In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:
```
cars=pd.read_csv('final_cars.csv')
```

In [4]:
```
y=cars['price']
X=cars.drop(columns=['price'])
```

In [5]:
```
X=pd.get_dummies(X)
```

In [6]:
```
from sklearn.model_selection import train_test_split
```

In [7]:
```
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=99)
```

Random Forest regressor

In [8]:
```
from sklearn.ensemble import RandomForestRegressor
```

In [11]:
```
model=RandomForestRegressor()
model.fit(X_train,y_train)
print(f'train score:{model.score(X_train,y_train)}')
```

```
train score:0.9842889830163025
```

In [12]:
```
from sklearn.metrics import mean_squared_error,r2_score,mean_absolute_error
```

In [13]:
```
import sklearn.metrics
y_pred=model.predict(X_test)
r2score=r2_score(y_test,y_pred)
```

In [14]:
```
print(f'test score:{r2score:0.2f}')
```

```
test score:0.93
```

In [15]:
```
mse = mean_squared_error(y_test,y_pred)
print("MSE  : ",mse)
print("RMSE : ", np.sqrt(mse))
```

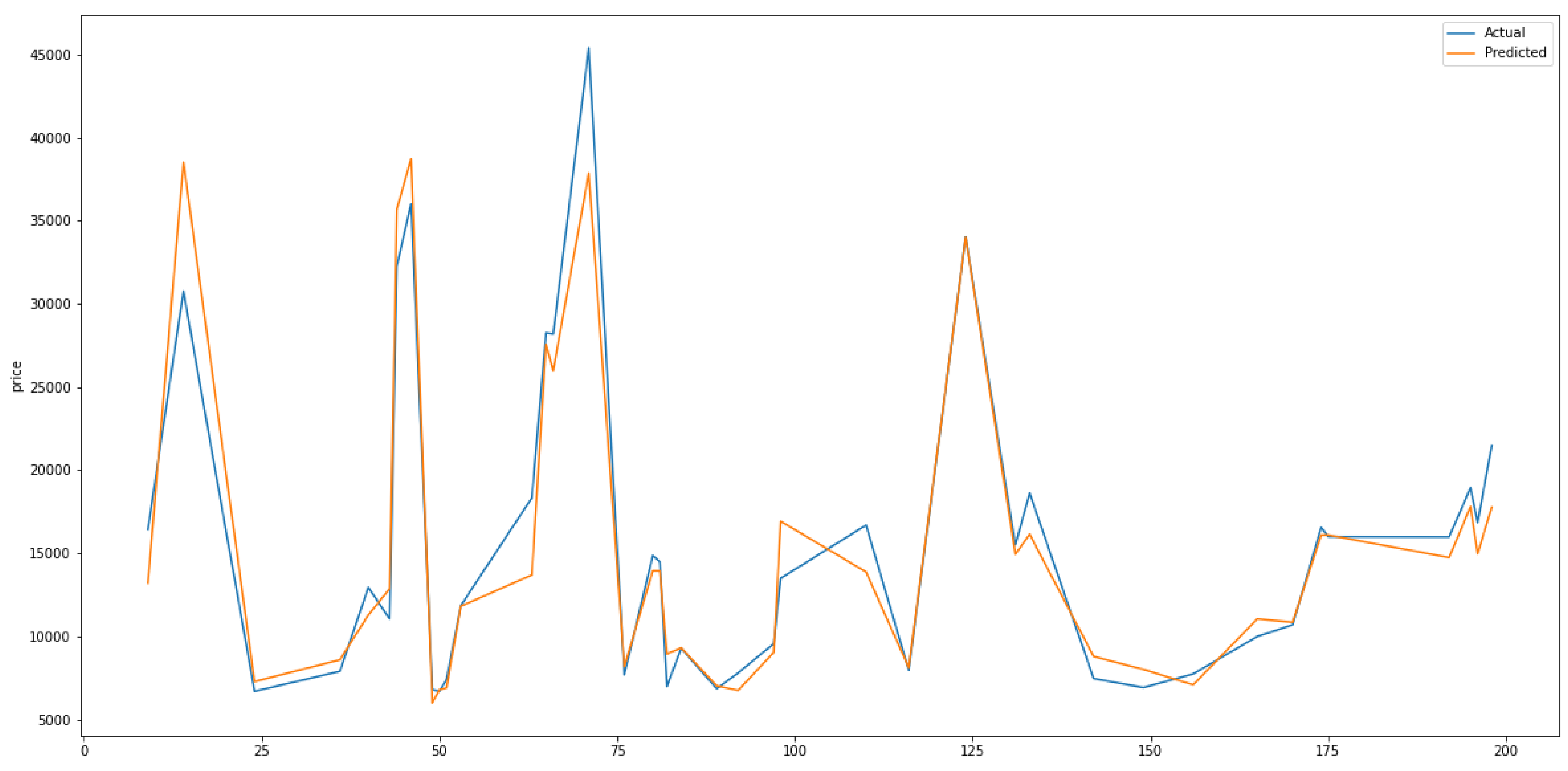
```
MSE  :  5826220.00377168
RMSE :  2413.756409369363
```

In [19]:
```
for f,v in zip (X_train.columns, model.feature_importances_):
      print(f"{f:30} {v:0.2f}")
```

```
curb-weight                    0.28
engine-size                    0.58
highway-mpg                    0.09
make_alfa-romero               0.00
make_audi                      0.00
make_bmw                       0.01
make_chevrolet                 0.00
make_dodge                     0.00
make_honda                     0.00
make_isuzu                     0.00
make_jaguar                    0.00
make_mazda                     0.00
make_mercedes-benz             0.00
make_mercury                   0.00
make_mitsubishi                0.00
make_nissan                    0.00
make_peugot                    0.00
make_plymouth                  0.00
make_porsche                   0.00
make_renault                   0.00
make_saab                      0.00
make_subaru                    0.00
make_toyota                    0.00
make_volkswagen                0.00
make_volvo                     0.00
fuel-type_diesel               0.00
fuel-type_gas                  0.00
num-of-doors_four              0.00
num-of-doors_two               0.00
body-style_convertible         0.00
body-style_hardtop             0.00
body-style_hatchback           0.00
body-style_sedan               0.00
body-style_wagon               0.00
drive-wheels_4wd               0.00
drive-wheels_fwd               0.00
drive-wheels_rwd               0.00
```

In [20]:
```
plt.gcf().set_size_inches(20,10)
sns.lineplot( y = y_test, x = X_test.index, label="Actual")
sns.lineplot( y = y_pred, x = X_test.index, label="Predicted")
```

Out[20]:
```
<AxesSubplot:ylabel='price'>
```



SGDRegressor

In [22]:
```
from sklearn.linear_model import SGDRegressor
```

In [25]:
```
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.2, random_state=99)
```

In [26]:
```
from sklearn.preprocessing import StandardScaler
scaler=StandardScaler()
```

scaler.fit(X_train) X_train=scaler.transform(X_train) X_test=scaler.transorm(X_test)

In [27]:
```
model = SGDRegressor(random_state=100)
model.fit(X_train, y_train)
print(f'Train score : {model.score(X_train,y_train):f}')
```

```
Train score : -1187559418988067028992.000000
```

In [28]:
```
y_pred = model.predict(X_test)
r2score = r2_score(y_test,y_pred)
print(f'Test Score  : {r2score:0.2f}')
```

```
Test Score  : -804186665125677105152.00
```

In [29]:
```
mse = mean_squared_error(y_test,y_pred)
print("MSE  : ",mse)
print("RMSE : ", np.sqrt(mse))
```

```
MSE  :  7.0908251652951476e+28
RMSE :  266286033529645.47
```

In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]: