

File Edit View Insert Cell Kernel Widgets Help

Not Trusted

Python 3

In [6]: `M import pandas as pd`

Two main datatypes

1) series

In [7]: `M series = pd.Series(["BMW", "Toyota", "Honda"])`In [8]: `M series`Out[8]:

0	BMW
1	Toyota
2	Honda
	dtype: object

In [9]: `M color = pd.Series(["Blue", "Red", "Black"])`

2) DataFrame

In [10]: `M car_data = pd.DataFrame({"Car_Maker":series , "color":color})`In [11]: `M car_data`Out[11]:

	Car_Maker	color
0	BMW	Blue
1	Toyota	Red
2	Honda	Black

import data

In [12]: `M car_sales = pd.read_csv("7.1 car-sales.csv")`In [13]: `M car_sales`Out[13]:

	Make	Colour	Odometer (KM)	Doors	Price
0	Toyota	White	150043	4	\$4,000.00
1	Honda	Red	87899	4	\$5,000.00
2	Toyota	Blue	32549	3	\$7,000.00
3	BMW	Black	11179	5	\$22,000.00
4	Nissan	White	213095	4	\$3,500.00
5	Toyota	Green	99213	4	\$4,500.00
6	Honda	Blue	45698	4	\$7,500.00
7	Honda	Blue	54738	4	\$7,000.00
8	Toyota	White	60000	4	\$6,250.00
9	Nissan	White	31600	4	\$9,700.00

Export the DataFrame

In [14]: `M car_sales.to_csv("Name-we-want-to-call-after-export.csv" , index=False)`In [15]: `M c = pd.read_csv("Name-we-want-to-call-after-export.csv")`In [16]: `M c`Out[16]:

	Make	Colour	Odometer (KM)	Doors	Price
0	Toyota	White	150043	4	\$4,000.00
1	Honda	Red	87899	4	\$5,000.00
2	Toyota	Blue	32549	3	\$7,000.00
3	BMW	Black	11179	5	\$22,000.00
4	Nissan	White	213095	4	\$3,500.00
5	Toyota	Green	99213	4	\$4,500.00
6	Honda	Blue	45698	4	\$7,500.00
7	Honda	Blue	54738	4	\$7,000.00
8	Toyota	White	60000	4	\$6,250.00
9	Nissan	White	31600	4	\$9,700.00

Describing data

In [17]: `M car_sales.dtypes`Out[17]:

Make	object
------	--------

```

    Colour      object
    Odometer (KM)   int64
    Doors       int64
    Price       object
    dtype: object

In [18]: M car_sales.columns
Out[18]: Index(['Make', 'Colour', 'Odometer (KM)', 'Doors', 'Price'], dtype='object')

In [19]: M car_sales.index
Out[19]: RangeIndex(start=0, stop=10, step=1)

In [20]: M car_sales.describe()
Out[20]:
              Odometer (KM)  Doors
count          10.000000  10.000000
mean        78601.400000  4.000000
std         61983.471735  0.471405
min        11179.000000  3.000000
25%        35836.250000  4.000000
50%        57369.000000  4.000000
75%        96384.500000  4.000000
max       213095.000000  5.000000

In [21]: M car_sales.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   Make        10 non-null    object 
 1   Colour      10 non-null    object 
 2   Odometer (KM) 10 non-null    int64  
 3   Doors        10 non-null    int64  
 4   Price        10 non-null    object 
dtypes: int64(2), object(3)
memory usage: 528.0+ bytes

In [22]: M car_sales.mean()
<ipython-input-22-04f02239622f>:1: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=True') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.
car_sales.mean()

Out[22]: Odometer (KM)    78601.4
Doors                  4.0
dtype: float64

In [23]: M car_prices = pd.Series([3000, 1202, 1859, 2589])
car_prices.mean()

Out[23]: 2162.5

In [24]: M car_sales.sum()
Out[24]:
      Make           ToyotaHondaToyotaBMWNissanToyotaHondaHondaToyo...
      Colour          WhiteRedBlueBlackWhiteGreenBlueBlueWhiteWhite
      Odometer (KM)      786014
      Doors            40
      Price           $4,000.00$5,000.00$7,000.00$22,000.00$3,500.00...
      dtype: object

In [25]: M car_sales["Doors"].sum()
Out[25]: 40

In [26]: M len(car_sales)
Out[26]: 10

```

Viewing and selecting data

```

In [27]: M car_sales.head()
Out[27]:
      Make  Colour  Odometer (KM)  Doors  Price
0   Toyota  White     150043      4 $4,000.00
1   Honda   Red      87899      4 $5,000.00
2   Toyota  Blue      32549      3 $7,000.00
3    BMW   Black     11179      5 $22,000.00
4   Nissan  White     213095      4 $3,500.00

In [28]: M car_sales.head(7)
Out[28]:
      Make  Colour  Odometer (KM)  Doors  Price
0   Toyota  White     150043      4 $4,000.00
1   Honda   Red      87899      4 $5,000.00
2   Toyota  Blue      32549      3 $7,000.00
3    BMW   Black     11179      5 $22,000.00
4   Nissan  White     213095      4 $3,500.00

```

```
4 Nissan White 213095 4 $3,500.00
5 Toyota Green 99213 4 $4,500.00
6 Honda Blue 45698 4 $7,500.00
```

```
In [29]: M car_sales.tail()
```

```
Out[29]:   Make Colour Odometer (KM) Doors Price
5 Toyota Green 99213 4 $4,500.00
6 Honda Blue 45698 4 $7,500.00
7 Honda Blue 54738 4 $7,000.00
8 Toyota White 60000 4 $6,250.00
9 Nissan White 31600 4 $9,700.00
```

```
In [30]: M animals=pd.Series(["dog","Cat","panda","snake","Lion"], index=[0,2,5,7,2])
animals.loc[2]
```

```
Out[30]: 2    Cat
2    Lion
dtype: object
```

```
In [31]: M car_sales.loc[3]
```

```
Out[31]: Make           BMW
Colour          Black
Odometer (KM) 11179
Doors          5
Price          $22,000.00
Name: 3, dtype: object
```

```
In [32]: M animals.iloc[2]
```

```
Out[32]: 'panda'
```

```
In [33]: M car_sales.iloc[3]
```

```
Out[33]: Make           BMW
Colour          Black
Odometer (KM) 11179
Doors          5
Price          $22,000.00
Name: 3, dtype: object
```

```
In [34]: M animals.iloc[:5]
```

```
Out[34]: 0    dog
2    Cat
5    panda
7    snake
2    Lion
dtype: object
```

```
In [35]: M animals.loc[:5]
```

```
Out[35]: 0    dog
2    Cat
5    panda
dtype: object
```

```
In [36]: M car_sales["Make"]
```

```
Out[36]: 0    Toyota
1    Honda
2    Toyota
3    BMW
4    Nissan
5    Toyota
6    Honda
7    Honda
8    Toyota
9    Nissan
Name: Make, dtype: object
```

```
In [37]: M car_sales.Make
```

```
Out[37]: 0    Toyota
1    Honda
2    Toyota
3    BMW
4    Nissan
5    Toyota
6    Honda
7    Honda
8    Toyota
9    Nissan
Name: Make, dtype: object
```

```
In [38]: M car_sales[car_sales["Make"] == "Toyota"]
```

```
Out[38]:   Make Colour Odometer (KM) Doors Price
0  Toyota  White  150043  4 $4,000.00
2  Toyota  Blue   32549  3 $7,000.00
5  Toyota  Green  99213  4 $4,500.00
8  Toyota  White  60000  4 $6,250.00
```

```
In [39]: M car_sales[car_sales["Odometer (KM)"] > 100000]
```

```
Out[39]:
```

	Make	Colour	Odometer (KM)	Doors	Price
0	Toyota	White	150043	4	\$4,000.00
4	Nissan	White	213095	4	\$3,500.00

```
In [40]:
```

```
car_sales
```

```
Out[40]:
```

	Make	Colour	Odometer (KM)	Doors	Price
0	Toyota	White	150043	4	\$4,000.00
1	Honda	Red	87899	4	\$5,000.00
2	Toyota	Blue	32549	3	\$7,000.00
3	BMW	Black	11179	5	\$22,000.00
4	Nissan	White	213095	4	\$3,500.00
5	Toyota	Green	99213	4	\$4,500.00
6	Honda	Blue	45698	4	\$7,500.00
7	Honda	Blue	54738	4	\$7,000.00
8	Toyota	White	60000	4	\$6,250.00
9	Nissan	White	31600	4	\$9,700.00

```
In [41]:
```

```
pd.crosstab(car_sales["Make"], car_sales["Doors"])
```

```
Out[41]:
```

Doors	3	4	5
Make			
BMW	0	0	1
Honda	0	3	0
Nissan	0	2	0
Toyota	1	3	0

```
In [42]:
```

```
car_sales.groupby(["Make"]).mean()
```

```
Out[42]:
```

Make	Odometer (KM)	Doors
BMW	11179.000000	5.00
Honda	62778.333333	4.00
Nissan	122347.500000	4.00
Toyota	85451.250000	3.75

```
In [43]:
```

```
car_sales["Odometer (KM)"].plot()
```

```
Out[43]:
```

Doors	Odometer (KM)
0	150043
1	87899
2	32549
3	11179
4	213095
5	99213
6	45698
7	54738
8	60000
9	31600

```
In [44]:
```

```
car_sales["Odometer (KM)"].hist()
```

```
Out[44]:
```

Bin Range (KM)	Frequency
0-25,000	10
25,000-50,000	30
50,000-75,000	20
75,000-100,000	10
100,000-125,000	10
125,000-150,000	10
150,000-175,000	10
175,000-200,000	10

```
In [45]:
```

```
car_sales['Price'] = car_sales['Price'].str.replace('[$\,\.\,\.]','').astype(int)
```

<ipython-input-45-2f8a152b777>:1: FutureWarning: The default value of regex will change from True to False in a future version.

```
car_sales['Price'] = car_sales['Price'].str.replace('[$\,\.\,\.]','').astype(int)
```

```
In [46]:
```

```
car_sales
```

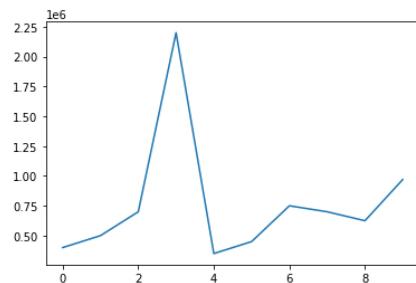
```
Out[46]:
```

	Make	Colour	Odometer (KM)	Doors	Price
0	Toyota	White	150043	4	400000
1	Honda	Red	87899	4	500000
2	Toyota	Blue	32549	3	700000
3	BMW	Black	11179	5	2200000

```
4 Nissan White 213095 4 350000
5 Toyota Green 99213 4 450000
6 Honda Blue 45698 4 750000
7 Honda Blue 54738 4 700000
8 Toyota White 60000 4 625000
9 Nissan White 31600 4 970000
```

```
In [47]: car_sales["Price"].plot()
```

```
Out[47]: <AxesSubplot:>
```



Manipulating Data

```
In [48]: car_sales["Make"].str.lower()
```

```
Out[48]: 0    toyota
1     honda
2    toyota
3     bmw
4   nissan
5    toyota
6     honda
7     honda
8    toyota
9   nissan
Name: Make, dtype: object
```

```
In [49]: car_sales
```

```
Out[49]:   Make Colour Odometer (KM)  Doors  Price
0  Toyota  White      150043      4  400000
1  Honda    Red       87899      4  500000
2  Toyota   Blue      32549      3  700000
3   BMW    Black      11179      5 2200000
4  Nissan  White     213095      4  350000
5  Toyota  Green      99213      4  450000
6  Honda   Blue      45698      4  750000
7  Honda   Blue      54738      4  700000
8  Toyota  White      60000      4  625000
9  Nissan  White      31600      4  970000
```

```
In [50]: car_sales["Make"] = car_sales["Make"].str.lower()
```

```
In [51]: car_sales
```

```
Out[51]:   Make Colour Odometer (KM)  Doors  Price
0  toyota  White      150043      4  400000
1  honda    Red       87899      4  500000
2  toyota   Blue      32549      3  700000
3   bmw    Black      11179      5 2200000
4  nissan  White     213095      4  350000
5  toyota  Green      99213      4  450000
6  honda   Blue      45698      4  750000
7  honda   Blue      54738      4  700000
8  toyota  White      60000      4  625000
9  nissan  White      31600      4  970000
```

```
In [52]: car_sales_missing = pd.read_csv('9.1_car-sales-missing-data.csv')
```

```
In [53]: car_sales_missing
```

```
Out[53]:   Make Colour Odometer  Doors  Price
0  Toyota  White  150043.0  4.0 $4,000
1  Honda    Red   87899.0  4.0 $5,000
2  Toyota   Blue    NaN  3.0 $7,000
3   BMW    Black  11179.0  5.0 $22,000
4  Nissan  White  213095.0  4.0 $3,500
5  Toyota  Green    NaN  4.0 $4,500
```

```

   Make Colour Odometer KM Doors Price
6 Honda   NaN     NaN  4.0 $7,500
7 Honda   Blue    NaN  4.0   NaN
8 Toyota  White   60000.0 NaN   NaN
9 NaN     White   31600.0 4.0 $9,700

```

```
In [54]: car_sales_missing["Odometer"].fillna(car_sales_missing["Odometer"].mean(), inplace=True)
```

```
In [55]: car_sales_missing
```

```
Out[55]:   Make Colour Odometer Doors Price
0 Toyota  White  150043.000000 4.0 $4,000
1 Honda   Red    87899.000000 4.0 $5,000
2 Toyota  Blue   92302.666667 3.0 $7,000
3 BMW     Black   11179.000000 5.0 $22,000
4 Nissan  White  213095.000000 4.0 $3,500
5 Toyota  Green  92302.666667 4.0 $4,500
6 Honda   NaN    92302.666667 4.0 $7,500
7 Honda   Blue   92302.666667 4.0   NaN
8 Toyota  White  60000.000000 NaN   NaN
9 NaN     White  31600.000000 4.0 $9,700
```

```
In [56]: car_sales_missing.dropna(inplace = True)
```

```
In [57]: car_sales_missing
```

```
Out[57]:   Make Colour Odometer Doors Price
0 Toyota  White  150043.000000 4.0 $4,000
1 Honda   Red    87899.000000 4.0 $5,000
2 Toyota  Blue   92302.666667 3.0 $7,000
3 BMW     Black   11179.000000 5.0 $22,000
4 Nissan  White  213095.000000 4.0 $3,500
5 Toyota  Green  92302.666667 4.0 $4,500
```

```
In [58]: ## column from series
seats_column = pd.Series([5,5,5,5,5,5])

## new column called seats
car_sales["Seats"] = seats_column
```

```
In [59]: car_sales
```

```
Out[59]:   Make Colour Odometer (KM) Doors Price Seats
0 toyota  White   150043   4 400000  5.0
1 honda   Red    87899    4 500000  5.0
2 toyota  Blue   32549    3 700000  5.0
3 bmw    Black   11179    5 2200000 5.0
4 nissan  White  213095   4 350000  5.0
5 toyota  Green  99213    4 450000  5.0
6 honda   Blue   45698    4 750000  NaN
7 honda   Blue   54738    4 700000  NaN
8 toyota  White  60000    4 625000  NaN
9 nissan  White  31600    4 970000  NaN
```

```
In [60]: ## Adding column from python List
fuel_economy = [1,5,3,8,9,6,7,8,9,2]
car_sales["Fuel per 100KM"] = fuel_economy
```

```
In [61]: car_sales
```

```
Out[61]:   Make Colour Odometer (KM) Doors Price Seats Fuel per 100KM
0 toyota  White   150043   4 400000  5.0      1
1 honda   Red    87899    4 500000  5.0      5
2 toyota  Blue   32549    3 700000  5.0      3
3 bmw    Black   11179    5 2200000 5.0      8
4 nissan  White  213095   4 350000  5.0      9
5 toyota  Green  99213    4 450000  5.0      6
6 honda   Blue   45698    4 750000  NaN      7
7 honda   Blue   54738    4 700000  NaN      8
8 toyota  White  60000    4 625000  NaN      9
9 nissan  White  31600    4 970000  NaN      2
```

```
In [62]: car_sales["Total fuel used"] = car_sales["Odometer (KM)"]/100 * car_sales["Fuel per 100KM"]
```

```
In [63]: car_sales
```

```
Out[63]:   Make Colour Odometer (KM) Doors Price Seats Fuel per 100KM Total fuel used
0 toyota  White   150043   4 400000  5.0      1      1500.43
1 honda   Red    87899    4 500000  5.0      5      4394.95
2 toyota  Blue   32549    3 700000  5.0      3      2477.97
3 bmw    Black   11179    5 2200000 5.0      8      17895.92
4 nissan  White  213095   4 350000  5.0      9      32875.71
5 toyota  Green  99213    4 450000  5.0      6      22346.89
6 honda   Blue   45698    4 750000  NaN      7      34789.07
7 honda   Blue   54738    4 700000  NaN      8      38318.60
8 toyota  White  60000    4 625000  NaN      9      37500.00
9 nissan  White  31600    4 970000  NaN      2      60320.00
```

```

      0 toyota White 150043 4 400000 5.0 1 1500.43
      1 honda Red 87899 4 500000 5.0 5 4394.95
      2 toyota Blue 32549 3 700000 5.0 3 976.47
      3 bmw Black 11179 5 2200000 5.0 8 894.32
      4 nissan White 213095 4 350000 5.0 9 19178.55
      5 toyota Green 99213 4 450000 5.0 6 5952.78
      6 honda Blue 45698 4 750000 NaN 7 3198.86
      7 honda Blue 54738 4 700000 NaN 8 4379.04
      8 toyota White 60000 4 625000 NaN 9 5400.00
      9 nissan White 31600 4 970000 NaN 2 632.00

```

```
In [64]: ## Creating a column from a single value
car_sales["Number of wheels"] = 4
```

```
In [65]: car_sales
```

```
Out[65]:   Make Colour Odometer (KM) Doors Price Seats Fuel per 100KM Total fuel used Number of wheels
      0 toyota White 150043 4 400000 5.0 1 1500.43 4
      1 honda Red 87899 4 500000 5.0 5 4394.95 4
      2 toyota Blue 32549 3 700000 5.0 3 976.47 4
      3 bmw Black 11179 5 2200000 5.0 8 894.32 4
      4 nissan White 213095 4 350000 5.0 9 19178.55 4
      5 toyota Green 99213 4 450000 5.0 6 5952.78 4
      6 honda Blue 45698 4 750000 NaN 7 3198.86 4
      7 honda Blue 54738 4 700000 NaN 8 4379.04 4
      8 toyota White 60000 4 625000 NaN 9 5400.00 4
      9 nissan White 31600 4 970000 NaN 2 632.00 4
```

```
In [66]: car_sales.drop("Number of wheels", axis=1,inplace=True)
```

```
In [67]: car_sales
```

```
Out[67]:   Make Colour Odometer (KM) Doors Price Seats Fuel per 100KM Total fuel used
      0 toyota White 150043 4 400000 5.0 1 1500.43
      1 honda Red 87899 4 500000 5.0 5 4394.95
      2 toyota Blue 32549 3 700000 5.0 3 976.47
      3 bmw Black 11179 5 2200000 5.0 8 894.32
      4 nissan White 213095 4 350000 5.0 9 19178.55
      5 toyota Green 99213 4 450000 5.0 6 5952.78
      6 honda Blue 45698 4 750000 NaN 7 3198.86
      7 honda Blue 54738 4 700000 NaN 8 4379.04
      8 toyota White 60000 4 625000 NaN 9 5400.00
      9 nissan White 31600 4 970000 NaN 2 632.00
```

```
In [68]: car_sales_shuffled = car_sales.sample(frac=1)
```

```
In [69]: car_sales_shuffled
```

```
Out[69]:   Make Colour Odometer (KM) Doors Price Seats Fuel per 100KM Total fuel used
      5 toyota Green 99213 4 450000 5.0 6 5952.78
      6 honda Blue 45698 4 750000 NaN 7 3198.86
      7 honda Blue 54738 4 700000 NaN 8 4379.04
      9 nissan White 31600 4 970000 NaN 2 632.00
      2 toyota Blue 32549 3 700000 5.0 3 976.47
      0 toyota White 150043 4 400000 5.0 1 1500.43
      8 toyota White 60000 4 625000 NaN 9 5400.00
      4 nissan White 213095 4 350000 5.0 9 19178.55
      3 bmw Black 11179 5 2200000 5.0 8 894.32
      1 honda Red 87899 4 500000 5.0 5 4394.95
```

```
In [70]: car_sales_shuffled
```

```
Out[70]:   Make Colour Odometer (KM) Doors Price Seats Fuel per 100KM Total fuel used
      5 toyota Green 99213 4 450000 5.0 6 5952.78
      6 honda Blue 45698 4 750000 NaN 7 3198.86
      7 honda Blue 54738 4 700000 NaN 8 4379.04
      9 nissan White 31600 4 970000 NaN 2 632.00
      2 toyota Blue 32549 3 700000 5.0 3 976.47
      0 toyota White 150043 4 400000 5.0 1 1500.43
      8 toyota White 60000 4 625000 NaN 9 5400.00
      4 nissan White 213095 4 350000 5.0 9 19178.55
      3 bmw Black 11179 5 2200000 5.0 8 894.32
      1 honda Red 87899 4 500000 5.0 5 4394.95
```

```
In [71]: ## Only select 20% of data
```

```
car_sales_shuffled.sample(frac=0.2)
```

```
Out[71]:
```

	Make	Colour	Odometer (KM)	Doors	Price	Seats	Fuel per 100KM	Total fuel used
8	toyota	White	60000	4	625000	NaN	9	5400.00
4	nissan	White	213095	4	350000	5.0	9	19178.55

```
In [72]: car_sales_shuffled.reset_index(drop = True , inplace=True)
```

```
In [73]: car_sales_shuffled
```

```
Out[73]:
```

	Make	Colour	Odometer (KM)	Doors	Price	Seats	Fuel per 100KM	Total fuel used
0	toyota	Green	99213	4	450000	5.0	6	5952.78
1	honda	Blue	45698	4	750000	NaN	7	3198.86
2	honda	Blue	54738	4	700000	NaN	8	4379.04
3	nissan	White	31600	4	970000	NaN	2	632.00
4	toyota	Blue	32549	3	700000	5.0	3	976.47
5	toyota	White	150043	4	400000	5.0	1	1500.43
6	toyota	White	60000	4	625000	NaN	9	5400.00
7	nissan	White	213095	4	350000	5.0	9	19178.55
8	bmw	Black	11179	5	2200000	5.0	8	894.32
9	honda	Red	87899	4	500000	5.0	5	4394.95

```
In [74]: car_sales
```

```
Out[74]:
```

	Make	Colour	Odometer (KM)	Doors	Price	Seats	Fuel per 100KM	Total fuel used
0	toyota	White	150043	4	400000	5.0	1	1500.43
1	honda	Red	87899	4	500000	5.0	5	4394.95
2	toyota	Blue	32549	3	700000	5.0	3	976.47
3	bmw	Black	11179	5	2200000	5.0	8	894.32
4	nissan	White	213095	4	350000	5.0	9	19178.55
5	toyota	Green	99213	4	450000	5.0	6	5952.78
6	honda	Blue	45698	4	750000	NaN	7	3198.86
7	honda	Blue	54738	4	700000	NaN	8	4379.04
8	toyota	White	60000	4	625000	NaN	9	5400.00
9	nissan	White	31600	4	970000	NaN	2	632.00

```
In [75]: car_sales["Odometer (KM)"] = car_sales["Odometer (KM)"].apply(lambda x: x / 1.6)
```

```
In [76]: car_sales
```

```
Out[76]:
```

	Make	Colour	Odometer (KM)	Doors	Price	Seats	Fuel per 100KM	Total fuel used
0	toyota	White	93776.875	4	400000	5.0	1	1500.43
1	honda	Red	54936.875	4	500000	5.0	5	4394.95
2	toyota	Blue	20343.125	3	700000	5.0	3	976.47
3	bmw	Black	6986.875	5	2200000	5.0	8	894.32
4	nissan	White	133184.375	4	350000	5.0	9	19178.55
5	toyota	Green	62008.125	4	450000	5.0	6	5952.78
6	honda	Blue	28561.250	4	750000	NaN	7	3198.86
7	honda	Blue	34211.250	4	700000	NaN	8	4379.04
8	toyota	White	37500.000	4	625000	NaN	9	5400.00
9	nissan	White	19750.000	4	970000	NaN	2	632.00

```
In [ ]:
```