

SAMUEL FERREIRA ALVES

Gere um script em **Python** que:

1. Utilize a biblioteca **pandas** para carregar um arquivo chamado `churn_data.csv`.
 - Caso o arquivo não esteja no mesmo diretório, permita passar um **caminho completo**.
 - Trate erros comuns, como `FileNotFoundError`, e exiba uma mensagem amigável ao usuário.
 - Caso o separador não seja vírgula, tente também `sep=' ; '`.
 - Caso haja problema de acentuação, tente `encoding='utf-8'` e `encoding='latin-1'`.
2. Valide se a coluna **Churn** existe no dataset.
 - Caso não exista, tente localizar colunas com nome semelhante (`churn`, `CHURN`) e renomeá-la automaticamente para **Churn**.
 - Se não encontrar, mostre mensagem de erro clara listando as colunas disponíveis.
3. Calcule:
 - Total de clientes.
 - Quantos clientes cancelaram (`Churn = 'Yes'`).
 - Quantos clientes não cancelaram (`Churn = 'No'`).
 - **Taxa de churn** = (clientes que cancelaram ÷ total) × 100.

Exiba os resultados de forma clara no console, por exemplo:

```
Taxa de Churn: 24.00%
Clientes que cancelaram: 120
Clientes que permaneceram: 380
Total de clientes: 500
```

4.

5. Gere também dois **gráficos com matplotlib**:
 - Um **gráfico de pizza** mostrando a proporção entre cancelados e não cancelados, com cores diferentes e porcentagens.
 - Um **gráfico de barras** comparando o número absoluto de clientes que cancelaram e não cancelaram.
6. Organize o código em etapas com **comentários explicativos** para facilitar o entendimento.
7. Formate o código de forma profissional, garantindo que qualquer pessoa consiga rodar sem ajustes grandes."

#PARTE DO CÓDIGO

```
# =====
# ANÁLISE DE CHURN COMPLETA
# =====

import pandas as pd
import matplotlib.pyplot as plt
from google.colab import files

# -----
# 1. Upload do arquivo
# -----
print("📁 Faça o upload do arquivo 'churn_data.csv'")
uploaded = files.upload()

# Pega o nome do arquivo enviado
csv_name = list(uploaded.keys())[0]

# -----
# 2. Tenta carregar o dataset
# -----
df = None
tentativas = [
    {"sep": ",", "encoding": "utf-8"},
    {"sep": ";", "encoding": "utf-8"},
    {"sep": ",", "encoding": "latin-1"},
    {"sep": ";", "encoding": "latin-1"},
]
```

```

for t in tentativas:
    try:
        df = pd.read_csv(csv_name, **t)
        print(f"✅ Arquivo lido com sucesso usando {t}")
        break
    except Exception as e:
        print(f"Tentativa falhou: {t} -> {e}")

if df is None:
    raise ValueError("❌ Não foi possível carregar o arquivo CSV.
Verifique o separador e encoding.")

# -----
# 3. Validar coluna 'Churn'
# -----
if "Churn" not in df.columns:
    possiveis = [c for c in df.columns if c.lower() == "churn"]
    if possiveis:
        df = df.rename(columns={possiveis[0]: "Churn"})
        print(f"⚠️ Coluna renomeada de {possiveis[0]} para 'Churn'")
    else:
        raise ValueError(f"❌ A coluna 'Churn' não foi encontrada.
Colunas disponíveis: {list(df.columns)}")

# -----
# 4. Cálculos principais
# -----
total_clientes = len(df)
cancelaram = (df["Churn"] == "Yes").sum()
nao_cancelaram = (df["Churn"] == "No").sum()
taxa_churn = (cancelaram / total_clientes) * 100

# -----
# 5. Resultado no console
# -----
print("\n📊 RESULTADO DA ANÁLISE DE CHURN")
print(f"Taxa de Churn: {taxa_churn:.2f}%")
print(f"Clientes que cancelaram: {cancelaram}")
print(f"Clientes que permaneceram: {nao_cancelaram}")
print(f"Total de clientes: {total_clientes}")

# -----

```

```

# 6. Gráfico de pizza
# -----
plt.figure(figsize=(6,6))
plt.pie(
    [cancelaram, nao_cancelaram],
    labels=["Cancelaram (Yes)", "Não Cancelaram (No)"],
    autopct="%1.2f%",
    startangle=90,
    colors=["#FF6B6B", "#4ECDC4"]
)
plt.title("Distribuição de Cancelamentos (Churn)")
plt.show()

# -----
# 7. Gráfico de barras
# -----
plt.figure(figsize=(6,4))
plt.bar(
    ["Cancelaram", "Não Cancelaram"],
    [cancelaram, nao_cancelaram],
    color=["#FF6B6B", "#4ECDC4"]
)
plt.title("Quantidade de Clientes por Status de Churn")
plt.ylabel("Número de Clientes")
plt.show()

```

2 PARTE - DADOS - CSV

```

CustomerID,Nome,Idade,Genero,Churn
1,Ana,28,F,No
2,Bruno,34,M,Yes
3,Carlos,25,M,No
4,Diana,30,F,Yes
5,Eduardo,40,M,No
6,Fernanda,22,F,No
7,Gustavo,31,M,Yes
8,Heloisa,29,F,No
9,Ivan,27,M,Yes
10,Juliana,33,F,No
11,Karen,26,F,Yes
12,Luis,38,M,No

```

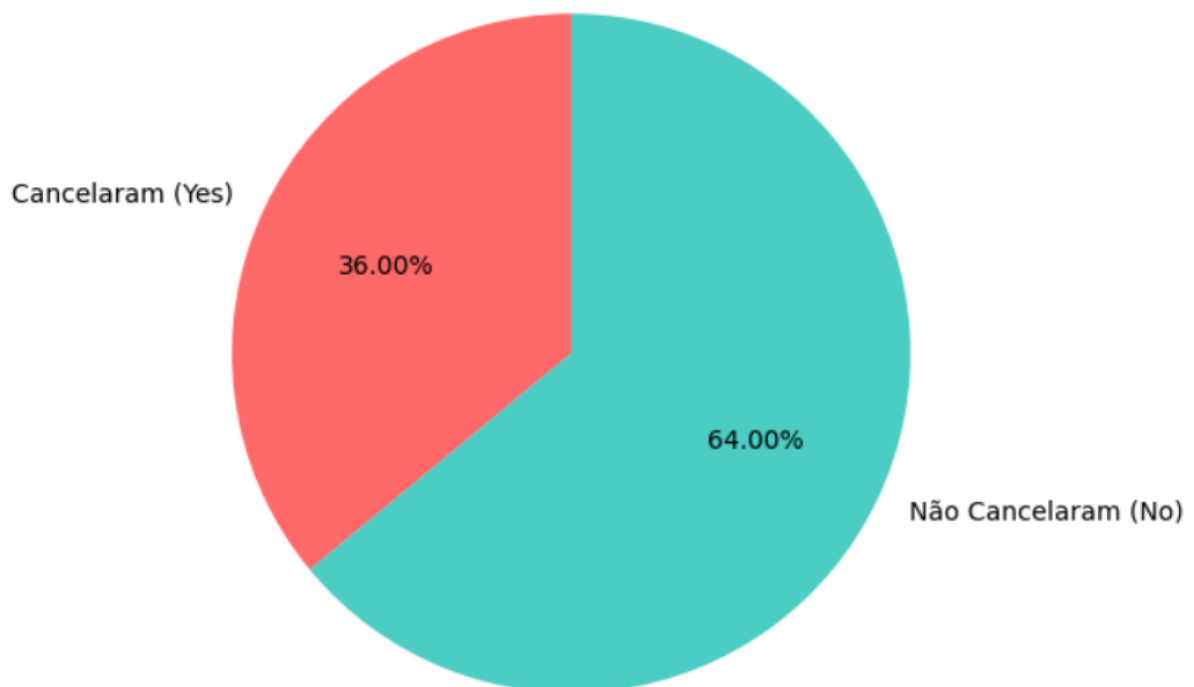
13, Mariana, 24, F, No
14, Nathan, 35, M, Yes
15, Olivia, 32, F, No
16, Paulo, 28, M, Yes
17, Quenia, 30, F, No
18, Rafael, 36, M, No
19, Sofia, 27, F, Yes
20, Thiago, 29, M, No
21, Vanessa, 31, F, Yes
22, William, 33, M, No
23, Xuxa, 25, F, No
24, Yuri, 37, M, Yes
25, Zara, 26, F, No
26, Alice, 28, F, No
27, Breno, 34, M, Yes
28, Carla, 22, F, No
29, Daniel, 30, M, No
30, Elisa, 27, F, Yes
31, Felipe, 32, M, No
32, Giovana, 29, F, No
33, Hugo, 35, M, Yes
34, Isabela, 24, F, No
35, João, 31, M, No
36, Katia, 33, F, Yes
37, Leonardo, 26, M, No
38, Marcos, 28, M, No
39, Nadia, 30, F, Yes
40, Otávio, 37, M, No
41, Patrícia, 25, F, No
42, Roberto, 36, M, Yes
43, Sandra, 27, F, No
44, Tiago, 29, M, No
45, Ursula, 31, F, Yes
46, Vinicius, 32, M, No
47, Wesley, 26, M, No
48, Yasmin, 28, F, Yes
49, Zeca, 30, M, No
50, Amanda, 33, F, No

RESULTADO + GRÁFICO

```
📁 Faça o upload do arquivo 'churn_data.csv'
📁 Escolher arquivos churn_data.csv
churn_data.csv(text/csv) - 985 bytes, last modified: 09/09/2025 - 100% done
Saving churn_data.csv to churn_data (1).csv
✅ Arquivo lido com sucesso usando {'sep': ',', 'encoding': 'utf-8'}

📊 RESULTADO DA ANÁLISE DE CHURN
Taxa de Churn: 36.00%
Clientes que cancelaram: 18
Clientes que permaneceram: 32
Total de clientes: 50
```

Distribuição de Cancelamentos (Churn)



Quantidade de Clientes por Status de Churn

