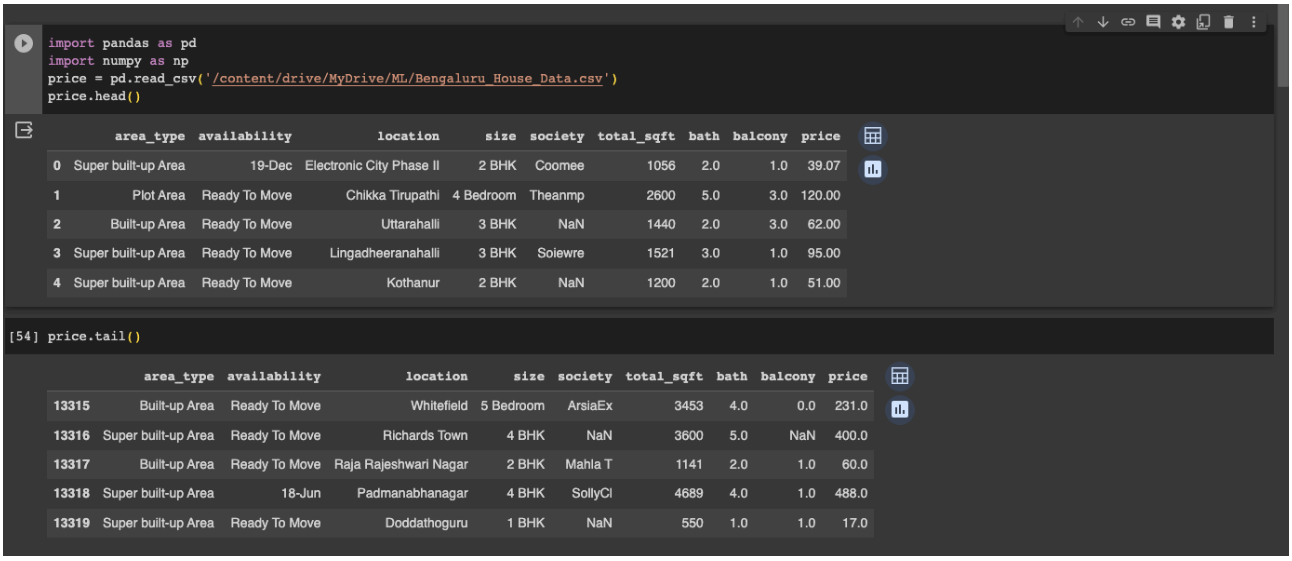
Assignment 4

U21CS089 | Garvit Shah

Checking for Missing Values using isnull() and total sum of null values.

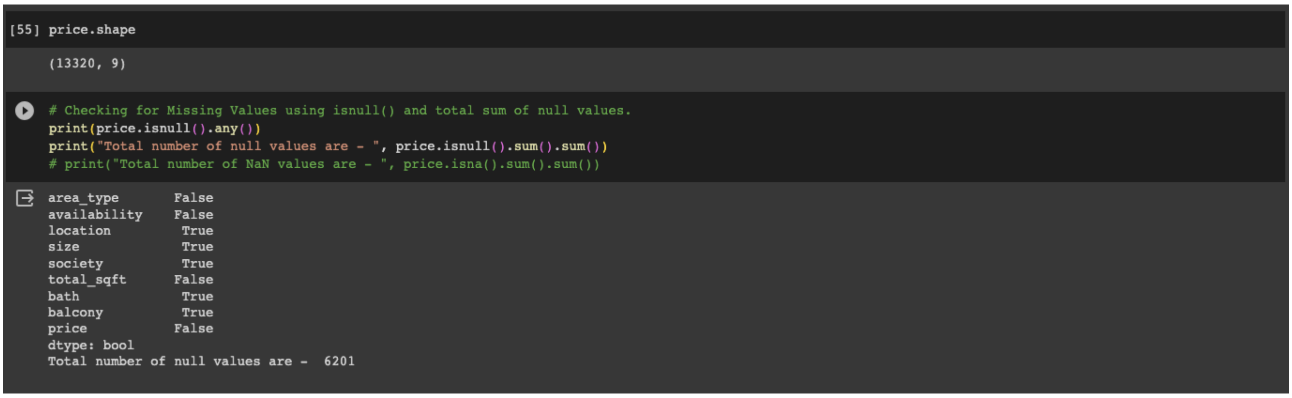


```
import pandas as pd
import numpy as np
price = pd.read_csv('/content/drive/MyDrive/ML/Bengaluru_House_Data.csv')
price.head()
```

	area_type	availability	location	size	society	total_sqft	bath	balcony	price
0	Super built-up Area	19-Dec	Electronic City Phase II	2 BHK	Coomee	1056	2.0	1.0	39.07
1	Plot Area	Ready To Move	Chikka Tirupathi	4 Bedroom	Theanmp	2600	5.0	3.0	120.00
2	Built-up Area	Ready To Move	Uttarahalli	3 BHK	NaN	1440	2.0	3.0	62.00
3	Super built-up Area	Ready To Move	Lingadheeranahalli	3 BHK	Soiewre	1521	3.0	1.0	95.00
4	Super built-up Area	Ready To Move	Kothanur	2 BHK	NaN	1200	2.0	1.0	51.00

```
[54] price.tail()
```

	area_type	availability	location	size	society	total_sqft	bath	balcony	price
13315	Built-up Area	Ready To Move	Whitefield	5 Bedroom	ArsiaEx	3453	4.0	0.0	231.0
13316	Super built-up Area	Ready To Move	Richards Town	4 BHK	NaN	3600	5.0	NaN	400.0
13317	Built-up Area	Ready To Move	Raja Rajeshwari Nagar	2 BHK	Mahla T	1141	2.0	1.0	60.0
13318	Super built-up Area	18-Jun	Padmanabhanagar	4 BHK	SollyCl	4689	4.0	1.0	488.0
13319	Super built-up Area	Ready To Move	Doddathoguru	1 BHK	NaN	550	1.0	1.0	17.0



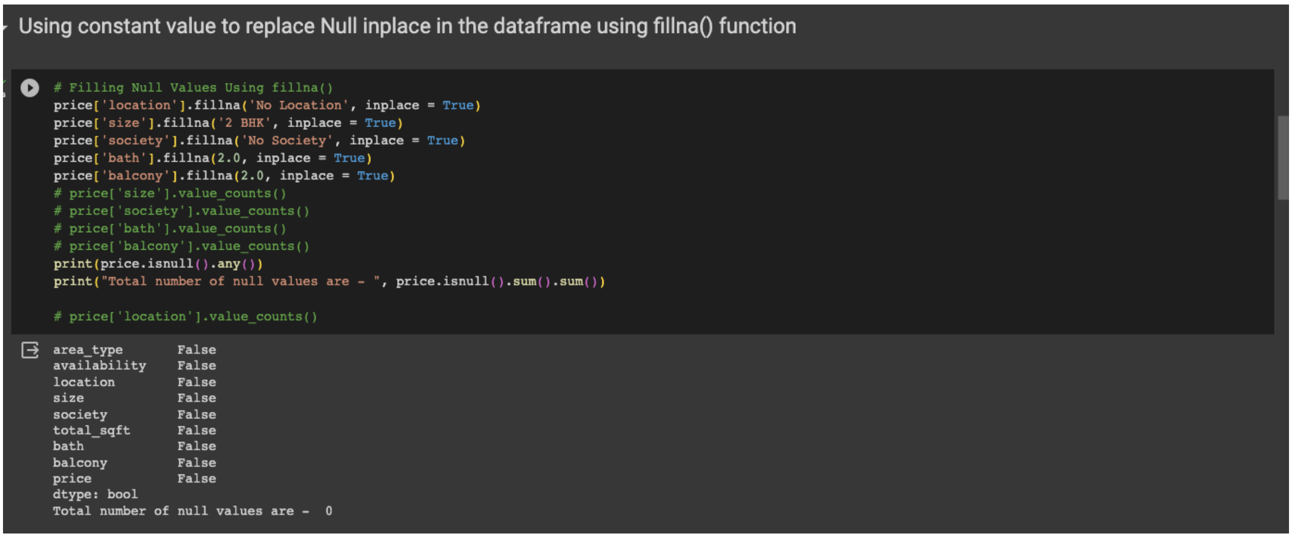
```
[55] price.shape

(13320, 9)
```

```
# Checking for Missing Values using isnull() and total sum of null values.
print(price.isnull().any())
print("Total number of null values are - ", price.isnull().sum().sum())
# print("Total number of NaN values are - ", price.isna().sum().sum())
```

```
area_type       False
availability    False
location         True
size             True
society          True
total_sqft      False
bath             True
balcony          True
price           False
dtype: bool
Total number of null values are -  6201
```

● Filling Null Values Using fillna()



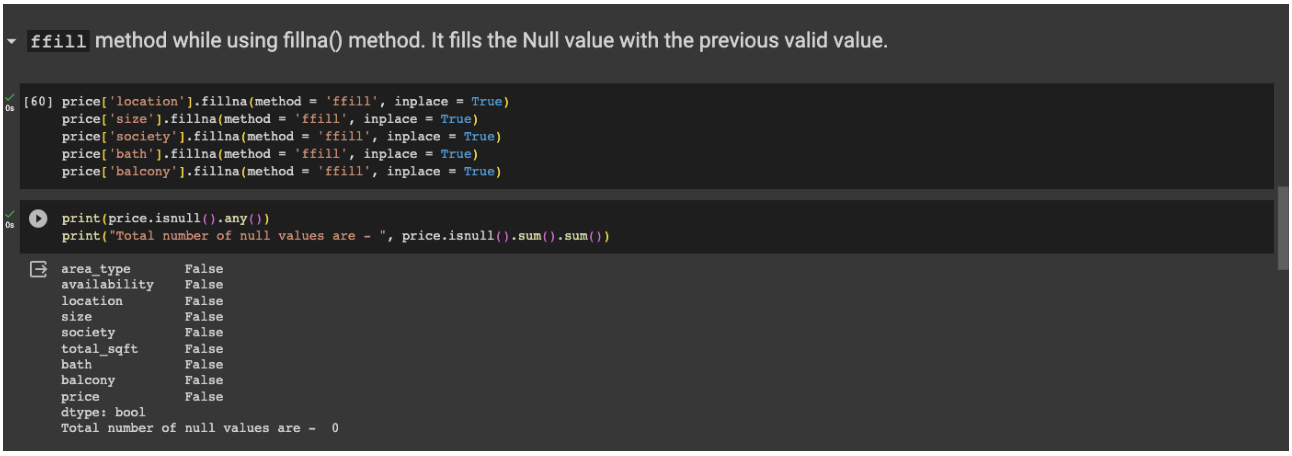
Using constant value to replace Null inplace in the dataframe using fillna() function

```
# Filling Null Values Using fillna()
price['location'].fillna('No Location', inplace = True)
price['size'].fillna('2 BHK', inplace = True)
price['society'].fillna('No Society', inplace = True)
price['bath'].fillna(2.0, inplace = True)
price['balcony'].fillna(2.0, inplace = True)
# price['size'].value_counts()
# price['society'].value_counts()
# price['bath'].value_counts()
# price['balcony'].value_counts()
print(price.isnull().any())
print("Total number of null values are - ", price.isnull().sum().sum())

# price['location'].value_counts()
```

```
area_type       False
availability    False
location        False
size            False
society         False
total_sqft      False
bath            False
balcony         False
price           False
dtype: bool
Total number of null values are -  0
```

- **Filling Null Values Using fillna(method = 'pad')**



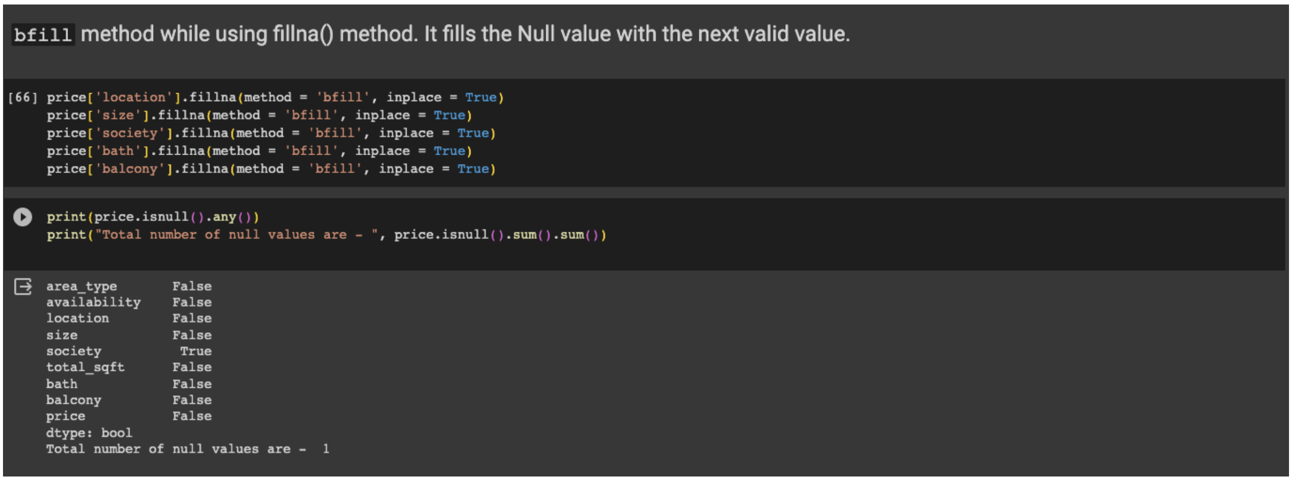
`ffill` method while using fillna() method. It fills the Null value with the previous valid value.

```
[60] price['location'].fillna(method = 'ffill', inplace = True)
     price['size'].fillna(method = 'ffill', inplace = True)
     price['society'].fillna(method = 'ffill', inplace = True)
     price['bath'].fillna(method = 'ffill', inplace = True)
     price['balcony'].fillna(method = 'ffill', inplace = True)
```

```
print(price.isnull().any())
print("Total number of null values are - ", price.isnull().sum().sum())
```

```
area_type       False
availability    False
location        False
size            False
society         False
total_sqft      False
bath            False
balcony         False
price           False
dtype: bool
Total number of null values are -  0
```

- **Filling Null Values Using fillna(method = 'bfill')**



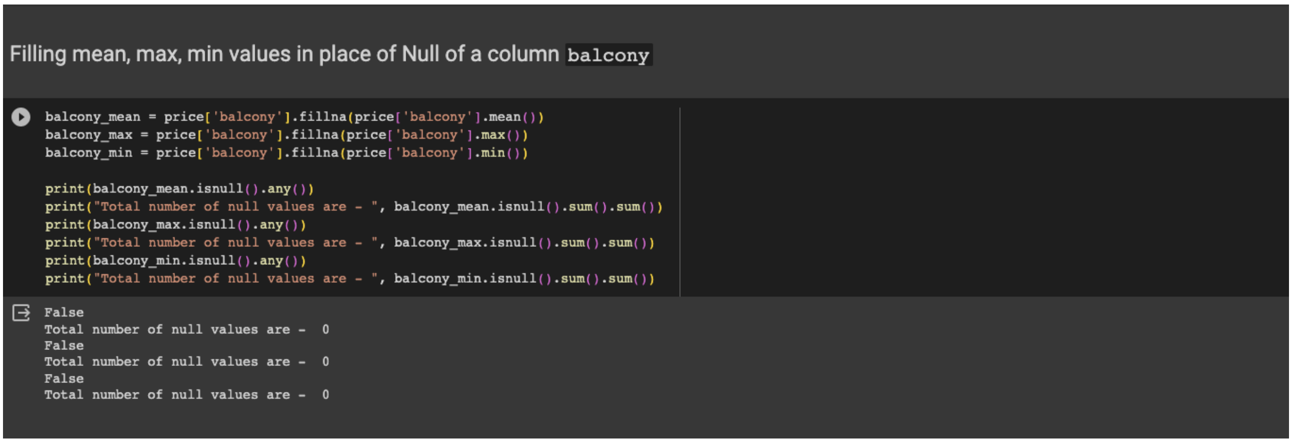
`bfill` method while using fillna() method. It fills the Null value with the next valid value.

```
[66] price['location'].fillna(method = 'bfill', inplace = True)
     price['size'].fillna(method = 'bfill', inplace = True)
     price['society'].fillna(method = 'bfill', inplace = True)
     price['bath'].fillna(method = 'bfill', inplace = True)
     price['balcony'].fillna(method = 'bfill', inplace = True)
```

```
print(price.isnull().any())
print("Total number of null values are - ", price.isnull().sum().sum())
```

```
area_type       False
availability    False
location        False
size            False
society          True
total_sqft      False
bath            False
balcony         False
price           False
dtype: bool
Total number of null values are -  1
```

- **Filling Null Values with the Mean, Max or Min of a Column**



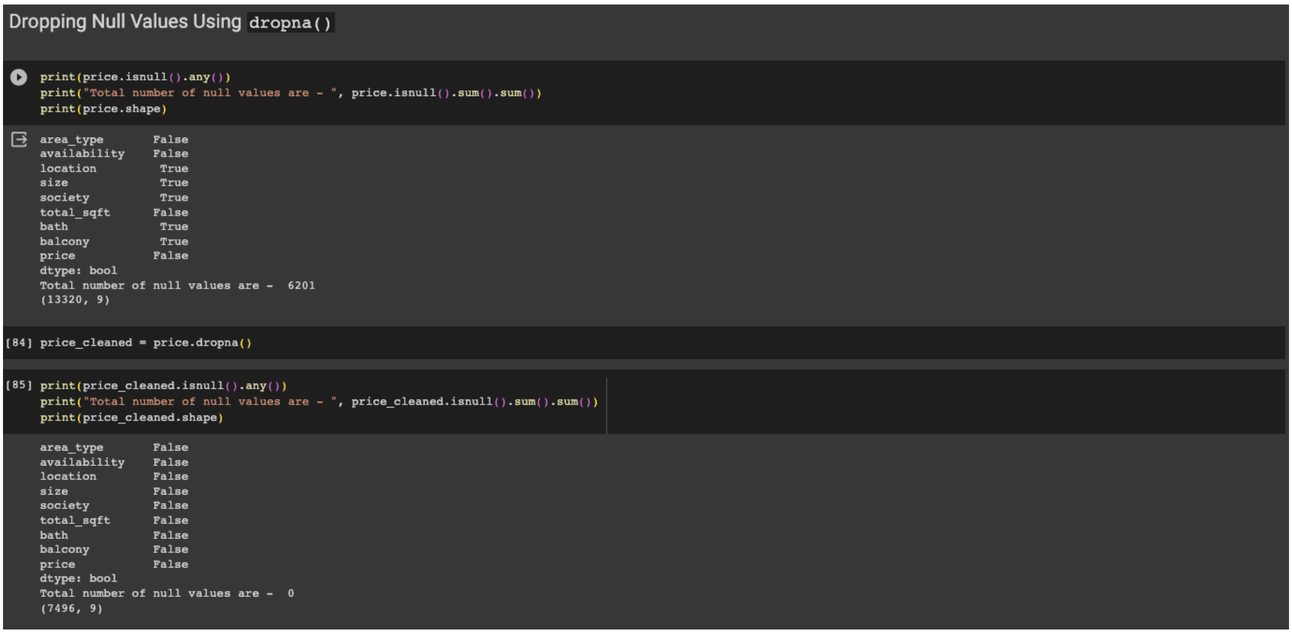
Filling mean, max, min values in place of Null of a column `balcony`

```
balcony_mean = price['balcony'].fillna(price['balcony'].mean())
balcony_max = price['balcony'].fillna(price['balcony'].max())
balcony_min = price['balcony'].fillna(price['balcony'].min())

print(balcony_mean.isnull().any())
print("Total number of null values are - ", balcony_mean.isnull().sum().sum())
print(balcony_max.isnull().any())
print("Total number of null values are - ", balcony_max.isnull().sum().sum())
print(balcony_min.isnull().any())
print("Total number of null values are - ", balcony_min.isnull().sum().sum())
```

```
False
Total number of null values are -  0
False
Total number of null values are -  0
False
Total number of null values are -  0
```

- **Dropping Null Values Using dropna()**



Dropping Null Values Using `dropna()`

```
print(price.isnull().any())
print("Total number of null values are - ", price.isnull().sum().sum())
print(price.shape)
```

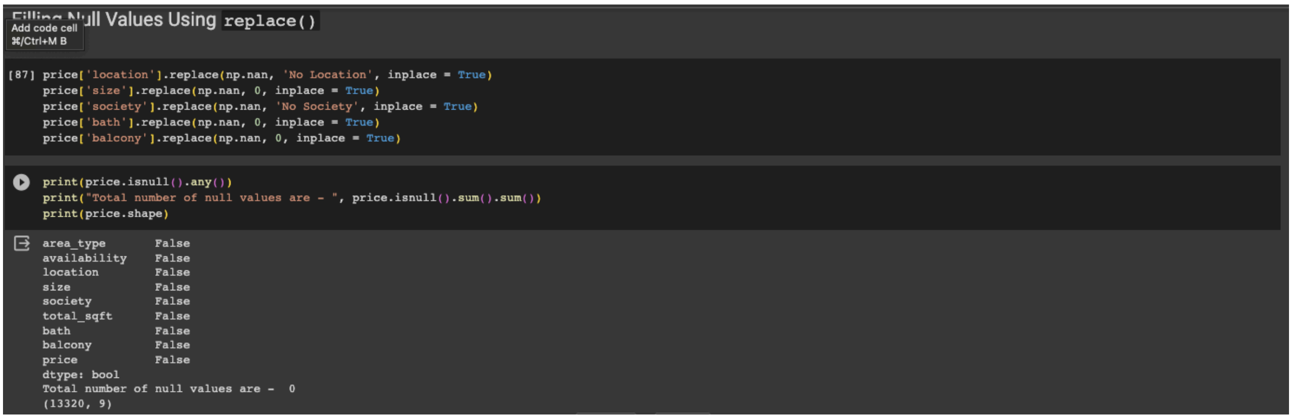
```
area_type       False
availability    False
location         True
size             True
society          True
total_sqft      False
bath             True
balcony          True
price           False
dtype: bool
Total number of null values are -  6201
(13320, 9)
```

```
[84] price_cleaned = price.dropna()
```

```
[85] print(price_cleaned.isnull().any())
     print("Total number of null values are - ", price_cleaned.isnull().sum().sum())
     print(price_cleaned.shape)
```

```
area_type       False
availability    False
location        False
size            False
society         False
total_sqft      False
bath            False
balcony         False
price           False
dtype: bool
Total number of null values are -  0
(7496, 9)
```

- **Filling Null Values Using replace()**



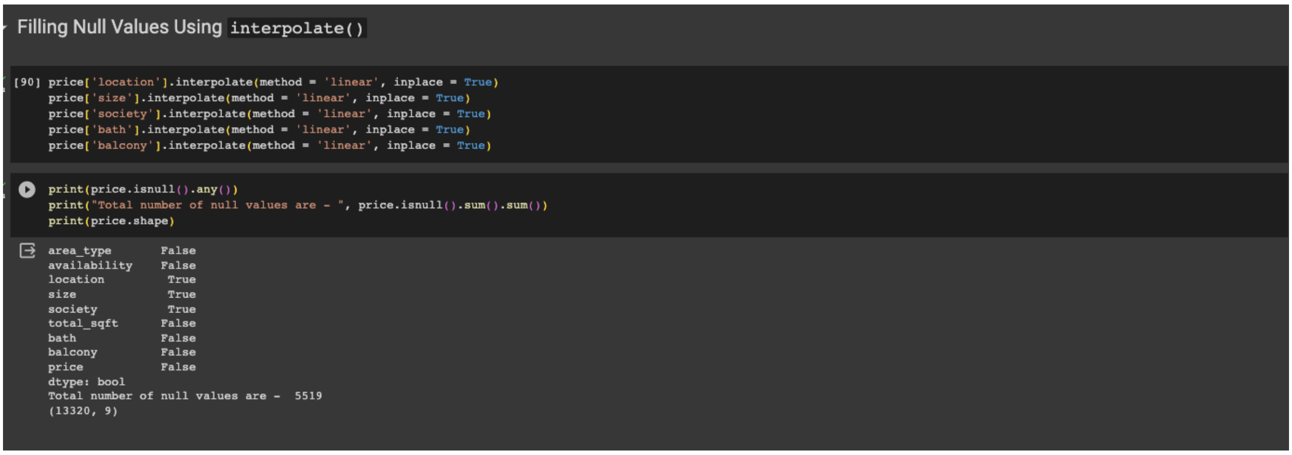
Filling Null Values Using `replace()`

```
[87] price['location'].replace(np.nan, 'No Location', inplace = True)
     price['size'].replace(np.nan, 0, inplace = True)
     price['society'].replace(np.nan, 'No Society', inplace = True)
     price['bath'].replace(np.nan, 0, inplace = True)
     price['balcony'].replace(np.nan, 0, inplace = True)
```

```
print(price.isnull().any())
print("Total number of null values are - ", price.isnull().sum().sum())
print(price.shape)
```

```
area_type       False
availability    False
location        False
size            False
society         False
total_sqft      False
bath            False
balcony         False
price           False
dtype: bool
Total number of null values are -  0
(13320, 9)
```

- **Filling Null Values Using interpolate()**



Filling Null Values Using `interpolate()`

```
[90] price['location'].interpolate(method = 'linear', inplace = True)
     price['size'].interpolate(method = 'linear', inplace = True)
     price['society'].interpolate(method = 'linear', inplace = True)
     price['bath'].interpolate(method = 'linear', inplace = True)
     price['balcony'].interpolate(method = 'linear', inplace = True)
```

```
print(price.isnull().any())
print("Total number of null values are - ", price.isnull().sum().sum())
print(price.shape)
```

```
area_type       False
availability    False
location         True
size             True
society          True
total_sqft      False
bath            False
balcony         False
price           False
dtype: bool
Total number of null values are -  5519
(13320, 9)
```