

Raeah Woodson  
HW Assignment Week 3

**Code:**

```
from sklearn import datasets
iris = datasets.load_iris()
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
data = { "weight": [4.17, 5.58, 5.18, 6.11, 4.50, 4.61, 5.17, 4.53, 5.33, 5.14, 4.81, 4.17, 4.41,
3.59, 5.87, 3.83, 6.03, 4.89, 4.32, 4.69, 6.31, 5.12, 5.54, 5.50, 5.37, 5.29, 4.92, 6.15, 5.80, 5.26],
"group": ["ctrl"] * 10 + ["trt1"] * 10 + ["trt2"] * 10}
PlantGrowth = pd.DataFrame(data)
```

```
#Q1 a
df = pd.DataFrame(iris.data, columns=iris.feature_names)
df['species'] = pd.Categorical.from_codes(iris.target, iris.target_names)
plt.hist(df['sepal width (cm)'], color='skyblue', edgecolor='black')
plt.show()
```

#Q1 b - I would expect the mean to be slightly higher than the median; there's a lot of observations between 3 & 3.5.

```
#Q1 c - Mean: 3.057, Median: 3.0
mean = np.mean(df['sepal width (cm)'])
median = np.median(df['sepal width (cm)'])
print("Mean:", mean, "Median:", median)
```

```
#Q1 d - 3.3 cm
percentile_73 = np.percentile(df['sepal width (cm)'], 73)
print("Only 27% of the flowers have a sepal width higher than:", percentile_73, "cm")
```

#Q1 e-

```
fig, axes = plt.subplots(2, 3, figsize=(15, 10))
```

```
axes[0, 0].scatter(df['sepal length (cm)'], df['sepal width (cm)'], color='purple')
axes[0, 0].set_xlabel('Sepal Length (cm)')
axes[0, 0].set_ylabel('Sepal Width (cm)')
axes[0, 0].set_title('Sepal Length vs Sepal Width')
```

```
axes[0, 1].scatter(df['sepal length (cm)'], df['petal length (cm)'], color='purple')
axes[0, 1].set_xlabel('Sepal Length (cm)')
axes[0, 1].set_ylabel('Petal Length (cm)')
axes[0, 1].set_title('Sepal Length vs Petal Length')
```

```
axes[0, 2].scatter(df['sepal length (cm)'], df['petal width (cm)'], color='purple')
axes[0, 2].set_xlabel('Sepal Length (cm)')
axes[0, 2].set_ylabel('Petal Width (cm)')
axes[0, 2].set_title('Sepal Length vs Petal Width')
```

```
axes[1, 0].scatter(df['sepal width (cm)'], df['petal length (cm)'], color='purple')
axes[1, 0].set_xlabel('Sepal Width (cm)')
axes[1, 0].set_ylabel('Petal Length (cm)')
axes[1, 0].set_title('Sepal Width vs Petal Length')
```

```
axes[1, 1].scatter(df['sepal width (cm)'], df['petal width (cm)'], color='purple')
axes[1, 1].set_xlabel('Sepal Width (cm)')
axes[1, 1].set_ylabel('Petal Width (cm)')
axes[1, 1].set_title('Sepal Width vs Petal Width')
```

```
axes[1, 2].scatter(df['petal length (cm)'], df['petal width (cm)'], color='purple')
axes[1, 2].set_xlabel('Petal Length (cm)')
axes[1, 2].set_ylabel('Petal Width (cm)')
axes[1, 2].set_title('Petal Length vs Petal Width')
```

```
plt.tight_layout()
plt.show()
```

#Q1 f - Sepal length vs. petal length and Petal Length vs. Petal Width appear to have the strongest positive correlation.  
#Sepal Length vs. Sepal Width and Sepal Width vs. Petal Width appear to have the weakest correlation.

#Q2 a -

```
bins = np.arange(3.3, PlantGrowth['weight'].max() + 0.3, 0.3)
plt.hist(PlantGrowth['weight'], bins=bins, color='skyblue', edgecolor='black')
plt.xlabel('Weight')
plt.ylabel('Frequency')
```

```
plt.title('Histogram of PlantGrowth Weight')
plt.show()
```

#Q2 b-

```
PlantGrowth.boxplot(column='weight', by='group', grid=False)
plt.title('Boxplot of Weight by Group')
plt.xlabel('Group')
plt.ylabel('Weight')
plt.show()
```

#Q2 c- Approximately 80% of trt1 weights are below the minimum trt2 weight, since there's two outliers above.

#Q2 d- 80%

```
min_trt2 = PlantGrowth[PlantGrowth['group'] == 'trt2']['weight'].min()
```

```
trt1_weights = PlantGrowth[PlantGrowth['group'] == 'trt1']['weight']
below = (trt1_weights < min_trt2).sum()
percent = (below / len(trt1_weights)) * 100
```

```
print(f"Approximately {percent:.0f}% of trt1 weights are below the minimum trt2 weight.")
```

#Q2 e-

```
filtered = PlantGrowth[PlantGrowth['weight'] > 5.5]
```

```
filteredtable = filtered['group'].value_counts()
labels_int = filteredtable.index.tolist()
labels = list(map(str, labels_int))
values = filteredtable.values
```

```
sns.barplot(x=labels, y=values, palette='coolwarm')
plt.xlabel('Group')
plt.ylabel('Count')
plt.title('Plants with Weight > 5.5')
plt.show()
```

## Output

Mean: 3.057333333333337 Median: 3.0

Only 27% of the flowers have a sepal width higher than: 3.3 cm





