```
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from sklearn import linear_model
import quandl
import numpy as np
```

```
def prepare_data(df,forecast_col,forecast_out,test_size):
    label = df[forecast_col].shift(-forecast_out);#creating new column called label wi
    X = np.array(df[[forecast_col]]) #creating the feature array
    X = preprocessing.scale(X) #processing the feature array
    X_lately = X[-forecast_out:] #creating the column i want to use later in the predi
    X = X[:-forecast_out] # X that will contain the training and testing
    label.dropna(inplace=True) #dropping na values
    y = np.array(label)  # assigning Y
    X_train, X_test, Y_train, Y_test = train_test_split(X, y, test_size=test_size) #cr

    response = [X_train,X_test , Y_train, Y_test , X_lately]
    return response;
```

```
df = quandl.get('WIKI/AMZN')
```

```
df.head()
```

Date	Open	High	Low	Close	Volume	Ex-Dividend	Split Ratio	Adj. Open	Adj. High	
1997-05-16	22.38	23.75	20.50	20.75	1225000.0	0.0	1.0	1.865000	1.979167	1.7(
1997-05-19	20.50	21.25	19.50	20.50	508900.0	0.0	1.0	1.708333	1.770833	1.6(
1997-05-20	20.75	21.00	19.63	19.63	455600.0	0.0	1.0	1.729167	1.750000	1.6(

```
forecast_col = 'Close'#choosing which column to forecast
forecast_out = 5 #how far to forecast
test_size = 0.2; #the size of my test set

X_train, X_test, Y_train, Y_test , X_lately =prepare_data(df,forecast_col,forecast_out

learner = linear_model.LinearRegression(); #initializing linear regression model

learner.fit(X_train,Y_train); #training the linear regression model
score=learner.score(X_test,Y_test);#testing the linear regression model

forecast= learner.predict(X_lately); #set that will contain the forecasted data

print(score, forecast)
```

```
0.9973452525769135 [1597.2022418  1559.06317935 1510.03589939 1570.94123481 1511.54
```

```

```