In [3]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [4]:
```python
df=pd.read_csv(r"C:\Users\Admin\Downloads\fiat500_VehicleSelection_Dataset - fiat500_VehicleSelecti:
df
```

Out[4]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon	price	Unnamed: 9	
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.611559868	8900	NaN	
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.24188995	8800	NaN	
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11.41784	4200	NaN	
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.63460922	6000	NaN	
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.49565029	5700	NaN	
...	...	...	...	...	...	...	...	...	...	...	
1544	NaN	NaN	NaN	NaN	NaN	NaN	NaN	length	5	NaN	
1545	NaN	NaN	NaN	NaN	NaN	NaN	NaN	concat	lonprice	NaN	
1546	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Null values	NO	NaN	
1547	NaN	NaN	NaN	NaN	NaN	NaN	NaN	find	1	NaN	
1548	NaN	NaN	NaN	NaN	NaN	NaN	NaN	search	1	NaN	

1549 rows × 11 columns

In [5]:
```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1549 entries, 0 to 1548
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   ID               1538 non-null   float64
 1   model            1538 non-null   object
 2   engine_power     1538 non-null   float64
 3   age_in_days      1538 non-null   float64
 4   km               1538 non-null   float64
 5   previous_owners  1538 non-null   float64
 6   lat              1538 non-null   float64
 7   lon              1549 non-null   object
 8   price            1549 non-null   object
 9   Unnamed: 9       0 non-null      float64
 10  Unnamed: 10      1 non-null      object
dtypes: float64(7), object(4)
memory usage: 133.2+ KB
```

In [6]: `df.describe()`

Out[6]:

	ID	engine_power	age_in_days	km	previous_owners	lat	Unnamed: 9
count	1538.000000	1538.000000	1538.000000	1538.000000	1538.000000	1538.000000	0.0
mean	769.500000	51.904421	1650.980494	53396.011704	1.123537	43.541361	NaN
std	444.126671	3.988023	1289.522278	40046.830723	0.416423	2.133518	NaN
min	1.000000	51.000000	366.000000	1232.000000	1.000000	36.855839	NaN
25%	385.250000	51.000000	670.000000	20006.250000	1.000000	41.802990	NaN
50%	769.500000	51.000000	1035.000000	39031.000000	1.000000	44.394096	NaN
75%	1153.750000	51.000000	2616.000000	79667.750000	1.000000	45.467960	NaN
max	1538.000000	77.000000	4658.000000	235000.000000	4.000000	46.795612	NaN

In [7]: `df.columns`

Out[7]:
```
Index(['ID', 'model', 'engine_power', 'age_in_days', 'km', 'previous_owners',
       'lat', 'lon', 'price', 'Unnamed: 9', 'Unnamed: 10'],
      dtype='object')
```

In [8]: `df.fillna(value=1)`

Out[8]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon	price	Unnamed: 9	U
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.611559868	8900	1.0	
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.24188995	8800	1.0	
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11.41784	4200	1.0	
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.63460922	6000	1.0	
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.49565029	5700	1.0	
...	...	...	...	...	...	...	...	...	...	...	
1544	1.0	1	1.0	1.0	1.0	1.0	1.000000	length	5	1.0	
1545	1.0	1	1.0	1.0	1.0	1.0	1.000000	concat	lonprice	1.0	
1546	1.0	1	1.0	1.0	1.0	1.0	1.000000	Null values	NO	1.0	
1547	1.0	1	1.0	1.0	1.0	1.0	1.000000	find	1	1.0	
1548	1.0	1	1.0	1.0	1.0	1.0	1.000000	search	1	1.0	

1549 rows × 11 columns

In [9]: `sns.pairplot(df)`
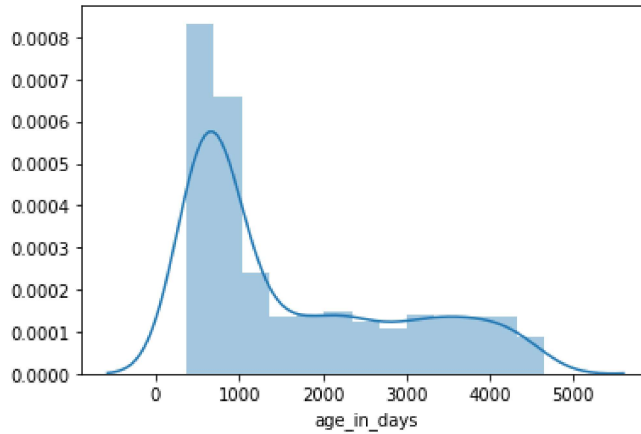
Out[9]: `<seaborn.axisgrid.PairGrid at 0x152e9099580>`

In [10]: `sns.distplot(df['age_in_days'])`

Out[10]: `<matplotlib.axes._subplots.AxesSubplot at 0x152ea694f10>`

In [11]:
```
df1=df[['ID', 'engine_power', 'age_in_days', 'km', 'previous_owners',
        'lat']]
df1
```

Out[11]:

	ID	engine_power	age_in_days	km	previous_owners	lat
0	1.0	51.0	882.0	25000.0	1.0	44.907242
1	2.0	51.0	1186.0	32500.0	1.0	45.666359
2	3.0	74.0	4658.0	142228.0	1.0	45.503300
3	4.0	51.0	2739.0	160000.0	1.0	40.633171
4	5.0	73.0	3074.0	106880.0	1.0	41.903221
...	...	...	...	...	...	...
1544	NaN	NaN	NaN	NaN	NaN	NaN
1545	NaN	NaN	NaN	NaN	NaN	NaN
1546	NaN	NaN	NaN	NaN	NaN	NaN
1547	NaN	NaN	NaN	NaN	NaN	NaN
1548	NaN	NaN	NaN	NaN	NaN	NaN

1549 rows × 6 columns

In [12]: `df2=df1.head(100)`

In [14]:
```python
sns.heatmap(df2.corr())
```

Out[14]: <matplotlib.axes._subplots.AxesSubplot at 0x152ebe14790>
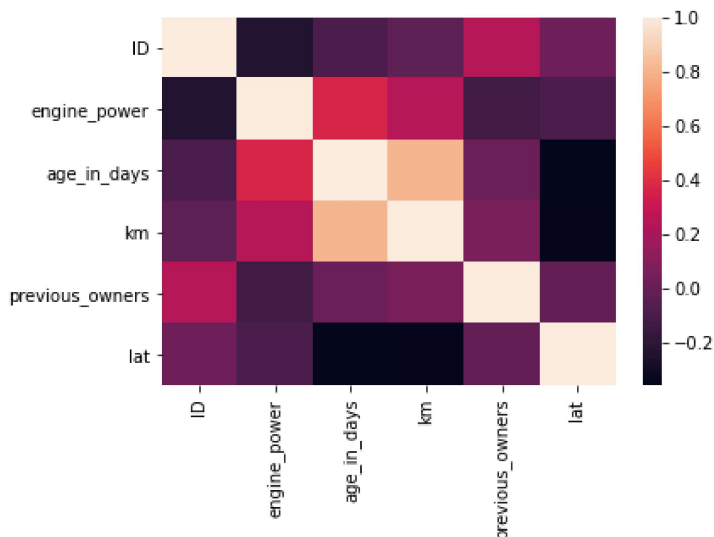
In [15]:
```python
x=df2[['ID', 'engine_power', 'km', 'previous_owners',
       'lat']]
y=df2[['age_in_days']]
```

In [16]:
```python
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3)
```
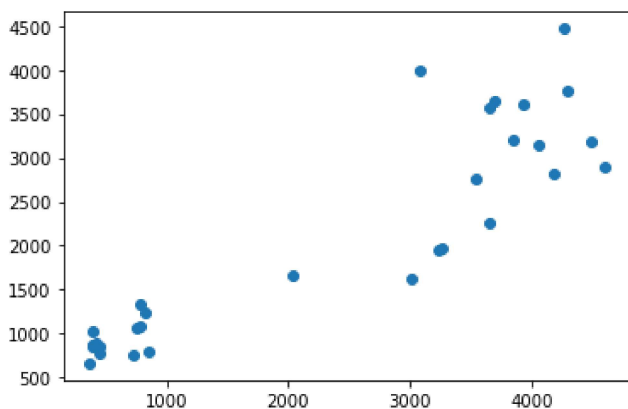
In [17]:
```python
from sklearn.linear_model import LinearRegression
lr= LinearRegression()
lr.fit(x_train,y_train)
```

Out[17]: LinearRegression()

In [18]:
```python
print(lr.intercept_)
```

[824.69460532]

In [19]:
```python
prediction= lr.predict(x_test)
plt.scatter(y_test,prediction)
```

Out[19]: <matplotlib.collections.PathCollection at 0x152ebeff340>

In [20]:
```python
print(lr.score(x_test,y_test))
```

0.7582312271487226

In [ ]: