In [2]:
```
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")
```

In [3]:
```
data= pd.read_csv('/home/placement/Desktop/fiat500.csv')
```

In [4]:
```
print(data)
```

```
        ID   model  engine_power  age_in_days      km  previous_owners  \
0        1  lounge            51          882   25000                1
1        2     pop            51         1186   32500                1
2        3   sport            74         4658  142228                1
3        4  lounge            51         2739  160000                1
4        5     pop            73         3074  106880                1
...    ...     ...           ...          ...     ...              ...
1533  1534   sport            51         3712  115280                1
1534  1535  lounge            74         3835  112000                1
1535  1536     pop            51         2223   60457                1
1536  1537  lounge            51         2557   80750                1
1537  1538     pop            51         1766   54276                1

            lat        lon  price
0     44.907242   8.611560   8900
1     45.666359  12.241890   8800
2     45.503300  11.417840   4200
3     40.633171  17.634609   6000
4     41.903221  12.495650   5700
...         ...        ...    ...
1533  45.069679   7.704920   5200
1534  45.845692   8.666870   4600
1535  45.481541   9.413480   7500
1536  45.000702   7.682270   5990
1537  40.323410  17.568270   7900

[1538 rows x 9 columns]
```

In [5]:

```
data1=data[data['km']<=50000]
data1
```

Out[5]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon	price
0	1	lounge	51	882	25000	1	44.907242	8.61156	8900
1	2	pop	51	1186	32500	1	45.666359	12.24189	8800
6	7	lounge	51	731	11600	1	44.907242	8.61156	10750
7	8	lounge	51	1521	49076	1	41.903221	12.49565	9190
10	11	pop	51	790	43286	1	40.871429	14.43896	8950
...	...	...	...	...	...	...	...	...	...
1525	1526	lounge	51	790	41870	1	45.707249	11.47760	9500
1526	1527	lounge	51	1705	23600	1	38.122070	13.36112	9300
1527	1528	pop	51	517	3000	1	40.748241	14.52835	9999
1529	1530	lounge	51	731	22551	1	38.122070	13.36112	9900
1530	1531	lounge	51	670	29000	1	45.764648	8.99450	10800

907 rows × 9 columns

grouping the data

In [6]:

```
data2=data.groupby(['model']).count()
```

In [7]:

```
data2
```

Out[7]:

	ID	engine_power	age_in_days	km	previous_owners	lat	lon	price
model								
lounge	1094	1094	1094	1094	1094	1094	1094	1094
pop	358	358	358	358	358	358	358	358
sport	86	86	86	86	86	86	86	86

renaming the column

In [8]:

```
data3=data.rename(columns={'model':'model_name'})
list(data3)
```

Out[8]:

```
['ID',
 'model_name',
 'engine_power',
 'age_in_days',
 'km',
 'previous_owners',
 'lat',
 'lon',
 'price']
```

In [9]:
```
print(data3)
```

```
        ID model_name  engine_power  age_in_days      km  previous_owners  \
0        1     lounge            51          882   25000                1
1        2        pop            51         1186   32500                1
2        3      sport            74         4658  142228                1
3        4     lounge            51         2739  160000                1
4        5        pop            73         3074  106880                1
...    ...        ...           ...          ...     ...              ...
1533  1534      sport            51         3712  115280                1
1534  1535     lounge            74         3835  112000                1
1535  1536        pop            51         2223   60457                1
1536  1537     lounge            51         2557   80750                1
1537  1538        pop            51         1766   54276                1

            lat        lon  price
0     44.907242   8.611560   8900
1     45.666359  12.241890   8800
2     45.503300  11.417840   4200
3     40.633171  17.634609   6000
4     41.903221  12.495650   5700
...         ...        ...    ...
1533  45.069679   7.704920   5200
1534  45.845692   8.666870   4600
1535  45.481541   9.413480   7500
1536  45.000702   7.682270   5990
1537  40.323410  17.568270   7900

[1538 rows x 9 columns]
```

removing the unwanted data

In [10]:
```
data4=data.drop(columns='lat')
```

In [11]:

```
data4
```

Out[11]:

	ID	model	engine_power	age_in_days	km	previous_owners	lon	price
0	1	lounge	51	882	25000	1	8.611560	8900
1	2	pop	51	1186	32500	1	12.241890	8800
2	3	sport	74	4658	142228	1	11.417840	4200
3	4	lounge	51	2739	160000	1	17.634609	6000
4	5	pop	73	3074	106880	1	12.495650	5700
...	...	...	...	...	...	...	...	...
1533	1534	sport	51	3712	115280	1	7.704920	5200
1534	1535	lounge	74	3835	112000	1	8.666870	4600
1535	1536	pop	51	2223	60457	1	9.413480	7500
1536	1537	lounge	51	2557	80750	1	7.682270	5990
1537	1538	pop	51	1766	54276	1	17.568270	7900

1538 rows × 8 columns

grouping the data

In [12]:

```
data5=data.groupby(['engine_power']).count()
```

In [13]:

```
data5
```

Out[13]:

	ID	model	age_in_days	km	previous_owners	lat	lon	price
engine_power								
51	1453	1453	1453	1453	1453	1453	1453	1453
58	1	1	1	1	1	1	1	1
62	43	43	43	43	43	43	43	43
63	1	1	1	1	1	1	1	1
66	1	1	1	1	1	1	1	1
73	22	22	22	22	22	22	22	22
74	14	14	14	14	14	14	14	14
77	3	3	3	3	3	3	3	3

In [14]:

```
data1=data.drop(columns='model')
data1
```

Out[14]:

	ID	engine_power	age_in_days	km	previous_owners	lat	lon	price
0	1	51	882	25000	1	44.907242	8.611560	8900
1	2	51	1186	32500	1	45.666359	12.241890	8800
2	3	74	4658	142228	1	45.503300	11.417840	4200
3	4	51	2739	160000	1	40.633171	17.634609	6000
4	5	73	3074	106880	1	41.903221	12.495650	5700
...	...	...	...	...	...	...	...	...
1533	1534	51	3712	115280	1	45.069679	7.704920	5200
1534	1535	74	3835	112000	1	45.845692	8.666870	4600
1535	1536	51	2223	60457	1	45.481541	9.413480	7500
1536	1537	51	2557	80750	1	45.000702	7.682270	5990
1537	1538	51	1766	54276	1	40.323410	17.568270	7900

1538 rows × 8 columns

correlation to the data

In [15]:

```
cor=data1.corr()
cor
```

Out[15]:

	ID	engine_power	age_in_days	km	previous_owners	lat	lon	price
ID	1.000000	-0.034059	-0.060753	-0.006537	0.007803	-0.058207	0.058941	0.028516
engine_power	-0.034059	1.000000	0.319190	0.285495	-0.005030	0.005721	-0.005032	-0.277235
age_in_days	-0.060753	0.319190	1.000000	0.833890	0.075775	0.062982	-0.042667	-0.893328
km	-0.006537	0.285495	0.833890	1.000000	0.097539	0.035519	0.004839	-0.859373
previous_owners	0.007803	-0.005030	0.075775	0.097539	1.000000	0.001697	-0.026836	-0.076274
lat	-0.058207	0.005721	0.062982	0.035519	0.001697	1.000000	-0.766646	-0.011733
lon	0.058941	-0.005032	-0.042667	0.004839	-0.026836	-0.766646	1.000000	-0.003541
price	0.028516	-0.277235	-0.893328	-0.859373	-0.076274	-0.011733	-0.003541	1.000000

sorting the data by using map function

In [16]:

```
data['model']=data['model'].map({'lounge':1,'pop':2,'sport':3})
```

In [17]:
```
data
```

Out[17]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon	price
0	1	1	51	882	25000	1	44.907242	8.611560	8900
1	2	2	51	1186	32500	1	45.666359	12.241890	8800
2	3	3	74	4658	142228	1	45.503300	11.417840	4200
3	4	1	51	2739	160000	1	40.633171	17.634609	6000
4	5	2	73	3074	106880	1	41.903221	12.495650	5700
...	...	...	...	...	...	...	...	...	...
1533	1534	3	51	3712	115280	1	45.069679	7.704920	5200
1534	1535	1	74	3835	112000	1	45.845692	8.666870	4600
1535	1536	2	51	2223	60457	1	45.481541	9.413480	7500
1536	1537	1	51	2557	80750	1	45.000702	7.682270	5990
1537	1538	2	51	1766	54276	1	40.323410	17.568270	7900

1538 rows × 9 columns

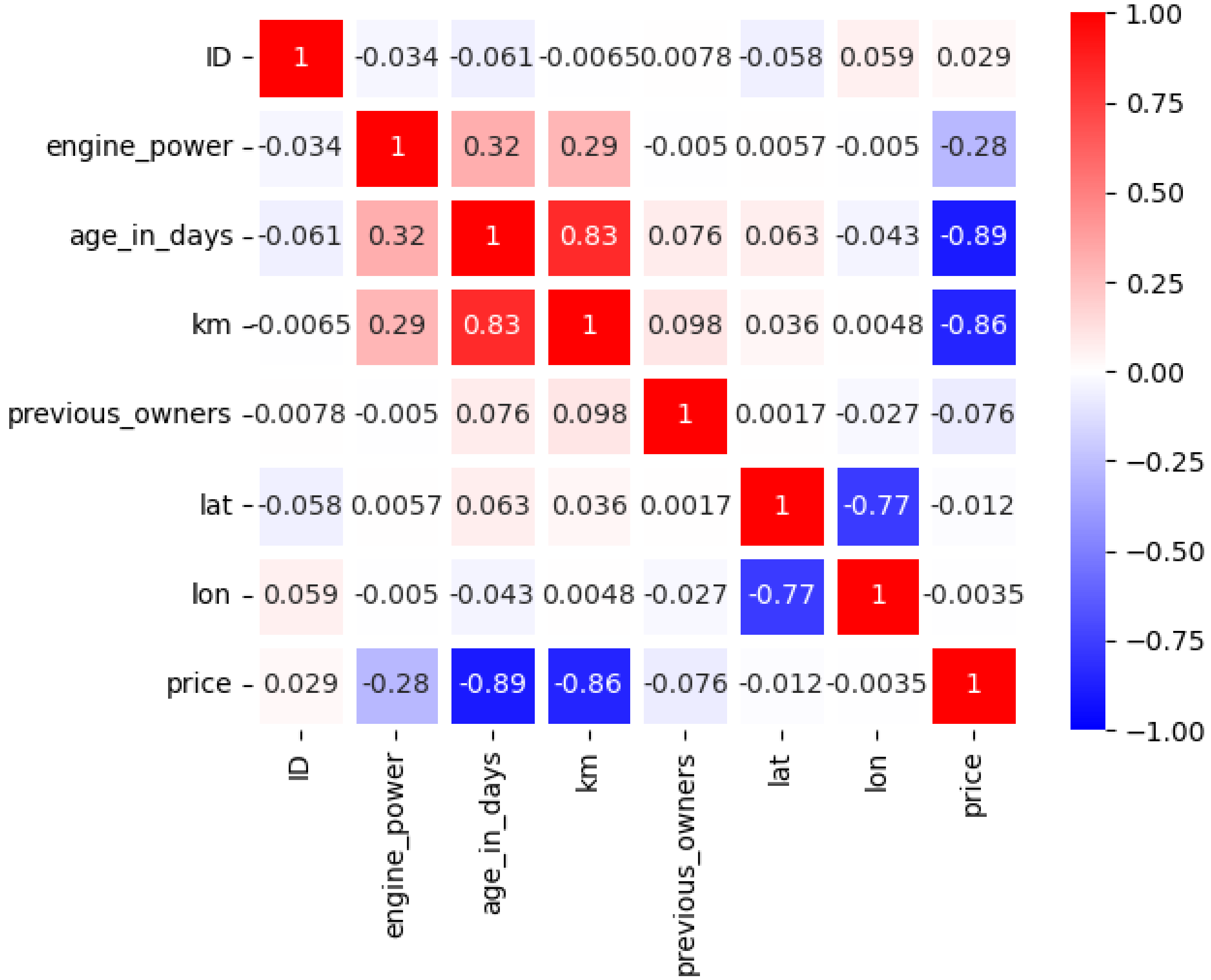
In [18]:
```
list(data)
```

Out[18]:
```
['ID',
 'model',
 'engine_power',
 'age_in_days',
 'km',
 'previous_owners',
 'lat',
 'lon',
 'price']
```

importing seaborn abd getting the heat map of the data

In [36]:
```
import seaborn as sns
sns.heatmap(cor,vmax=1,vmin=-1,annot=True,linewidths=5,cmap='bwr')
```

Out[36]:
```
<Axes: >
```



In [ ]:

In [ ]: