EDA

January 29, 2025

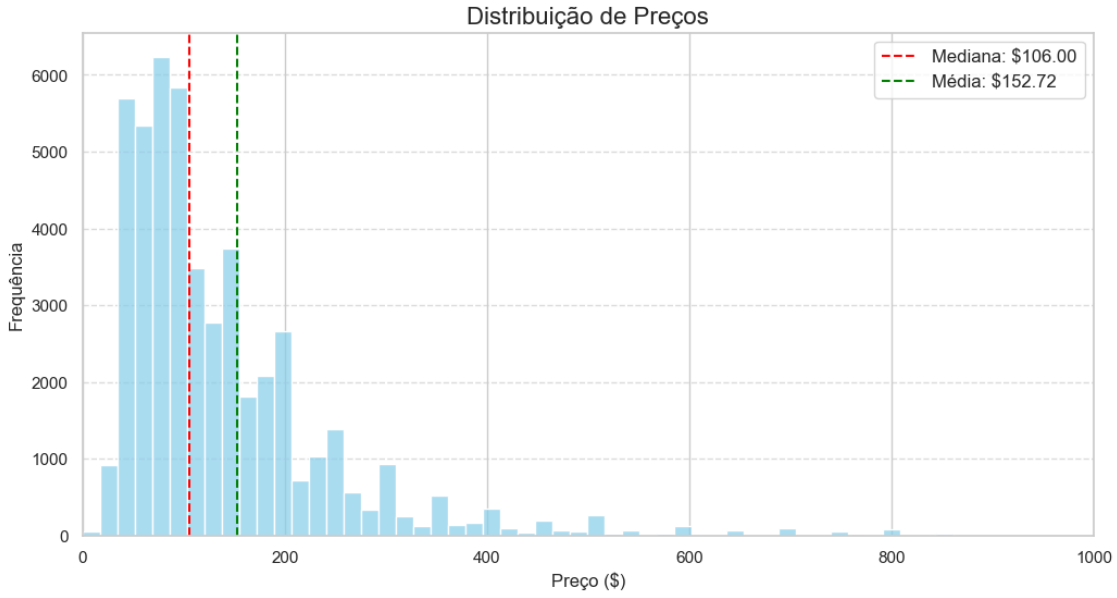
[1]:
```
#1.preparando as bibliotecas que serão usadoas
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import matplotlib.colors as mcolors
import cartopy.crs as ccrs
import cartopy.feature as cfeature

# Uso seabornn para configurar o estilo dos grafico
sns.set(style="whitegrid")
```

[2]:
```
#carregar dados
df = pd.read_csv("teste_indicium_precificacao.csv")
```

[3]:
```
# Usei a media e o desvio padrão para eliminar alguns outliers dos dados, assim␣
 ↪garantindo resultados mais precisos
mean_price = df['price'].mean()
std_price = df['price'].std()

lower_limit = mean_price - 3 * std_price
upper_limit = mean_price + 3 * std_price

# Filtrar o dataset
df_filtrado = df[(df['price'] >= lower_limit) & (df['price'] <= upper_limit)]
```

[4]:
```
#analise geral da distribuição de preços
plt.figure(figsize=(12, 6))

sns.histplot(df_filtrado['price'], bins=50, color='skyblue', alpha=0.7)


median_price =df['price'].median()
plt.axvline(median_price, color='red', linestyle='--', label=f"Mediana:␣
 ↪${median_price:.2f}")
```

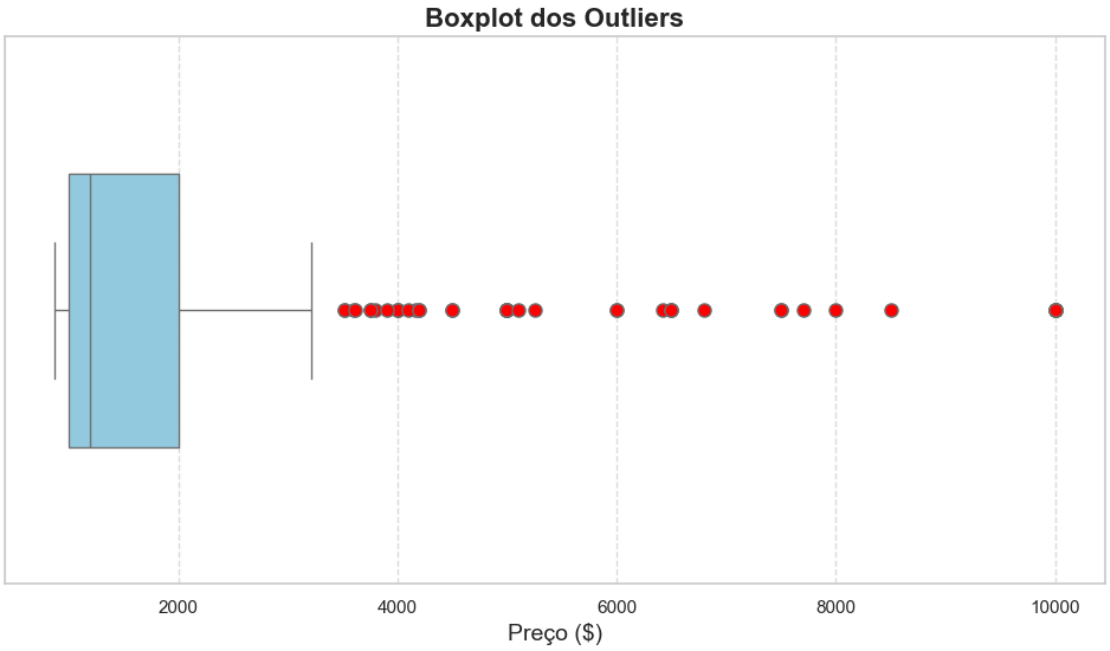
```
mean_price = df['price'].mean()
plt.axvline(mean_price, color='green', linestyle='--', label=f"Média:␣
 ↪${mean_price:.2f}")

# Configurações do gráfico
plt.title("Distribuição de Preços", fontsize=16)
plt.xlabel("Preço ($)", fontsize=12)
plt.ylabel("Frequência", fontsize=12)
plt.legend(fontsize=12)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.xlim(0, 1000)
plt.show()
```



[ ]:
```
'''
Podemos perceber que a grande maioria dos preços se concentra em volta de $0␣
 ↪ate $400.
Foram identificados alguns valores atípicos muito altos, que poderiam distorcer␣
 ↪a análise.
'''
```

[5]:
```
#boxplot dos valores que saem do padrão, ou seja aqueles acima de $1000
outliers = df[(df['price'] < lower_limit) | (df['price'] > upper_limit)]



plt.figure(figsize=(12, 6))
```

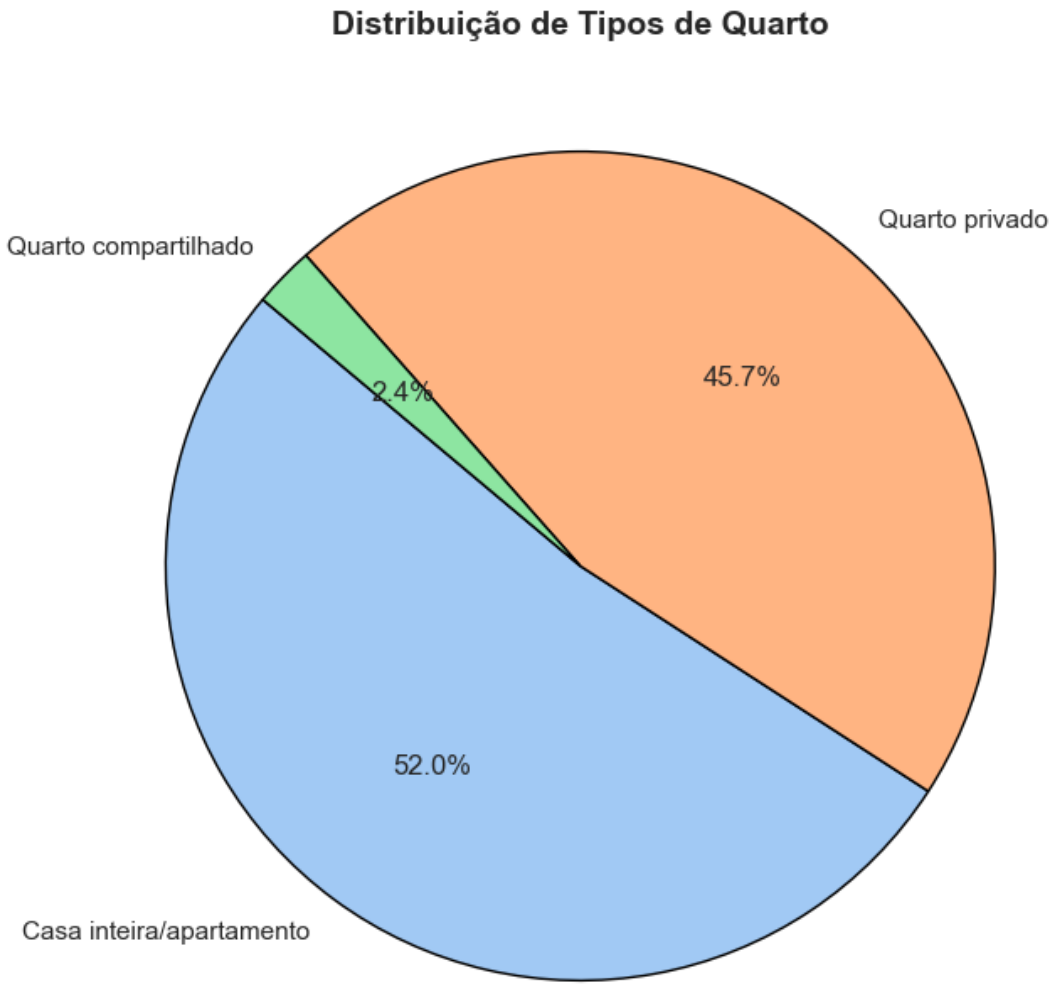
```
# Criar o boxplot
sns.boxplot(
    x=outliers['price'],
    color="skyblue",
    width=0.5,
    flierprops={"marker": "o", "markerfacecolor": "red", "markersize": 8,␣
 ↪"linestyle": "none"}
)

# Configurações do título e eixos
plt.title("Boxplot dos Outliers", fontsize=16, fontweight="bold")
plt.xlabel("Preço ($)", fontsize=14)
plt.grid(axis="x", linestyle="--", alpha=0.7)

plt.show()
```



[ ]:
```
'''
Valores superiores a $1000 foram considerados fora do padrão e removidos para␣
 ↪evitar que influenciassem negativamente na média e nos gráficos.
O boxplot mostrou que esses outliers são poucos, mas destoam bastante do␣
 ↪restante dos dados.
'''
```

[6]:
```
#traduzir os tipos de quarto para portugues

df['room_type'] = df['room_type'].replace({
    'Entire home/apt': 'Casa inteira/apartamento',
    'Private room': 'Quarto privado',
    'Shared room': 'Quarto compartilhado'
})
```

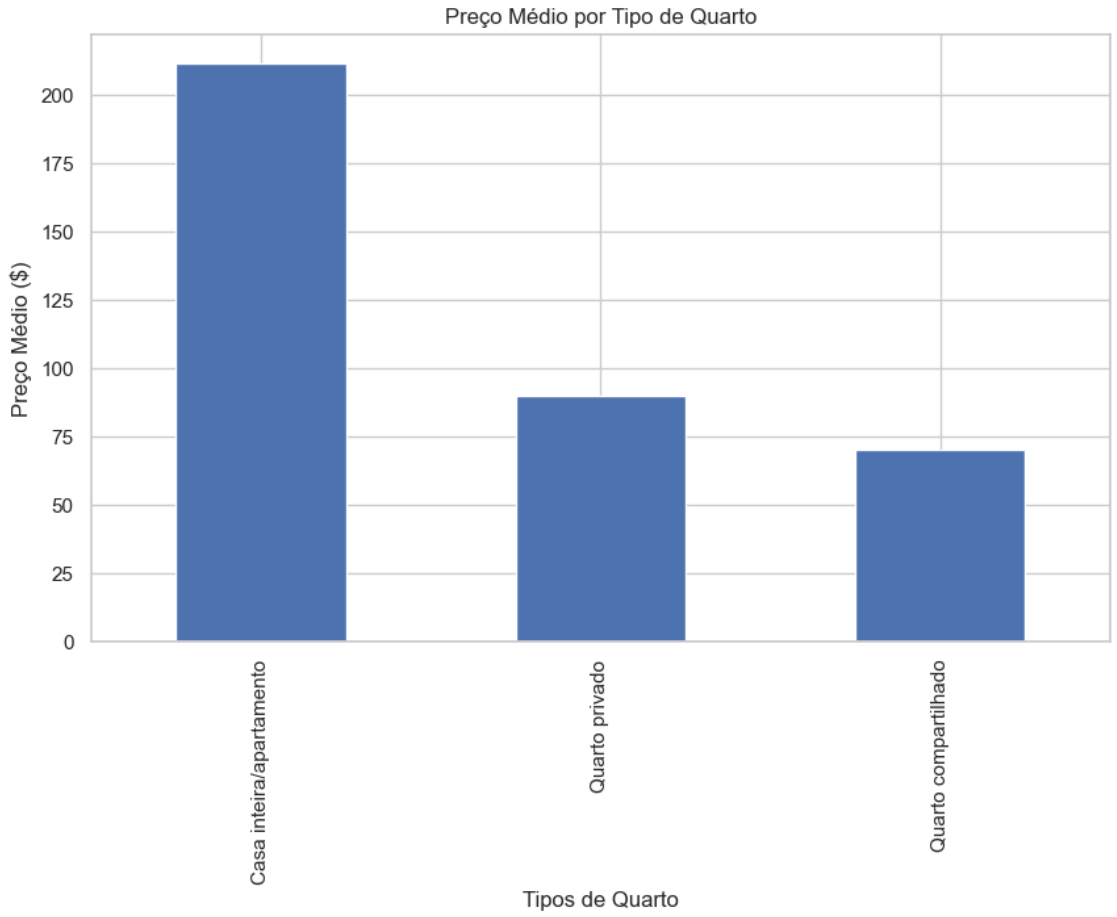
[8]:
```
#analise da frequencia por tipo de quarto
room_counts = df['room_type'].value_counts()

plt.figure(figsize=(8, 8))
plt.pie(
    room_counts, labels=room_counts.index, autopct='%1.1f%%',
    startangle=140, colors=sns.color_palette("pastel"), wedgeprops={'edgecolor':
 ↪ 'black'}
)

plt.title("Distribuição de Tipos de Quarto", fontsize=14, fontweight="bold")
plt.show()
```



[ ]:
```
'''
Os tipos de acomodações mais frequentes foram "Casa inteira/apartamento" e␣
↪"Quarto privado".
'''
```

[9]:
```
#analise do preço medio por tipo de quarto


df.groupby('room_type')['price'].mean().sort_values(ascending=False).plot(
    kind='bar', figsize=(10, 6))
plt.title("Preço Médio por Tipo de Quarto")
plt.ylabel("Preço Médio ($)")
plt.xlabel("Tipos de Quarto")
```

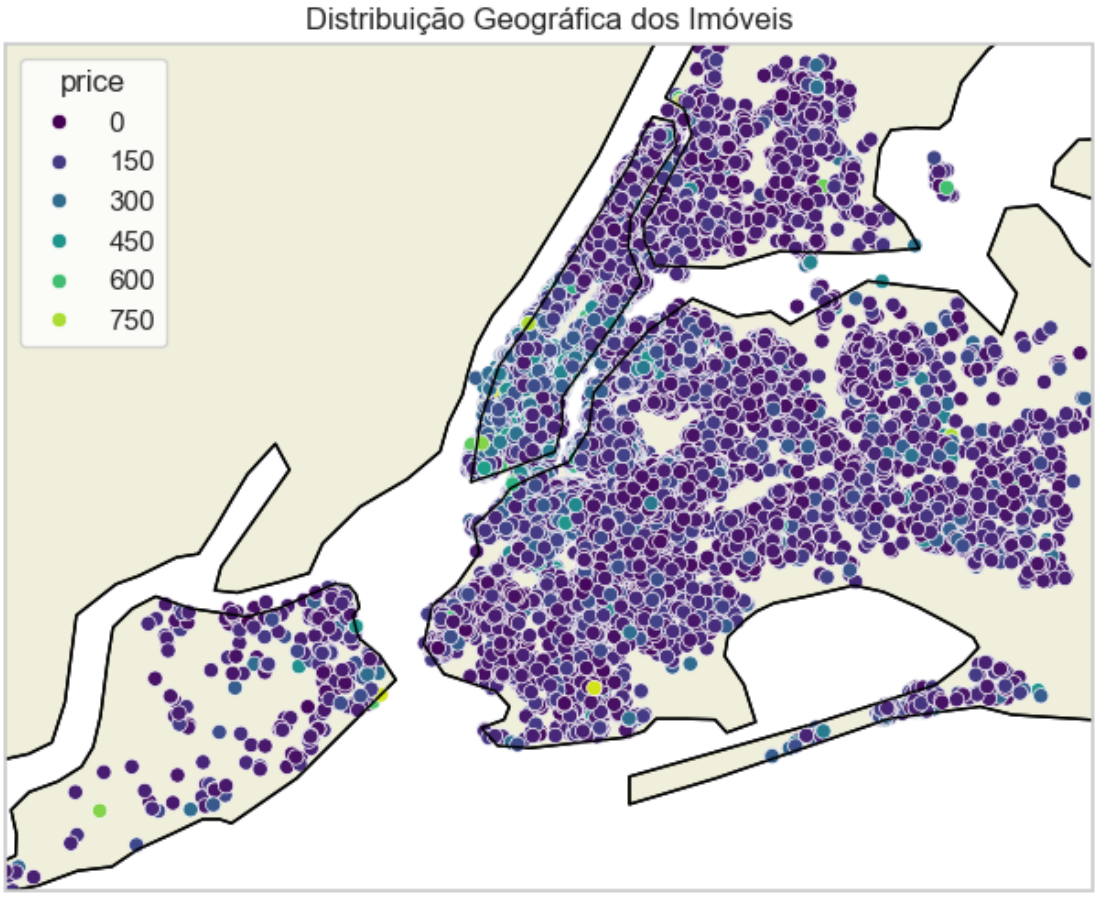
```
plt.show()
```



[ ]:
```
'''
O preço médio dos imóveis variou bastante de acordo com o tipo de quarto, sendo␣
↪as casas/apartamentos inteiros os mais caros.
'''
```

[10]:
```
#Analise do preço pela distribuição geografica
plt.figure(figsize=(10, 6))

# Ver os valores mínimos e máximos de latitude e longitude
min_longitude = df['longitude'].min()
max_longitude = df['longitude'].max()
min_latitude = df['latitude'].min()
max_latitude = df['latitude'].max()

# Usando Cartopy para adicionar o mapa
ax = plt.axes(projection=ccrs.PlateCarree())  # Usando a projeção PlateCarree
```

```
ax.set_extent([min_longitude, max_longitude, min_latitude, max_latitude],␣
 ↪crs=ccrs.PlateCarree())

# Adicionando características do mapa (litoral, países, etc.)
ax.add_feature(cfeature.LAND, edgecolor='black')
ax.add_feature(cfeature.COASTLINE)
ax.add_feature(cfeature.BORDERS)



scatter = sns.scatterplot(data=df_filtrado, x='longitude', y='latitude',␣
 ↪hue='price', palette='viridis', legend=True)

# Adicionando título e rótulos aos eixos
plt.title("Distribuição Geográfica dos Imóveis")
plt.xlabel("Longitude")
plt.ylabel("Latitude")



plt.show()
```



[ ]:
```
'''
A análise espacial revelou que certas áreas concentram imóveis com faixas de␣
 ↪preço semelhantes.
Isso reforça que a localização exerce um impacto direto no valor das␣
 ↪acomodações.
'''
```

[ ]:
```
#Hipótese 1: Imóveis em Manhattan tendem a ser mais caros.
```

[13]:
```
df.groupby('bairro_group')['price'].mean().sort_values(ascending=False)
```

[13]:
```
bairro_group
Manhattan        196.875814
Brooklyn         124.381983
Staten Island    114.812332
Queens            99.517649
Bronx             87.496792
Name: price, dtype: float64
```

[ ]:
```
'''
Confirmado.
O preço médio em Manhattan ficou em $196.88, o mais alto dentre todas as␣
 ↪regiões analisadas.
Já no Bronx, por exemplo, a média foi de apenas $87.49.
'''
```

[ ]:
```
#Hipótese 2: Tipos de quartos inteiros (Entire home/apt) têm maior demanda e␣
 ↪preço.
```

[14]:
```
df.groupby('room_type')['numero_de_reviews'].mean().sort_values(ascending=False)
```

[14]:
```
room_type
Quarto privado                24.113639
Casa inteira/apartamento      22.842418
Quarto compartilhado          16.600000
Name: numero_de_reviews, dtype: float64
```

[15]:
```
df.groupby('room_type')['reviews_por_mes'].mean().sort_values(ascending=False)
```

[15]:
```
room_type
Quarto compartilhado          1.471726
Quarto privado                1.445279
Casa inteira/apartamento      1.306578
Name: reviews_por_mes, dtype: float64
```

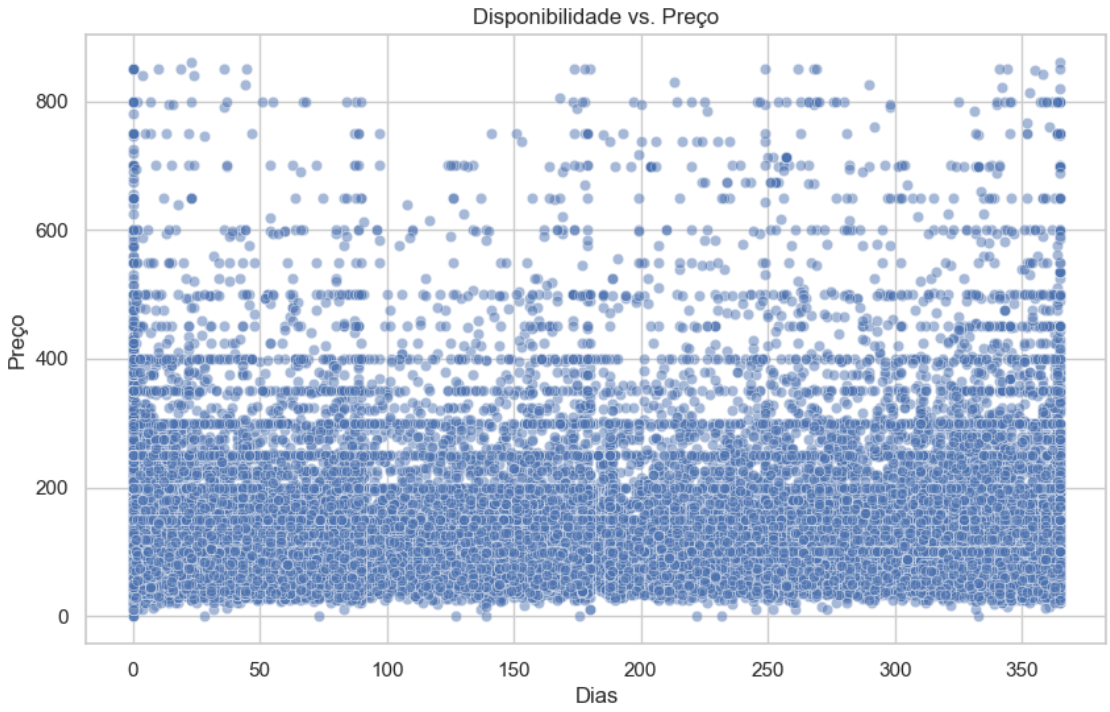
[16]:
```
df.groupby('room_type')['disponibilidade_365'].mean().
 ↪sort_values(ascending=True)
```

[16]:
```
room_type
Quarto privado               111.192564
Casa inteira/apartamento     111.920304
Quarto compartilhado         162.000862
Name: disponibilidade_365, dtype: float64
```

[ ]:
```
'''
Falso.
Podemos perceber que na verdade quarto privado tem a maior quantidade de notas␣
 ↪totais, a menor disponibilidade e a segunda maior quantidade de reviwes
por mês.
isso indica que quarto privado é o tipo com maior busca.
'''
```

[ ]:
```
#Hipótese 3: Anúncios com alta disponibilidade (acima de 300 dias/ano) são mais␣
 ↪baratos devido à alta competição.
```

[23]:
```
#Analise de como a disponibilidade pode afetar o preço

plt.figure(figsize=(10, 6))
sns.scatterplot(data=df_filtrado, x='disponibilidade_365', y='price', alpha=0.5)
plt.title("Disponibilidade vs. Preço")
plt.xlabel("Dias")
plt.ylabel("Preço")
plt.show()
```



[ ]:
```
'''
Falso
Não houve relação clara entre disponibilidade e preço.
O número de dias disponíveis ao longo do ano não parece impactar␣
 ↪significativamente os valores cobrados.
'''
```

[ ]:
```
#Hipótese 4:O preço impacta na quantidade de avaliações ?
```

[11]:
```
# Criando faixas de reviews
bins = [0, 100, 250, 400, float("inf")]
labels = ["Baixo", "Médio", "Alto", "Muito Alto"]

df["categoria_reviews"] = pd.cut(df["numero_de_reviews"], bins=bins,␣
 ↪labels=labels)

df.groupby('categoria_reviews')['price'].mean().sort_values(ascending=True)
```

```
C:\Users\gusta\AppData\Local\Temp\ipykernel_4680\80218672.py:7: FutureWarning:
The default of observed=False is deprecated and will be changed to True in a
future version of pandas. Pass observed=False to retain current behavior or
observed=True to adopt the future default and silence this warning.
  df.groupby('categoria_reviews')['price'].mean().sort_values(ascending=True)
```

[11]:
```
categoria_reviews
Muito Alto     88.871795
Alto          113.610619
Médio         128.009524
Baixo         143.646581
Name: price, dtype: float64
```

[ ]:
```
'''
Confirmado.
A análise indica que imóveis mais caros tendem a ter menos avaliações, enquanto␣
 ↪os mais baratos acumulam mais reviews ao longo do tempo.
Isso tambem pode ser indicativo da quantidade de clientes recebidos
'''
```