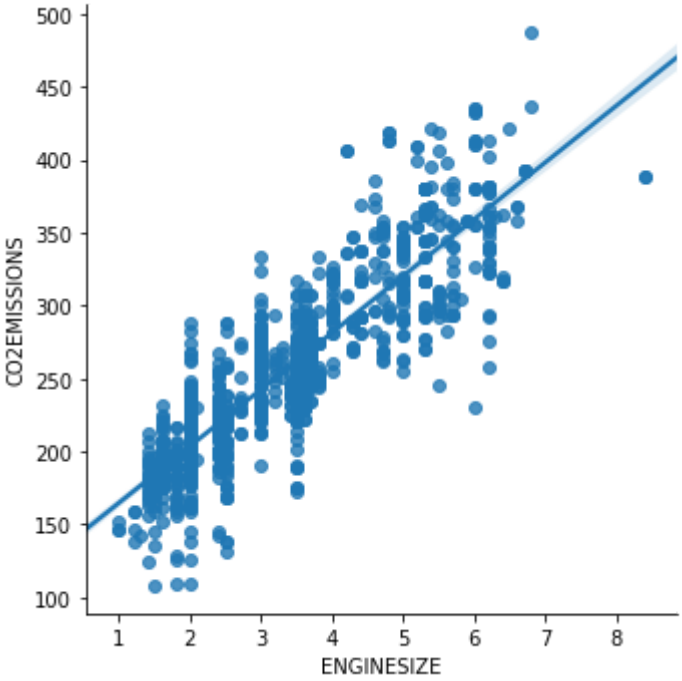
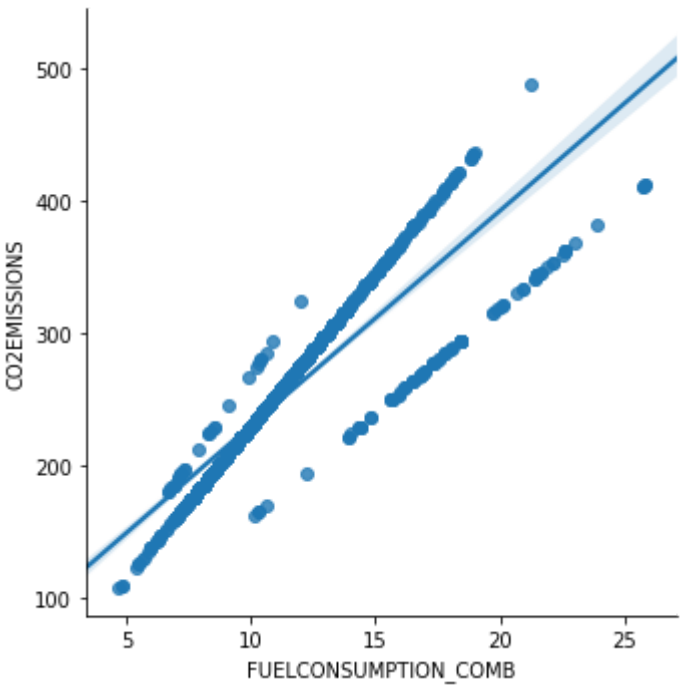
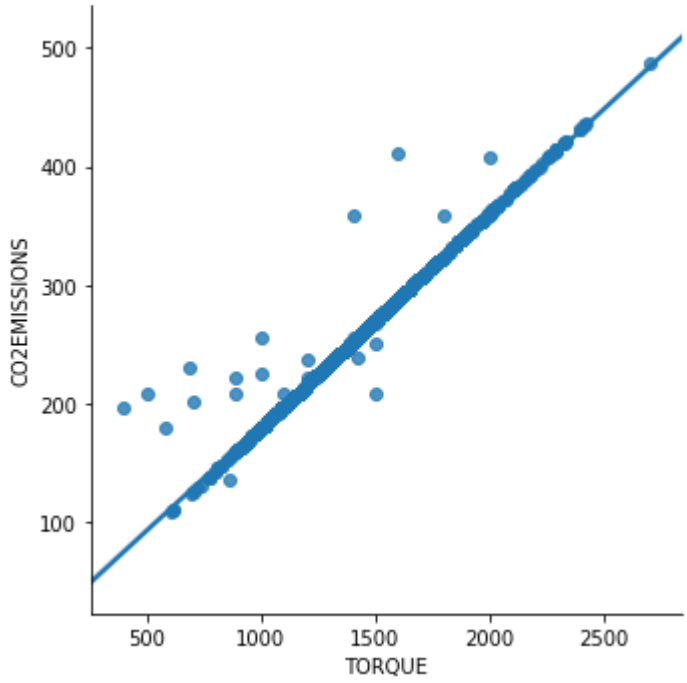
In [10]:

```
sns.lmplot(x = 'ENGINESIZE' , y = 'CO2EMISSIONS' , data = emission_defined)
sns.lmplot(x = 'FUELCONSUMPTION_COMB' , y = 'CO2EMISSIONS' , data = emission_defined)
sns.lmplot(x = 'TORQUE' , y = 'CO2EMISSIONS' , data = emission_defined)
plt.show()
```

```
F:\Anaconda3\lib\site-packages\scipy\stats\stats.py:1713: FutureWarning: Usi
ng a non-tuple sequence for multidimensional indexing is deprecated; use `ar
r[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted
as an array index, `arr[np.array(seq)]`, which will result either in an erro
r or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```







In [ ]:

```
```

In [11]:

```
from sklearn.model_selection import train_test_split
```

In [ ]:

```
```

In [12]:

```
x = emission[['TORQUE']]
y = emission['CO2EMISSIONS']
train_x , test_x , train_y , test_y = train_test_split(x,y,test_size=0.2)
```

In [13]:

```
regre = linear_model.LinearRegression()
```

In [14]:

```
regre.fit(train_x,train_y)
```

Out[14]:

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
         normalize=False)
```

In [17]:

```
regre.score(train_x,train_y)
```

Out[17]:

```
0.9747626930292009
```

In [19]:

```
print("Coefficient or Slope of regression line is: " , regre.coef_[0])
print("Intercept of regression line is: ",regre.intercept_)
```

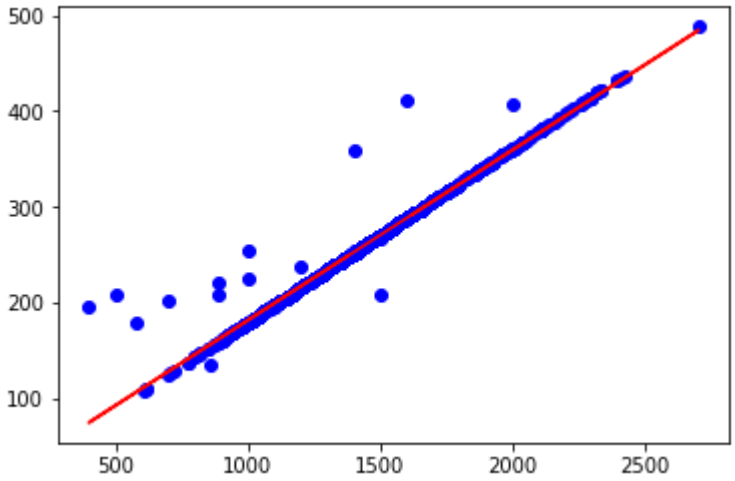
```
Coefficient or Slope of regression line is:  0.17791308013018053
Intercept of regression line is:  3.783245190266257
```

In [20]:

```
predict_y = regre.predict(test_x)
```

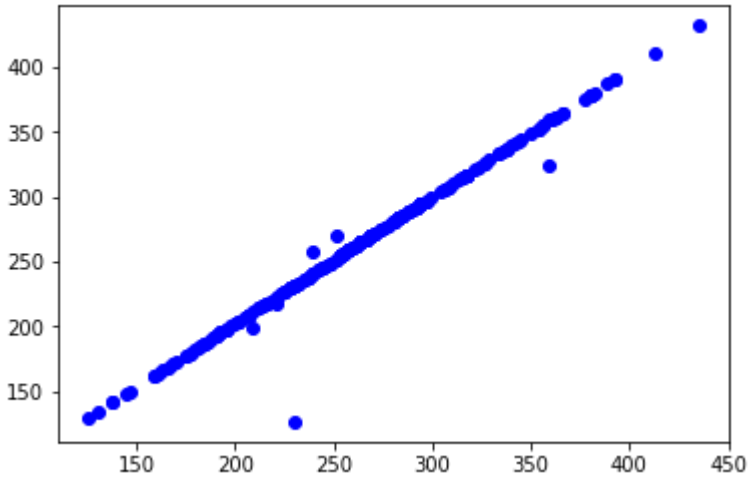
In [21]:

```
plt.scatter(train_x,train_y , color = 'blue')
plt.plot(train_x, regre.coef_[0]*train_x + regre.intercept_, '-r')
plt.show()
```



In [22]:

```
plt.scatter(test_y,predict_y , color = 'blue')
plt.show()
```



In [23]:

```
print(metrics.mean_squared_error(test_y,predict_y))
```

```
61.96703642789677
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: