Relatório de Análise VIII

Identificando e Removendo Outliers

In [1]:
```
%matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
# configura a plotagem nas dimensões desejadas
plt.rc('figure', figsize=(14, 6))
```

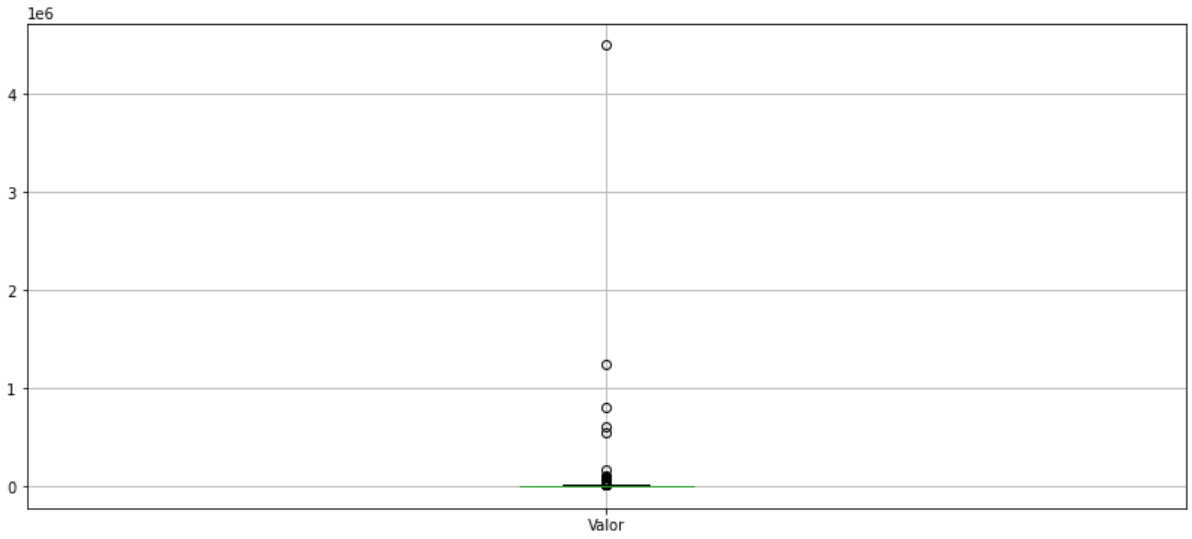
In [2]:
```
dados = pd.read_csv('../dados/aluguel_residencial_final.csv', sep=';')
```

Representação Box-Plot

Usando o Boxplot

In [3]:
```
dados.boxplot('Valor')
```

Out[3]: <AxesSubplot:>



- visualização é comprometida por haver dados muitos discrepantes

Fazendo uma seleção para verificar alguns dados discrepantes

In [4]:
```
dados[dados['Valor'] >= 500000]
```

Out[4]:

	Tipo	Bairro	Quartos	Vagas	Suites	Area	Valor	Condominio	IPTU
7629	Apartamento	Barra da Tijuca	1	1	0	65	600000.0	980.0	120.0
10636	Casa de Condomínio	Freguesia (Jacarepaguá)	4	2	3	163	800000.0	900.0	0.0
12661	Apartamento	Freguesia (Jacarepaguá)	2	2	1	150	550000.0	850.0	150.0
13846	Apartamento	Recreio dos Bandeirantes	3	2	1	167	1250000.0	1186.0	320.0
15520	Apartamento	Botafogo	4	1	1	300	4500000.0	1100.0	0.0

Criando uma Series

In [5]:
```
valor = dados['Valor']
```

Removendo Outliers

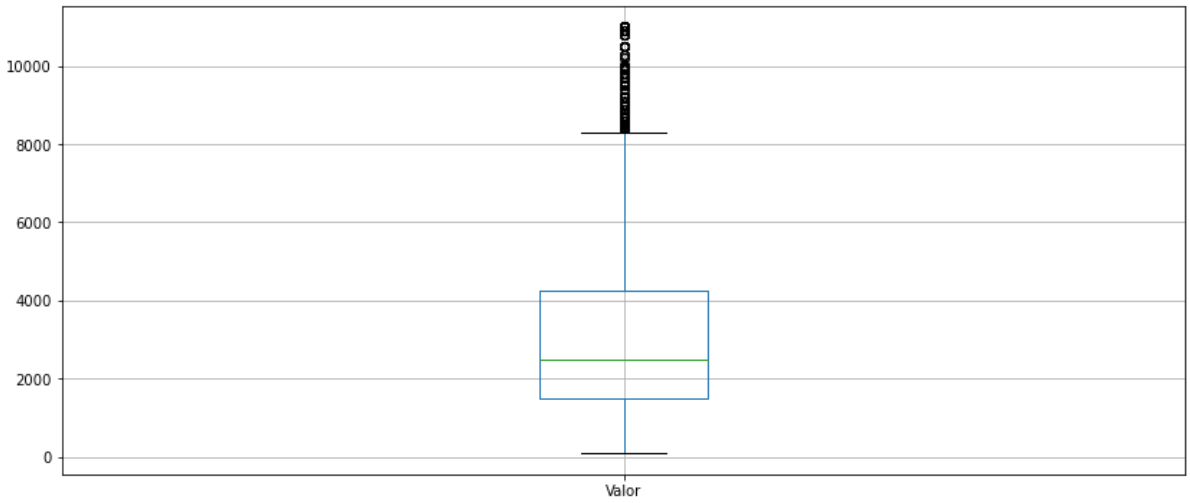
Observando o Modelo do Boxplot e Calculando os quartis

In [6]:
```
Q1 = valor.quantile(.25)
Q3 = valor.quantile(.75)
IIQ = Q3 - Q1
limite_inferior = Q1 - 1.5 * IIQ
limite_superior = Q3 + 1.5 * IIQ
```

Remeovendo os Outliers através de uma seleção

In [7]:
```
selecao = (valor >= limite_inferior) & (valor <= limite_superior)
dados_new = dados[selecao]
dados_new.boxplot('Valor')
```

Out[7]: <AxesSubplot:>



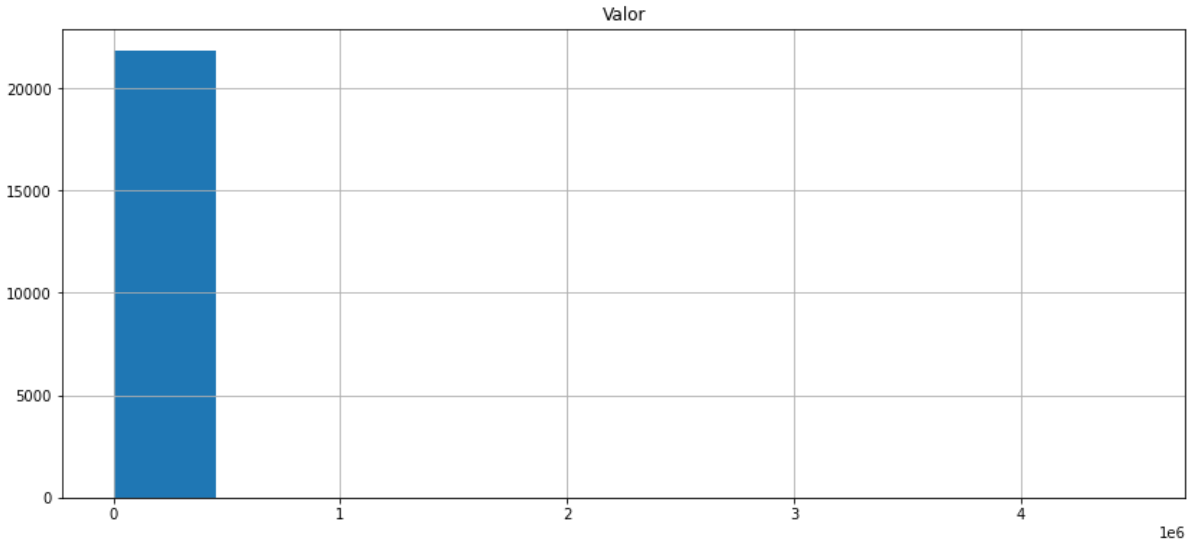
- O Boxplot ficou muito mais visível com a remoção dos outliers

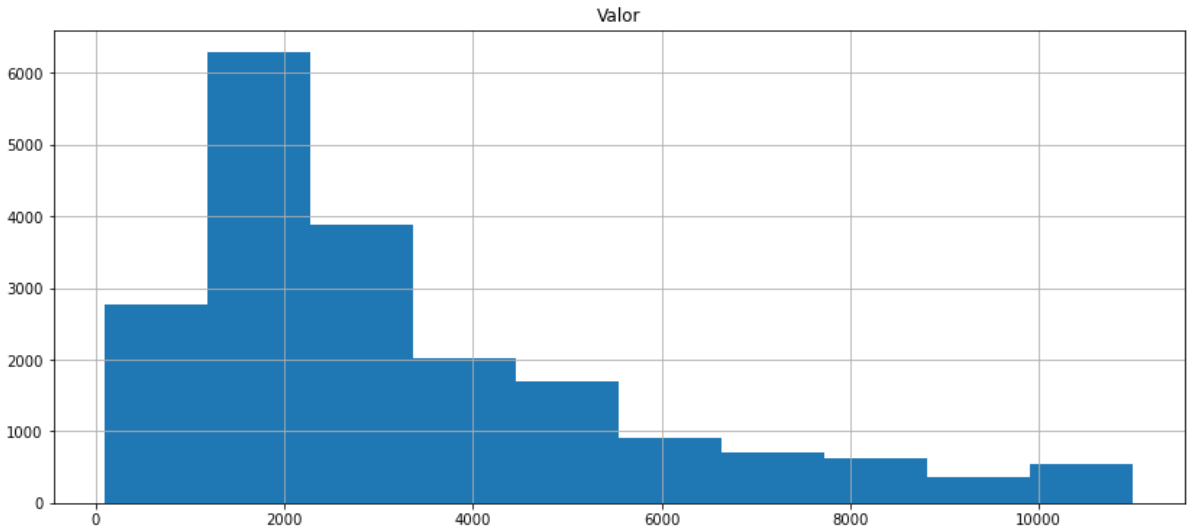
Comparando com histogramas

- o histograma mostra a distribuição da frequência dos dados
- observando as duas variáveis, é possível ver um comportamento melhor da segunda, após o tratamento, com a remoção dos outliers

```
In [8]: dados.hist('Valor')
        dados_new.hist('Valor')
```

```
Out[8]: array([[<AxesSubplot:title={'center':'Valor'}>]], dtype=object)
```





Exercício

Obtenha o conjunto de estatísticas representado na figura acima.
Para isso, utilize o arquivo aluguel_amostra.csv, e realize suas análises utilizando como variável alvo o Valor m2 (valor do metro quadrado).
Lembrando que Q1 representa o 1º quartil e Q3 o 3º quartil, selecione o item com a resposta correta (considere somente duas casas decimais):

In [9]:
```
data = pd.read_csv('../dados/aluguel_amostra.csv', sep=';')
Q1 = data['Valor m2'].quantile(.25)
Q3 = data['Valor m2'].quantile(.75)
IIQ = Q3 - Q1
limite_inferior = Q1 - 1.5 * IIQ
limite_superior = Q3 + 1.5 * IIQ
```

In [10]:
```
data['Valor m2'].describe().round(2)
```

Out[10]:
```
count    10000.00
mean        37.08
std        175.30
min          2.78
25%         21.25
50%         30.00
75%         42.31
max      15000.00
Name: Valor m2, dtype: float64
```

Resposta

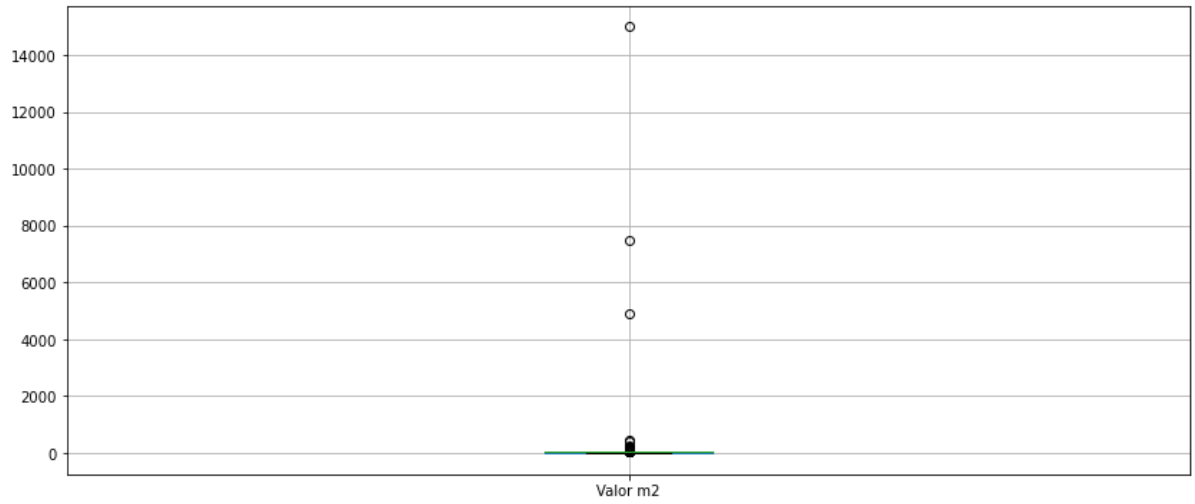
In [11]:
```
print(f'[Q1] -> {Q1}')
print(f'[Q3] -> {Q3}')
print(f'[IIQ] -> {IIQ:.2f}')
print(f'[Q1 - 1.5 * IIQ] -> {limite_inferior:.2f}')
print(f'[Q3 + 1.5 * IIQ] -> {limite_superior:.2f}')
```

```
[Q1] -> 21.25
[Q3] -> 42.31
[IIQ] -> 21.06
[Q1 - 1.5 * IIQ] -> -10.34
[Q3 + 1.5 * IIQ] -> 73.90
```

Observando o Boxplot

In [12]:
```
data.boxplot('Valor m2')
```

Out[12]: <AxesSubplot:>

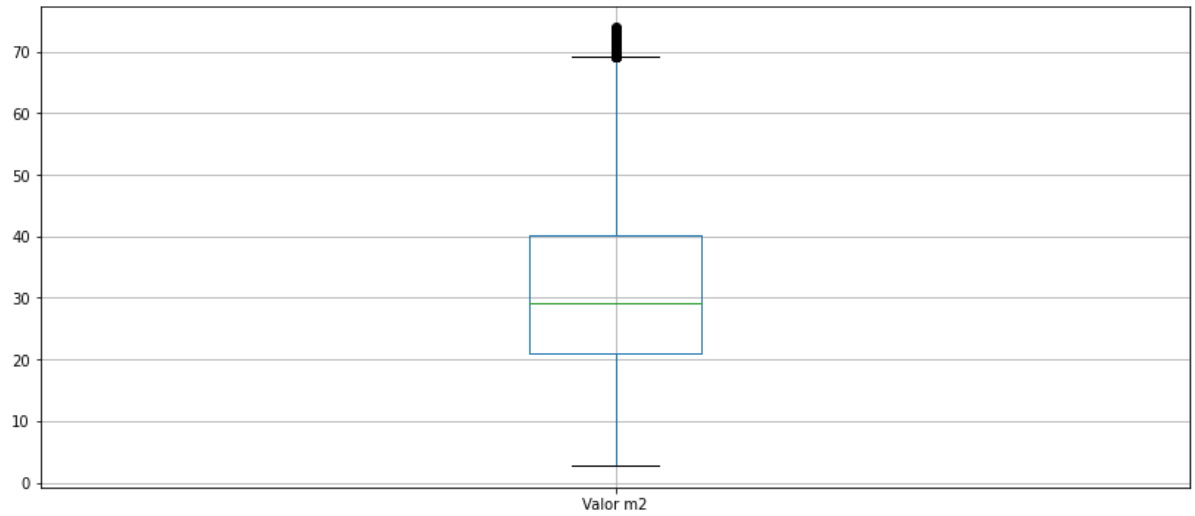


Excluindo os outliers e observando o boxplot

In [13]:
```
selecao = (data['Valor m2'] >= limite_inferior) & (data['Valor m2'] <= limite_
superior)
data = data[selecao]
```

In [14]:
```
data.boxplot('Valor m2')
```

Out[14]: <AxesSubplot:>



Comparando a visualização com o Seaborn

In [15]:
```
import seaborn as sns
```

In [16]:
```
sns.boxplot(x=data['Valor m2'])
```

Out[16]:
```
<AxesSubplot:xlabel='Valor m2'>
```