

Relatório de Análise V

Tratamento de Dados Faltantes

```
In [55]: # importando Pandas
import pandas as pd

In [56]: # importando a base de dados
dados = pd.read_csv('data/aluguel_residencial.csv', sep = ';')

In [57]: # exibindo os primeiros dez itens
dados.head(10)

Out[57]:
```

	Tipo	Bairro	Quartos	Vagas	Suites	Area	Valor	Condominio	IPTU
0	Quitinete	Copacabana	1	0	0	40	1700.0	500.0	60.0
1	Casa	Jardim Botânico	2	0	1	100	7000.0	NaN	NaN
2	Apartamento	Centro	1	0	0	15	800.0	390.0	20.0
3	Apartamento	Higienópolis	1	0	0	48	800.0	230.0	NaN
4	Apartamento	Vista Alegre	3	1	0	70	1200.0	NaN	NaN
5	Apartamento	Cachambi	2	0	0	50	1300.0	301.0	17.0
6	Casa de Condomínio	Barra da Tijuca	5	4	5	750	22000.0	NaN	NaN
7	Casa de Condomínio	Ramos	2	2	0	65	1000.0	NaN	NaN
8	Apartamento	Centro	1	0	0	36	1200.0	NaN	NaN
9	Apartamento	Grajaú	2	1	0	70	1500.0	642.0	74.0

```
In [58]: # método que nos ajuda a realizar a seleção dos dados faltantes
# método irá gerar um DataFrame booleano, em que a observação marcada como 'True' caracteriz
a um dado nulo
dados.isnull()
```

```
Out[58]:
```

	Tipo	Bairro	Quartos	Vagas	Suites	Area	Valor	Condominio	IPTU
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	True	True
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	True
4	False	False	False	False	False	False	False	True	True
...
22575	False	False	False	False	False	False	False	False	False
22576	False	False	False	False	False	False	False	False	False
22577	False	False	False	False	False	False	False	False	False
22578	False	False	False	False	False	False	False	False	False
22579	False	False	False	False	False	False	False	False	True

22580 rows × 9 columns

```
In [59]: # outra notação
# método funciona exatamente da maneira inversa ao isnull()
# se a informação for nula, será marcada como 'False'
dados.notnull()
```

```
Out[59]:
```

	Tipo	Bairro	Quartos	Vagas	Suites	Area	Valor	Condominio	IPTU
0	True	True	True	True	True	True	True	True	True
1	True	True	True	True	True	True	True	False	False
2	True	True	True	True	True	True	True	True	True
3	True	True	True	True	True	True	True	True	False
4	True	True	True	True	True	True	True	False	False
...
22575	True	True	True	True	True	True	True	True	True
22576	True	True	True	True	True	True	True	True	True
22577	True	True	True	True	True	True	True	True	True
22578	True	True	True	True	True	True	True	True	True
22579	True	True	True	True	True	True	True	True	False

22580 rows × 9 columns

```
In [60]: # analisando integralmente as informações nulas
# a segunda linha nos mostra a quantidade de registros disponíveis no DataFrame
# para cada variável, também há um número de registros não nulos
# isso nos ajuda a identificar os problemas e onde devemos aplicar soluções
dados.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 22580 entries, 0 to 22579
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   Tipo        22580 non-null  object
1   Bairro      22580 non-null  object
2   Quartos     22580 non-null  int64
3   Vagas       22580 non-null  int64
4   Suites      22580 non-null  int64
5   Area        22580 non-null  int64
6   Valor       22571 non-null  float64
7   Condominio  20765 non-null  float64
8   IPTU        15795 non-null  float64
dtypes: float64(3), int64(4), object(2)
memory usage: 1.6+ MB
```

```
In [61]: # observando a variável 'Valor' que apresenta dados nulos
dados[dados['Valor'].isnull()]

Out[61]:
```

	Tipo	Bairro	Quartos	Vagas	Suites	Area	Valor	Condominio	IPTU
58	Apartamento	Barra da Tijuca	2	1	1	70	NaN	970.0	68.0
1492	Apartamento	Leme	2	0	0	75	NaN	878.0	NaN
1683	Casa	Campo Grande	3	4	3	363	NaN	NaN	NaN
2012	Apartamento	Botafogo	2	0	0	95	NaN	1010.0	170.0
2034	Apartamento	Copacabana	2	0	0	72	NaN	850.0	NaN
4941	Casa	Campo Grande	3	2	1	100	NaN	NaN	NaN
8568	Apartamento	Leme	2	0	1	75	NaN	878.0	NaN
8947	Apartamento	Glória	3	0	1	135	NaN	910.0	228.0
9149	Apartamento	Gávea	3	1	1	105	NaN	880.0	221.0

```
In [62]: # eliminando os dados nulos da variável 'Valor'
A = dados.shape[0]
dados.dropna(subset = ['Valor'], inplace = True)
B = dados.shape[0]
A - B

Out[62]: 9

In [63]: # verificando a saída após ter eliminado
dados[dados['Valor'].isnull()]

Out[63]:
```

	Tipo	Bairro	Quartos	Vagas	Suites	Area	Valor	Condominio	IPTU
--	------	--------	---------	-------	--------	------	-------	------------	------

```
In [64]: # verificando quantas assinaturas nulas temos em Condominio
dados[dados['Condominio'].isnull()].shape[0]

Out[64]: 1813

In [65]: # filtrando os dados de acordo com o tipo Apartamento
selecao = (dados['Tipo'] == 'Apartamento') & (dados['Condominio'].isnull())

In [66]: # eliminando os dados nulos da variável 'Condominio'
A = dados.shape[0]
dados = dados[~selecao]
B = dados.shape[0]
A - B

Out[66]: 745

In [67]: # verificando a saída após ter eliminado
dados[dados['Condominio'].isnull()].shape[0]

Out[67]: 1068

In [68]: # substituindo valores nulos dos DataFrame por 0
dados = dados.fillna({'Condominio': 0, 'IPTU': 0})

In [69]: # verificando a saída de dados nulos da variável 'Condominio'
dados[dados['Condominio'].isnull()].shape[0]

Out[69]: 0

In [70]: # verificando a saída de dados nulos da variável 'IPTU'
dados[dados['IPTU'].isnull()].shape[0]

Out[70]: 0

In [71]: # analisando integralmente as informações nulas
# a segunda linha nos mostra a quantidade de registros disponíveis no DataFrame
# para cada variável, também há um número de registros não nulos
dados.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 21826 entries, 0 to 22579
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   Tipo        21826 non-null  object
1   Bairro      21826 non-null  object
2   Quartos     21826 non-null  int64
3   Vagas       21826 non-null  int64
4   Suites      21826 non-null  int64
5   Area        21826 non-null  int64
6   Valor       21826 non-null  float64
7   Condominio  21826 non-null  float64
8   IPTU        21826 non-null  float64
dtypes: float64(3), int64(4), object(2)
memory usage: 1.7+ MB
```

```
In [72]: # exportando o DataFrame
# DataFrame atualizado com as modificações realizadas
dados.to_csv('data/aluguel_residencial.csv', sep = ';', index = False)
```