Basic Analysis using Numpy and Pandas

In [1]:
```
import numpy as np
```

In [2]:
```
import pandas as pd
```

Pre-processing

In [4]:
```
data=pd.read_csv(r"C:\Users\user\Downloads\fiat500_VehicleSelection_Dataset (1).csv
data
```

Out[4]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon	
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.611559868	
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.24188995	
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11.41784	
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.63460922	
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.49565029	
...	...	...	...	...	...	...	...	...	
1544	NaN	NaN	NaN	NaN	NaN	NaN	NaN	length	
1545	NaN	NaN	NaN	NaN	NaN	NaN	NaN	concat	l
1546	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Null values	
1547	NaN	NaN	NaN	NaN	NaN	NaN	NaN	find	
1548	NaN	NaN	NaN	NaN	NaN	NaN	NaN	search	

1549 rows × 11 columns

In [5]:
```
data.isnull()
```

Out[5]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon	price	Unname
0	False	False	False	False	False	False	False	False	False	Tru
1	False	False	False	False	False	False	False	False	False	Tru
2	False	False	False	False	False	False	False	False	False	Tru
3	False	False	False	False	False	False	False	False	False	Tru
4	False	False	False	False	False	False	False	False	False	Tru
...	...	...	...	...	...	...	...	...	...	
1544	True	True	True	True	True	True	True	False	False	Tru
1545	True	True	True	True	True	True	True	False	False	Tru
1546	True	True	True	True	True	True	True	False	False	Tru
1547	True	True	True	True	True	True	True	False	False	Tru
1548	True	True	True	True	True	True	True	False	False	Tru

1549 rows × 11 columns

In [12]:
```
da=data.fillna(value=5)
da
```

Out[12]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.611559868
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.24188995
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11.41784
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.63460922
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.49565029
...	...	...	...	...	...	...	...	...
1544	5.0	5	5.0	5.0	5.0	5.0	5.000000	length
1545	5.0	5	5.0	5.0	5.0	5.0	5.000000	concat lor
1546	5.0	5	5.0	5.0	5.0	5.0	5.000000	Null values
1547	5.0	5	5.0	5.0	5.0	5.0	5.000000	find
1548	5.0	5	5.0	5.0	5.0	5.0	5.000000	search

1549 rows × 11 columns

In [13]:
```
da.head()
```

Out[13]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon	price
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.611559868	8900
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.24188995	8800
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11.41784	4200
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.63460922	6000
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.49565029	5700

In [14]:
```
da.tail()
```

Out[14]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon	price	Unnamed: 9
1544	5.0	5	5.0	5.0	5.0	5.0	5.0	length	5	5.0
1545	5.0	5	5.0	5.0	5.0	5.0	5.0	concat	lonprice	5.0
1546	5.0	5	5.0	5.0	5.0	5.0	5.0	Null values	NO	5.0
1547	5.0	5	5.0	5.0	5.0	5.0	5.0	find	1	5.0
1548	5.0	5	5.0	5.0	5.0	5.0	5.0	search	1	5.0

In [15]:
```
da.describe()
```

Out[15]:

	ID	engine_power	age_in_days	km	previous_owners	lat	Unnan
count	1549.000000	1549.000000	1549.000000	1549.000000	1549.000000	1549.000000	15
mean	764.071014	51.571336	1639.291801	53016.863138	1.151065	43.267665	
std	447.180625	5.595851	1292.349232	40155.508467	0.527445	3.872986	
min	1.000000	5.000000	5.000000	5.000000	1.000000	5.000000	
25%	377.000000	51.000000	670.000000	19956.000000	1.000000	41.770081	
50%	764.000000	51.000000	1035.000000	38800.000000	1.000000	44.332401	
75%	1151.000000	51.000000	2616.000000	79000.000000	1.000000	45.467960	
max	1538.000000	77.000000	4658.000000	235000.000000	5.000000	46.795612	

In [16]:
```
print(np.shape(da))
```

```
(1549, 11)
```

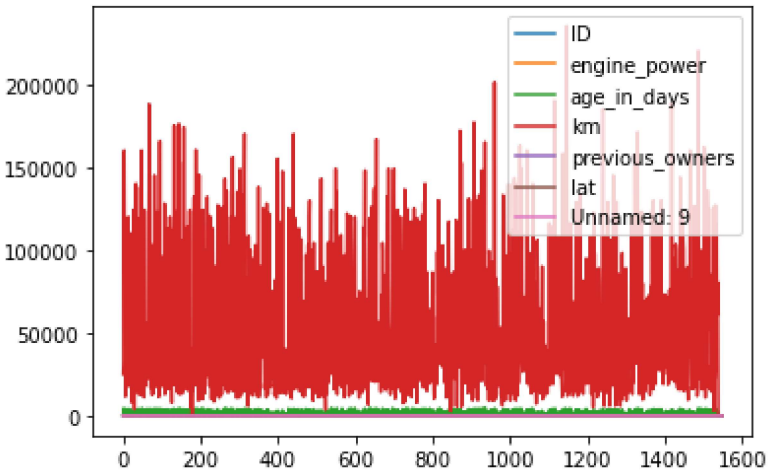
In [17]:
```
print(np.size(da))
```

```
17039
```

Visualization

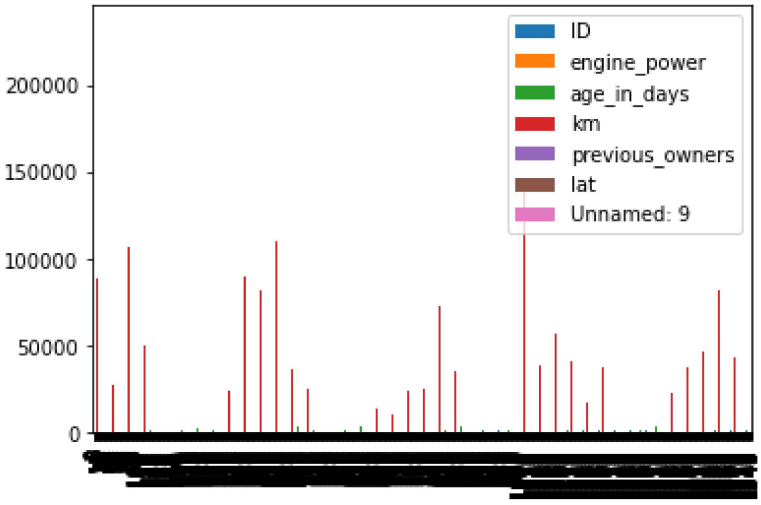
In [23]:
```
da.plot.line()
```

Out[23]: <AxesSubplot:>



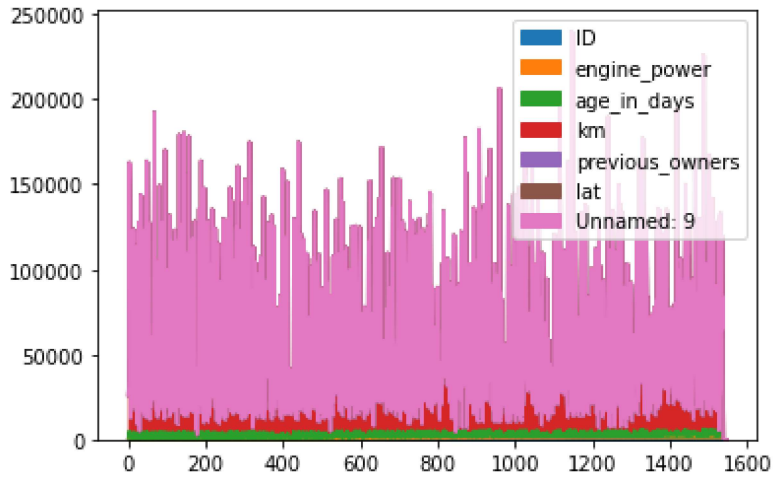
In [24]:
```
da.plot.bar()
```

Out[24]: <AxesSubplot:>



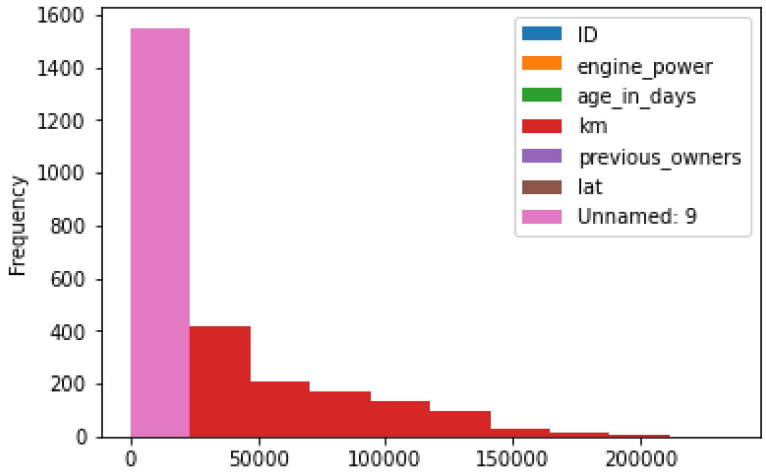
In [25]:
```
da.plot.area()
```

Out[25]: <AxesSubplot:>



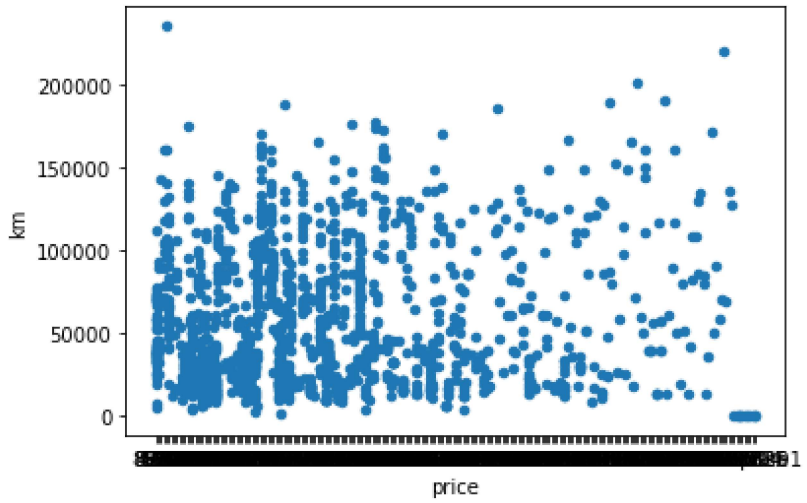
In [26]:
```
da.plot.hist()
```

Out[26]: <AxesSubplot:ylabel='Frequency'>



In [27]:
```
da.plot.scatter(x='price',y='km')
```

Out[27]: <AxesSubplot:xlabel='price', ylabel='km'>



Vector

In [31]:
```
from numpy import linalg as la
```

In [32]:
```
A1=np.array([[2,6,7],[9,6,5]])
print(A1)
```

```
[[2 6 7]
 [9 6 5]]
```

In [33]:
```
print(la.matrix_rank(A1))
```

```
2
```

In [34]:
```
print(np.trace(A1))
```

```
8
```

In [36]:
```
print(A1.T)
```

```
[[2 9]
 [6 6]
 [7 5]]
```

In [38]:
```
A=np.array([[2,3],[4,5]])
print(la.inv(A))
```

```
[[-2.5  1.5]
 [ 2.  -1. ]]
```

In [39]:
```
print(la.eig(A))
```

```
(array([-0.27491722,  7.27491722]), array([[-0.79681209, -0.49436913],
       [ 0.60422718, -0.86925207]]))
```

In [40]:
```
print(la.matrix_power(A,0))
```

```
[[1 0]
 [0 1]]
```

In [41]:
```
print(la.det(A))
```

```
-2.0
```

In [42]:
```
print(np.diag(A))
```

```
[2 5]
```

In [ ]:
```

```