In [1]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn import preprocessing ,svm
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [2]:

```
df=pd.read_csv(r"C:\Users\91628\Desktop\vehicle.csv")
df
```

Out[2]:

	ID	model	engine_power	age_in_days	km	previous_owners	lat	
0	1	lounge	51	882	25000	1	44.907242	8.611
1	2	pop	51	1186	32500	1	45.666359	12.241
2	3	sport	74	4658	142228	1	45.503300	11.417
3	4	lounge	51	2739	160000	1	40.633171	17.634
4	5	pop	73	3074	106880	1	41.903221	12.495
...	...	...	...	...	...	...	...	
1533	1534	sport	51	3712	115280	1	45.069679	7.704
1534	1535	lounge	74	3835	112000	1	45.845692	8.666
1535	1536	pop	51	2223	60457	1	45.481541	9.413
1536	1537	lounge	51	2557	80750	1	45.000702	7.682
1537	1538	pop	51	1766	54276	1	40.323410	17.568

1538 rows × 9 columns

In [3]:

```
df=df[['price','km']]
df.columns=['P','K']
```

In [4]:

```
df.head(10)
```

Out[4]:

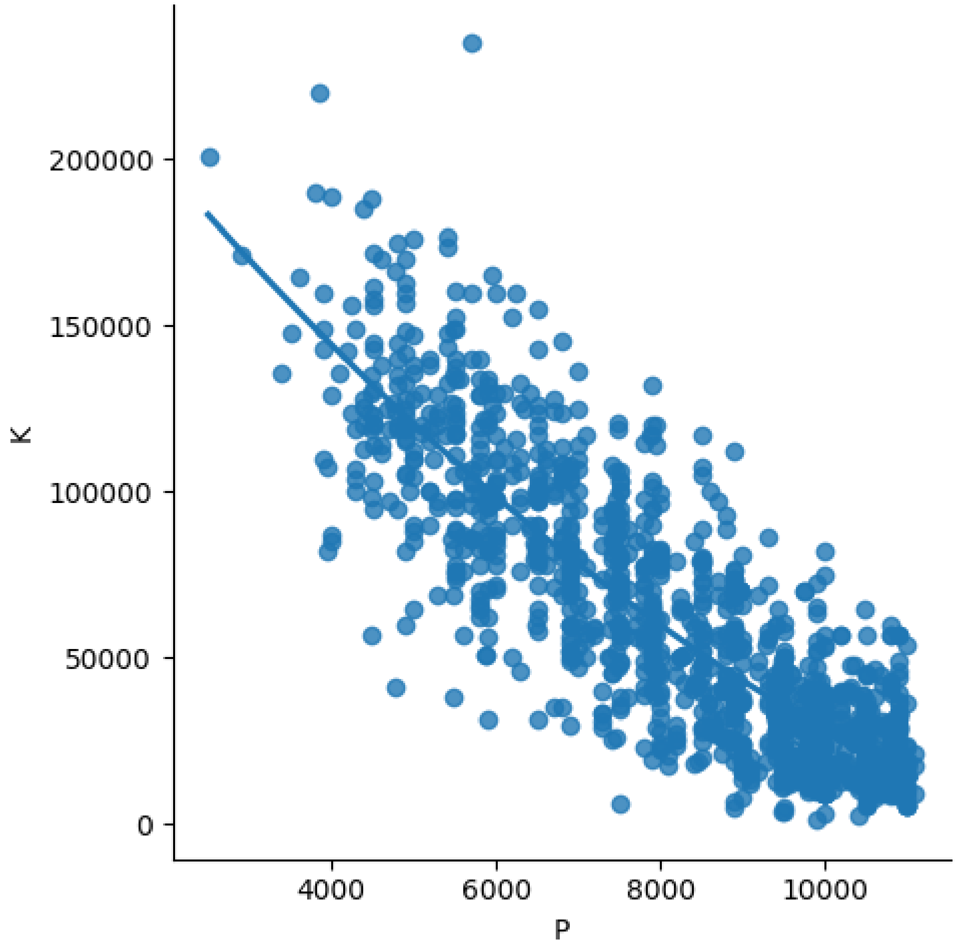
	P	K
0	8900	25000
1	8800	32500
2	4200	142228
3	6000	160000
4	5700	106880
5	7900	70225
6	10750	11600
7	9190	49076
8	5600	76000
9	6000	89000

In [5]:

```
sns.lmplot(x="P",y="K",data=df,order=2,ci=None)
```

Out[5]:

```
<seaborn.axisgrid.FacetGrid at 0x23b81b22a90>
```



In [6]:

```
df.describe()
```

Out[6]:

	P	K
count	1538.000000	1538.000000
mean	8576.003901	53396.011704
std	1939.958641	40046.830723
min	2500.000000	1232.000000
25%	7122.500000	20006.250000
50%	9000.000000	39031.000000
75%	10000.000000	79667.750000
max	11100.000000	235000.000000

In [7]:

```
df.fillna(method='ffill',inplace=True)
```

```
C:\Users\91628\AppData\Local\Temp\ipykernel_9624\4116506308.py:1: SettingW
ithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://
pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-
view-versus-a-copy)
  df.fillna(method='ffill',inplace=True)
```

In [8]:

```
x=np.array(df['P']).reshape(-1,1)
y=np.array(df['K']).reshape(-1,1)
```

In [9]:

```
df.dropna(inplace=True)
```

```
C:\Users\91628\AppData\Local\Temp\ipykernel_9624\1379821321.py:1: SettingW
ithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://
pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-
view-versus-a-copy)
  df.dropna(inplace=True)
```

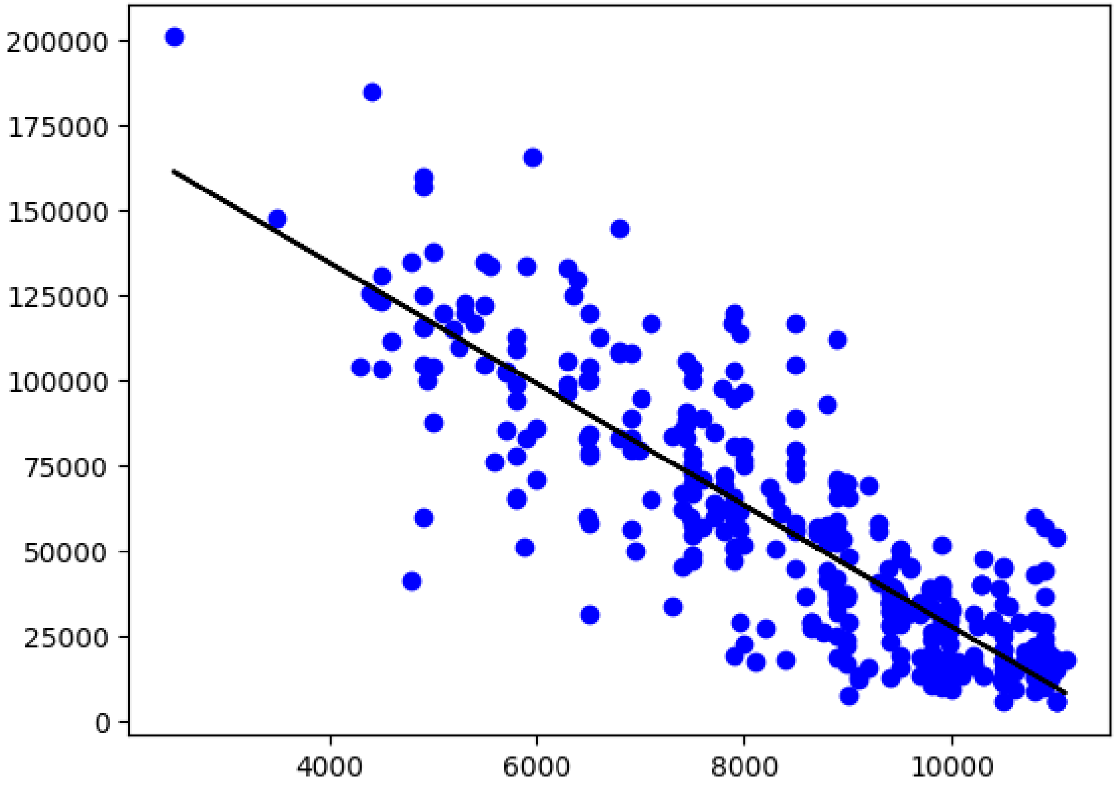
In [10]:

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.25)
regr=LinearRegression()
regr.fit(x_train,y_train)
print(regr.score(x_test,y_test))
```

```
0.7172125125413794
```

In [11]:

```
y_pred=regr.predict(x_test)
plt.scatter (x_test,y_test,color='b')
plt.plot(x_test,y_pred,color='k')
plt.show()
```

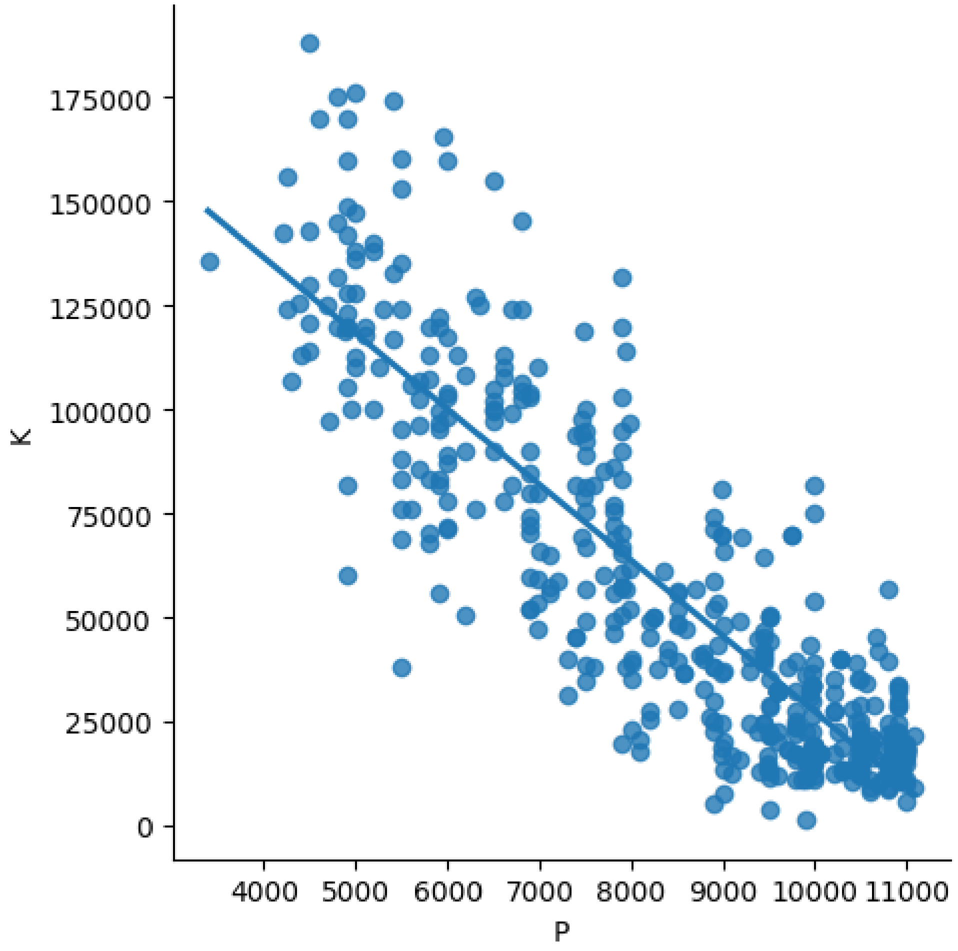


In [12]:

```
df500=df[:][:500]
sns.lmplot(x="P",y="K",data=df500,order=1,ci=None)
```

Out[12]:

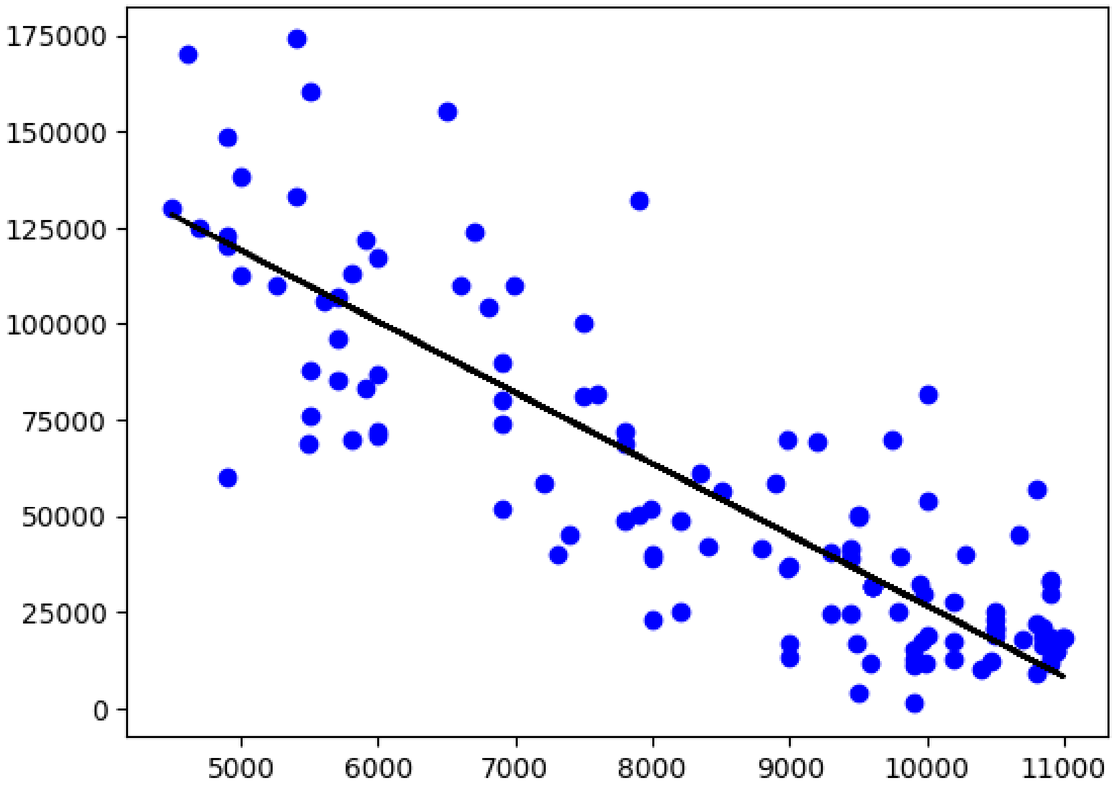
```
<seaborn.axisgrid.FacetGrid at 0x23b81b8f450>
```



In [13]:

```
df500.fillna(method='ffill',inplace=True)
x=np.array(df500['P']).reshape(-1,1)
y=np.array(df500['K']).reshape(-1,1)
df500.dropna(inplace=True)
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.25)
regr=LinearRegression()
regr.fit(x_train,y_train)
print("Regression:",regr.score(x_test,y_test))
y_pred=regr.predict(x_test)
plt.scatter(x_test,y_test,color='b')
plt.plot(x_test,y_pred,color='k')
plt.show()
```

```
Regression: 0.713702913133478
```



In [17]:

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
mode1=LinearRegression()
mode1.fit(x_train,y_train)
y_pred=mode1.predict(x_test)
r2=r2_score(y_test,y_pred)
print("R2 score:",r2)
```

```
R2 score: 0.713702913133478
```

In [ ]: