Importing The Data

In [7]:
```
import pandas as pd
hw=pd.read_csv("https://gist.githubusercontent.com/EconometricsBySimulation/5735039/raw/c7
hw.head()
```

Out[7]:

	heightIn	weightLb
0	56.3	85.0
1	62.3	105.0
2	63.3	108.0
3	59.0	92.0
4	62.5	112.5

Finding Null Values

In [2]:
```
hw.isnull().sum()
```

Out[2]:
```
sex         0
ageYear     0
ageMonth    0
heightIn    0
weightLb    0
dtype: int64
```

Shapiro Test

In [8]:
```
from scipy import stats
stats.shapiro(hw['heightIn'])
```

Out[8]:
```
ShapiroResult(statistic=0.9962847828865051, pvalue=0.8473318219184875)
```

In [9]:
```
from scipy import stats
stats.shapiro(hw['weightLb'])
```

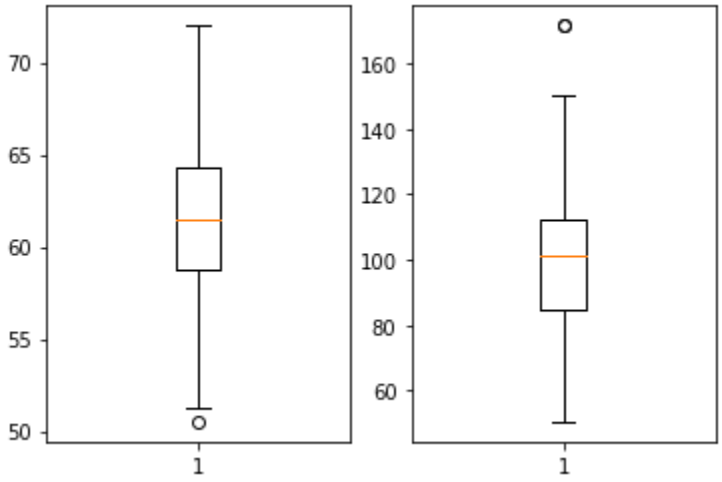
Out[9]:
```
ShapiroResult(statistic=0.9699857234954834, pvalue=6.59234938211739e-05)
```

Boxplot

In [3]:
```
import matplotlib.pyplot as plt
plt.subplot(1,2,1)
plt.boxplot(hw['heightIn'])
plt.subplot(1,2,2)
plt.boxplot(hw['weightLb'])
```

Out[3]:
```
{'whiskers': [<matplotlib.lines.Line2D at 0x1c95cbc50d0>,
  <matplotlib.lines.Line2D at 0x1c95cbc5460>],
 'caps': [<matplotlib.lines.Line2D at 0x1c95cbc57f0>,
  lines.Line2D at 0x1c95cbc5b80>],
```

```
 'boxes': [<matplotlib.lines.Line2D at 0x1c95cbb3d00>],
 'medians': [<matplotlib.lines.Line2D at 0x1c95cbc5f10>],
 'fliers': [<matplotlib.lines.Line2D at 0x1c95cbcf2e0>],
 'means': []}
```



Shape Of The Data

In [4]:
```
hw.shape
```

Out[4]:
```
(237, 5)
```

Train And Test Split

In [5]:
```
train=hw.iloc[:189]
train.shape
```

Out[5]:
```
(189, 5)
```

In [6]:
```
test=hw.iloc[189:237]
test.shape
```

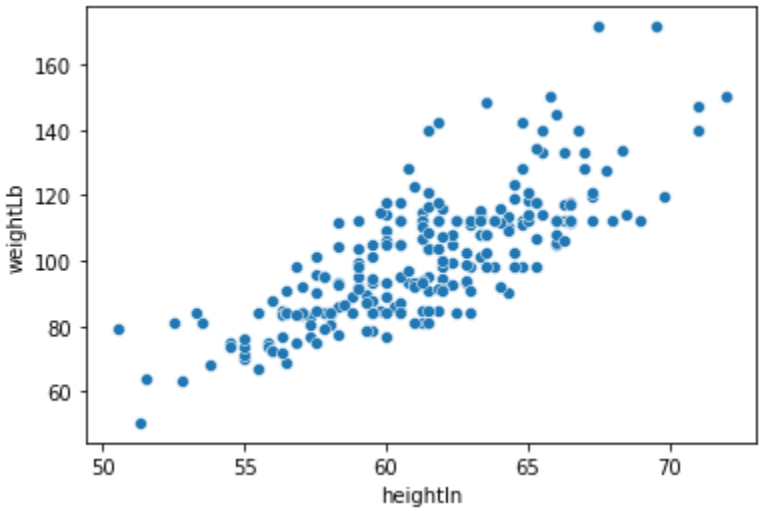
Out[6]:
```
(48, 5)
```

Scatterplot

In [10]:
```
import seaborn as sns
sns.scatterplot('heightIn','weightLb',data=hw)
```

```
C:\Users\admin\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass
the following variables as keyword args: x, y. From version 0.12, the only valid positiona
l argument will be `data`, and passing other arguments without an explicit keyword will re
sult in an error or misinterpretation.
  warnings.warn(
```

Out[10]:
```
<AxesSubplot:xlabel='heightIn', ylabel='weightLb'>
```



Correlation between two Variables

In [11]:
```
from scipy import stats
from scipy.stats import pearsonr
stats.pearsonr(hw.heightIn,hw.weightLb)
```

Out[11]:
```
(0.7748761066276015, 1.028685831403233e-48)
```

Ordinary Least Square

In [12]:
```
import statsmodels.api as sm
train_x=train.heightIn
train_y=train.weightLb
train_x=sm.add_constant(train_x)
model=sm.OLS(train_y,train_x).fit()
model.summary()
```

```
C:\Users\admin\anaconda3\lib\site-packages\statsmodels\tsa\tsatools.py:142: FutureWarning:
In a future version of pandas all arguments of concat except for the argument 'objs' will
be keyword-only
  x = pd.concat(x[::order], 1)
```

Out[12]:

OLS Regression Results

Dep. Variable:	weightLb	**R-squared:**	0.574
Model:	OLS	**Adj. R-squared:**	0.572
Method:	Least Squares	**F-statistic:**	252.2
Date:	Thu, 31 Mar 2022	**Prob (F-statistic):**	1.61e-36
Time:	14:49:54	**Log-Likelihood:**	-743.32
No. Observations:	189	**AIC:**	1491.
Df Residuals:	187	**BIC:**	1497.
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-123.6766	14.137	-8.748	0.000	-151.565	-95.788
	5	0.230	15.882	0.000	3.205	4.114

Omnibus:	28.024	**Durbin-Watson:**	2.012
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	37.777
Skew:	0.900	**Prob(JB):**	6.26e-09
Kurtosis:	4.247	**Cond. No.**	960.

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

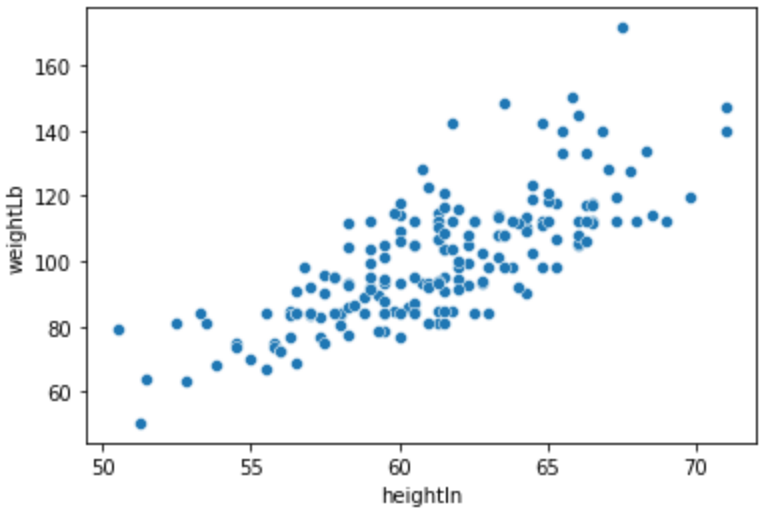
In [13]:
```
test_x=test.heightIn
test_y=test.weightLb
test_x=sm.add_constant(test_x)
model=sm.OLS(test_y,test_x).fit()
```

```
C:\Users\admin\anaconda3\lib\site-packages\statsmodels\tsa\tsatools.py:142: FutureWarning:
In a future version of pandas all arguments of concat except for the argument 'objs' will
be keyword-only
  x = pd.concat(x[::order], 1)
```

Train VS Test Scatterplot

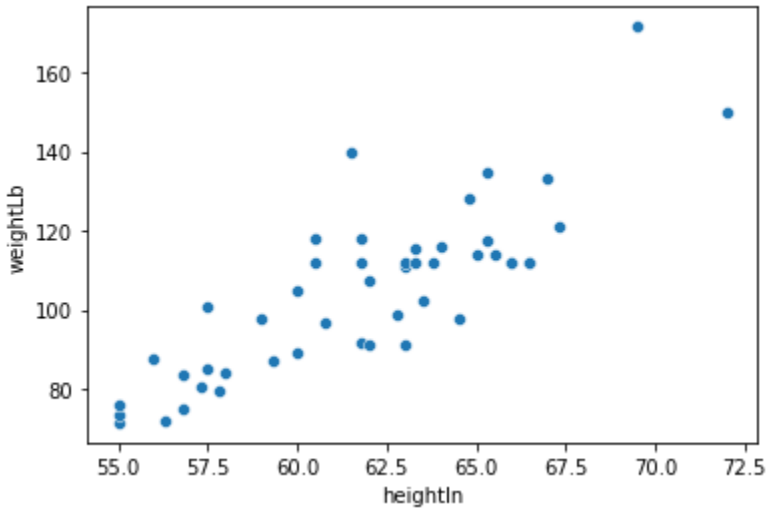
In [14]:
```
sns.scatterplot(x=train.heightIn,y=train.weightLb)
```

Out[14]:
```
<AxesSubplot:xlabel='heightIn', ylabel='weightLb'>
```



In [15]:
```
sns.scatterplot(x=test.heightIn,y=test.weightLb)
```

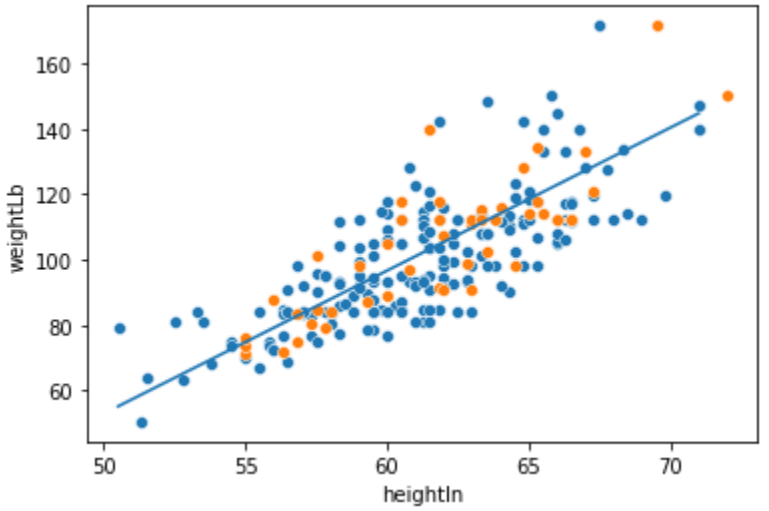
Out[15]:
```
<AxesSubplot:xlabel='heightIn', ylabel='weightLb'>
```



Simple Linear Regression

In [16]:
```
sns.scatterplot(x=train.heightIn,y=train.weightLb)
sns.scatterplot(x=test.heightIn,y=test.weightLb)
predicted_y=model.predict(train_x)
sns.lineplot(x=train.heightIn,y=predicted_y)
```

Out[16]:
```
<AxesSubplot:xlabel='heightIn', ylabel='weightLb'>
```



In [ ]:
```

```