[diferenca\_media.idmax(), 'shopping\_mall']

print("O shopping que possui o maior ticket médio de compras é:", max\_categoria)



****

**Destaca-se os shoppings kanyon e mall of istanbul representa 40% das vendas no período analisado.**

* 1. **Como se dá a comparação de gastos totais por gênero e por faixa etária? E dos gastos médios?**

#criando uma nova coluna na tabela

df2 = df

df2['consumption'] = df2['price']\*df2['quantity']

df3 = df2

df3['age\_group'] = 0

def age\_group(age):

    if age <= 19:

        age = '18-19'

    elif 20 <= age <= 29:

        age = '20-29'

    elif 30 <= age <= 39:

        age = '30-39'

    elif 40 <= age <= 49:

        age = '40-49'

    elif 50 <= age <= 59:

        age = '50-59'

    else:

        age = '60-69'

    return age

# criar novas colunas para age\_group

df3['age\_group'] = df3['age'].apply(age\_group)

df3['age\_group'].value\_counts()

#agrupando e verificando a soma e a média

df4 = df3.groupby(['gender','age\_group'])['consumption'].agg(['sum', 'mean'])

df4 = df4.sort\_values(['gender'], ascending=True)

df4 = df4.reset\_index()

df4

#montagem de filtros

filt = (df4['gender'] == 'Female')

filtro = (df4['gender'] == 'Male')

df5 = df4[filt]

df6 = df4[filtro]

#criação da primeira tabela

n1 = list(df5['sum'])

n2 = list(df6['sum'])

barWidth = 0.34

r1 = np.arange(len(n1))

r2 = [x + barWidth for x in r1]

plt.bar(r1, n1, color='#6A5ACD', width = barWidth, label='Female')

plt.bar(r2, n2, color='#6495ED', width = barWidth, label='Male')

plt.xlabel('Idade')

plt.xticks([r + barWidth for r in range(len(n1))], list(df5['age\_group']))

plt.ylabel('Total Gastos')

plt.title('Gasto total por faixa etária')

plt.legend(loc='lower right')

plt.show()

#criação da segunda tabela

m1 = list(df5['mean'])

m2 = list(df6['mean'])

barWidth = 0.34

s1 = np.arange(len(m1))

s2 = [x + barWidth for x in r1]

#criação da segunda tabela

plt.bar(s1, m1, color='#6A5ACD', width = barWidth, label='Female')

plt.bar(s2, m2, color='#6495ED', width = barWidth, label='Male')

plt.xlabel('Idade')

plt.xticks([r + barWidth for r in range(len(n1))], list(df5['age\_group']))

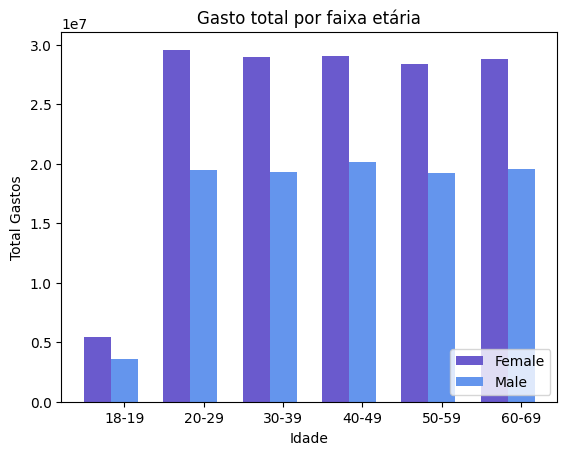
plt.ylabel('Total Gastos')

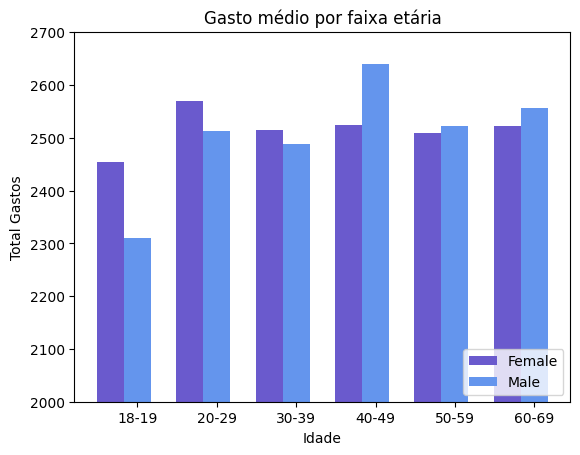
plt.title('Gasto médio por faixa etária')

plt.ylim([2000, 2700])

plt.legend(loc='lower right')

plt.show()

****

****

**Análise:** Através dos gráficos abaixo fica perceptível o grande volume de compras realizadas em shoppings centers de Istambul, porém o gráfico da soma total por gênero e idade não nos traz uma ideia clara dos gastos por cada um desses grupos, enquanto o gráfico dos gastos médios nos indica uma proporção mais realística dos grupos que mais e menos gastaram.

* 1. **Qual faixa etária mais consome artigos de leitura?**

df2 = df.groupby('category')['quantity'].agg([sum])

df2 = df2.sort\_values(['sum'], ascending=False)

df2

df3 = df

df3['age\_group'] = 0

df3

def age\_group(age):

    if age <= 19:

        age = '18-19'

    elif 20 <= age <= 29:

        age = '20-29'

    elif 30 <= age <= 39:

        age = '30-39'

    elif 40 <= age <= 49:

        age = '40-49'

    elif 50 <= age <= 59:

        age = '50-59'

    else:

        age = '60-69'

    return age

# criar novas colunas para age\_group

df3['age\_group'] = df3['age'].apply(age\_group)

df3['age\_group'].value\_counts()

#Agrupando por categoría, faixa etária e quantidade consumida

df2 = df3.groupby(['category', 'age\_group'])['quantity'].agg([sum])

df2 = df2.sort\_values(['category'], ascending=True)

df2 = df2.reset\_index()

df2

#filtrando apenas artigos de leitura

filt = (df2['category'] == 'Books')

df4 = df2[filt]

df4 = df4.sort\_values(['sum'], ascending=False)

df4

mv = (df4['sum'] == df4['sum'].max())

df5 = df4[mv]

mv2 = list(df5['age\_group'])

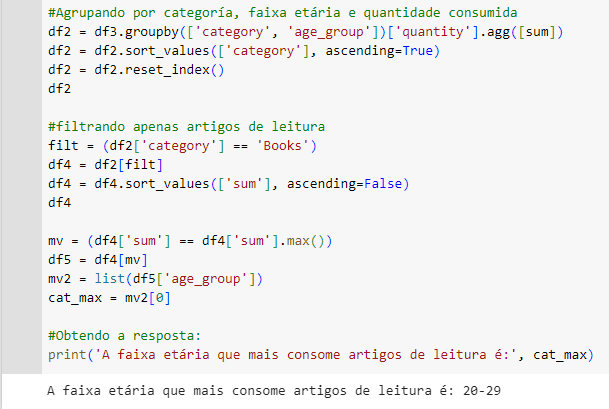
cat\_max = mv2[0]

#Obtendo a resposta:

print('A faixa etária que mais consome artigos de leitura é:', cat\_max)

Resposta: A faixa etária que mais consome artigos de leitura é: 20-29





* 1. **Qual a localidade de shopping tem o maior consumo por categoria?**

# Agrupar o DataFrame por "region" e "category"

grupo = df.groupby(['Regiao', 'category'])

# Calcular o total de vendas para cada categoria e shopping\_mall

total\_vendas = grupo['vendas'].sum()

# Transformar a série resultante em um DataFrame e ordenar em ordem decrescente de vendas

total\_vendas\_df = total\_vendas.reset\_index().sort\_values(by=['category','vendas'], ascending=[True, False])

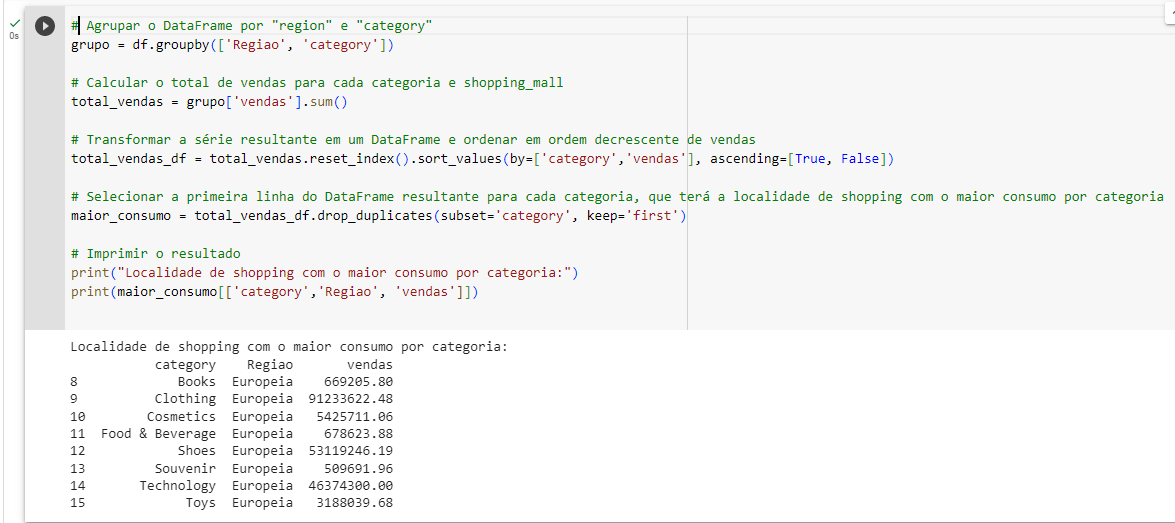
# Selecionar a primeira linha do DataFrame resultante para cada categoria, que terá a localidade de shopping com o maior consumo por categoria

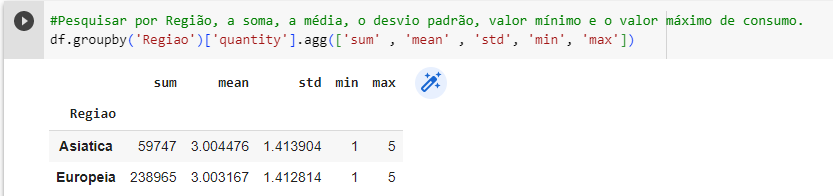
maior\_consumo = total\_vendas\_df.drop\_duplicates(subset='category', keep='first')

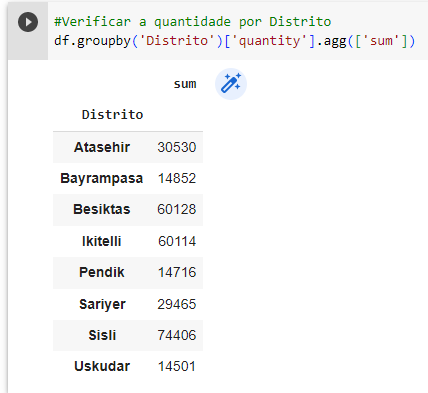
# Imprimir o resultado

print("Localidade de shopping com o maior consumo por categoria:")

print(maior\_consumo[['category','Regiao', 'vendas']])









* 1. **Qual produto tem o maior e menor ticket médio por região?**

# Agrupamos o DataFrame por "region" e "category"

grupo = df.groupby(['Regiao', 'category'])

# Calcular o ticket médio para cada categoria e região

ticket\_medio = grupo['vendas'].mean()

# Transformar a série resultante em um DataFrame e ordenar em ordem decrescente de ticket médio

ticket\_medio\_df = ticket\_medio.reset\_index().sort\_values(by=['Regiao', 'vendas'], ascending=[True, False])

# Selecionar a primeira e a última linha do DataFrame resultante para cada região, que terá o produto com o maior e menor ticket médio por região, respectivamente

maior\_ticket\_medio = ticket\_medio\_df.drop\_duplicates(subset='Regiao', keep='first')

menor\_ticket\_medio = ticket\_medio\_df.drop\_duplicates(subset='Regiao', keep='last')

# Imprimir o resultado

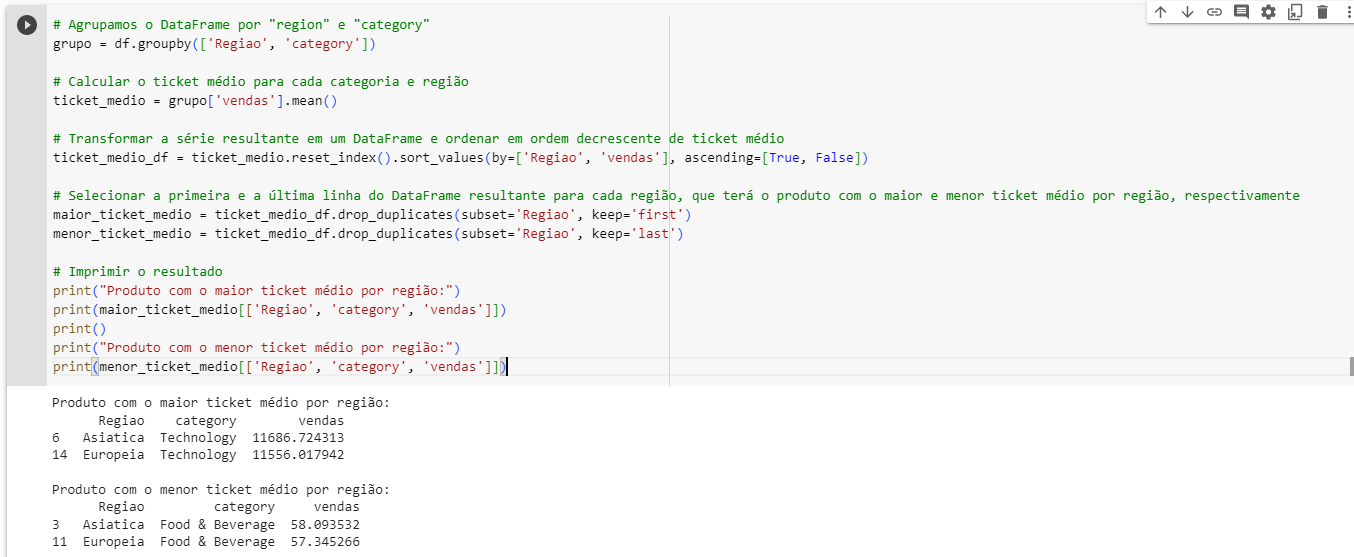
print("Produto com o maior ticket médio por região:")

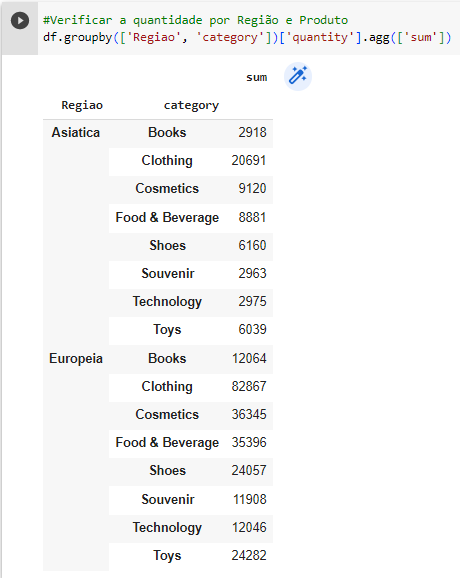
print(maior\_ticket\_medio[['Regiao', 'category', 'vendas']])

print()

print("Produto com o menor ticket médio por região:")

print(menor\_ticket\_medio[['Regiao', 'category', 'vendas']])





# Agrupamos o DataFrame por "region" e "category"

grupo = df.groupby(['Regiao', 'category'])

# Calcular o ticket médio para cada categoria e região

ticket\_medio = grupo['vendas'].mean()

# Transformar a série resultante em um DataFrame e ordenar em ordem decrescente de ticket médio

ticket\_medio\_df = ticket\_medio.reset\_index().sort\_values(by=['Regiao', 'vendas'], ascending=[True, False])

# Selecionar a primeira e a última linha do DataFrame resultante para cada região, que terá o produto com o maior e menor ticket médio por região, respectivamente

maior\_ticket\_medio = ticket\_medio\_df.drop\_duplicates(subset='Regiao', keep='first')

menor\_ticket\_medio = ticket\_medio\_df.drop\_duplicates(subset='Regiao', keep='last')

# Imprimir o resultado

print("Produto com o maior ticket médio por região:")

print(maior\_ticket\_medio[['Regiao', 'category', 'vendas']])

print()

print("Produto com o menor ticket médio por região:")

print(menor\_ticket\_medio[['Regiao', 'category', 'vendas']])

df = ticket\_medio\_df

# Criar o gráfico

plt.figure(figsize=(10, 6))

ax = sns.barplot(x='Regiao', y='vendas', hue='category', data=df)

plt.xlabel('Região')

plt.ylabel('Vendas')

plt.title('Vendas por Região e Categoria')

plt.xticks(rotation=45)

# Adicionar os valores dos preços acima das barras

for p in ax.patches:

    ax.annotate(format(p.get\_height(), '.2f'),

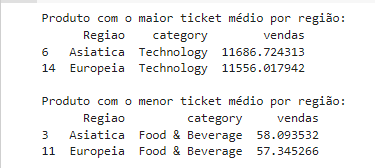
                (p.get\_x() + p.get\_width() / 2, p.get\_height()),

                ha='center', va='bottom')

plt.tight\_layout()

# Mostrar o gráfico

plt.show()



#Criar o calculo de quantidade de Distrito

Distrito\_qtd = df['Distrito'].value\_counts()

Distrito\_qtd.plot(kind='bar', color='green')

# Inserir titulo e nomes dos eixos X e Y

plt.xlabel('Distrito')

plt.ylabel('Quantidade')

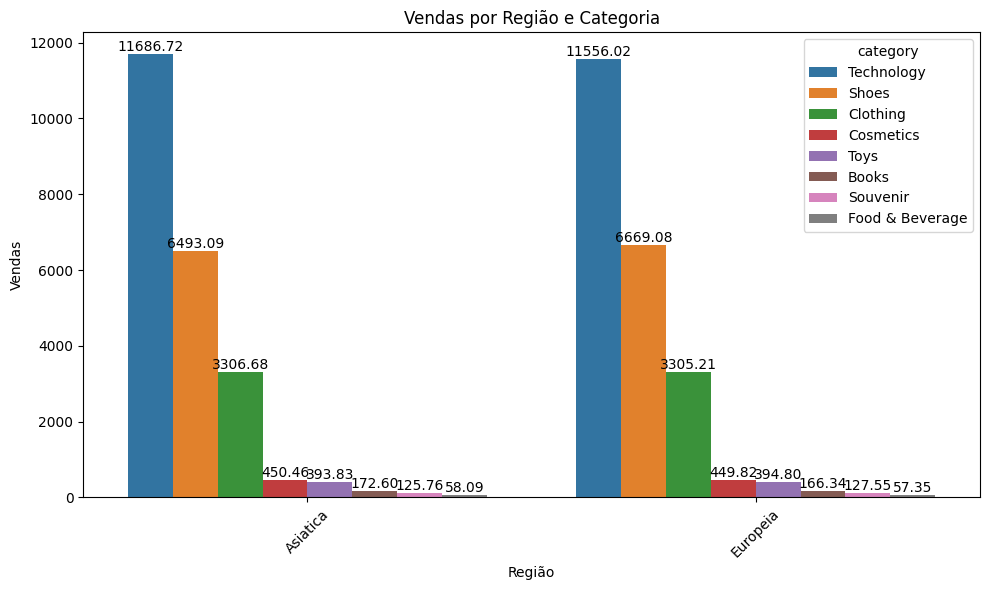
plt.title('Contagem de Distrito')

# Inserir rótulos nas barras

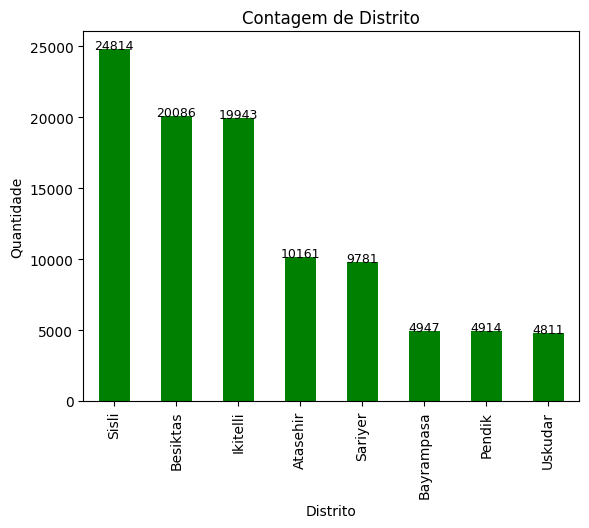
for i, count in enumerate(Distrito\_qtd):

    plt.text(i, count+10, str(count), ha='center', fontsize=9)

plt.show()

****

Analisando o gráfico tanto para os shoppings da região asiática e europeia o maior ticket médio são para produtos da categoria Technology (Tecnologia) e os menores ticket médio foram para Food & Beverage (Comidas e Bebidas).

****

# CONCLUSÃO

Com base na análise exploratória dos dados de compras em 10 shoppings em Istambul, alguns aspectos relevantes podem ser extraídos.

Inicialmente, observamos uma clara disparidade de consumo entre os gêneros. O público feminino representa a maioria das compras, com uma proporção de 59,9%, enquanto os homens respondem por 40,1% do consumo. Essa diferença pode ser influenciada pela compra específica de cada grupo ou fatores socioeconômicos e culturais que influenciam o comportamento de consumo.

Ao analisar os produtos mais consumidos, identificamos as seguintes categorias principais: roupas, cosméticos, comidas e bebidas, brinquedos, sapatos, lembranças, tecnologia e livros. Dentre esses, as roupas é o item mais popular, com um total de 34.487 unidades adquiridas, seguidas pelos cosméticos (15.097) e comidas e bebidas (14.776). Esses resultados podem indicar as necessidades e desejos dos consumidores em Istambul.

No que diz respeito às formas de pagamento, constatamos que o dinheiro é a opção mais utilizada tanto por homens quanto por mulheres. Isso pode ser atribuído a fatores como preferência pessoal, questões culturais ou mesmo restrições na aceitação de outros métodos de pagamento em determinados estabelecimentos comerciais.

Ao analisar o dia da semana com maior número de compras, verificamos que a segunda-feira se destaca discretamente em relação aos demais dias. Essa informação pode ser útil para os varejistas, que podem direcionar suas estratégias promocionais ou ajustar a disponibilidade de funcionários com base nos padrões de consumo ao longo da semana.

O estudo também revelou que o Emaar Square Mall é o shopping com o maior ticket médio de compras. Isso sugere que esse centro comercial pode atrair um público com maior poder aquisitivo ou oferecer produtos e marcas de maior valor agregado.

Destaca-se ainda que os shoppings Kanyon e Mall of Istanbul representam uma parcela significativa das vendas no período analisado, totalizando 40% do volume total. Esses shoppings parecem ter um papel importante no cenário de compras em Istambul, e estratégias direcionadas a esses estabelecimentos podem trazer resultados positivos para os varejistas.

Ao considerar o ticket médio por categoria de produtos, notamos que a tecnologia possui o maior valor, indicando que os consumidores estão dispostos a investir mais nesse tipo de item. Por outro lado, as comidas e bebidas, apresentam o menor ticket médio, o que pode ser explicado pela natureza mais acessível e de consumo cotidiano desses produtos.

Em relação ao gasto médio por faixa etária, identificamos diferenças entre homens e mulheres. Para os homens, a faixa etária de 40 a 49 anos apresenta o maior gasto médio, enquanto para as mulheres, a faixa etária de 20 a 29 anos se destaca nesse aspecto. Esses resultados podem indicar padrões de consumo distintos e úteis para segmentar o mercado e direcionar estratégias de marketing mais eficazes.

Em suma, com base nos dados analisados, podemos concluir que o consumo em shoppings em Istambul é influenciado pelo gênero, influenciado de produtos, formas de pagamento, dias da semana, ticket médio por shopping e categoria de produtos, bem como pela faixa etária dos consumidores. Essas informações podem ser valiosas para os varejistas e profissionais de marketing que buscam compreender e atender melhor às necessidades e desejos dos consumidores em Istambul, permitindo o desenvolvimento de estratégias mais eficientes e direcionadas.

# REFERÊNCIAS

* Aulas e textos de apoio dos componentes de Análise Exploratória de Dados, Algoritmo e Programação e Pensamento Computacional.
* Pesquisas diversas para construção de códigos de exploração de dados e de criação de gráficos.
* Dataset:

[**https://www.kaggle.com/datasets/mehmettahiraslan/customer-shopping-dataset**](https://www.kaggle.com/datasets/mehmettahiraslan/customer-shopping-dataset)

* Dados de apoio da região dos Shopping, Google Maps.

# LINK VÍDEO

Para a apresentação deste projeto realizamos a gravação de um vídeo relatando um resumo geral do desenvolvimento deste componente, onde houve a participação de todos os integrantes do grupo. O vídeo estará disponível no youtube, que poderá ser acessado pelo link:

<https://www.youtube.com/watch?v=zn3LurrLf2k>

O link do código Python consta no Colab:

<https://colab.research.google.com/drive/19Kxyv0W2HZ8yjcBo5znpmmnj7rfHd4-5>