In [1]:

```
#Análise Exploratória de Dados Contábeis
#Tratamento de Dados Ausentes e Outliers
```

In [1]:

```
!pip install -q -U watermark
```

In [3]:

```
#Bibliotecas utilizadas
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

In [4]:

```
df=pd.read_csv('dataset.csv')
```

In [6]:

```
df.info('dataset.csv')
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1200 entries, 0 to 1199
Data columns (total 11 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   id                 1200 non-null   int64  
 1   data_lancamento    1200 non-null   object 
 2   conta_debito       1200 non-null   object 
 3   conta_credito      1200 non-null   object 
 4   valor              1200 non-null   float64
 5   documento          1078 non-null   object 
 6   natureza_operacao  1080 non-null   object 
 7   centro_custo       1200 non-null   object 
 8   impostos           1020 non-null   float64
 9   moeda              947 non-null    object 
 10  taxa_conversao     982 non-null    float64
dtypes: float64(3), int64(1), object(7)
memory usage: 103.3+ KB
```

In [8]:

```
df.head()
```

Out[8]:

	id	data_lancamento	conta_debito	conta_credito	valor	documento	natureza_operacao
0	1	2022-02-27	DWAVRL	CIOVQ6	5533.25	DOCPAXGQ	OP7JDVK
1	2	2022-05-11	D8TF53	CV9Y0V	7180.37	DOCBXZXG	OPXSY64
2	3	2020-03-23	D0TZCE	CELQSH	6067.36	DOCF5ITC	OPTDE9B
3	4	2021-06-14	DOGLK7	CDFEMS	5494.34	DOCZRS1U	NaN
4	5	2022-11-13	DHL0I5	CRU97G	4294.18	NaN	OP62LG1

In [9]:

```
df.shape
```

Out[9]:

```
(1200, 11)
```

In [10]:
```
df.columns
```

Out[10]:
```
Index(['id', 'data_lancamento', 'conta_debito', 'conta_credito', 'valor',
       'documento', 'natureza_operacao', 'centro_custo', 'impostos', 'moed
a',
       'taxa_conversao'],
      dtype='object')
```

In [11]:
```
df.sample(10)
```

Out[11]:

	id	data_lancamento	conta_debito	conta_credito	valor	documento	natureza_op
769	770	2021-12-27	DCK9N5	CQNNF0	7939.61	DOCD4Q88	OP
855	856	2021-12-25	DZ56WN	CP2TYA	7133.51	DOCUNAYO	
618	619	2022-08-14	DJF4GE	CNMA97	4032.84	DOCFKGI4	OPV
283	284	2020-11-18	DZPTO6	CBKS6K	231.04	DOCQOHPL	OP
48	49	2020-01-03	DUMUUK	CQKYIZ	3222.74	DOCDKGCA	OPN
676	677	2022-06-01	D21EY5	CNTA33	7290.79	DOC14QIH	OP
494	495	2022-10-18	DWFFUU	CJKOSO	5565.57	DOC674LH	OP(
505	506	2020-03-13	DJ608L	CH0W8U	9249.17	DOCOFDIX	OP
1061	1062	2021-01-19	D7O3S8	CXJEEY	4244.50	DOCY4C4P	OP
775	776	2023-07-28	DM83W5	CWSFQ5	3011.36	DOC24UL9	OF

In [12]:
```
df.describe(include = object)
```

Out[12]:

	data_lancamento	conta_debito	conta_credito	documento	natureza_operacao	centr
count	1200	1200	1200	1078	1080	
unique	808	1200	1197	1078	1080	
top	2023-07-18	DWAVRL	?	DOCPAXGQ	OP7JDVK	C
freq	5	1	4	1	1	

In [13]:
```
df.describe()
```

Out[13]:

	id	valor	impostos	taxa_conversao
count	1200.000000	1200.000000	1020.000000	982.000000
mean	600.500000	10094.975148	604.264546	2.601499
std	346.554469	25595.942955	1116.015868	0.853906
min	1.000000	105.410000	154.263980	1.248029
25%	300.750000	2631.245000	326.499880	2.135300
50%	600.500000	5092.510000	430.155339	2.568117
75%	900.250000	7881.407500	444.132520	3.475606
max	1200.000000	187297.686041	6779.970522	3.523287

In [14]:
```
duplicatas=df.duplicated()
```

In [15]:
```
df[duplicatas]
```

Out[15]:

id	data_lancamento	conta_debito	conta_credito	valor	documento	natureza_operacao	ce

In [16]:
```
print(df.duplicated())
```

```
0       False
1       False
2       False
3       False
4       False
        ...  
1195    False
1196    False
1197    False
1198    False
1199    False
Length: 1200, dtype: bool
```

In [17]:
```
df.isna().any()
```

Out[17]:
```
id                   False
data_lancamento      False
conta_debito         False
conta_credito        False
valor                False
documento             True
natureza_operacao     True
centro_custo         False
impostos              True
moeda                 True
taxa_conversao        True
dtype: bool
```

In [18]:
```
df.isna().sum()
```

Out[18]:
```
id                     0
data_lancamento        0
conta_debito           0
conta_credito          0
valor                  0
documento            122
natureza_operacao    120
centro_custo           0
impostos             180
moeda                253
taxa_conversao       218
dtype: int64
```

In [19]:

```
total_valores_ausentes=df.isna().sum()
total_linha=len(df)
proporcao_valores_ausentes=total_valores_ausentes/total_linha
print(proporcao_valores_ausentes)
```

```
id                    0.000000
data_lancamento       0.000000
conta_debito          0.000000
conta_credito         0.000000
valor                 0.000000
documento             0.101667
natureza_operacao     0.100000
centro_custo          0.000000
impostos              0.150000
moeda                 0.210833
taxa_conversao        0.181667
dtype: float64
```

In [20]:

```
Valores=['?']
df.isin(Valores)
```

Out[20]:

	id	data_lancamento	conta_debito	conta_credito	valor	documento	natureza_opera
0	False	False	False	False	False	False	F
1	False	False	False	False	False	False	F
2	False	False	False	False	False	False	F
3	False	False	False	False	False	False	F
4	False	False	False	False	False	False	F
...	...	...	...	...	...	...	
1195	False	False	False	False	False	False	F
1196	False	False	False	False	False	False	F
1197	False	False	False	False	False	False	F
1198	False	False	False	False	False	False	F
1199	False	False	False	False	False	False	F

1200 rows × 11 columns

In [21]:

```
Valores=[0]
df.isin(Valores)
```

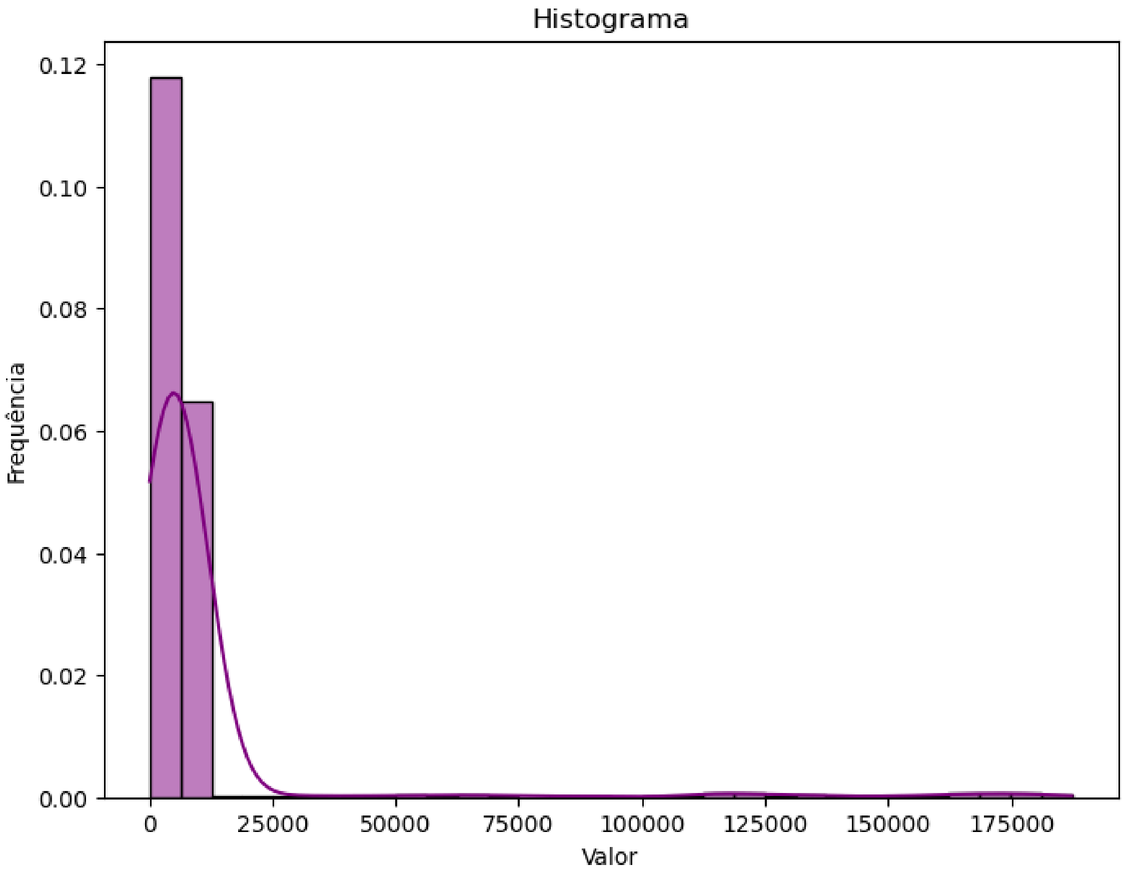
Out[21]:

```
         id  data_lancamento  conta_debito  conta_credito  valor  documento  natureza_opera
   0  False            False         False          False  False      False               F
   1  False            False         False          False  False      False               F
   2  False            False         False          False  False      False               F
   3  False            False         False          False  False      False               F
   4  False            False         False          False  False      False               F
 ...    ...              ...           ...            ...    ...        ...
1195  False            False         False          False  False      False               F
1196  False            False         False          False  False      False               F
1197  False            False         False          False  False      False               F
1198  False            False         False          False  False      False               F
1199  False            False         False          False  False      False               F

1200 rows × 11 columns
```

In [22]:
```
plt.figure(figsize = (8,6))
sns.histplot(df['valor'], bins=30, kde=True, stat='frequency', color='purpl
plt.title('Histograma')
plt.xlabel('Valor')
plt.ylabel('Frequência')
plt.show
```

Out[22]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```

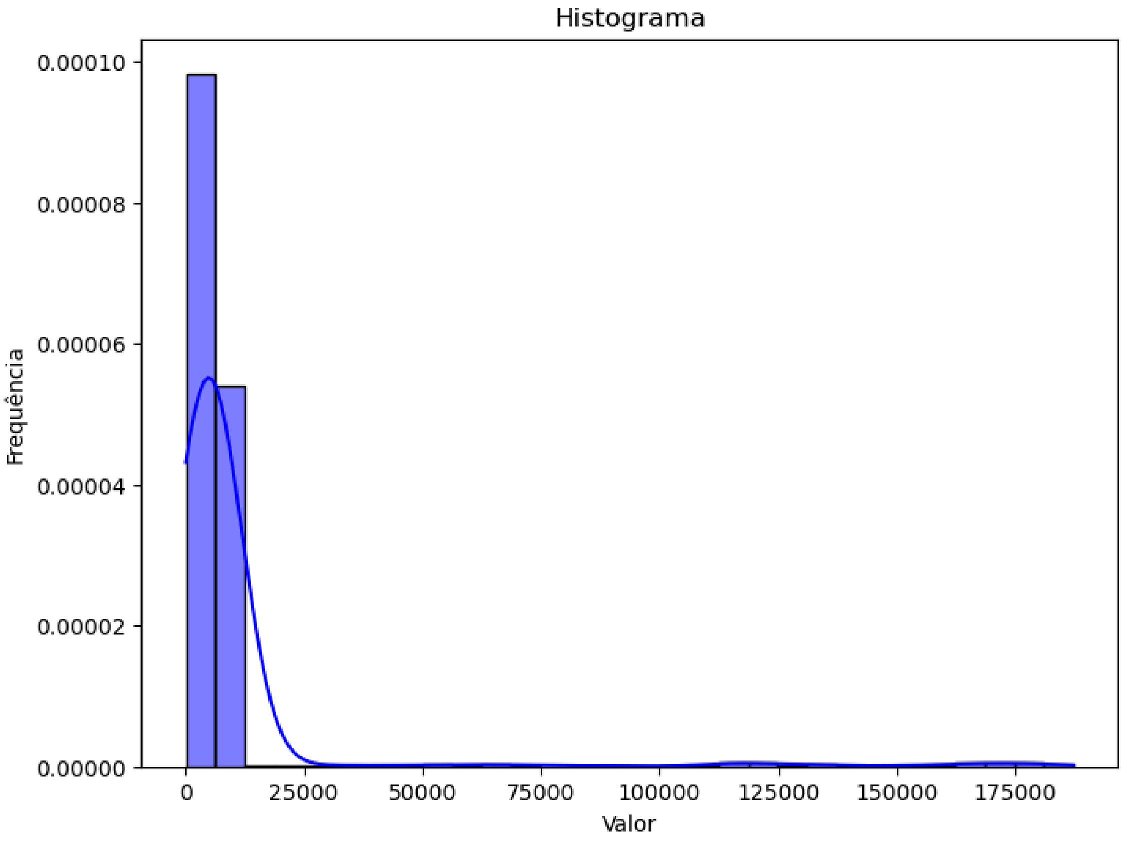


In [23]:

```
plt.figure(figsize = (8,6))
sns.histplot(df['valor'], bins=30, kde=True, stat='density', color='blue')
plt.title('Histograma')
plt.xlabel('Valor')
plt.ylabel('Frequência')
plt.show
```

Out[23]:

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

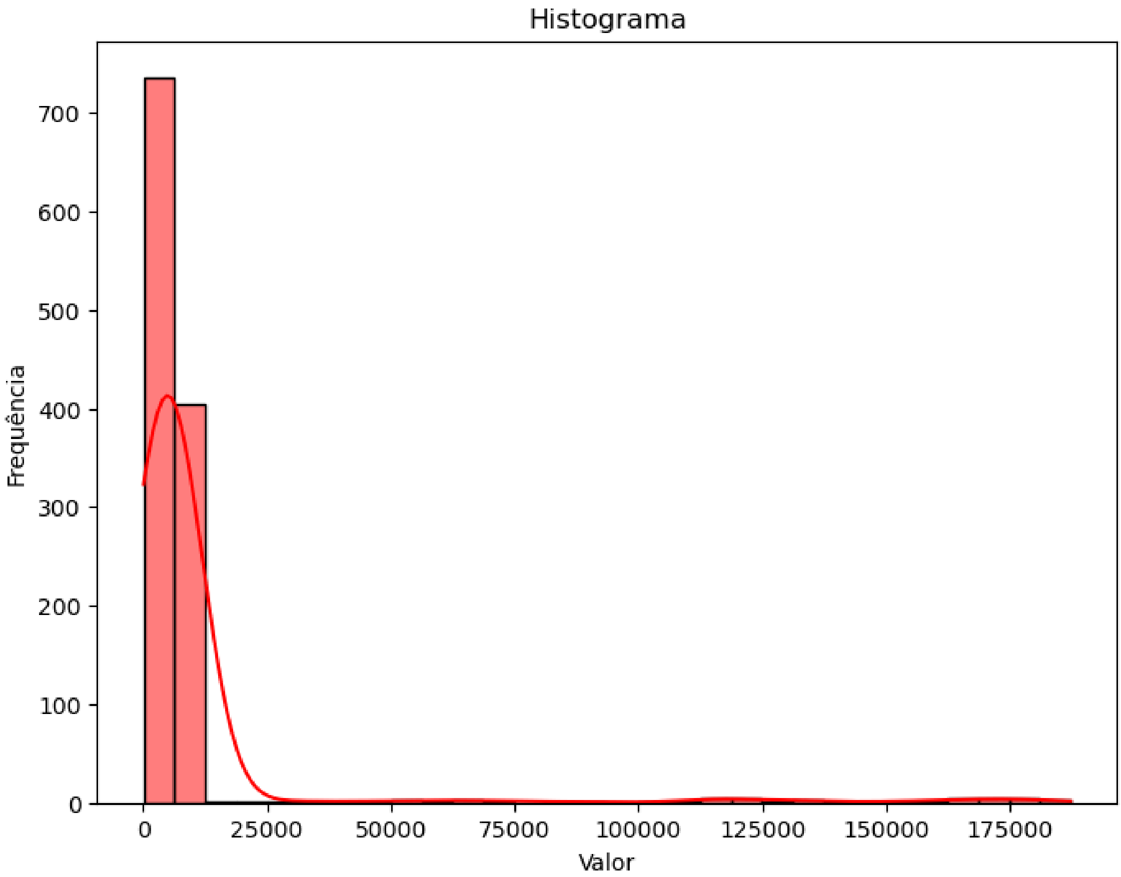


In [24]:

```
plt.figure(figsize = (8,6))
sns.histplot(df['valor'], bins=30, kde=True, stat='count', color='red')
plt.title('Histograma')
plt.xlabel('Valor')
plt.ylabel('Frequência')
plt.show
```

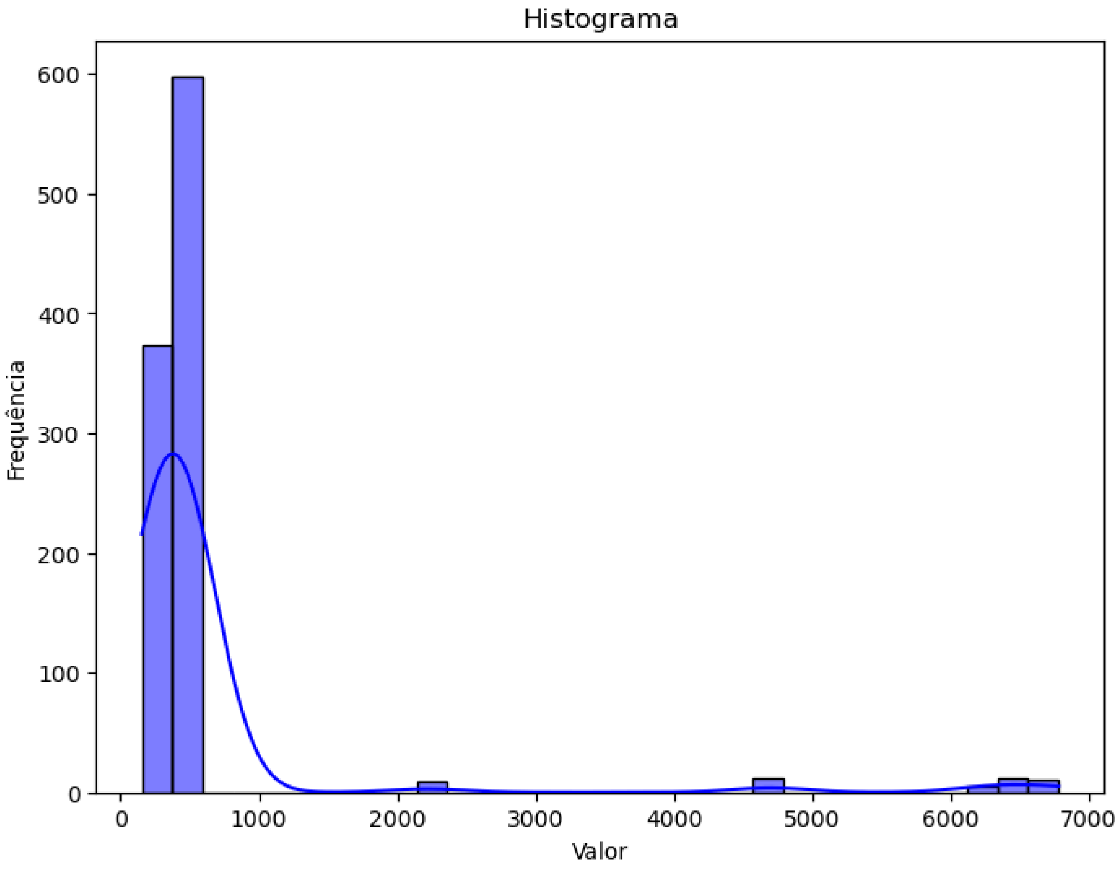
Out[24]:

```
<function matplotlib.pyplot.show(close=None, block=None)>
```



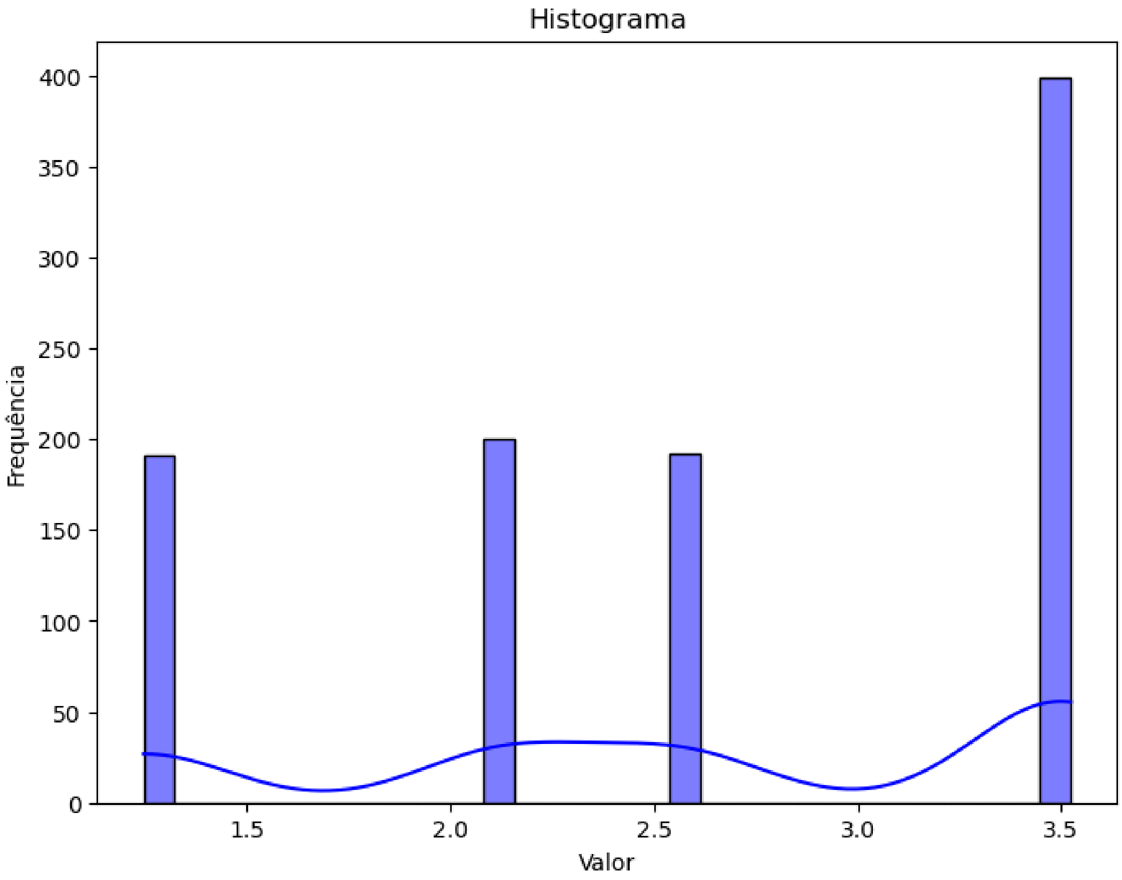
```
In [25]: plt.figure(figsize = (8,6))
         sns.histplot(df['impostos'], bins=30, kde=True, stat='count', color='blue')
         plt.title('Histograma')
         plt.xlabel('Valor')
         plt.ylabel('Frequência')
         plt.show

Out[25]: <function matplotlib.pyplot.show(close=None, block=None)>
```



```
In [26]: plt.figure(figsize = (8,6))
         sns.histplot(df['taxa_conversao'], bins=30, kde=True, stat='count', color='
         plt.title('Histograma')
         plt.xlabel('Valor')
         plt.ylabel('Frequência')
         plt.show
```

```
Out[26]: <function matplotlib.pyplot.show(close=None, block=None)>
```



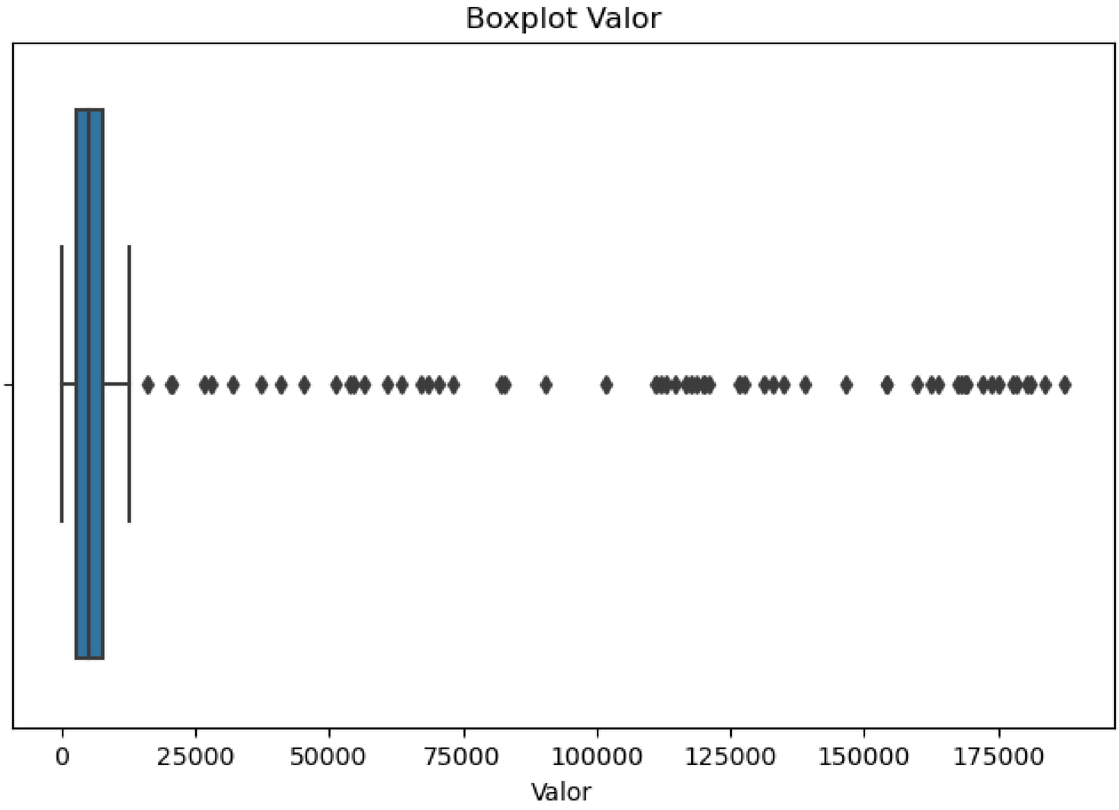
```
In [27]: from scipy.stats import skew
```

```
In [28]: skew(df['valor'])
```

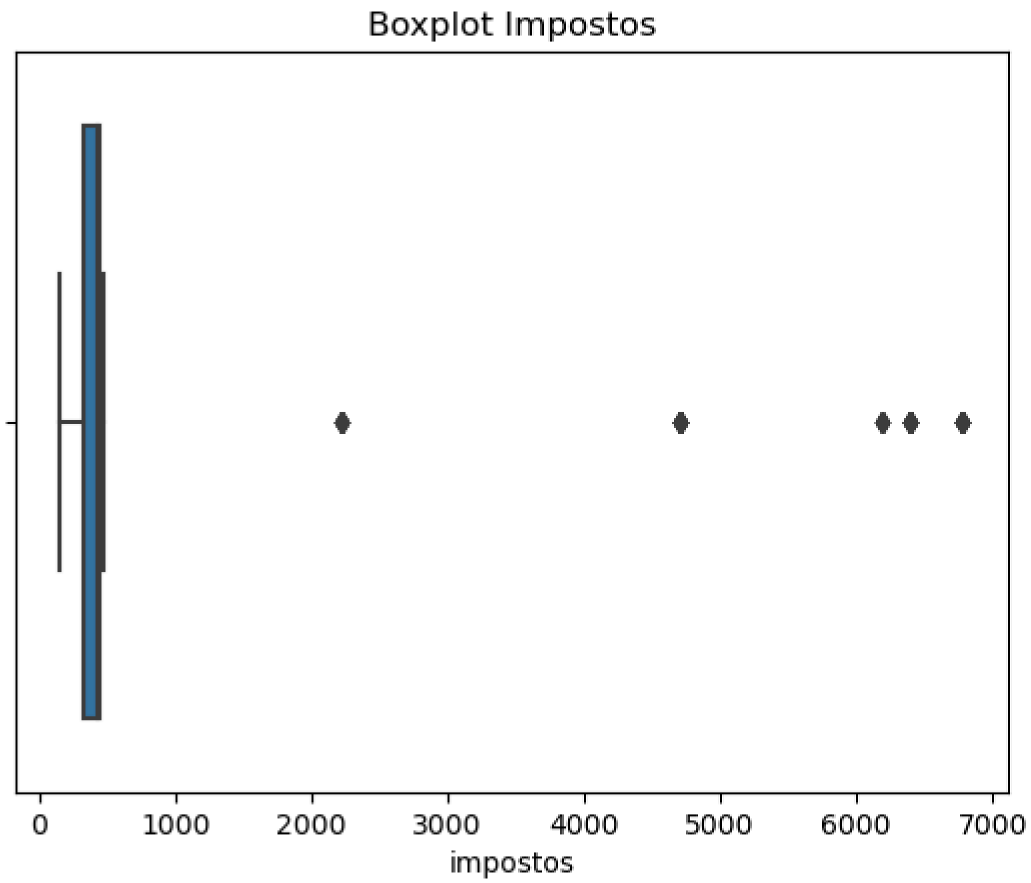
```
Out[28]: 5.207837830710742
```

In [31]:

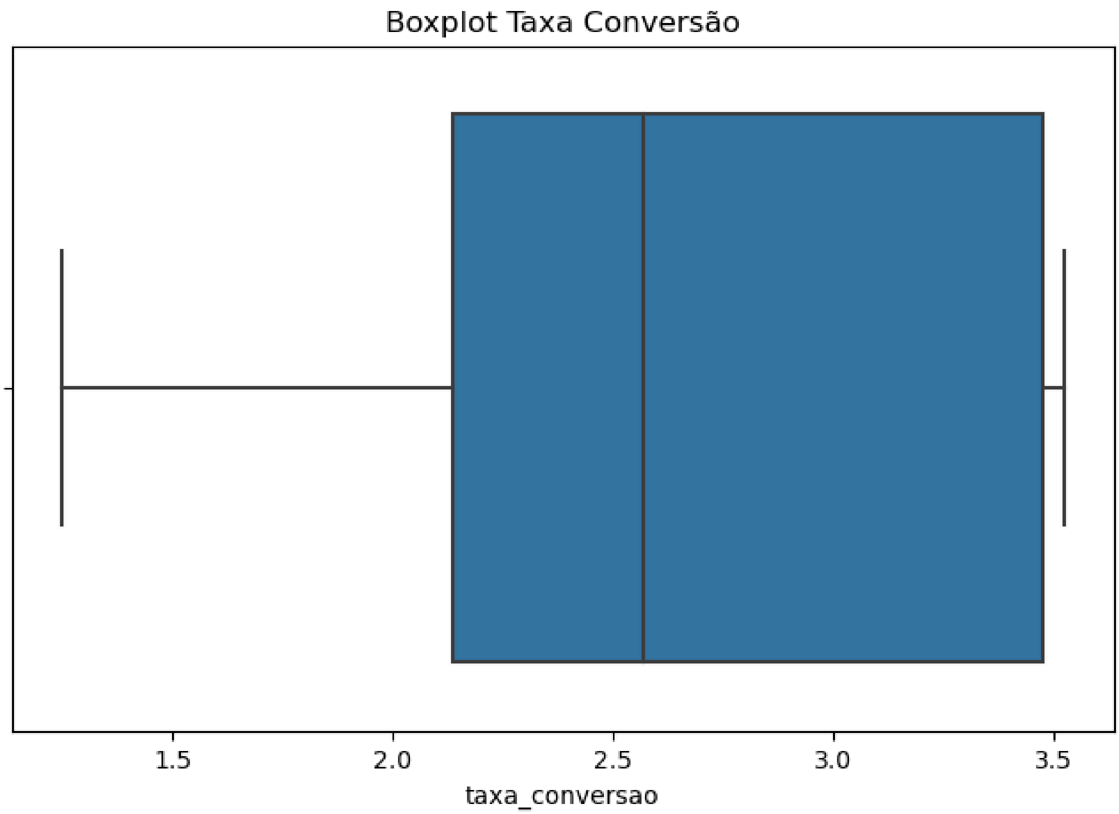
```
plt.figure(figsize = (8, 5))
sns.boxplot(x = df['valor'])
plt.title('Boxplot Valor')
plt.xlabel('Valor')
plt.show()
```



In [32]:
```
sns.boxplot(x = df['impostos'])
plt.title('Boxplot Impostos')
plt.xlabel('impostos')
plt.show()
```



In [33]:
```
plt.figure(figsize = (8, 5))
sns.boxplot(x = df['taxa_conversao'])
plt.title('Boxplot Taxa Conversão')
plt.xlabel('taxa_conversao')
plt.show()
```



In [34]:
```
df.isna().sum()
```

Out[34]:
```
id                     0
data_lancamento        0
conta_debito           0
conta_credito          0
valor                  0
documento            122
natureza_operacao    120
centro_custo           0
impostos             180
moeda                253
taxa_conversao       218
dtype: int64
```

In [35]:
```
df['impostos'].mean()
```

Out[35]:
```
604.264545965864
```

In [36]:
```
df['impostos'].median()
```

Out[36]:
```
430.1553391717098
```

In [37]:
```
df['impostos'].fillna(df['impostos'].median(), inplace=True)
```

```
In [38]:  df['impostos'].isna().sum()

Out[38]:  0

In [39]:  df['taxa_conversao'].mean()

Out[39]:  2.601498735918867

In [40]:  df['taxa_conversao'].median()

Out[40]:  2.5681167953894297

In [41]:  df['taxa_conversao'].fillna(df['taxa_conversao'].mean(), inplace=True)

In [42]:  df['taxa_conversao'].isna().sum()

Out[42]:  0

In [43]:  df['moeda'].mode()[0]

Out[43]:  'BRL'

In [44]:  df['moeda'].fillna(df['moeda'].mode()[0], inplace=True)

In [45]:  df['moeda'].isna().sum()

Out[45]:  0

In [46]:  df['documento'].fillna('Outro', inplace=True)

In [47]:  df['documento'].isna().sum()

Out[47]:  0

In [49]:  df['natureza_operacao'].fillna(method = 'bfill', inplace=True)

In [51]:  df['natureza_operacao'].isna().sum()

Out[51]:  0
```

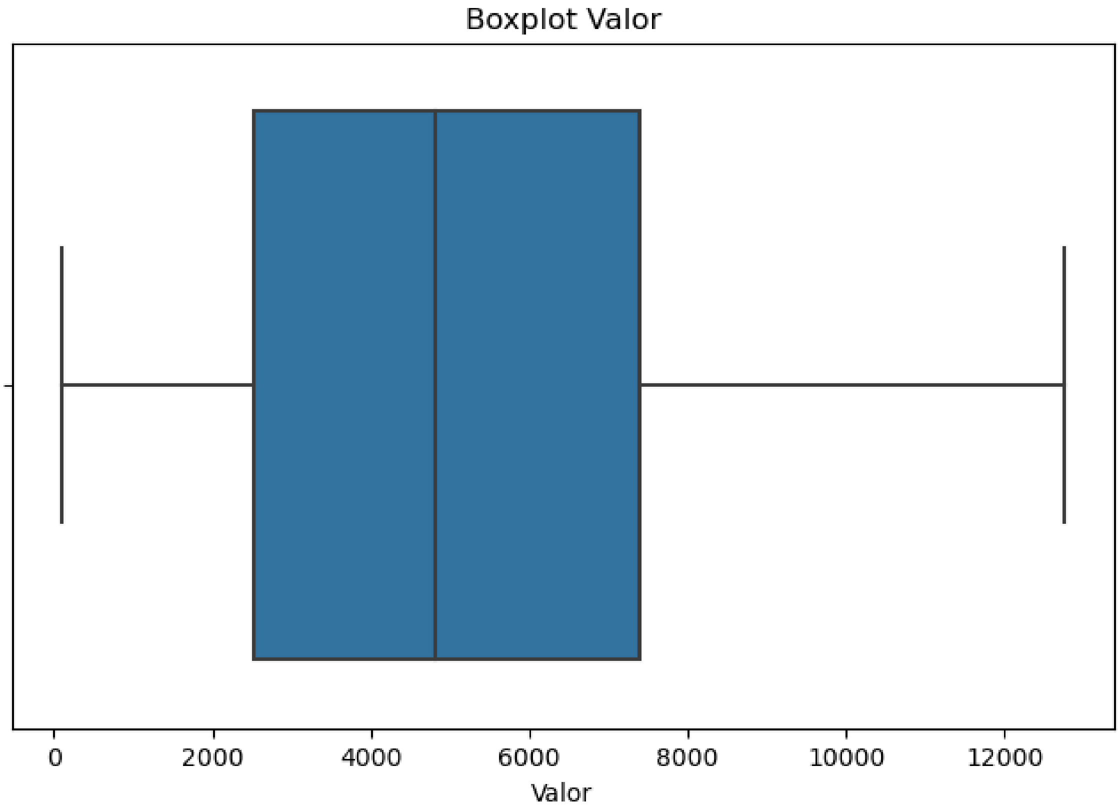
In [52]:
```
df.isna().sum()
```

Out[52]:
```
id                   0
data_lancamento      0
conta_debito         0
conta_credito        0
valor                0
documento            0
natureza_operacao    0
centro_custo         0
impostos             0
moeda                0
taxa_conversao       0
dtype: int64
```

In [53]:
```
Q1 = df['valor'].quantile(0.25)
Q3 = df['valor'].quantile(0.75)
IQR = Q3 - Q1
Limite_inferior = Q1 - 1.5 * IQR
Limite_superior = Q3 + 1.5 * IQR
```

In [55]:
```
df_sem_outlier = df[~((df['valor']< Limite_inferior) | (df['valor'] > Limit
```

In [56]:
```
plt.figure(figsize = (8, 5))
sns.boxplot(x = df_sem_outlier['valor'])
plt.title('Boxplot Valor')
plt.xlabel('Valor')
plt.show()
```



In [57]:

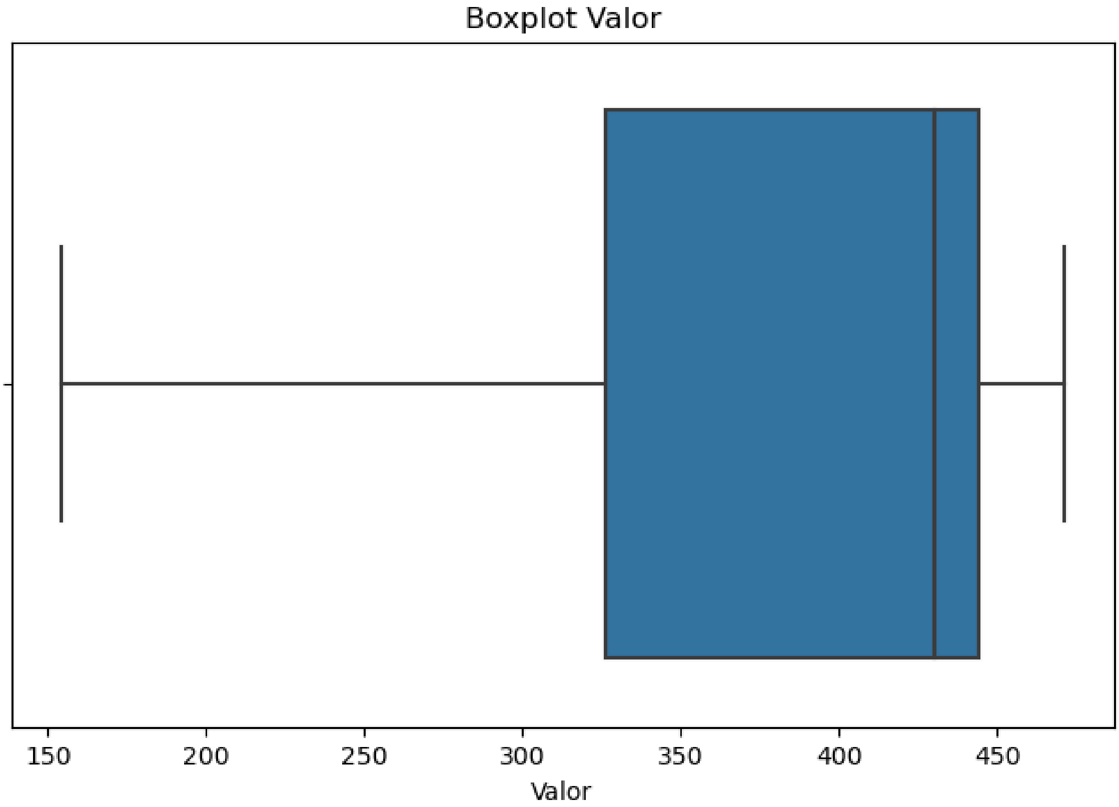
```
Q1 = df_sem_outlier['impostos'].quantile(0.25)
Q3 = df_sem_outlier['impostos'].quantile(0.75)
IQR = Q3 - Q1
Limite_inferior = Q1 - 1.5 * IQR
Limite_superior = Q3 + 1.5 * IQR
```

In [58]:

```
df_sem_outlier2 = df_sem_outlier[~((df_sem_outlier['impostos'] < Limite_inf
```

In [59]:

```
plt.figure(figsize = (8, 5))
sns.boxplot(x = df_sem_outlier2['impostos'])
plt.title('Boxplot Valor')
plt.xlabel('Valor')
plt.show()
```


Boxplot Valor
Valor