AUTOMOBILE ASSIGNMENT

by Munjaji Ingole

READING DATA

In [2]:
```
data = pd.read_csv('/Users/saurabhingole/Downloads/bepec/pandas/8. Automobile price data _Raw_.csv')
```

In [3]:
```
data
```

Out[3]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base	...	engine-size	fuel-system	bore	stroke	compression-ratio	horsep
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0	
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0	
4	2	164	audi	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	9.5	
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	8.7	
202	-1	95	volvo	gas	std	four	sedan	rwd	front	109.1	...	173	mpfi	3.58	2.87	8.8	
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	109.1	...	145	idi	3.01	3.40	23.0	
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	9.5	

205 rows × 26 columns

UNDERSTANDING DATA

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 205 entries, 0 to 204
Data columns (total 26 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   symboling          205 non-null    int64
 1   normalized-losses  164 non-null    object
 2   make               205 non-null    object
 3   fuel-type          205 non-null    object
 4   aspiration         205 non-null    object
 5   num-of-doors       205 non-null    int64
 6   body-style         205 non-null    object
 7   drive-wheels       205 non-null    object
 8   engine-location    205 non-null    object
 9   wheel-base         205 non-null    float64
 10  length             205 non-null    float64
 11  width              205 non-null    float64
 12  height             205 non-null    float64
 13  curb-weight        205 non-null    int64
 14  engine-type        205 non-null    object
 15  num-of-cylinders   205 non-null    int64
 16  engine-size        205 non-null    int64
 17  fuel-system        205 non-null    object
 18  bore               201 non-null    object
 19  stroke             201 non-null    object
 20  compression-ratio  205 non-null    float64
 21  horsepower         203 non-null    object
 22  peak-rpm           203 non-null    object
 23  city-mpg           205 non-null    int64
 24  highway-mpg        205 non-null    int64
 25  price              201 non-null    object
dtypes: float64(5), int64(7), object(14)
memory usage: 41.8+ KB
```

understanding the data types and non null count of particular columns

CLEANING DATA

```
data['num-of-cylinders'].unique()
```

```
array(['four', 'six', 'five', 'three', 'twelve', 'two', 'eight'],
      dtype=object)
```

```
data['num-of-cylinders'] = data['num-of-cylinders'].replace(['four', 'six', 'five', 'three', 'twelve', 'two', 'eight
```

```
data['num-of-doors'].unique()
```

```
array(['two', 'four', '?'], dtype=object)
```

```
data['num-of-doors'] = data['num-of-doors'].replace(['two', 'four', '?'],[2,4,4])
```

- converting string data into numeric values

```
data = data.replace('?',np.nan)
```

- replacing all ‘?’ with null values

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 205 entries, 0 to 204
Data columns (total 26 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   symboling          205 non-null    int64
 1   normalized-losses  164 non-null    float64
 2   make               205 non-null    object
 3   fuel-type          205 non-null    object
 4   aspiration         205 non-null    object
 5   num-of-doors       205 non-null    int64
 6   body-style         205 non-null    object
 7   drive-wheels       205 non-null    object
 8   engine-location    205 non-null    object
 9   wheel-base         205 non-null    float64
 10  length             205 non-null    float64
 11  width              205 non-null    float64
 12  height             205 non-null    float64
 13  curb-weight        205 non-null    int64
 14  engine-type        205 non-null    object
 15  num-of-cylinders   205 non-null    int64
 16  engine-size        205 non-null    int64
 17  fuel-system        205 non-null    object
 18  bore               201 non-null    float64
 19  stroke             201 non-null    float64
 20  compression-ratio  205 non-null    float64
 21  horsepower         203 non-null    float64
 22  peak-rpm           203 non-null    float64
 23  city-mpg           205 non-null    int64
 24  highway-mpg        205 non-null    int64
 25  price              201 non-null    float64
dtypes: float64(11), int64(7), object(8)
memory usage: 41.8+ KB
```

with the help of ‘pd.to_numeric()’ we convert all required object data type into numeric data type

```
data= data.replace(np.nan,data.mean())
```

```
/var/folders/3r/dntzkpvs20s26mlz3hyykz_h0000gn/T/ipykernel_35578/4137260971.py:1: FutureWarning: The default value
of numeric_only in DataFrame.mean is deprecated. In a future version, it will default to False. In addition, specif
ying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence t
his warning.
  data= data.replace(np.nan,data.mean())
```

- Replacing all null values with mean of that particular column

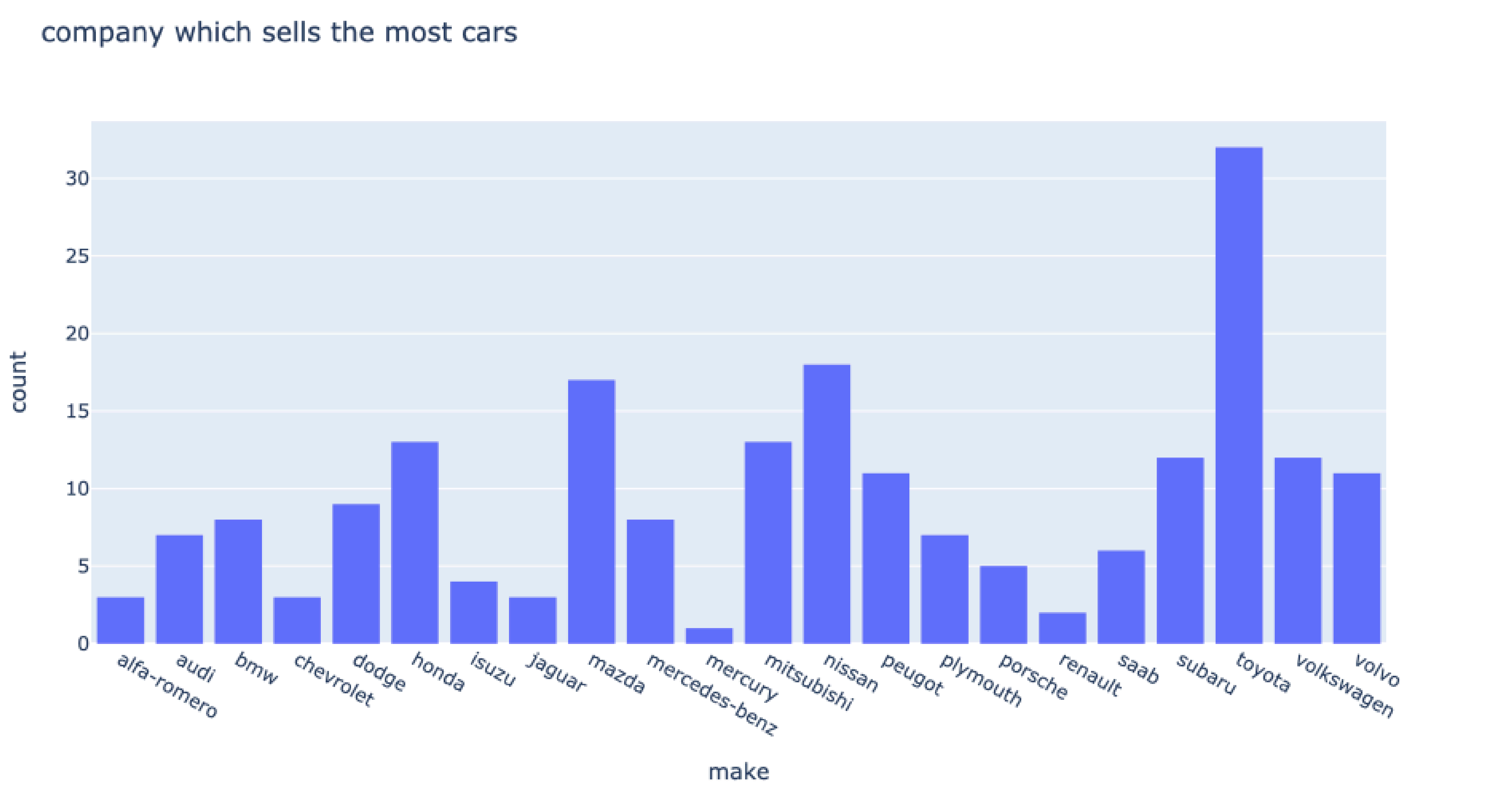
#Displaying the cleared data

```
data
```

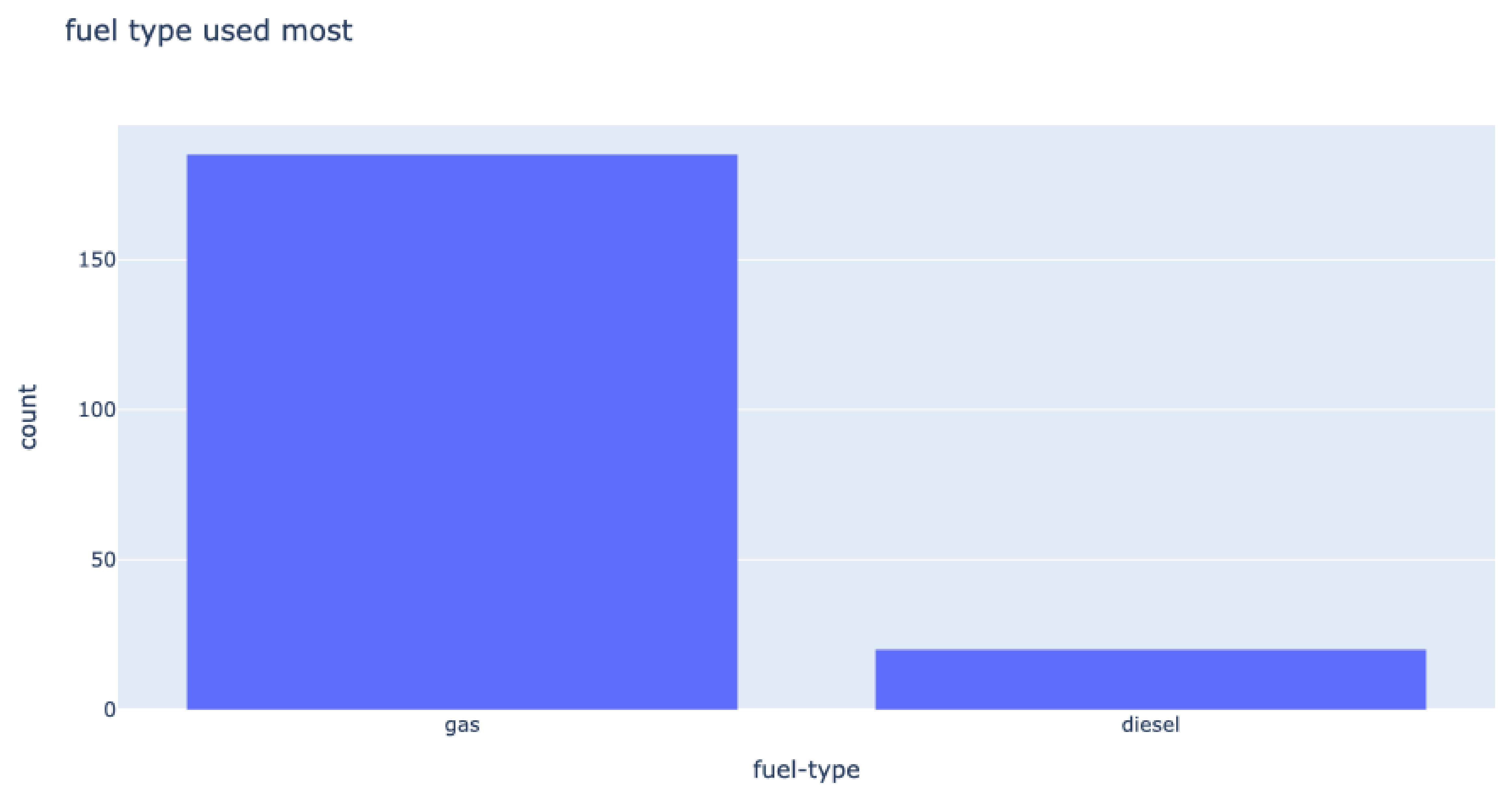
	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base	...	engine-size	fuel-system	bore	stroke	compression-ratio	horsep
0	3	122.0	alfa-romero	gas	std	2	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	
1	3	122.0	alfa-romero	gas	std	2	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	
2	1	122.0	alfa-romero	gas	std	2	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0	
3	2	164.0	audi	gas	std	4	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0	
4	2	164.0	audi	gas	std	4	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
200	4	95.0	volvo	gas	std	4	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	9.5	
201	4	95.0	volvo	gas	turbo	4	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	8.7	
202	4	95.0	volvo	gas	std	4	sedan	rwd	front	109.1	...	173	mpfi	3.58	2.87	8.8	
203	4	95.0	volvo	diesel	turbo	4	sedan	rwd	front	109.1	...	145	idi	3.01	3.40	23.0	
204	4	95.0	volvo	gas	turbo	4	sedan	rwd	front	109.1	...	141	mpfi	3.78	3.15	9.5	

205 rows × 26 columns

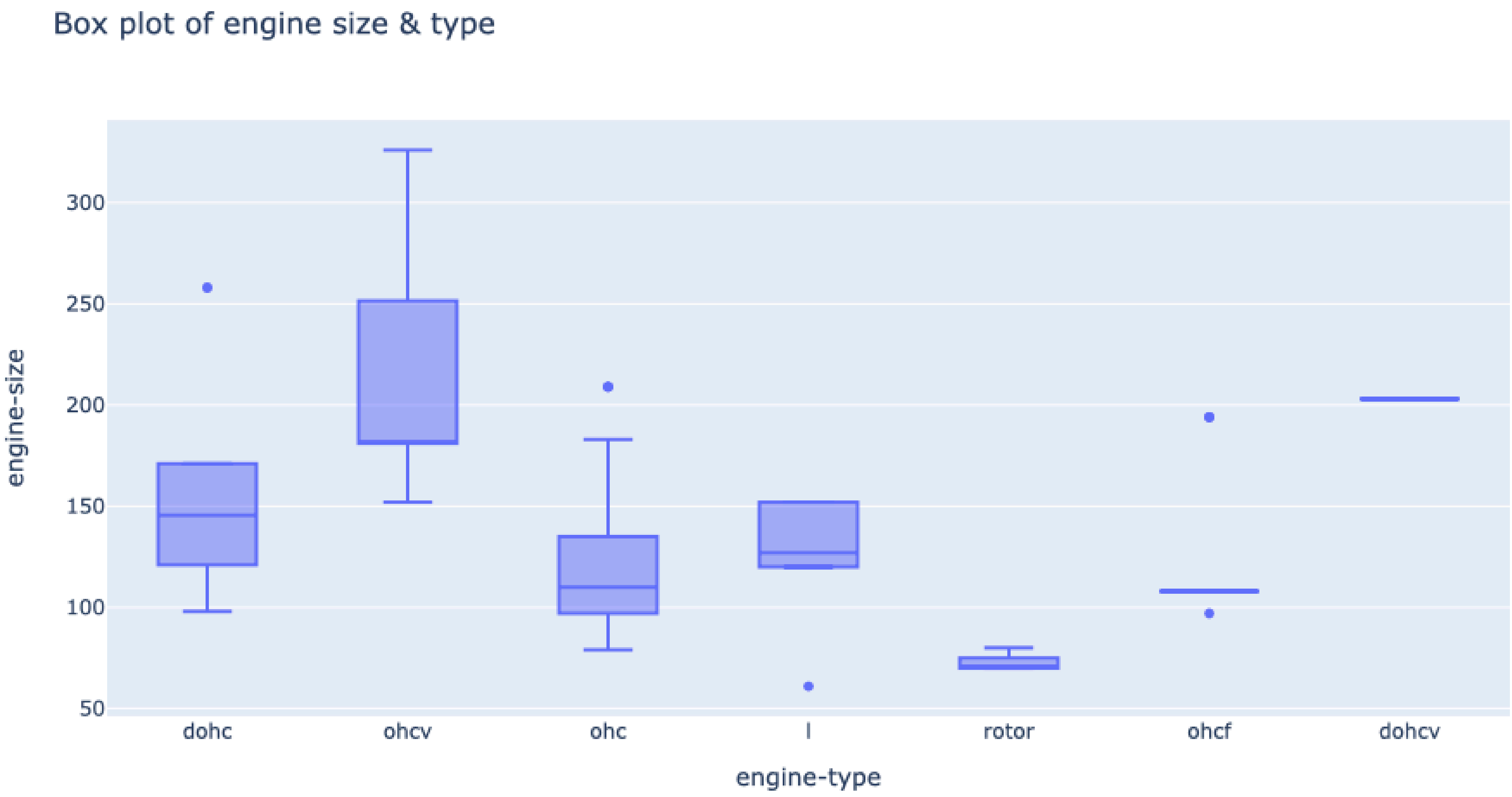
VISUALIZATION



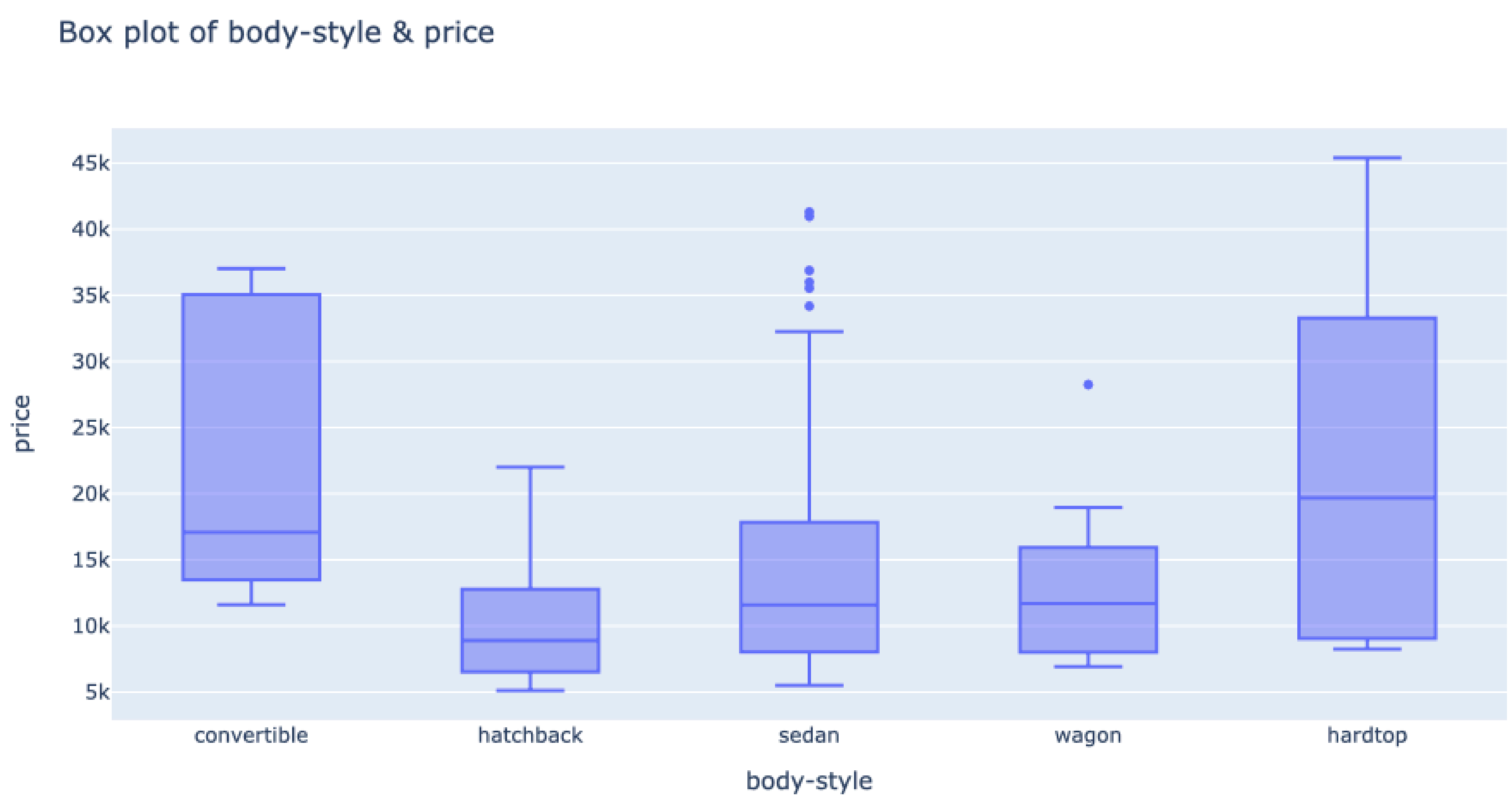
- histogram shows that TOYOTA has the most number of sales
- where as MERCURY is the least selling company among all companies



- Use of gas is most & use of diesel is very less as compaired to gas



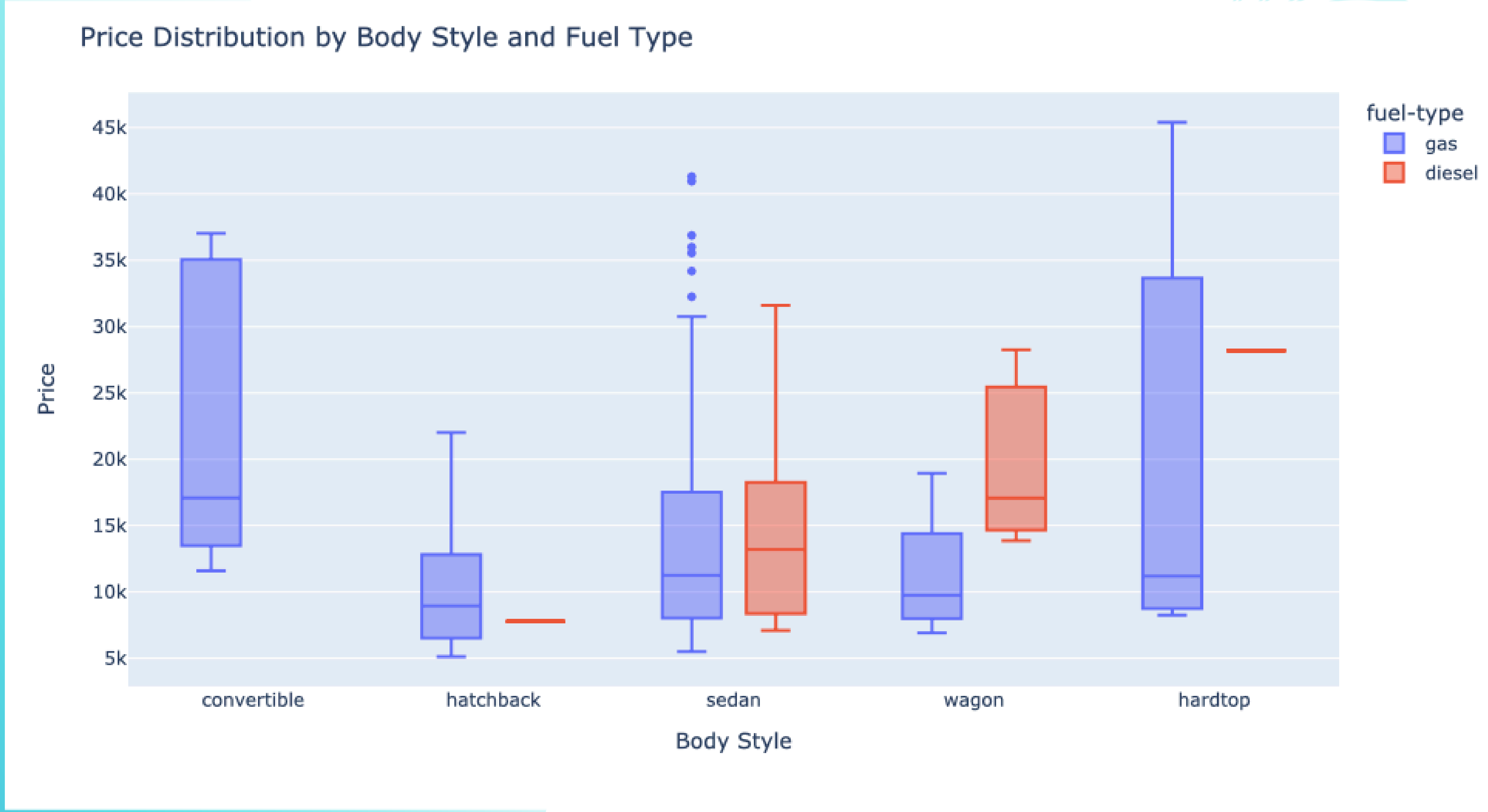
- there is a wide spread of engine sizes within OHCV , DOHC then OHC and I engine type
- engine size for ROTOR , OHCF & DOHCV engine type is relatively central there is no variation in the engine size
- among all OHCV has max engine size of 326 and the wide spread of sizes



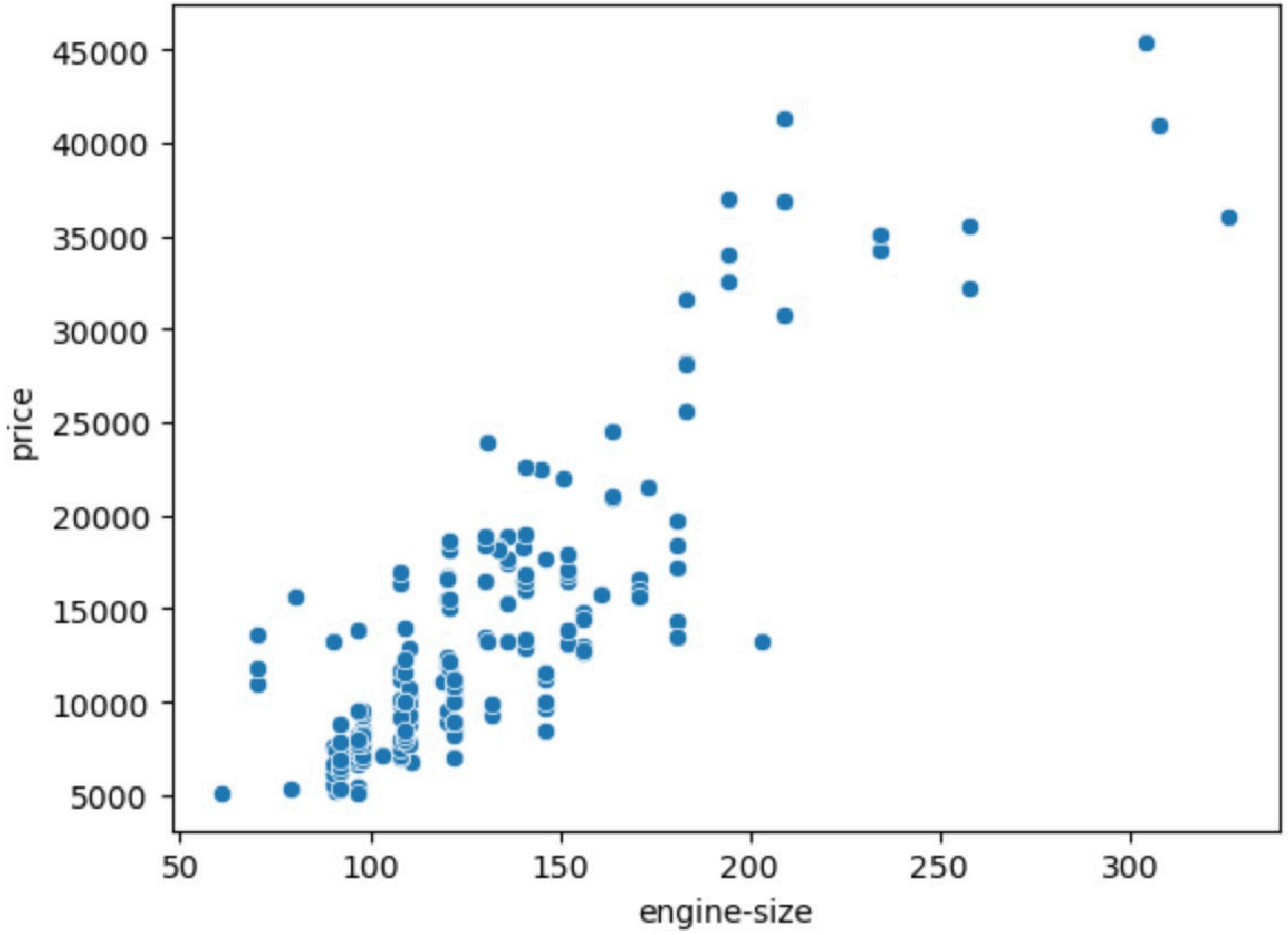
- Here the box for "hardtop & convertible" is positioned higher as compared to the box for "wagon , hatchback , sedan" it means that hardtop & convertible tend to have higher median prices than "wagon , hatchback , sedan".

- some of the sedan are sold at higher price from the typical price range for that style

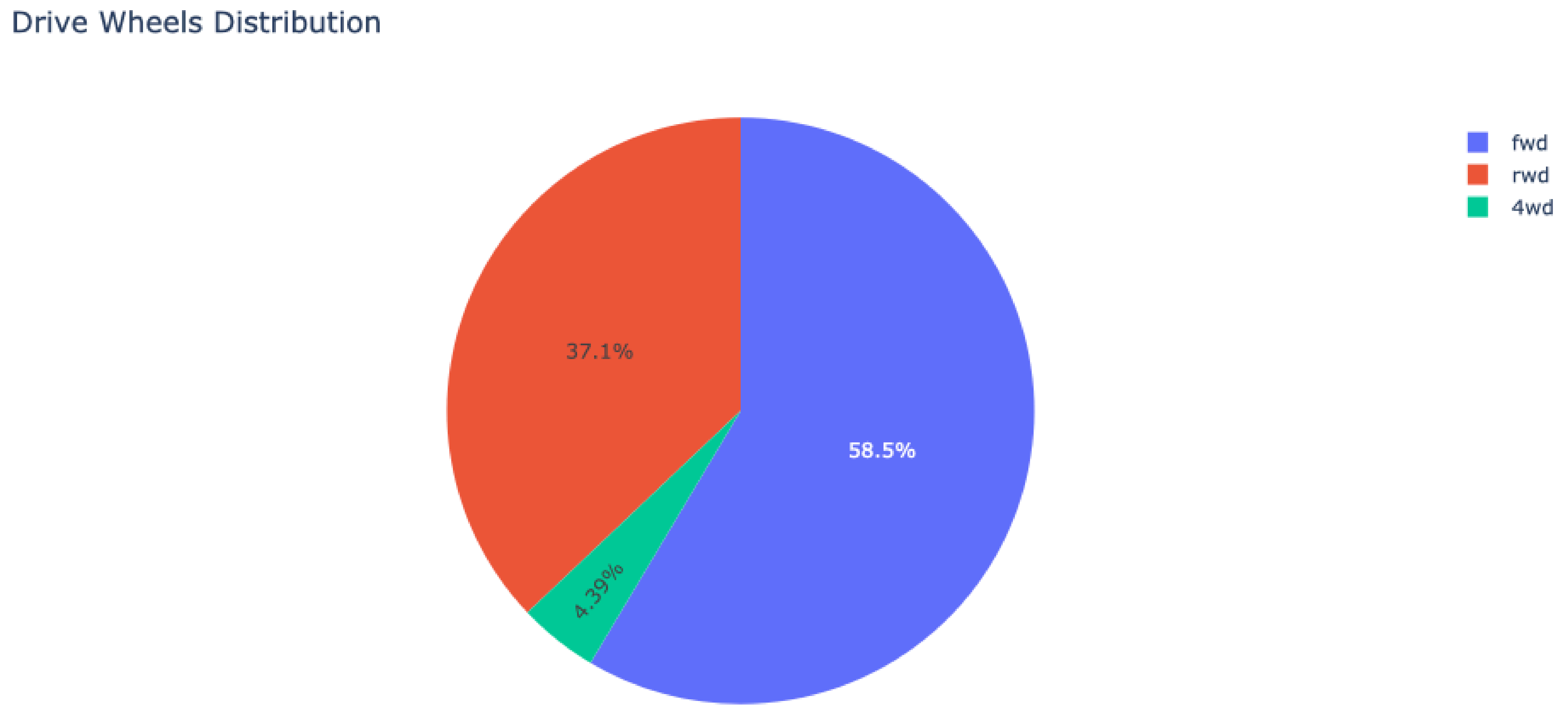
- A taller box suggesting greater price variability within that body style.a shorter box indicates a less price variability



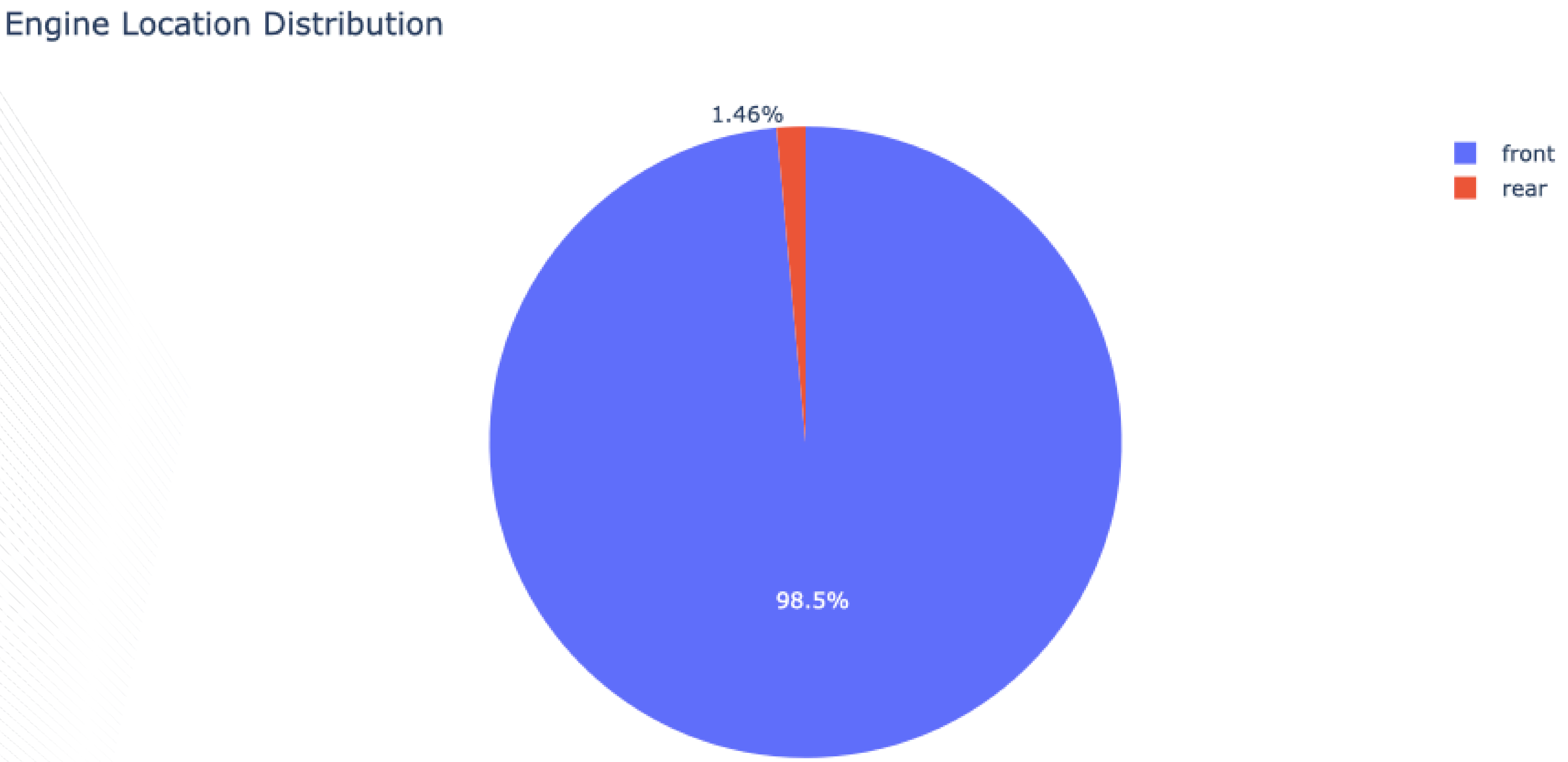
- this different colour boxes allow us to see how prices vary by both body style and fuel type.
- the "sedan" body style is divided into two subgroups gas and diesel , and the prices of cars in these two subgroups are relatively similar
- “convertible , hatchback & hardtop” very mostly sells only gas fuel-type cars



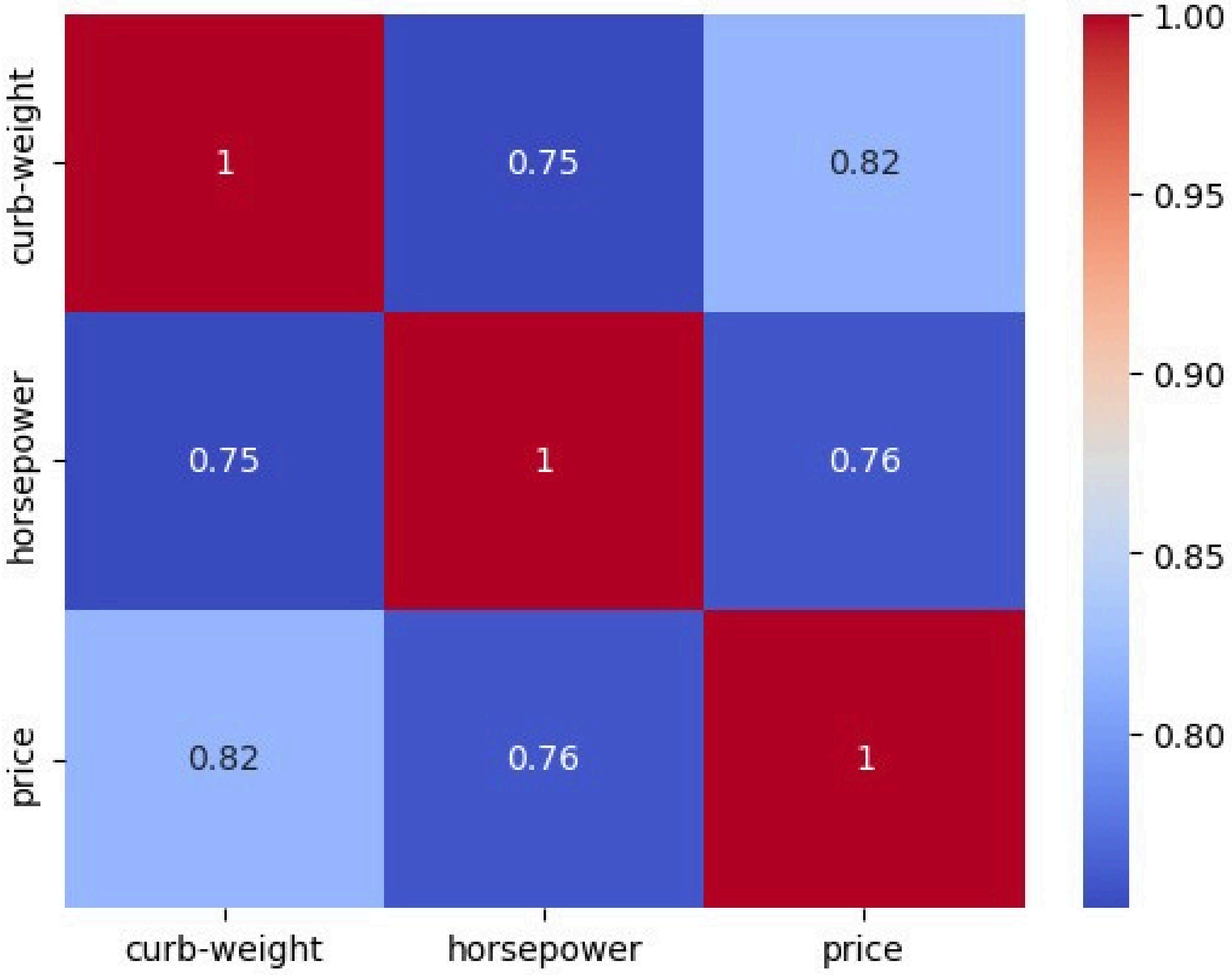
- it is showing the upward trend which means that as the engine-size increases the price also increases and most of the cars has the engine size between 50 to 200 and price from 5000 to 25000



- for most of the cars front wheel drive is prefered (58.5%)
- Down to FWD the second most prefered is rare wheel drive (37.3%)
- 4 wheel drive is prefered less



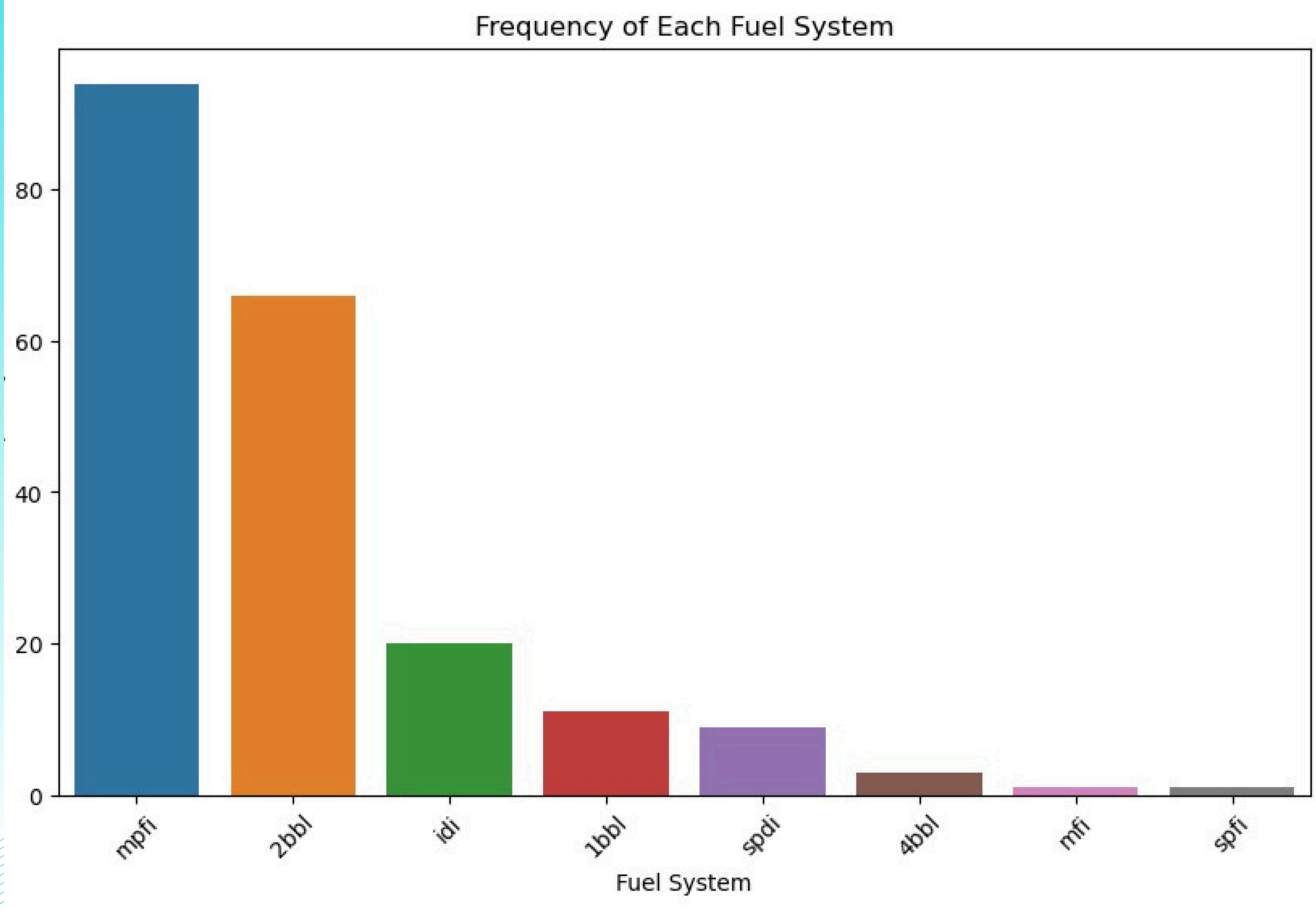
- Engine at front is most preferable in almost every car it is used



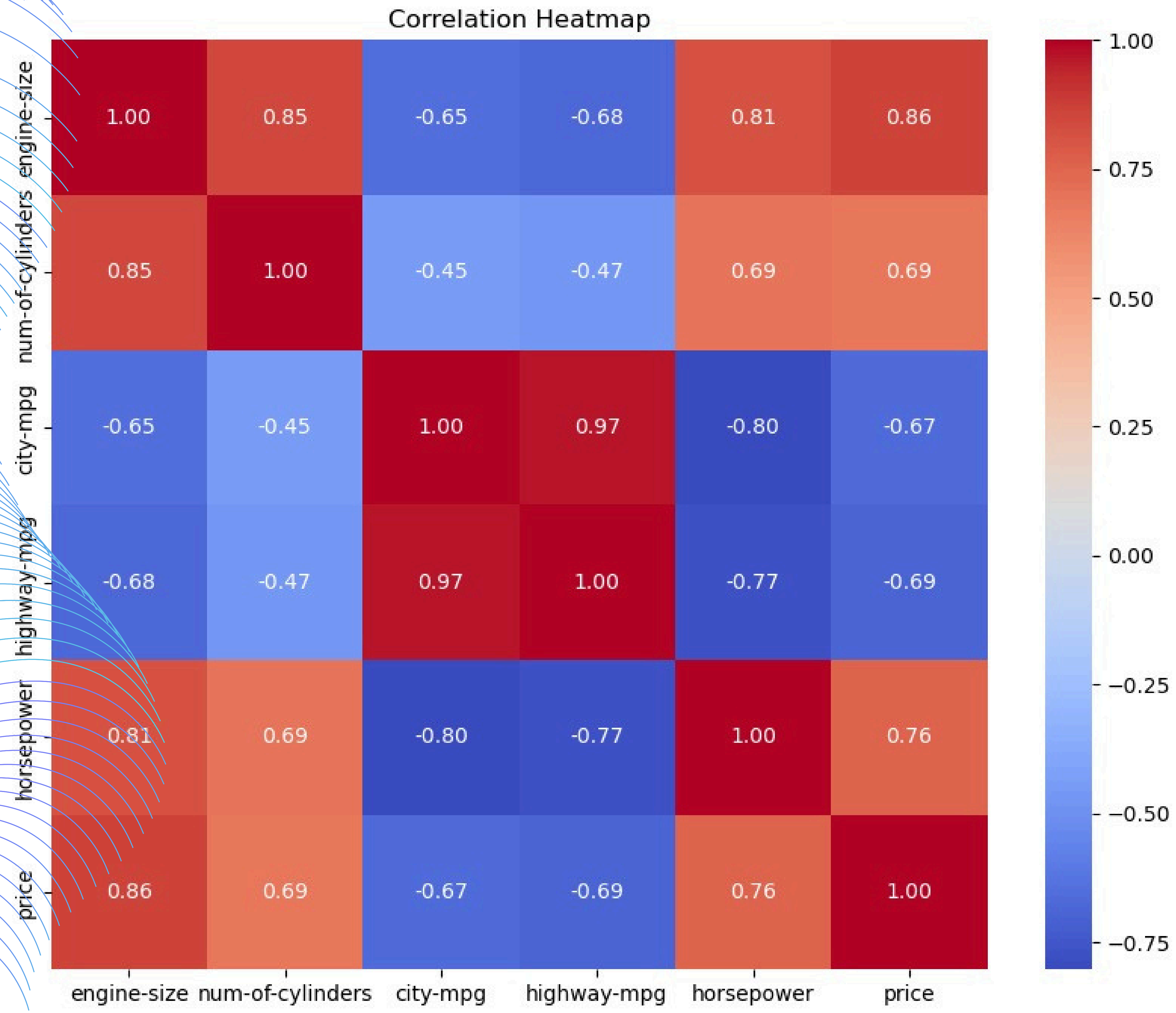
- the numbers inside the boxes shows the corelation between those 2 parameters

- positive correlation means that as one variable (in this case, "curb weight") increases,the other variable (in this case, "horsepower") tends to increase as well. In other words, when the curb weight of a vehicle goes up, its horsepoweralso tends to increase.

- crub-Weight & horsepower also has positive correlation with price



- “mpfi & 2bbl’’ fuel system are mostly used “mfi & spfi” are least perfered



- This heatmap gives the complete info abot the factors affecting milage and also their relation with the price and other features

Conclusion

1. **Sales Leaders:** Toyota leads, while Mercury lags behind.
2. **Fuel Types:** Gasoline-powered cars dominate; diesel is less common.
3. **Engine Sizes:** OHCV has the largest engines (up to 326), while other types vary.
4. **Body Style and Prices:** Hardtops and convertibles have higher median prices.
5. **Price Trend:** As engine size increases, so does the price.
6. **Engine Size Range:** Most cars fall within 50–200 engine size, priced $5000–$25000.
7. **Drive Preference:** FWD is most preferred, followed by RWD.
8. **Correlations:** Curb weight, horsepower, and price are positively correlated.
9. **Common Fuel Systems:** MPFI and 2BBL are commonly used.

Thank you!