

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

```
In [ ]: df= pd.read_csv('/content/lungdata.csv')
```

```
In [ ]: df
```

```
Out[ ]:
```

	LungCap	Age	Height	Smoke	Gender	Caesarean
0	6.475	6	62.1	no	male	no
1	10.125	18	74.7	yes	female	no
2	9.550	16	69.7	no	female	yes
3	11.125	14	71.0	no	male	no
4	4.800	5	56.9	no	male	no
...	...	...	...	...	...	...
720	5.725	9	56.0	no	female	no
721	9.050	18	72.0	yes	male	yes
722	3.850	11	60.5	yes	female	no
723	9.825	15	64.9	no	female	no
724	7.100	10	67.7	no	male	no

725 rows × 6 columns

## Histogram

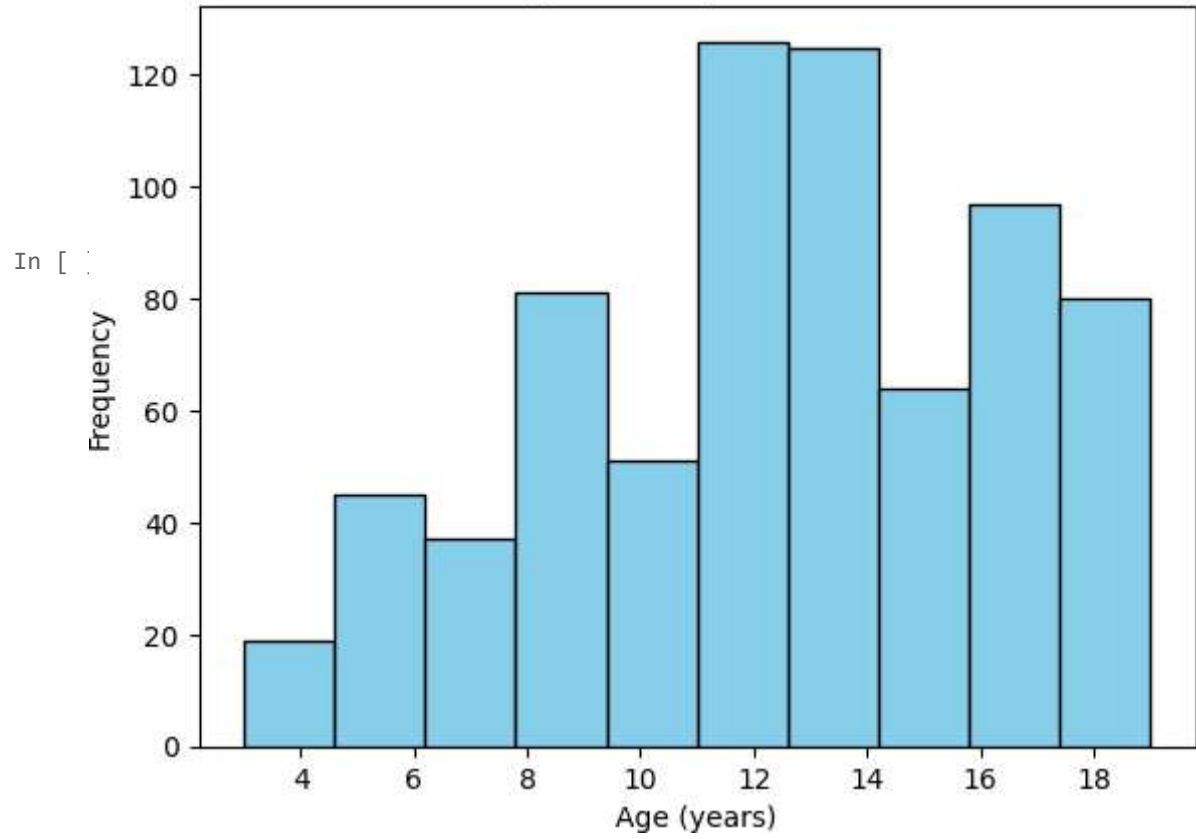
Draw a histogram of the Age column.  
What does the shape of the distribution  
tell you about the ages in the dataset?

```
In [ ]: # Draw a histogram of the Age column
plt.hist(df["Age"], bins=10, color='skyblue', edgecolor='black')

# Add labels and title
plt.title("Histogram of Age Distribution")
plt.xlabel("Age (years)")
plt.ylabel("Frequency")

# Show the plot
plt.show()
```

Histogram of Age Distribution

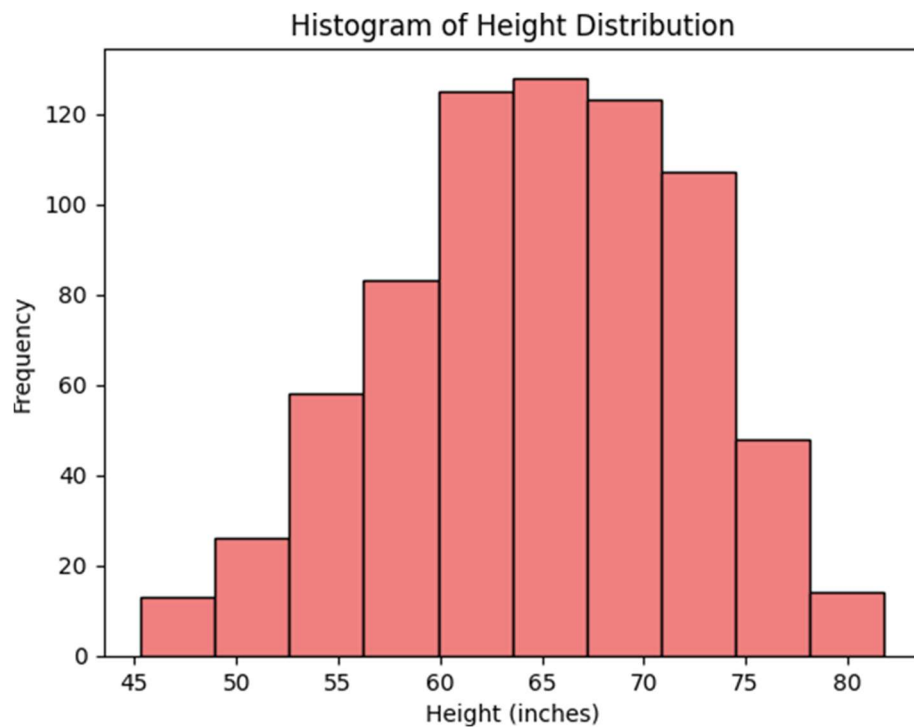


# Create a histogram of Height. What is the most common height range in the dataset?

```
In [ ]: # Plot histogram for Height
plt.hist(df["Height"], bins=10, color='lightcoral', edgecolor='black')

# Add title and labels
plt.title("Histogram of Height Distribution")
plt.xlabel("Height (inches)")
plt.ylabel("Frequency")

# Show the plot
plt.show()
```



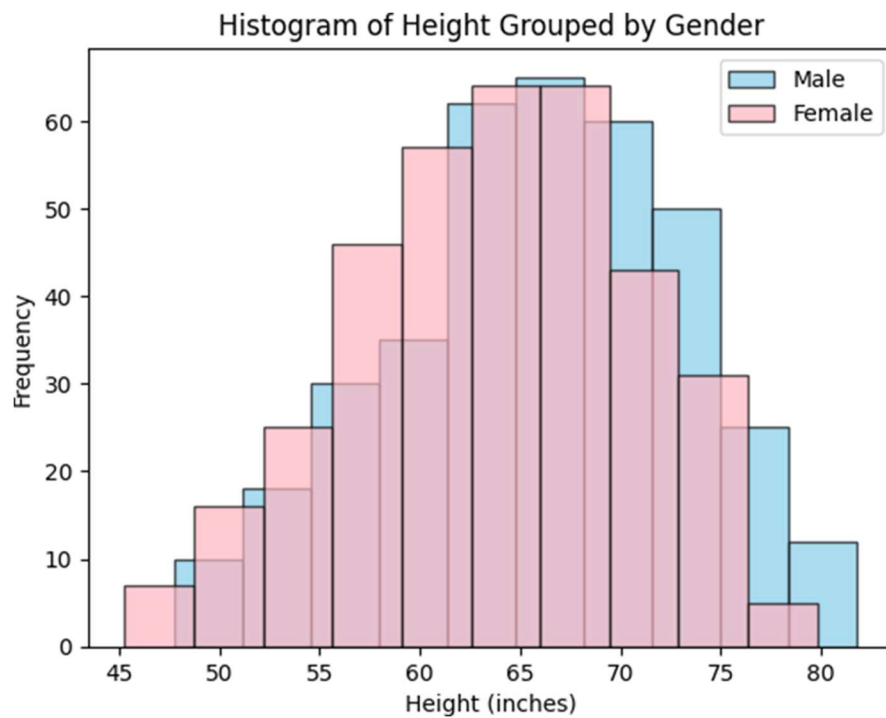
# Create a histogram for Height grouped by Gender. Are males generally taller than females?

```
In [ ]: # Plot histograms of Height grouped by Gender
plt.hist(df[df["Gender"] == "male"]["Height"], bins=10, alpha=0.7, label="Male",
plt.hist(df[df["Gender"] == "female"]["Height"], bins=10, alpha=0.7, label="Fema

# Add title and Labels
plt.title("Histogram of Height Grouped by Gender")
plt.xlabel("Height (inches)")
plt.ylabel("Frequency")

# Add Legend
plt.legend()

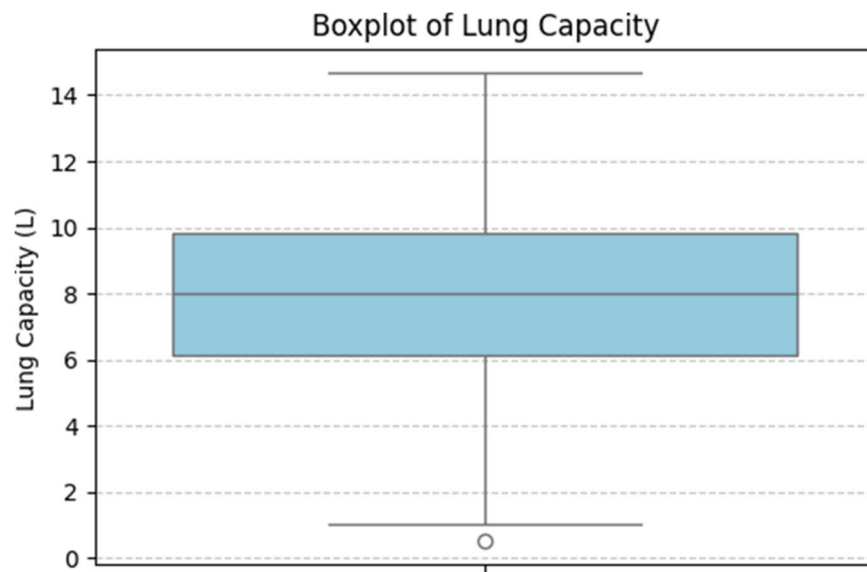
# Show the plot
plt.show()
```



# BoxPlot

Q1: Draw a boxplot of LungCap. What does the boxplot tell you about the spread of lung capacity values?

```
In [ ]: # Draw the boxplot for Lung Capacity
plt.figure(figsize=(6, 4))
sns.boxplot(y=df['LungCap'], color='skyblue')
plt.title('Boxplot of Lung Capacity')
plt.ylabel('Lung Capacity (L)')
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()
```



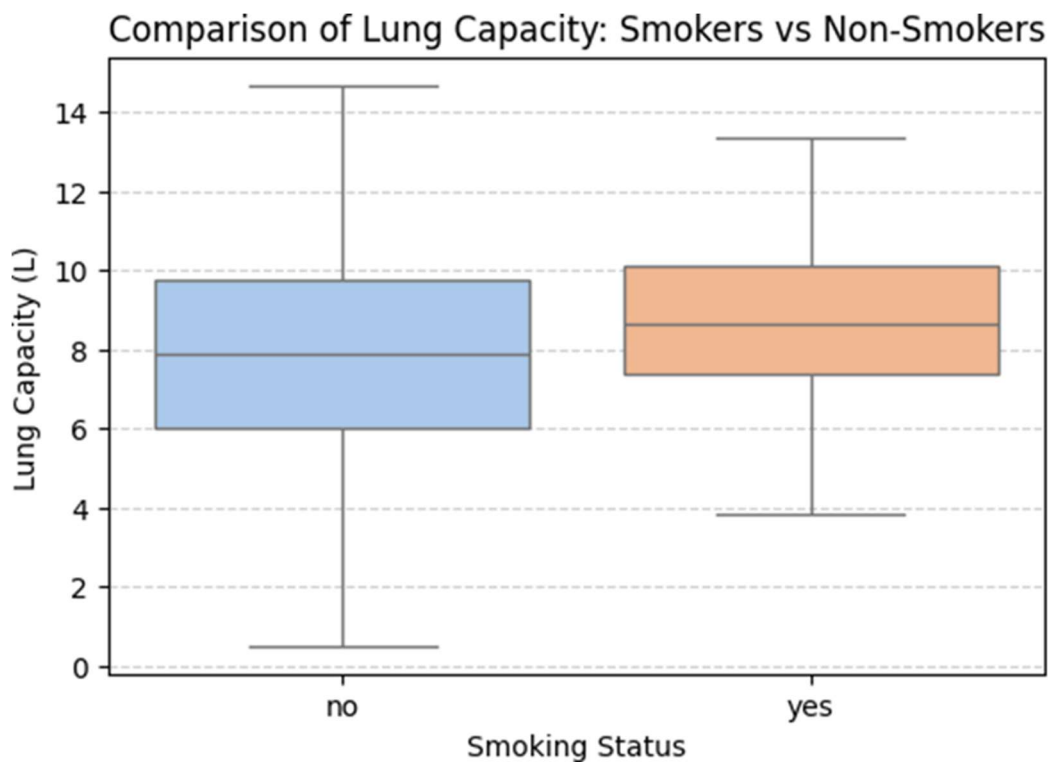
**Q2: Use a boxplot to compare LungCap for smokers (Smoke = yes) and non-smokers (Smoke = no). Do smokers tend to have higher or lower lung capacity?**

```
In [ ]: # Draw boxplot for Lung Capacity grouped by Smoke status
plt.figure(figsize=(6, 4))
sns.boxplot(x='Smoke', y='LungCap', data=df, palette='pastel')
plt.title('Comparison of Lung Capacity: Smokers vs Non-Smokers')
plt.xlabel('Smoking Status')
plt.ylabel('Lung Capacity (L)')
plt.grid(axis='y', linestyle='--', alpha=0.6)
plt.show()
```

/tmp/ipython-input-2804857682.py:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v 0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
sns.boxplot(x='Smoke', y='LungCap', data=df, palette='pastel')
```



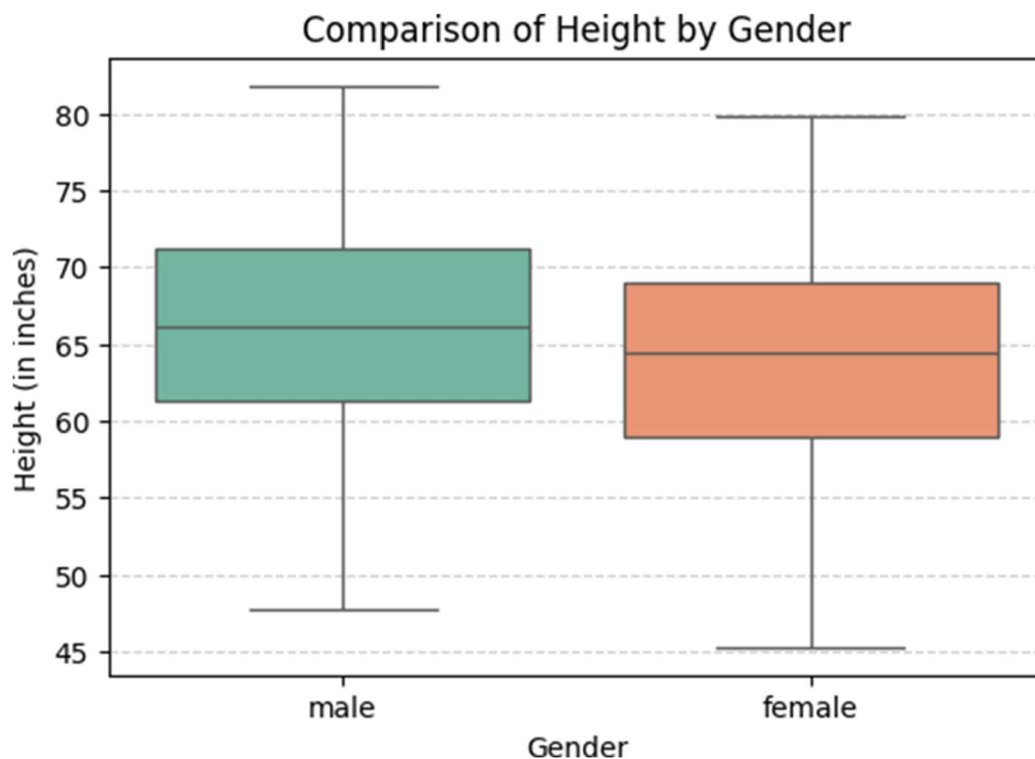
### Q3: Make a boxplot of Height grouped by Gender. Whose median height is greater, males or females?

```
In [ ]: # Draw boxplot for Height grouped by Gender
plt.figure(figsize=(6, 4))
sns.boxplot(x='Gender', y='Height', data=df, palette='Set2')
plt.title('Comparison of Height by Gender')
plt.xlabel('Gender')
plt.ylabel('Height (in inches)')
plt.grid(axis='y', linestyle='--', alpha=0.6)
plt.show()
```

/tmp/ipython-input-2944706162.py:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v 0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

```
sns.boxplot(x='Gender', y='Height', data=df, palette='Set2')
```



# Mean

What is the mean Lung Capacity (LungCap) of all individuals in the dataset?

```
In [ ]: # Calculate mean of LungCap
mean_lungcap = df['LungCap'].mean()

print("Mean Lung Capacity (LungCap):", mean_lungcap)
```

Mean Lung Capacity (LungCap): 7.863147586206895

## Find the mean Age of all participants

```
In [ ]: # Calculate mean Age
mean_age = df['Age'].mean()

print("Mean Age of participants:", mean_age)
```

Mean Age of participants: 12.326896551724138

## Calculate the mean LungCap for male and females separate

```
# Mean LungCap for males
mean_lungcap_male = df[df['Gender'] == 'male']['LungCap'].mean()

# Mean LungCap for females
mean_lungcap_female = df[df['Gender'] == 'female']['LungCap'].mean()

print("Mean LungCap for males:", mean_lungcap_male)
print("Mean LungCap for females:", mean_lungcap_female)
```



# Median

## What is the median Height for smokers?

```
In [ ]: # Filter smokers
smokers = df[df['Smoke'] == 'yes']

# Calculate median Height for smokers
median_height_smokers = smokers['Height'].median()

print("Median Height for smokers:", median_height_smokers)
```

Median Height for smokers: 69.0

## Compare the median LungCap between males and females.

```
In [ ]: # Median LungCap for males
median_lungcap_male = df[df['Gender'] == 'male']['LungCap'].median()

# Median LungCap for females
median_lungcap_female = df[df['Gender'] == 'female']['LungCap'].median()

print("Median LungCap for males:", median_lungcap_male)
print("Median LungCap for females:", median_lungcap_female)
```

Median LungCap for males: 8.35

Median LungCap for females: 7.75

# Mode

## What is the most frequent Caesarean status (yes/no)?

```
In [ ]: # Find the most frequent Caesarean status
most_frequent_caesarean = df['Caesarean'].mode()[0]

print("Most frequent Caesarean status:", most_frequent_caesarean)
```

Most frequent Caesarean status: no

# Quartiles

Find the first (Q1) and third (Q3) quartiles of the LungCap column.

```
In [ ]: # Calculate Q1 and Q3 for LungCap
Q1 = df['LungCap'].quantile(0.25)
Q3 = df['LungCap'].quantile(0.75)

print("First quartile (Q1) of LungCap:", Q1)
print("Third quartile (Q3) of LungCap:", Q3)
```

First quartile (Q1) of LungCap: 6.15  
Third quartile (Q3) of LungCap: 9.8

# Coefficient of Variation

Calculate the CV of LungCap for participants born by Caesarean versus those who were not

```
In [ ]: # Function to calculate Coefficient of Variation
def coefficient_of_variation(series):
    return (series.std() / series.mean()) * 100

# CV for Caesarean = yes
cv_caesarean_yes = coefficient_of_variation(df[df['Caesarean'] == 'yes']['LungCap'])

# CV for Caesarean = no
cv_caesarean_no = coefficient_of_variation(df[df['Caesarean'] == 'no']['LungCap'])

print("CV of LungCap (Caesarean = yes):", round(cv_caesarean_yes, 2), "%")
print("CV of LungCap (Caesarean = no):", round(cv_caesarean_no, 2), "%")
```

CV of LungCap (Caesarean = yes): 33.13 %  
CV of LungCap (Caesarean = no): 34.09 %