In [3]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import statsmodels.formula.api as smf
import statsmodels.api as sm
# Importing libraries
```

In [4]:
```
lung_data = pd.read_csv('LungCapData2.csv')
# Load File
```

In [5]:
```
# view data types and basic structure
print(lung_data.head())
```

```
   Age  LungCap  Height  Gender Smoke
0    9    3.124    57.0  female    no
1    8    3.172    67.5  female    no
2    7    3.160    54.5  female    no
3    9    2.674    53.0    male    no
4    9    3.685    57.0    male    no
```

In [6]:
```
print(lung_data.tail())
```

```
     Age  LungCap  Height  Gender Smoke
649   16   10.810    67.0    male   yes
650   15    9.181    68.0    male   yes
651   18    6.559    60.0  female    no
652   16    6.385    63.0  female   yes
653   15    7.633    66.5  female    no
```

In [7]:
```
lung_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 654 entries, 0 to 653
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   Age      654 non-null    int64  
 1   LungCap  654 non-null    float64
 2   Height   654 non-null    float64
 3   Gender   654 non-null    object 
 4   Smoke    654 non-null    object 
dtypes: float64(2), int64(1), object(2)
memory usage: 25.7+ KB
```

In [8]:
```
list (lung_data.columns.values)
```

Out[8]:
```
['Age', 'LungCap', 'Height', 'Gender', 'Smoke']
```

In [14]:
```
lung_data.describe().round(2)
# Descriptive statistics
```

Out[14]:

	Age	LungCap	Height
count	654.00	654.00	654.00
mean	9.93	5.91	61.14
std	2.95	2.60	5.70
min	3.00	0.37	46.00
25%	8.00	3.94	57.00
50%	10.00	5.64	61.50
75%	12.00	7.36	65.50
max	19.00	15.38	74.00

In [20]:
```
lung_data[['LungCap', 'Height']].describe().round(2)
# Descriptive statistics, LungCap & Height
```

Out[20]:

	LungCap	Height
count	654.00	654.00
mean	5.91	61.14
std	2.60	5.70
min	0.37	46.00
25%	3.94	57.00
50%	5.64	61.50
75%	7.36	65.50
max	15.38	74.00

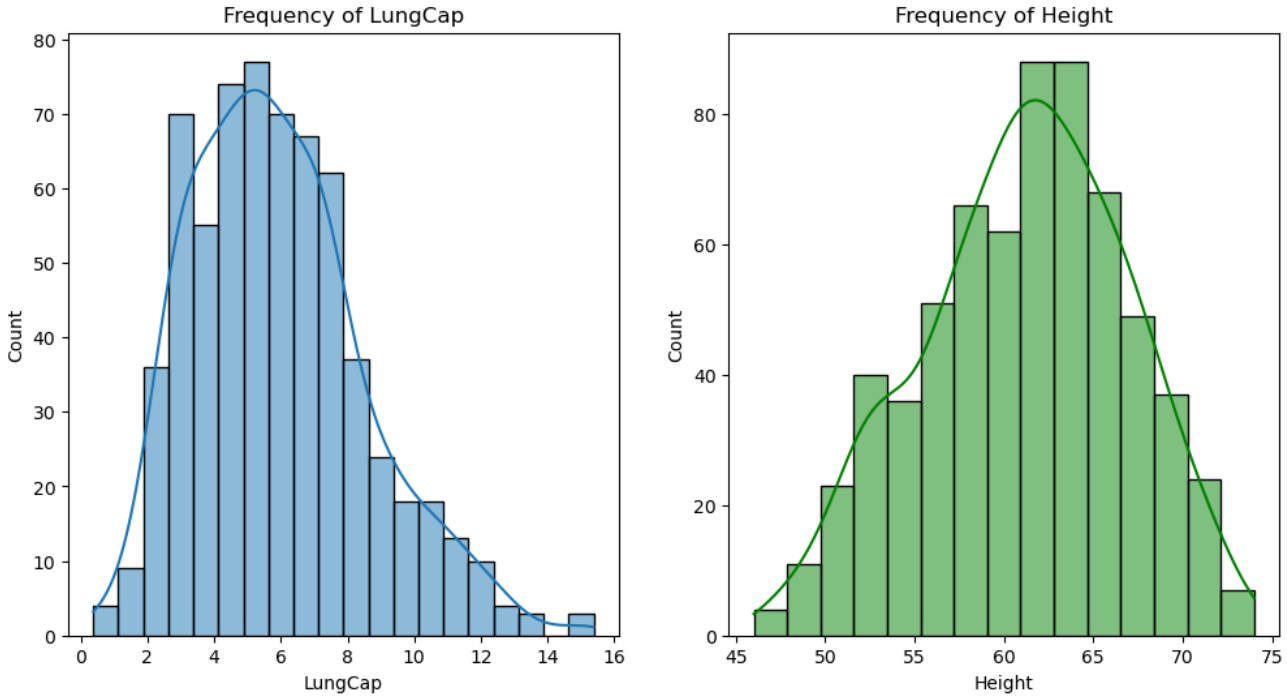
In [15]:
```
# Check if any null values
print(lung_data.isnull().sum())
```

```
Age        0
LungCap    0
Height     0
Gender     0
Smoke      0
dtype: int64
```

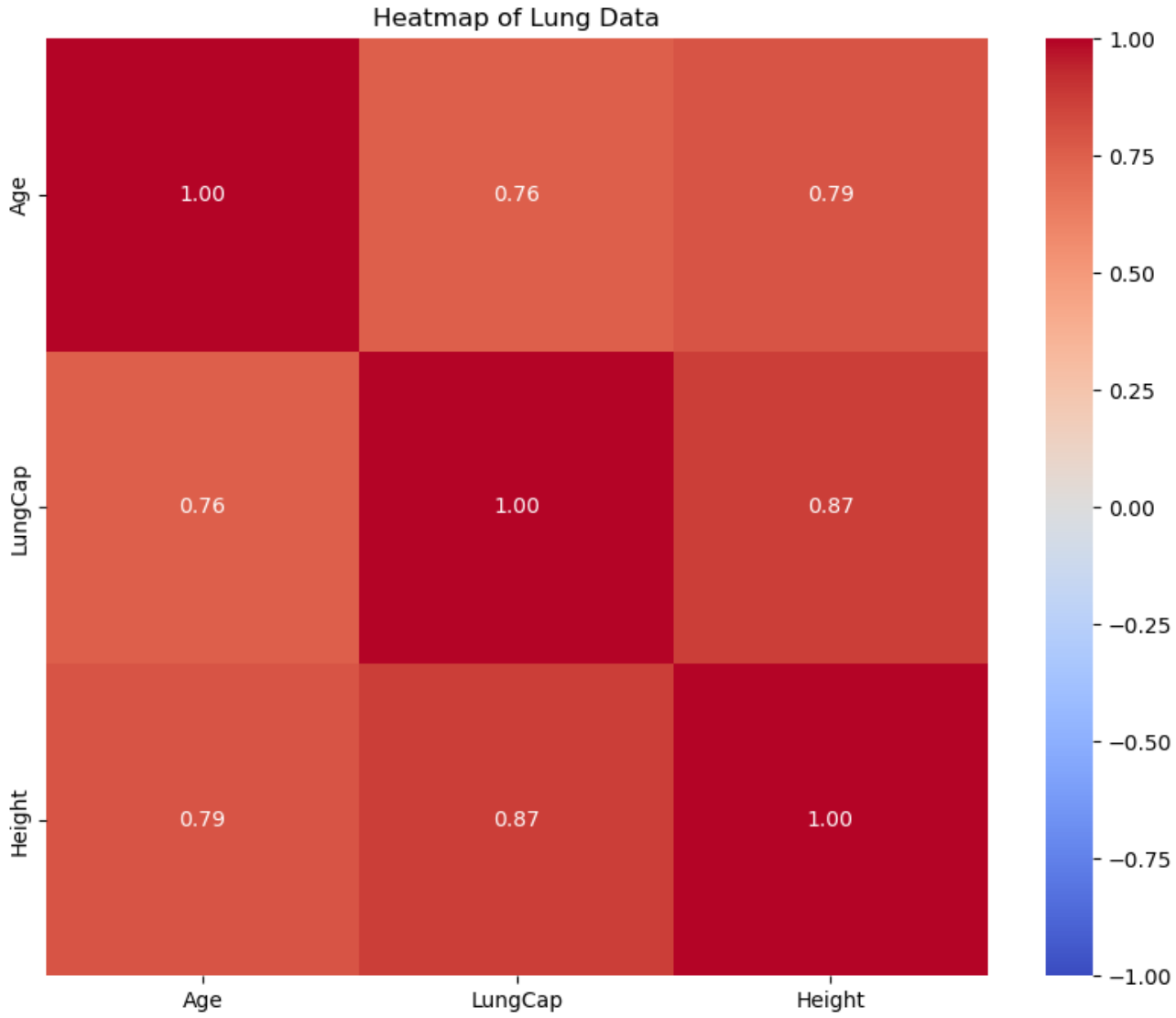
In [43]:
```
#frequency plots for LungCap
```

```
plt.figure(figsize=(12, 6))
plt.subplot(1, 2, 1)
sns.histplot(lung_data['LungCap'], kde=True)
plt.title('Frequency of LungCap')
plt.xlabel('LungCap')

#frequency plots for Height
plt.subplot(1, 2, 2)
sns.histplot(lung_data['Height'], kde=True,color='Green')
plt.title('Frequency of Height')
plt.xlabel('Height')
```

Out[43]: Text(0.5, 0, 'Height')



In [46]:
```
#frequency plots for LungCap # Select only numeric columns from the dataset
numeric_data = lung_data.select_dtypes(include=[np.number])

# Generate a correlation matrix
corr_lung_data = numeric_data.corr()

# Plot the heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(corr_lung_data, annot=True, cmap="coolwarm", vmin=-1.0, vmax=1.0
plt.title("Heatmap of Lung Data")
plt.show()
```



In [50]:
```
# Model 1: LungCap ~ Height
model1 = smf.ols('LungCap ~ Height', data=lung_data).fit()
```

In [51]:
```
# Model 2: LungCap ~ Height + Height^2
lung_data['Height_sq'] = lung_data['Height'] ** 2
model2 = smf.ols('LungCap ~ Height + Height_sq', data=lung_data).fit()
```

In [52]:
```
# Model 3: LungCap ~ Height + Height^2 + Height^3
lung_data['Height_cub'] = lung_data['Height'] ** 3
model3 = smf.ols('LungCap ~ Height + Height_sq + Height_cub', data=lung_data
```

In [53]:
```
# ANOVA for model1 vs model2
anova_result1 = sm.stats.anova_lm(model1, model2)
print('ANOVA between model1 and model2:')
print(anova_result1)

# ANOVA for model2 vs model3
```

```
anova_result2 = sm.stats.anova_lm(model2, model3)
print('\nANOVA between model2 and model3:')
print(anova_result2)
```

```
ANOVA between model1 and model2:
   df_resid          ssr  df_diff    ss_diff          F        Pr(>F)
0     652.0  1088.405828      0.0        NaN        NaN           NaN
1     651.0   998.092326      1.0  90.313502  58.906464  6.068515e-14

ANOVA between model2 and model3:
   df_resid         ssr  df_diff   ss_diff        F    Pr(>F)
0     651.0  998.092326      0.0       NaN      NaN       NaN
1     650.0  997.791669      1.0  0.300658  0.19586  0.658231
```

In [54]:
```
#ANOVA of model 2 and model 1-
#For ANOVA model 1 Vs Model2, Fis 58.91 and P-value is 6.68-14. with such sm
#statistically significant improvement over model 1


#ANOVA of model 2 and model 3-
# #For ANOVA model 2 Vs Model 3, F is 0.19  and P-value is 0.658. Pvalue is
#provide a statistically significant improvement
```

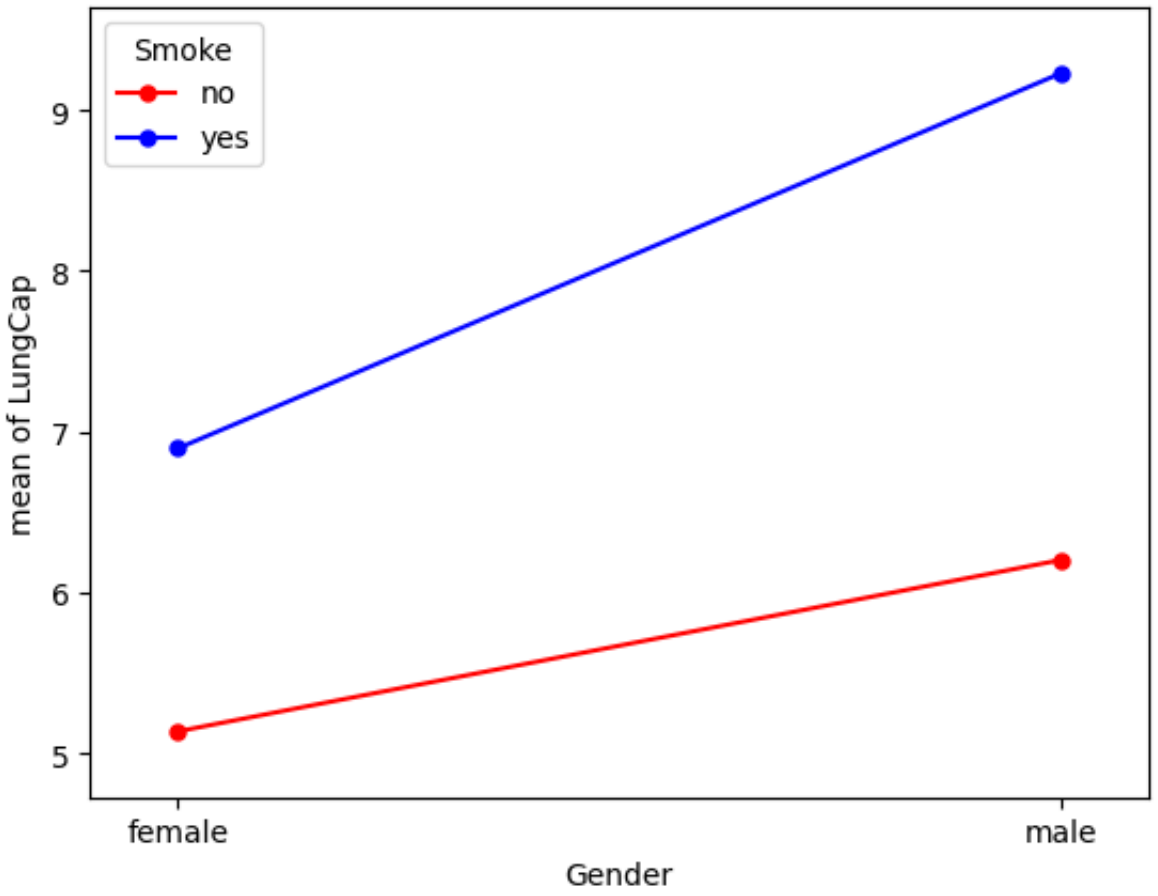
In [56]:
```
from statsmodels.formula.api import ols

formula = 'Q("LungCap") ~ Gender * Smoke * Age'

# Fit the model using OLS regression
model = ols(formula, data=lung_data).fit()

# Perform ANOVA with type 2 sum of squares
aov_table = sm.stats.anova_lm(model, typ=2)

# Display ANOVA
print(aov_table)
```

```
                     sum_sq     df           F         PR(>F)
Gender            150.553386    1.0   65.347581   3.099945e-15
Smoke               5.001144    1.0    2.170743   1.411453e-01
Gender:Smoke        0.011404    1.0    0.004950   9.439334e-01
Age              2189.564550    1.0  950.378807  5.023556e-129
Gender:Age        126.725933    1.0   55.005294   3.792228e-13
Smoke:Age          83.373857    1.0   36.188359   2.999855e-09
Gender:Smoke:Age    1.430190    1.0    0.620773   4.310493e-01
Residual         1488.310439  646.0         NaN           NaN
```

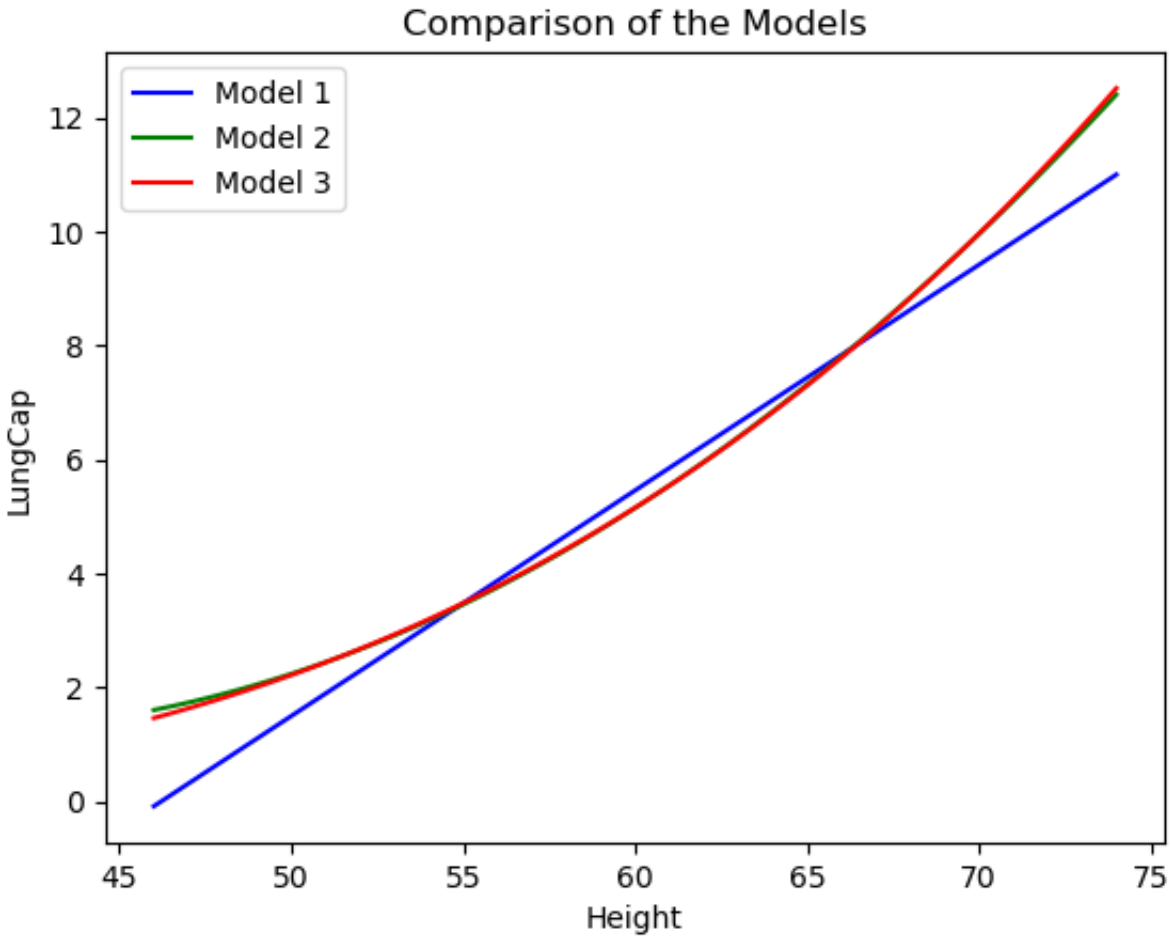
In [60]:
```
print(lung_data.columns)
```

```
Index(['Age', 'LungCap', 'Height', 'Gender', 'Smoke', 'Height_sq',
       'Height_cub'],
      dtype='object')
```

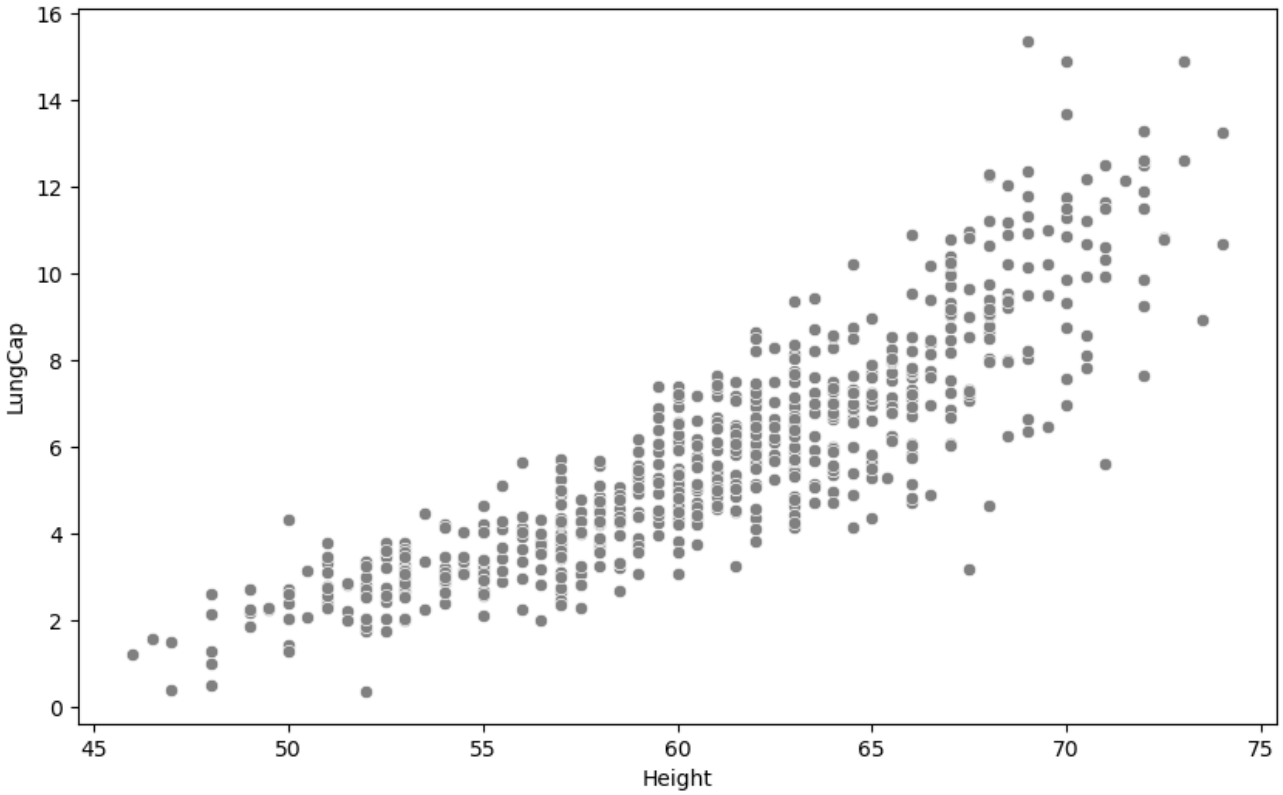
In [61]:
```
# plot for Gender and smoke against LungCap
sm.graphics.interaction_plot(lung_data['Gender'],  lung_data['Smoke'],  lung
    ms=10
)
plt.show()
```



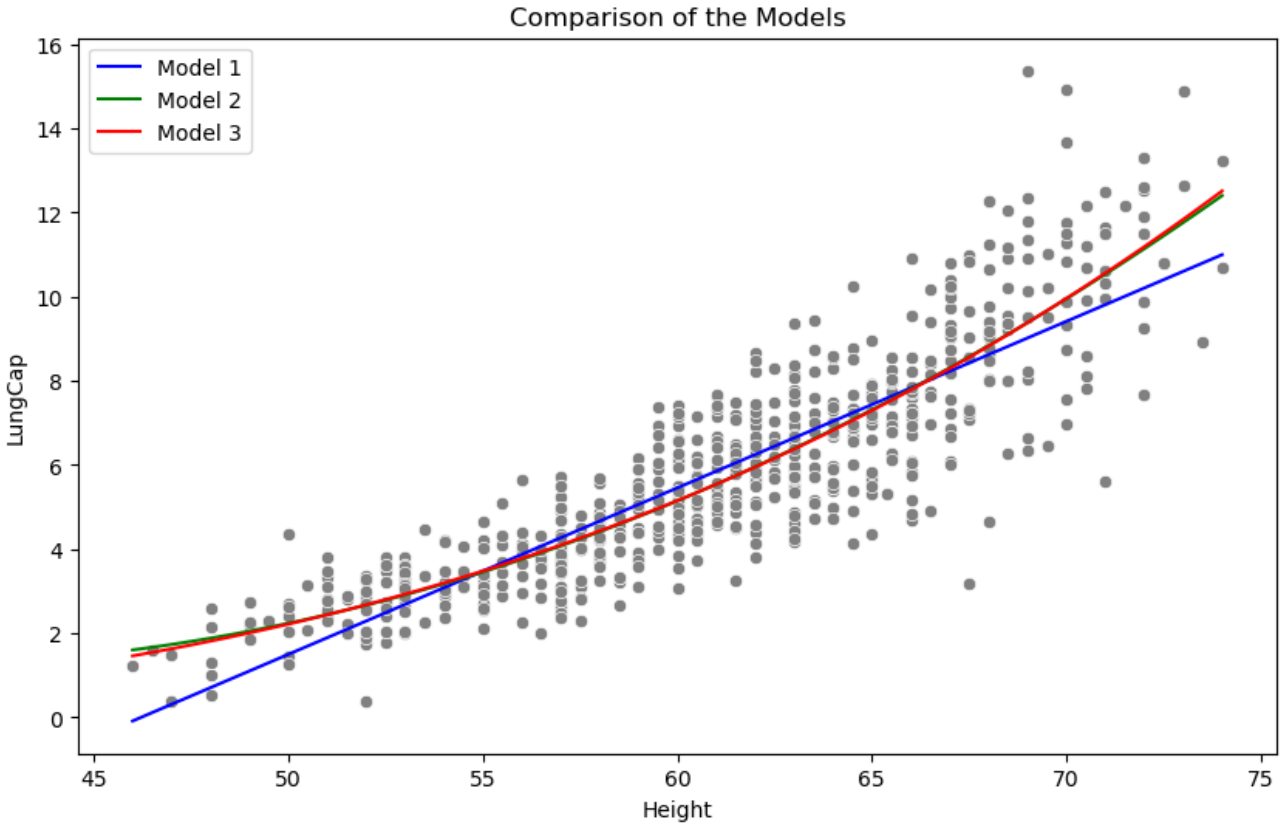
In [66]:
```
#single plot of all 3 models
plt.plot(x_vals, model1.predict(pd.DataFrame({'Height': x_vals})), label='Mo
plt.plot(x_vals, model2.predict(pd.DataFrame({'Height': x_vals, 'Height_sq':
plt.plot(x_vals, model3.predict(pd.DataFrame({'Height': x_vals, 'Height_sq':

plt.xlabel('Height')
plt.ylabel('LungCap')
plt.title('Plot of all 3 models')
plt.legend()
plt.show()
```



In [68]:
```
#scatter plots of the actual data
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Height', y='LungCap', data=lung_data, color='gray')

x_vals = np.linspace(lung_data['Height'].min(), lung_data['Height'].max(), 1

plt.show()
```



In [69]:
```
# single plot of all 3 models compared to scatter plots of the actual data
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Height', y='LungCap', data=lung_data, color='gray')

x_vals = np.linspace(lung_data['Height'].min(), lung_data['Height'].max(), 1


plt.plot(x_vals, model1.predict(pd.DataFrame({'Height': x_vals})), label='Mo
plt.plot(x_vals, model2.predict(pd.DataFrame({'Height': x_vals, 'Height_sq':
plt.plot(x_vals, model3.predict(pd.DataFrame({'Height': x_vals, 'Height_sq':

plt.xlabel('Height')
plt.ylabel('LungCap')
plt.title('Comparison of the Models')
plt.legend()
plt.show()
```



In [ ]:
```
#ANOVA of model 2 and model 1-
#For ANOVA model 1 Vs Model2, Fis 58.91 and P-value is 6.68-14. with such sm
#statistically significant improvement over model 1


#ANOVA of model 2 and model 3-
# #For ANOVA model 2 Vs Model 3, F is 0.19  and P-value is 0.658. Pvalue is
#provide a statistically significant improvement
```