Data set 1

In [1]:

```
import numpy as np
import pandas as pd
```

In [2]:

```
a=pd.read_csv(r"C:\Users\user\Downloads\Vehicle.csv")
```

To print top rows:

In [3]:

```
a.head()
```

Out[3]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lo
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.61155986
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.2418899
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11.4178
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.6346092
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.4956502

To print Last rows:

In [4]:

```
a.tail()
```

Out[4]:

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

Statistical Summary:

In [5]:

```
a.describe()
```

Out[5]:

	ID	engine_power	age_in_days	km	previous_owners	la
count	1538.000000	1538.000000	1538.000000	1538.000000	1538.000000	1538.00000
mean	769.500000	51.904421	1650.980494	53396.011704	1.123537	43.54136
std	444.126671	3.988023	1289.522278	40046.830723	0.416423	2.13351
min	1.000000	51.000000	366.000000	1232.000000	1.000000	36.85583
25%	385.250000	51.000000	670.000000	20006.250000	1.000000	41.80299
50%	769.500000	51.000000	1035.000000	39031.000000	1.000000	44.39409
75%	1153.750000	51.000000	2616.000000	79667.750000	1.000000	45.46796
max	1538.000000	77.000000	4658.000000	235000.000000	4.000000	46.79561

In [124]:

```
a.iloc[193:528]
```

Out[124]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	
193	194.0	pop	51.0	4353.0	76000.0	1.0	45.851662	9.391
194	195.0	lounge	51.0	3227.0	142000.0	1.0	44.294300	9.6744
195	196.0	lounge	51.0	517.0	9150.0	1.0	44.411758	12.20
196	197.0	pop	51.0	1552.0	52026.0	1.0	45.069679	7.704
197	198.0	lounge	51.0	2282.0	145150.0	2.0	45.386841	11.79
...	...	...	...	...	...	...	...	
523	524.0	lounge	51.0	425.0	3600.0	1.0	40.695560	14.48
524	525.0	lounge	51.0	3562.0	80646.0	1.0	41.903221	12.49
525	526.0	lounge	51.0	2616.0	45000.0	1.0	45.537251	9.371
526	527.0	lounge	51.0	2616.0	56400.0	1.0	45.783249	12.48
527	528.0	lounge	51.0	425.0	13111.0	1.0	45.022388	7.586

335 rows × 11 columns

To print no of rows and columns

In [6]:

```
a.shape
```

Out[6]:

```
(1549, 11)
```

To print no of elements

In [7]:

```
a.size
```

Out[7]:

```
17039
```

Missing no of values

In [8]:

```
pd.isna(a)
```

Out[8]:

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

1549 rows × 11 columns

In [31]:

```
c=a.fillna(value=23)
c
```

Out[31]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.6115
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.241
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.634
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.495
...	...	...	...	...	...	...	...	
1544	23.0	23	23.0	23.0	23.0	23.0	23.000000	
1545	23.0	23	23.0	23.0	23.0	23.0	23.000000	
1546	23.0	23	23.0	23.0	23.0	23.0	23.000000	Null
1547	23.0	23	23.0	23.0	23.0	23.0	23.000000	
1548	23.0	23	23.0	23.0	23.0	23.0	23.000000	

1549 rows × 11 columns

In [10]:

```
conda install matplotlib
```

```
Collecting package metadata (current_repodata.json): ...working... done
Solving environment: ...working... done

# All requested packages already installed.


Note: you may need to restart the kernel to use updated packages.



==> WARNING: A newer version of conda exists. <==
  current version: 4.10.1
  latest version: 23.5.2

Please update conda by running

    $ conda update -n base -c defaults conda
```

In [11]:

```
import matplotlib.pyplot as pp
```

In [12]:

```
b=a[['model','km']]
b
```

Out[12]:

	model	km
0	lounge	25000.0
1	pop	32500.0
2	sport	142228.0
3	lounge	160000.0
4	pop	106880.0
...	...	...
1544	NaN	NaN
1545	NaN	NaN
1546	NaN	NaN
1547	NaN	NaN
1548	NaN	NaN

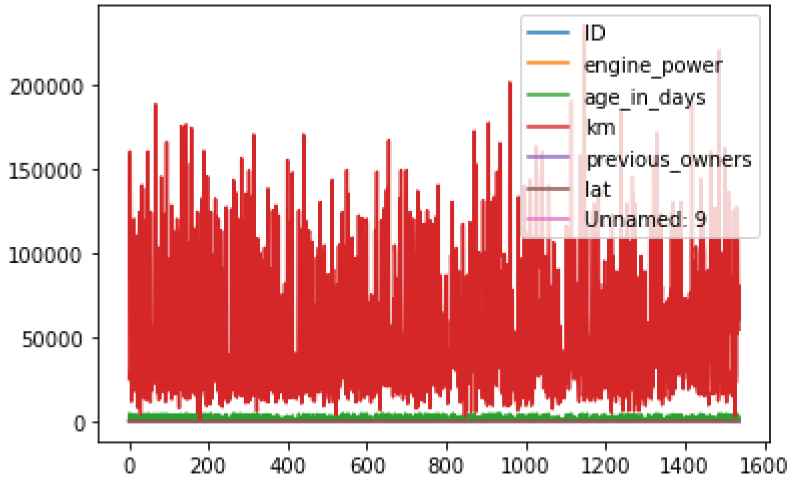
1549 rows × 2 columns

In [13]:

```
a.plot.line()
```

Out[13]:

```
<AxesSubplot:>
```

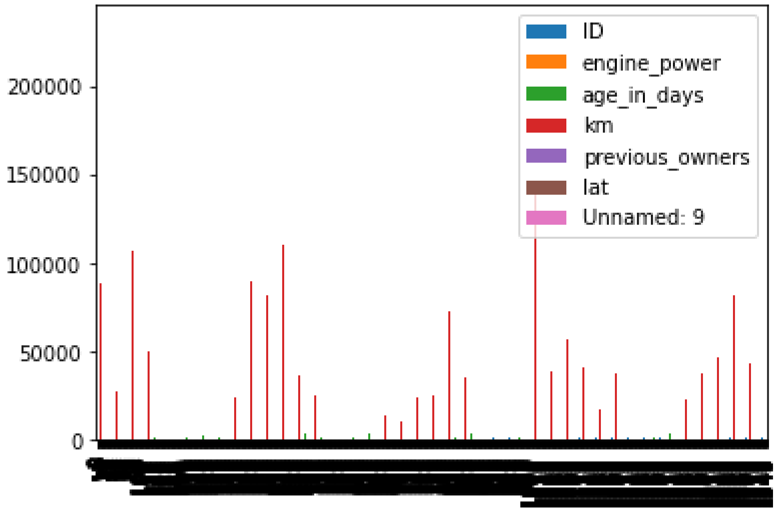


In [14]:

```
a.plot.bar()
```

Out[14]:

```
<AxesSubplot:>
```

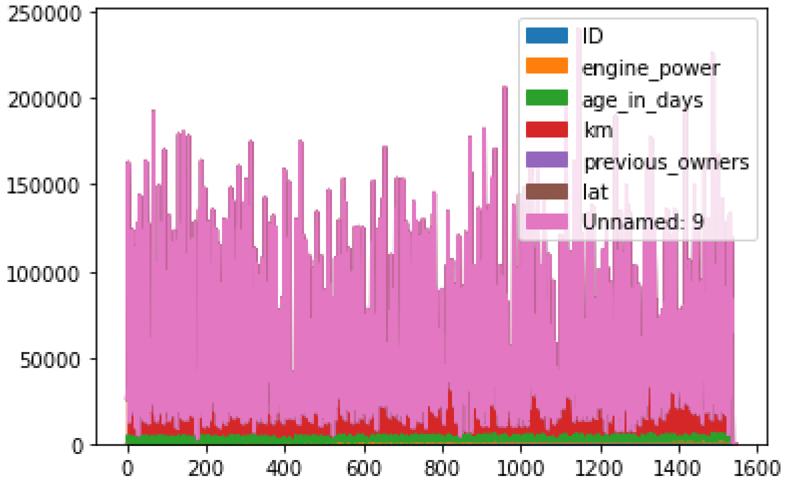


In [15]:

```
a.plot.area()
```

Out[15]:

```
<AxesSubplot:>
```

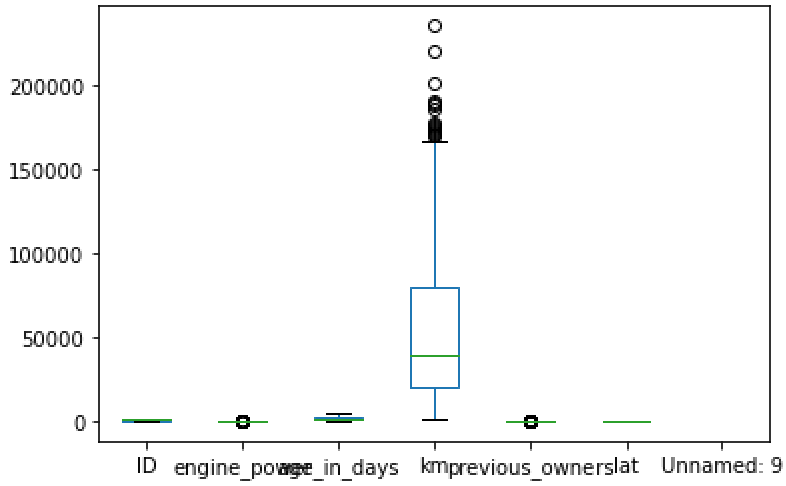


In [16]:

```
a.plot.box()
```

Out[16]:

```
<AxesSubplot:>
```

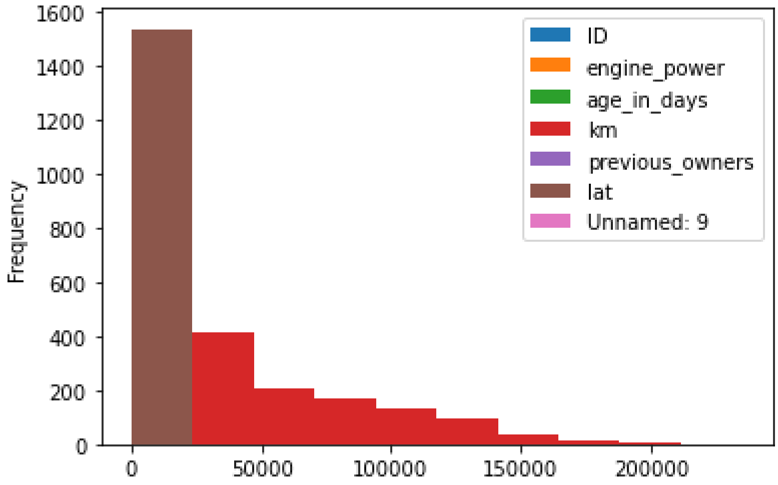


In [17]:

```
a.plot.hist()
```

Out[17]:

```
<AxesSubplot:ylabel='Frequency'>
```

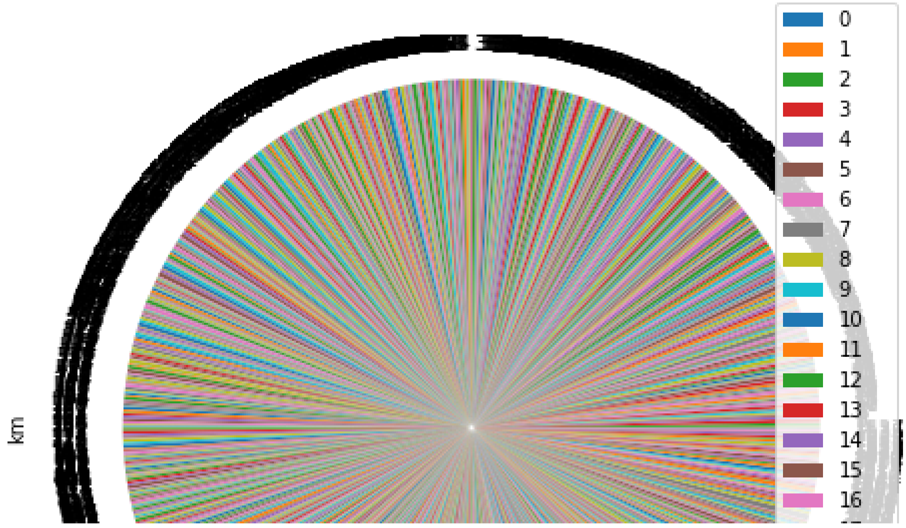


In [18]:

```
a.plot.pie(y='km',figsize=(8,8))
```

Out[18]:

```
<AxesSubplot:ylabel='km'>
```

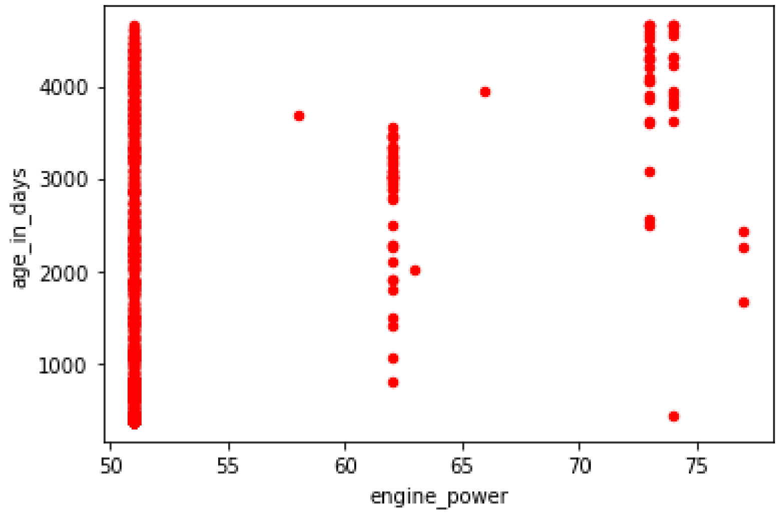


In [19]:

```
a.plot.scatter(x='engine_power',y='age_in_days',color='r')
```

Out[19]:

```
<AxesSubplot:xlabel='engine_power', ylabel='age_in_days'>
```



In [27]:

```
from numpy import linalg as la
```

In [32]:

```
b=c[['engine_power','age_in_days','km']]
b
```

Out[32]:

	engine_power	age_in_days	km
0	51.0	882.0	25000.0
1	51.0	1186.0	32500.0
2	74.0	4658.0	142228.0
3	51.0	2739.0	160000.0
4	73.0	3074.0	106880.0
...	...	...	...
1544	23.0	23.0	23.0
1545	23.0	23.0	23.0
1546	23.0	23.0	23.0
1547	23.0	23.0	23.0
1548	23.0	23.0	23.0

1549 rows × 3 columns

In [35]:

```
print(la.matrix_rank(b))
```

```
3
```

In [36]:

```
print(b.T)
```

```
                    0        1         2         3         4        5     \
engine_power     51.0     51.0      74.0      51.0      73.0     74.0   
age_in_days     882.0   1186.0    4658.0    2739.0    3074.0   3623.0   
km            25000.0  32500.0  142228.0  160000.0  106880.0  70225.0   

                    6        7        8        9     ...  1539  1540  1541  1
542  \
engine_power     51.0     51.0     73.0     51.0  ...  23.0  23.0  23.0  2
3.0   
age_in_days     731.0   1521.0   4049.0   3653.0  ...  23.0  23.0  23.0  2
3.0   
km            11600.0  49076.0  76000.0  89000.0  ...  23.0  23.0  23.0  2
3.0   

              1543  1544  1545  1546  1547  1548  
engine_power  23.0  23.0  23.0  23.0  23.0  23.0  
age_in_days   23.0  23.0  23.0  23.0  23.0  23.0  
km            23.0  23.0  23.0  23.0  23.0  23.0  

[3 rows x 1549 columns]
```

In [38]:

```
print(np.trace(b))
```

```
143465.0
```

In [41]:

```
print(np.diag(b))
```

```
[5.10000e+01 1.18600e+03 1.42228e+05]
```

In [20]:

```
print(a.mean())
```

```
ID                    769.500000
engine_power           51.904421
age_in_days          1650.980494
km                  53396.011704
previous_owners         1.123537
lat                    43.541361
Unnamed: 9                   NaN
dtype: float64
```

In [21]:

```
print(a.median())
```

```
ID                    769.500000
engine_power           51.000000
age_in_days          1035.000000
km                  39031.000000
previous_owners         1.000000
lat                    44.394096
Unnamed: 9                   NaN
dtype: float64
```

In [22]:

```
print(a.mode())
```

```
          ID   model  engine_power  age_in_days       km  previous_owners  
\
0        1.0  lounge          51.0        366.0  17000.0              1.0
1        2.0     NaN           NaN        790.0      NaN              NaN
2        3.0     NaN           NaN          NaN      NaN              NaN
3        4.0     NaN           NaN          NaN      NaN              NaN
4        5.0     NaN           NaN          NaN      NaN              NaN
...      ...     ...           ...          ...      ...              ...
1533  1534.0     NaN           NaN          NaN      NaN              NaN
1534  1535.0     NaN           NaN          NaN      NaN              NaN
1535  1536.0     NaN           NaN          NaN      NaN              NaN
1536  1537.0     NaN           NaN          NaN      NaN              NaN
1537  1538.0     NaN           NaN          NaN      NaN              NaN

            lat          lon  price  Unnamed: 9 Unnamed: 10
0     41.903221  12.49565029  10500         NaN      >10000
1           NaN          NaN    NaN         NaN         NaN
2           NaN          NaN    NaN         NaN         NaN
3           NaN          NaN    NaN         NaN         NaN
4           NaN          NaN    NaN         NaN         NaN
...         ...          ...    ...         ...         ...
1533        NaN          NaN    NaN         NaN         NaN
1534        NaN          NaN    NaN         NaN         NaN
1535        NaN          NaN    NaN         NaN         NaN
1536        NaN          NaN    NaN         NaN         NaN
1537        NaN          NaN    NaN         NaN         NaN

[1538 rows x 11 columns]
```

In [23]:

```
print(a.count())
```

```
ID                   1538
model                1538
engine_power         1538
age_in_days          1538
km                   1538
previous_owners      1538
lat                  1538
lon                  1549
price                1549
Unnamed: 9              0
Unnamed: 10             1
dtype: int64
```

In [24]:

```
print(a.min())
```

```
ID                         1.0
engine_power              51.0
age_in_days              366.0
km                      1232.0
previous_owners            1.0
lat                  36.855839
lon                10.00240993
price                        1
Unnamed: 9                 NaN
dtype: object
```

In [25]:

```
print(a.max())
```

```
ID                    1538.0
engine_power            77.0
age_in_days           4658.0
km                  235000.0
previous_owners          4.0
lat                46.795612
lon                    sumif
price               lonprice
Unnamed: 9               NaN
dtype: object
```

In [45]:

```
print(b.sum())
```

```
engine_power       80082.0
age_in_days      2539461.0
km              82123319.0
dtype: float64
```

In [46]:

```
print(b.cumsum())
```

```
      engine_power  age_in_days          km
0             51.0        882.0     25000.0
1            102.0       2068.0     57500.0
2            176.0       6726.0    199728.0
3            227.0       9465.0    359728.0
4            300.0      12539.0    466608.0
...            ...          ...         ...
1544       79990.0    2539369.0  82123227.0
1545       80013.0    2539392.0  82123250.0
1546       80036.0    2539415.0  82123273.0
1547       80059.0    2539438.0  82123296.0
1548       80082.0    2539461.0  82123319.0

[1549 rows x 3 columns]
```

In [47]:

```
print(b.describe())
```

```
       engine_power  age_in_days             km
count   1549.000000  1549.000000    1549.000000
mean      51.699161  1639.419626   53016.990962
std        4.656815  1292.188356   40155.339637
min       23.000000    23.000000      23.000000
25%       51.000000   670.000000   19956.000000
50%       51.000000  1035.000000   38800.000000
75%       51.000000  2616.000000   79000.000000
max       77.000000  4658.000000  235000.000000
```

In [49]:

```
from numpy import cov
```

In [50]:

```
x=b['engine_power']
y=b['age_in_days']
print(cov(x,y))
```

```
[[2.16859231e+01 1.96181818e+03]
 [1.96181818e+03 1.66975075e+06]]
```

In [51]:

```
from scipy.stats import pearsonr
from scipy.stats import spearmanr
```

In [52]:

```
print(pearsonr(x,y))
```

```
(0.32601979915182855, 1.1029299003522821e-39)
```

In [53]:

```
print(spearmanr(x,y))
```

```
SpearmanrResult(correlation=0.30435214309064385, pvalue=1.469623052988258e
-34)
```

Data set 2

In [54]:

```
a1=pd.read_csv(r"C:\Users\user\Downloads\Vehicle.csv")
```

In [55]:

```
a1.head(100)
```

Out[55]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.6115
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.241
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11.4
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.634
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.495
...	...	...	...	...	...	...	...	
95	96.0	sport	51.0	4292.0	165600.0	1.0	44.715408	11.308
96	97.0	pop	51.0	1066.0	28000.0	1.0	41.769051	12.662
97	98.0	sport	51.0	2009.0	86000.0	2.0	40.633171	17.634
98	99.0	lounge	51.0	456.0	18592.0	2.0	45.393600	10.482
99	100.0	pop	51.0	731.0	41558.0	2.0	45.571220	9.1591

100 rows × 11 columns

In [56]:

```
a1.tail(100)
```

Out[56]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	
1449	1450.0	lounge	51.0	3105.0	99999.0	1.0	40.947170	14.37
1450	1451.0	pop	51.0	397.0	15000.0	1.0	41.903221	12.49
1451	1452.0	sport	62.0	3166.0	89000.0	1.0	45.724380	11.76
1452	1453.0	lounge	51.0	1247.0	75000.0	1.0	41.683800	12.77
1453	1454.0	pop	51.0	1186.0	42900.0	1.0	39.214539	9.110
...	...	...	...	...	...	...	...	
1544	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1545	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1546	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Nu
1547	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1548	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

100 rows × 11 columns

In [57]:

```
a1.describe()
```

Out[57]:

	ID	engine_power	age_in_days	km	previous_owners	la
count	1538.000000	1538.000000	1538.000000	1538.000000	1538.000000	1538.00000
mean	769.500000	51.904421	1650.980494	53396.011704	1.123537	43.54136
std	444.126671	3.988023	1289.522278	40046.830723	0.416423	2.13351
min	1.000000	51.000000	366.000000	1232.000000	1.000000	36.85583
25%	385.250000	51.000000	670.000000	20006.250000	1.000000	41.80299
50%	769.500000	51.000000	1035.000000	39031.000000	1.000000	44.39409
75%	1153.750000	51.000000	2616.000000	79667.750000	1.000000	45.46796
max	1538.000000	77.000000	4658.000000	235000.000000	4.000000	46.79561

In [123]:

```
a1.iloc[500:1000]
```

Out[123]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	
500	501.0	lounge	51.0	2527.0	25191.0	1.0	41.903221	12.49
501	502.0	lounge	51.0	1155.0	38197.0	1.0	41.903221	12.49
502	503.0	pop	51.0	1461.0	87066.0	1.0	44.508839	11.46
503	504.0	lounge	51.0	456.0	21200.0	1.0	44.063129	12.44
504	505.0	lounge	51.0	3378.0	25800.0	1.0	41.462730	12.90
...	...	...	...	...	...	...	...	
995	996.0	pop	51.0	701.0	13947.0	1.0	45.438110	12.31
996	997.0	lounge	51.0	2192.0	106000.0	1.0	40.563889	17.33
997	998.0	sport	51.0	3470.0	139750.0	3.0	41.232948	16.29
998	999.0	pop	51.0	731.0	56000.0	3.0	40.840141	14.25
999	1000.0	pop	51.0	2070.0	97677.0	1.0	42.514408	14.14

500 rows × 11 columns

In [59]:

```
a1.shape
```

Out[59]:

```
(1549, 11)
```

In [60]:

```
a1.size
```

Out[60]:

```
17039
```

In [61]:

```
a1.isna()
```

Out[61]:

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

1549 rows × 11 columns

In [63]:

```
b1=a1.fillna(value=44)
b1
```

Out[63]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.6115
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.241
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.634
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.495
...	...	...	...	...	...	...	...	
1544	44.0	44	44.0	44.0	44.0	44.0	44.000000	
1545	44.0	44	44.0	44.0	44.0	44.0	44.000000	
1546	44.0	44	44.0	44.0	44.0	44.0	44.000000	Null
1547	44.0	44	44.0	44.0	44.0	44.0	44.000000	
1548	44.0	44	44.0	44.0	44.0	44.0	44.000000	

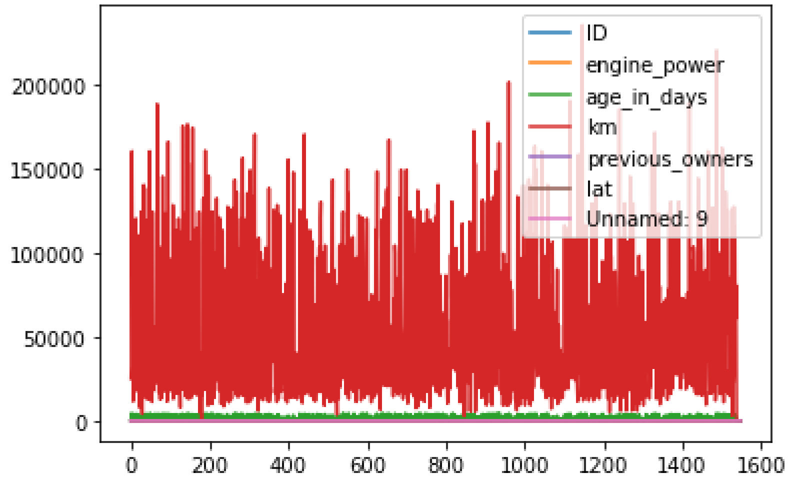
1549 rows × 11 columns

In [64]:

```
b1.plot.line()
```

Out[64]:

```
<AxesSubplot:>
```

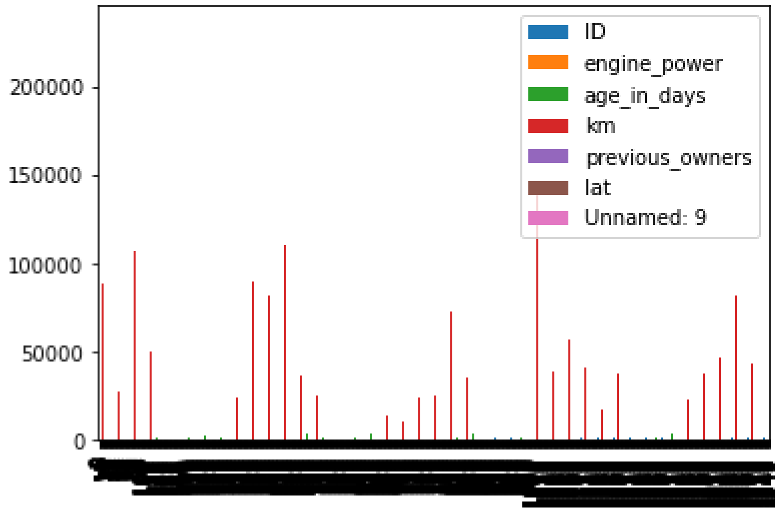


In [65]:

```
b1.plot.bar()
```

Out[65]:

```
<AxesSubplot:>
```

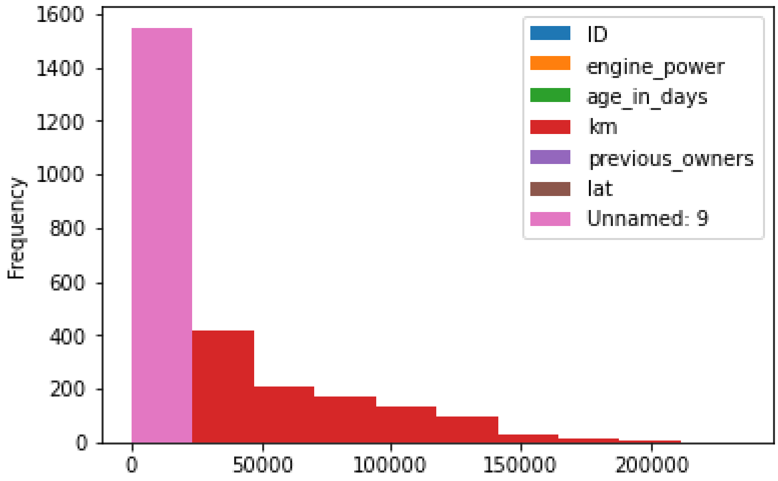


In [66]:

```
b1.plot.hist()
```

Out[66]:

```
<AxesSubplot:ylabel='Frequency'>
```

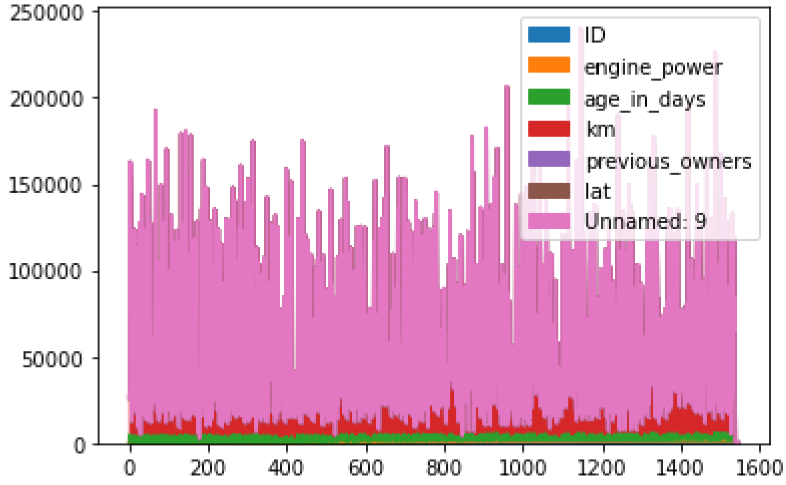


In [67]:

```
b1.plot.area()
```

Out[67]:

```
<AxesSubplot:>
```

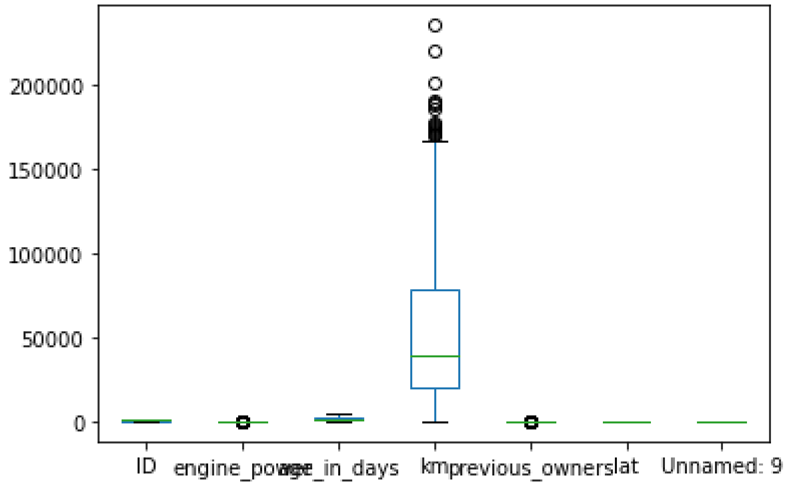


In [68]:

```
b1.plot.box()
```

Out[68]:

```
<AxesSubplot:>
```

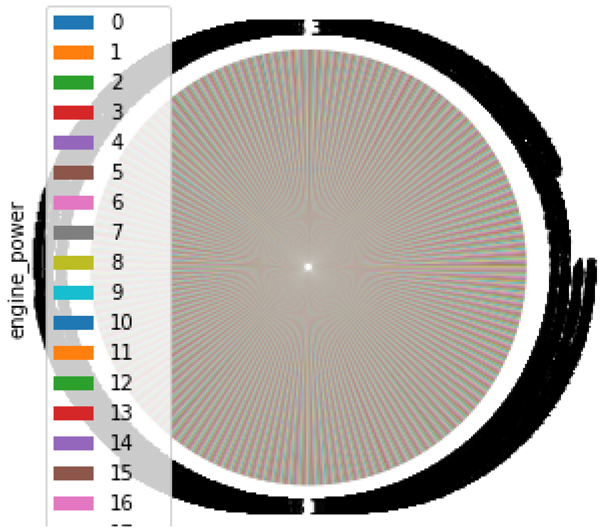


In [69]:

```
b1.plot.pie(y='engine_power',figsize=(5,5))
```

Out[69]:

```
<AxesSubplot:ylabel='engine_power'>
```

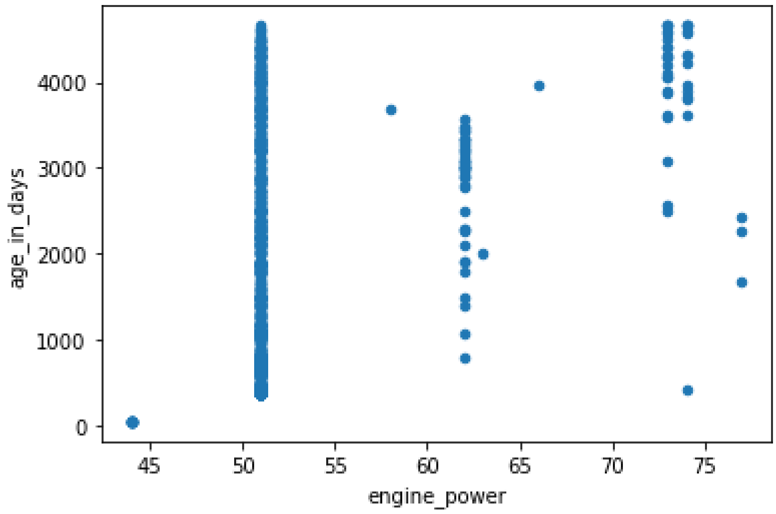


In [70]:

```
b1.plot.scatter(x='engine_power',y='age_in_days')
```

Out[70]:

```
<AxesSubplot:xlabel='engine_power', ylabel='age_in_days'>
```



In [72]:

```
c1=b1[['engine_power','age_in_days']]
c1
```

Out[72]:

	engine_power	age_in_days
0	51.0	882.0
1	51.0	1186.0
2	74.0	4658.0
3	51.0	2739.0
4	73.0	3074.0
...	...	...
1544	44.0	44.0
1545	44.0	44.0
1546	44.0	44.0
1547	44.0	44.0
1548	44.0	44.0

1549 rows × 2 columns

In [73]:

```
print(la.matrix_rank(c1))
```

```
2
```

In [74]:

```
print(c1.T)
```

```
               0       1       2       3       4       5      6       7
\
engine_power   51.0    51.0    74.0    51.0    73.0    74.0   51.0    51.0
age_in_days   882.0  1186.0  4658.0  2739.0  3074.0  3623.0  731.0  1521.0

                8       9   ...  1539  1540  1541  1542  1543  1544  154
5  \
engine_power    73.0    51.0  ...  44.0  44.0  44.0  44.0  44.0  44.0  44.
0
age_in_days   4049.0  3653.0  ...  44.0  44.0  44.0  44.0  44.0  44.0  44.
0

              1546  1547  1548
engine_power  44.0  44.0  44.0
age_in_days   44.0  44.0  44.0

[2 rows x 1549 columns]
```

In [76]:

```
print(np.trace(c1))
```

```
1237.0
```

In [77]:

```
print(np.diag(c1))
```

```
[  51. 1186.]
```

In [78]:

```
print(b1.mean())
```

```
ID                   764.347966
engine_power          51.848289
age_in_days         1639.568754
km                 53017.140090
previous_owners        1.428018
lat                   43.544618
Unnamed: 9            44.000000
dtype: float64
```

In [79]:

```
print(b1.median())
```

```
ID                   764.000000
engine_power          51.000000
age_in_days         1035.000000
km                 38800.000000
previous_owners        1.000000
lat                   44.332401
Unnamed: 9            44.000000
dtype: float64
```

In [80]:

```
print(b1.mode())
```

```
     ID   model  engine_power  age_in_days       km  previous_owners  \
0  44.0  lounge          51.0        366.0  17000.0              1.0
1   NaN     NaN           NaN        790.0      NaN              NaN

         lat          lon  price  Unnamed: 9 Unnamed: 10
0  41.903221  12.49565029  10500        44.0          44
1        NaN          NaN    NaN         NaN         NaN
```

In [81]:

```
print(c1.sum())
```

```
engine_power       80313.0
age_in_days      2539692.0
dtype: float64
```

In [82]:

```
print(c1.cumsum())
```

```
      engine_power  age_in_days
0             51.0        882.0
1            102.0       2068.0
2            176.0       6726.0
3            227.0       9465.0
4            300.0      12539.0
...            ...          ...
1544       80137.0    2539516.0
1545       80181.0    2539560.0
1546       80225.0    2539604.0
1547       80269.0    2539648.0
1548       80313.0    2539692.0

[1549 rows x 2 columns]
```

In [83]:

```
print(c1.count())
```

```
engine_power    1549
age_in_days     1549
dtype: int64
```

In [84]:

```
print(c1.min())
```

```
engine_power    44.0
age_in_days     44.0
dtype: float64
```

In [85]:

```
print(c1.max())
```

```
engine_power      77.0
age_in_days     4658.0
dtype: float64
```

In [86]:

```
x1=c['engine_power']
y1=c['age_in_days']
print(cov(x1,y1))
```

```
[[2.16859231e+01 1.96181818e+03]
 [1.96181818e+03 1.66975075e+06]]
```

In [87]:

```
print(pearsonr(x1,y1))
```

```
(0.32601979915182855, 1.1029299003522821e-39)
```

In [88]:

```
print(spearmanr(x1,y1))
```

```
SpearmanrResult(correlation=0.30435214309064385, pvalue=1.469623052988258e
-34)
```

Data Set 3

In [89]:

```
a2=pd.read_csv(r"C:\Users\user\Downloads\4_drug200.csv")
```

In [90]:

```
a2.head(100)
```

Out[90]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
...	...	...	...	...	...	...
95	36	M	LOW	NORMAL	11.424	drugX
96	58	F	LOW	HIGH	38.247	drugY
97	56	F	HIGH	HIGH	25.395	drugY
98	20	M	HIGH	NORMAL	35.639	drugY
99	15	F	HIGH	NORMAL	16.725	drugY

100 rows × 6 columns

In [91]:

```
a2.tail(100)
```

Out[91]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
100	31	M	HIGH	NORMAL	11.871	drugA
101	45	F	HIGH	HIGH	12.854	drugA
102	28	F	LOW	HIGH	13.127	drugC
103	56	M	NORMAL	HIGH	8.966	drugX
104	22	M	HIGH	NORMAL	28.294	drugY
...	...	...	...	...	...	...
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

100 rows × 6 columns

In [92]:

```
a2.describe()
```

Out[92]:

	Age	Na_to_K
count	200.000000	200.000000
mean	44.315000	16.084485
std	16.544315	7.223956
min	15.000000	6.269000
25%	31.000000	10.445500
50%	45.000000	13.936500
75%	58.000000	19.380000
max	74.000000	38.247000

In [122]:

```
a2.iloc[50:150]
```

Out[122]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
50	58	F	HIGH	HIGH	19.416	drugY
51	67	M	NORMAL	NORMAL	10.898	drugX
52	62	M	LOW	NORMAL	27.183	drugY
53	24	F	HIGH	NORMAL	18.457	drugY
54	68	F	HIGH	NORMAL	10.189	drugB
...	...	...	...	...	...	...
145	61	M	NORMAL	HIGH	9.443	drugX
146	37	F	LOW	NORMAL	12.006	drugX
147	26	F	HIGH	NORMAL	12.307	drugA
148	61	F	LOW	NORMAL	7.340	drugX
149	22	M	LOW	HIGH	8.151	drugC

100 rows × 6 columns

In [93]:

```
a2.shape
```

Out[93]:

```
(200, 6)
```

In [94]:

```
a2.size
```

Out[94]:

```
1200
```

In [95]:

```
a2.isna()
```

Out[95]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	False	False	False	False	False	False
1	False	False	False	False	False	False
2	False	False	False	False	False	False
3	False	False	False	False	False	False
4	False	False	False	False	False	False
...	...	...	...	...	...	...
195	False	False	False	False	False	False
196	False	False	False	False	False	False
197	False	False	False	False	False	False
198	False	False	False	False	False	False
199	False	False	False	False	False	False

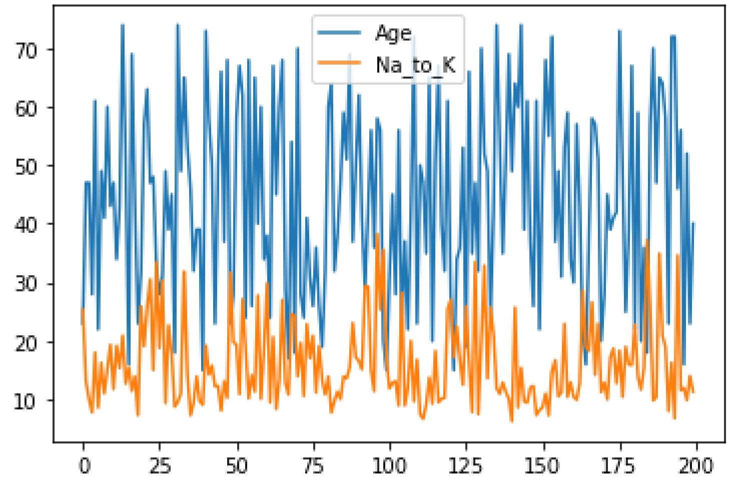
200 rows × 6 columns

In [96]:

```
a2.plot.line()
```

Out[96]:

```
<AxesSubplot:>
```

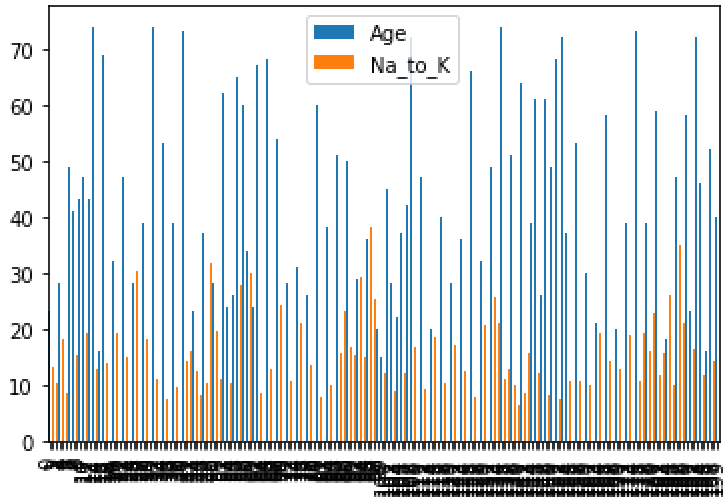


In [97]:

```
a2.plot.bar()
```

Out[97]:

```
<AxesSubplot:>
```

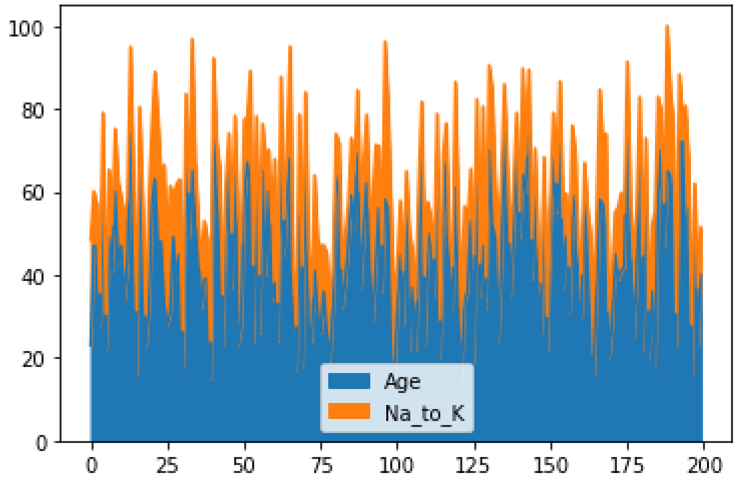


In [98]:

```
a2.plot.area()
```

Out[98]:

```
<AxesSubplot:>
```

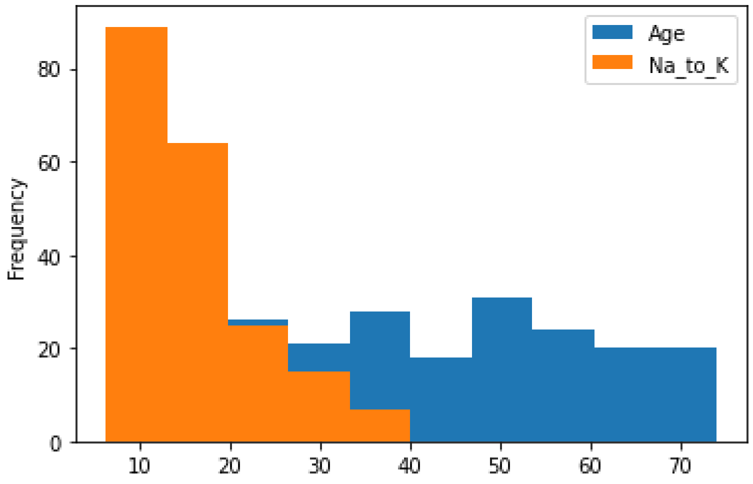


In [99]:

```
a2.plot.hist()
```

Out[99]:

```
<AxesSubplot:ylabel='Frequency'>
```

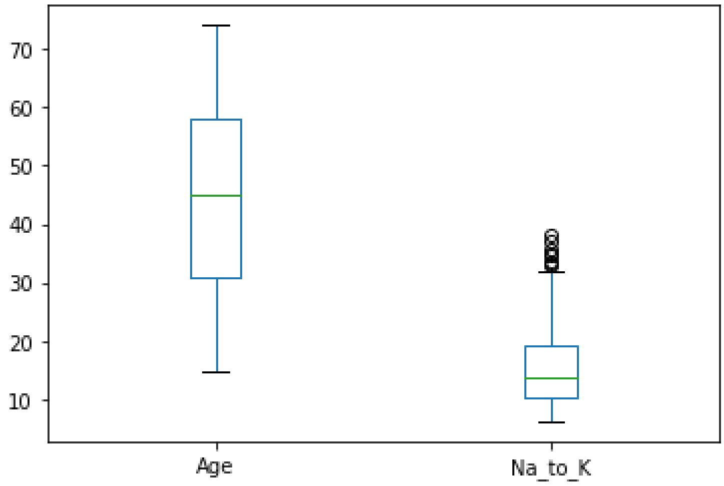


In [100]:

```
a2.plot.box()
```

Out[100]:

```
<AxesSubplot:>
```

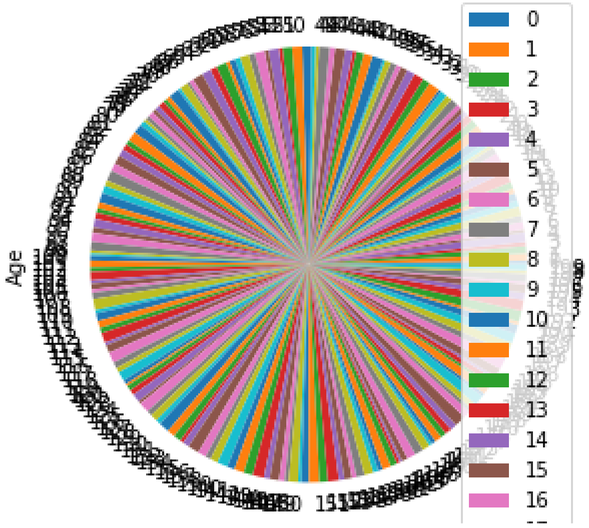


In [101]:

```
a2.plot.pie(y='Age',figsize=(5,5))
```

Out[101]:

```
<AxesSubplot:ylabel='Age'>
```

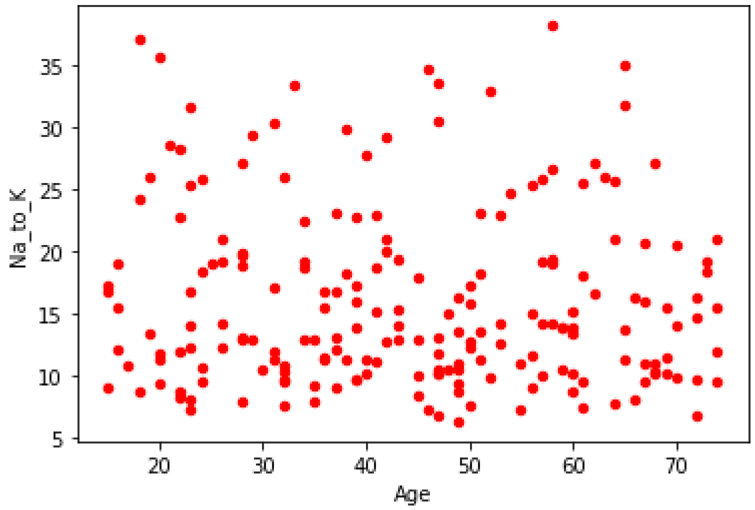


In [103]:

```
a2.plot.scatter(x='Age',y='Na_to_K',color='r')
```

Out[103]:

```
<AxesSubplot:xlabel='Age', ylabel='Na_to_K'>
```



In [106]:

```
b2=a2[['Age','Na_to_K']]
b2
```

Out[106]:

	Age	Na_to_K
0	23	25.355
1	47	13.093
2	47	10.114
3	28	7.798
4	61	18.043
...	...	...
195	56	11.567
196	16	12.006
197	52	9.894
198	23	14.020
199	40	11.349

200 rows × 2 columns

In [107]:

```
print(la.matrix_rank(b2))
```

```
2
```

In [108]:

```
print(b2.T)
```

```
            0       1       2       3       4       5       6       7    \
Age      23.000  47.000  47.000  28.000  61.000  22.000  49.000  41.000   
Na_to_K  25.355  13.093  10.114   7.798  18.043   8.607  16.275  11.037   

            8       9    ...     190     191    192     193     194     19
5  \
Age      60.000  43.000  ...  58.000  23.000  72.00  72.000  46.000  56.00
0   
Na_to_K  15.171  19.368  ...  18.991   8.011  16.31   6.769  34.686  11.56
7   

            196     197    198     199  
Age      16.000  52.000  23.00  40.000  
Na_to_K  12.006   9.894  14.02  11.349  

[2 rows x 200 columns]
```

In [109]:

```
print(np.trace(b2))
```

```
36.093
```

In [110]:

```
print(np.diag(b2))
```

```
[23.     13.093]
```

In [111]:

```
print(a2.mean())
```

```
Age          44.315000
Na_to_K      16.084485
dtype: float64
```

In [112]:

```
print(a2.median())
```

```
Age          45.0000
Na_to_K      13.9365
dtype: float64
```

In [113]:

```
print(a2.mode())
```

```
    Age  Sex    BP Cholesterol  Na_to_K   Drug
0  47.0    M  HIGH        HIGH   12.006  drugY
1   NaN  NaN   NaN         NaN   18.295    NaN
```

In [114]:

```
print(b2.sum())
```

```
Age          8863.000
Na_to_K      3216.897
dtype: float64
```

In [115]:

```
print(b2.cumsum())
```

```
      Age   Na_to_K
0      23    25.355
1      70    38.448
2     117    48.562
3     145    56.360
4     206    74.403
..    ...       ...
195  8732  3169.628
196  8748  3181.634
197  8800  3191.528
198  8823  3205.548
199  8863  3216.897

[200 rows x 2 columns]
```

In [116]:

```
print(b2.count())
```

```
Age        200
Na_to_K    200
dtype: int64
```

In [117]:

```
print(b2.min())
```

```
Age        15.000
Na_to_K     6.269
dtype: float64
```

In [118]:

```
print(b2.max())
```

```
Age        74.000
Na_to_K    38.247
dtype: float64
```

In [119]:

```
x2=b2['Age']
y2=b2['Na_to_K']
print(cov(x2,y2))
```

```
[[273.71434673  -7.54375153]
 [ -7.54375153  52.18553348]]
```

In [120]:

```
print(pearsonr(x2,y2))
```

```
(-0.06311949726772592, 0.3745756399034559)
```

In [121]:

```
print(spearmanr(x2,y2))
```

```
SpearmanrResult(correlation=-0.047273882688479915, pvalue=0.50622005813874
18)
```

In [ ]:

```

```