기초데이터과학 (01분반)

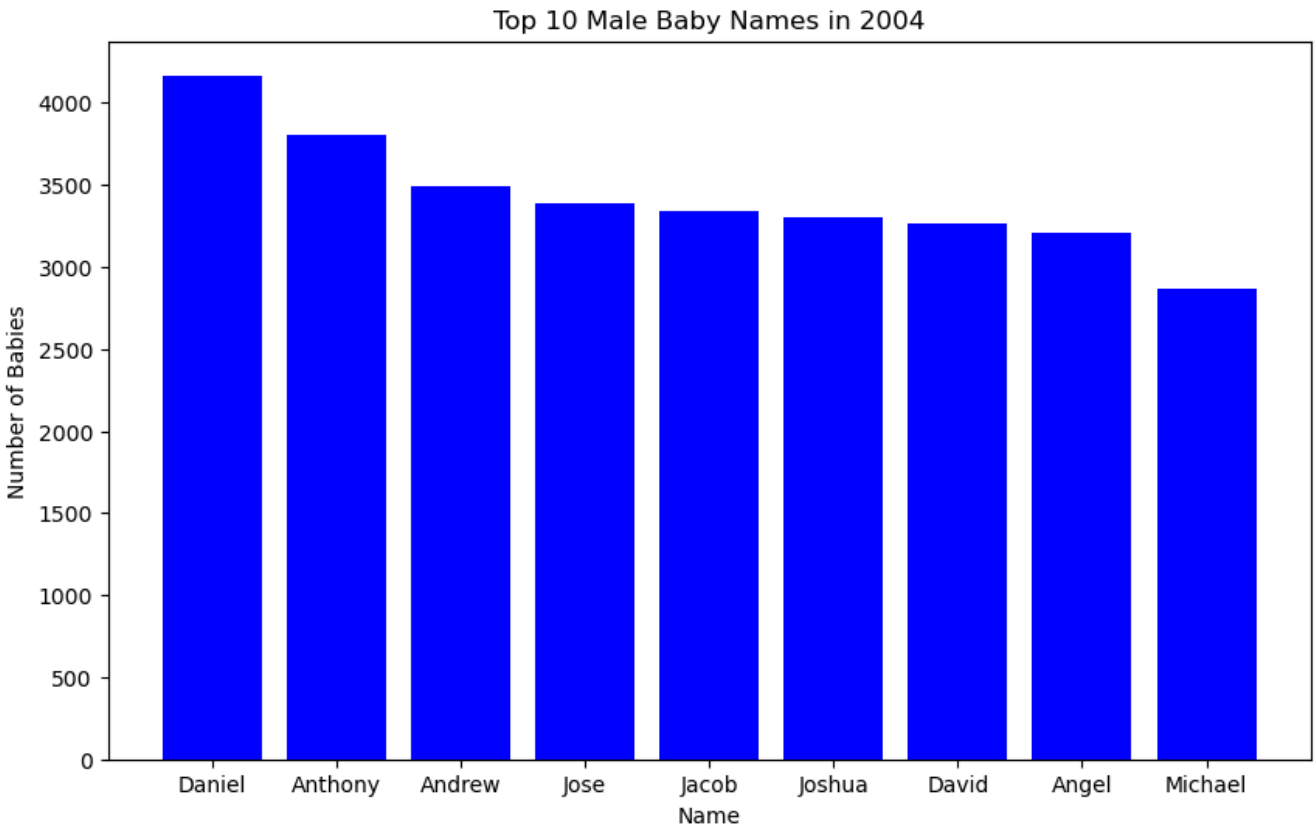
Programming assignment 04

1. 미국의 아기 이름 데이터로 아래 문제에 맞게 코드를 작성하여 그래프를 그리세요.

In [3]:
```
import pandas as pd
import matplotlib.pyplot as plt

# 주어진 데이터 생성
df = pd.read_csv('https://raw.githubusercontent.com/guipsamora/pandas_exercises/master/06_S
```

1-1. 2004년의 남아 아기 이름 중 가장 인기 있는 상위 10개 이름을 막대 그래프로 나타내세요.

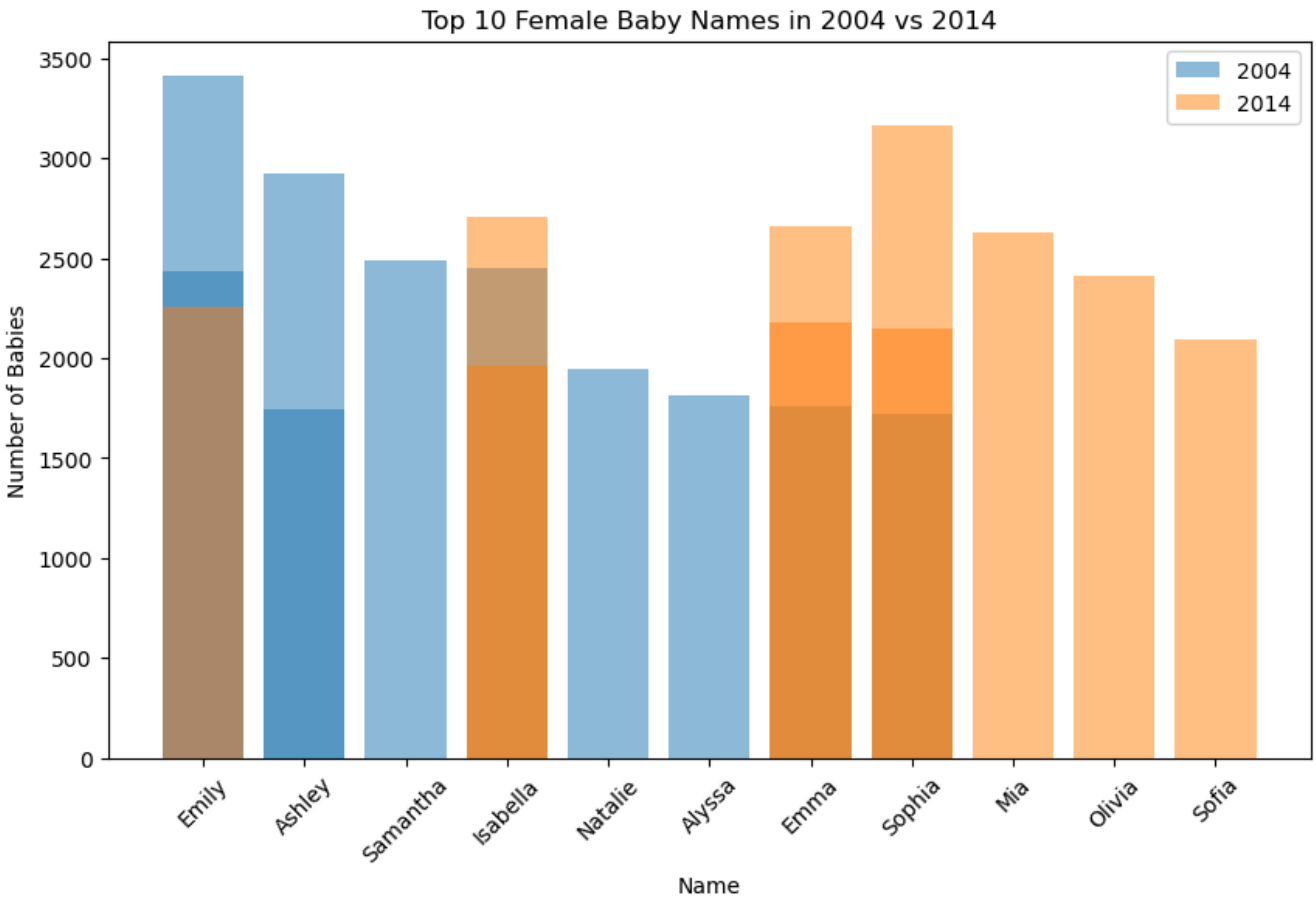
In [5]:
```
# 'Year'가 2004이고 'Gender'가 'M'인 dataframe 구분하여 변수에 저장
df_male = df[(df['Year'] == 2004) & (df['Gender'] == 'M')]

# 'Count'를 가준으로 상위 10개 구분하여 변수에 저장
top_10_male = df_male.nlargest(10, 'Count')

# figure size (10, 6)으로 설정
# matplotlib.pyplot.figure() 함수의 figsize 매개변수
plt.figure(figsize=(10, 6))

# x축은 'Name' y축은 'Count'로 'blue' color의 막대 그래프를 그리기
plt.bar(top_10_male['Name'], top_10_male['Count'], color='blue')

# title은 'Top 10 Male Baby Names in 2004'
plt.title('Top 10 Male Baby Names in 2004')

# xlabel과 ylabel ''Name'', ''Number of Babies''로 설정
plt.xlabel('Name')
plt.ylabel('Number of Babies')

# 그래프 출력
plt.show()
```



1-2. 2004년과 2014년 가장 인기 있는 상위 10개 여아 아기 이름 비교

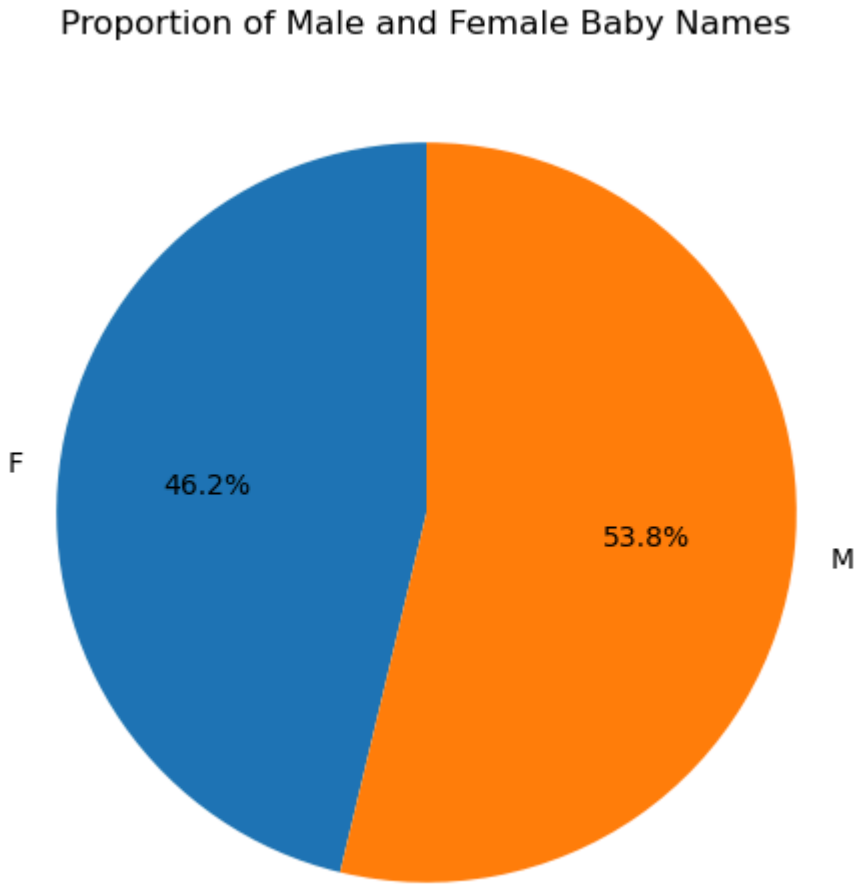
In [7]:

```
# 'Gender'가 'F'이고  'Year'가 2004인 dataframe과 'Year'가 2014 dataframe을 'Count'를 가준으
df_female = df[df['Gender'] == 'F']
top_10_female_2004 = df_female[df_female['Year'] == 2004].nlargest(10, 'Count')
top_10_female_2014 = df_female[df_female['Year'] == 2014].nlargest(10, 'Count')

# figure size (10, 6)으로 설정
plt.figure(figsize=(10, 6))

# x축은 'Name' y축은 'Count'로 2004년 2014년 막대 그래프 2개 그리기, 투명도를 조정해서 변화
# (투명도 조절은 matplotlib.pyplot.bar 함수의 alpha 매개변수 이용)
plt.bar(top_10_female_2004['Name'], top_10_female_2004['Count'], label='2004', alpha=0.5)
plt.bar(top_10_female_2014['Name'], top_10_female_2014['Count'], label='2014', alpha=0.5)

# title은 'Top 10 Male Baby Names in 2000'
plt.title('Top 10 Female Baby Names in 2004 vs 2014')

# xlabel과 ylabel은 'Name', 'Number of Babies'로 설정
plt.xlabel('Name')
plt.ylabel('Number of Babies')

# x축 항목 45도 기울이기 (matplotlib.pyplot.xticks 함수의 rotation 매개변수)
plt.xticks(rotation=45)

# 범례 표시
plt.legend()

# 그래프 출력
plt.show()
```



1-3. 남아와 여아의 이름 수를 파이그래프를 그려 비교하세요.

In [9]:
```
# 'Gender'를 기준으로 'Count' 속성을 더해 성별 인구수 합을 새로운 변수에 저장
names_count = df.groupby('Gender')['Count'].sum()

# figure size (10, 6)으로 설정
plt.figure(figsize=(10, 6))

# 파이 차트로 그리기
plt.pie(names_count, labels=names_count.index, autopct='%1.1f%%', startangle=90)

# title은 'Proportion of Male and Female Baby Names'
plt.title('Proportion of Male and Female Baby Names')

# 그래프 출력
plt.show()
```

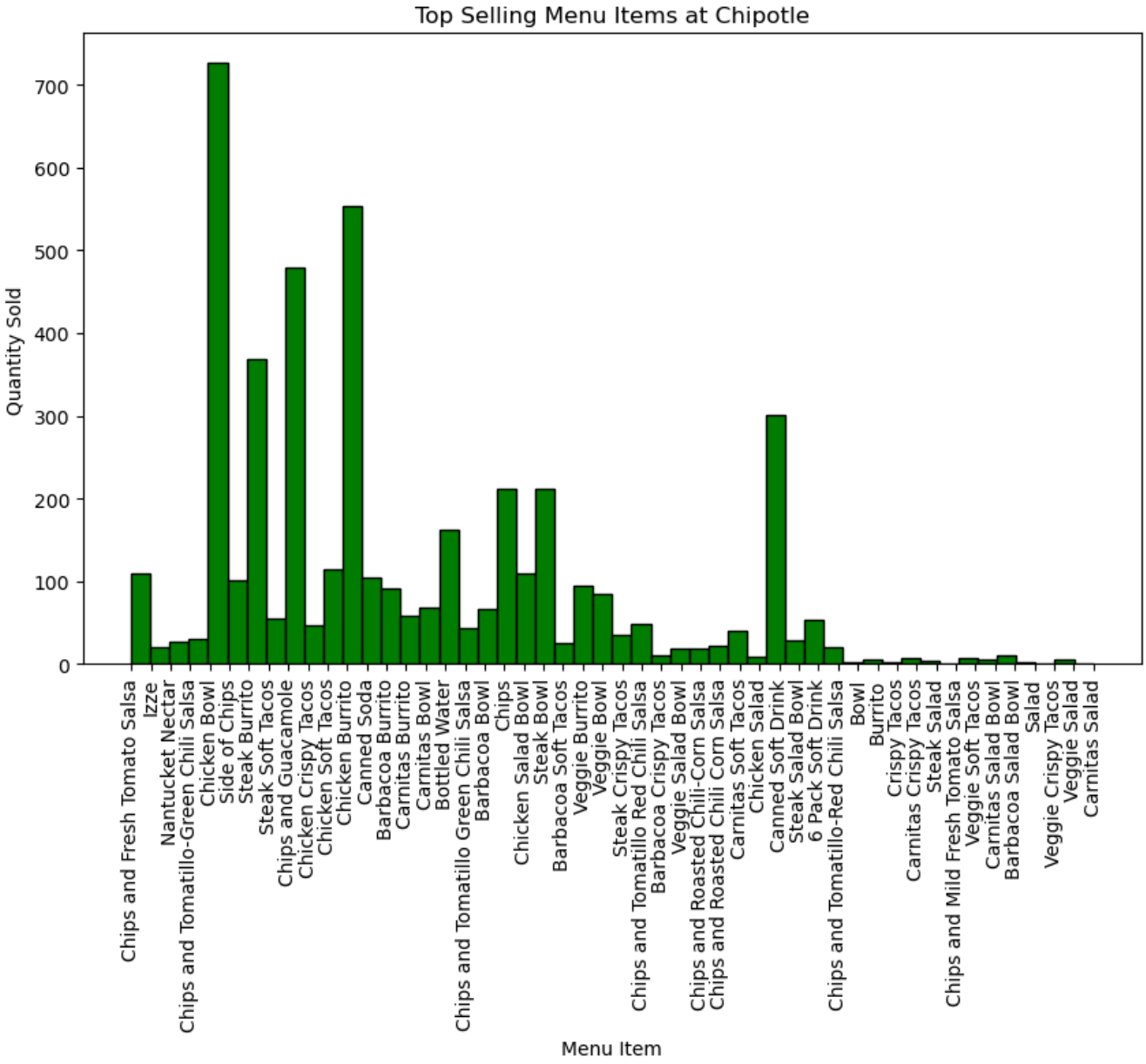


2. Chipotle 데이터로 아래 문제에 맞게 코드를 작성하여 그래프를 그리세요.

In [11]:
```
import pandas as pd
import matplotlib.pyplot as plt

# 주어진 데이터 생성
chipotle_url = 'https://raw.githubusercontent.com/justmarkham/DAT8/master/data/chipotle.tsv
df = pd.read_csv(chipotle_url, sep = '\t')
```

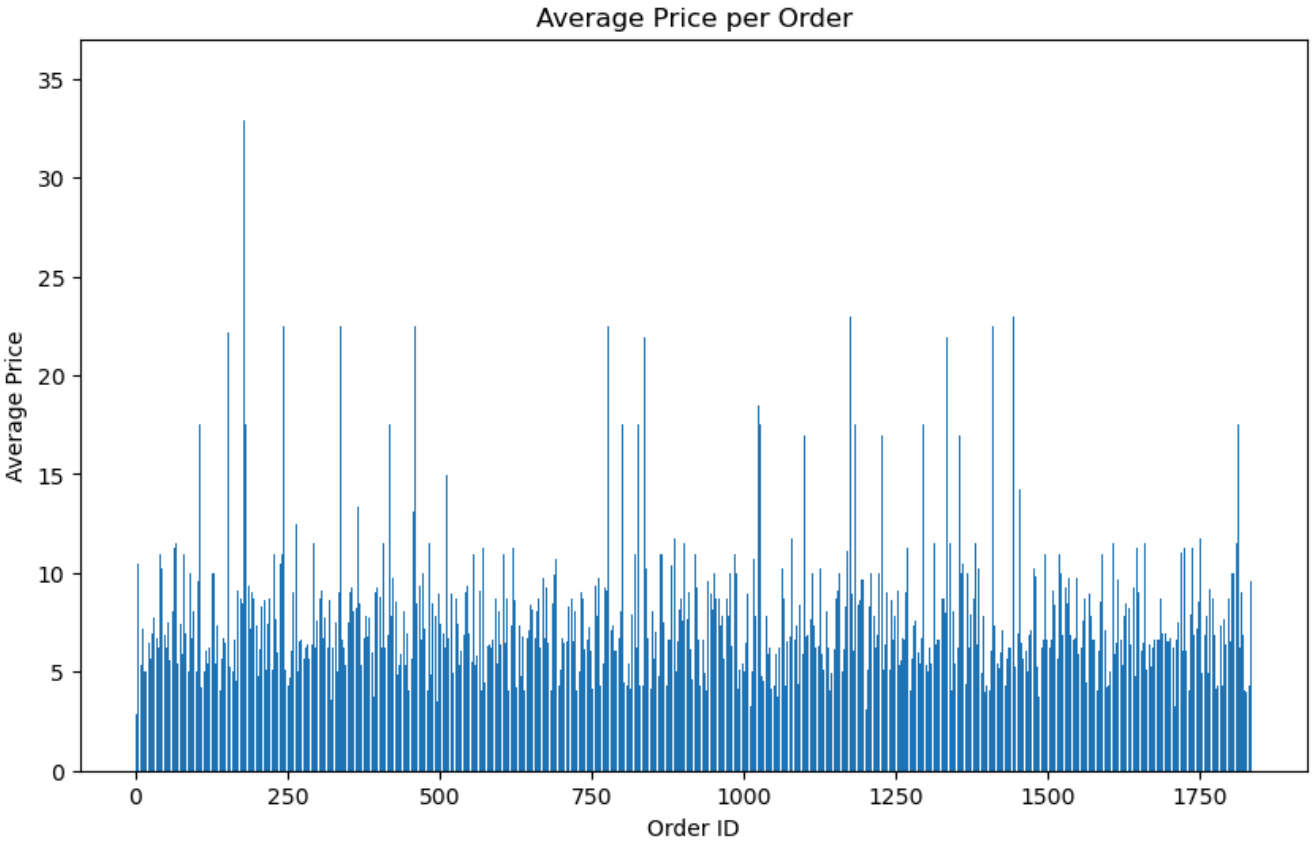
2-1. 히스토그램 그래프를 그려 메뉴 판매수 분포를 나타내세요.

In [13]:
```
# figure size (10, 6)으로 설정
plt.figure(figsize=(10, 6))

# 데이터셋에 팔린 메뉴 수가 어떻게 분포하는지 히스토그램 그래프 그리기
# 'green' 색으로 edgecolor는 'black'으로 bins는 50
plt.hist(df['item_name'], bins=50, color='green', edgecolor='black')

# title은 'Top Selling Menu Items at Chipotle'로 설정
plt.title('Top Selling Menu Items at Chipotle')

# xlabel과 ylabel은 'Menu Item', 'Quantity Sold'로 설정
plt.xlabel('Menu Item')
plt.ylabel('Quantity Sold')

# x축 항목 90도 기울이기
plt.xticks(rotation=90)

# 그래프 출력
plt.show()
```



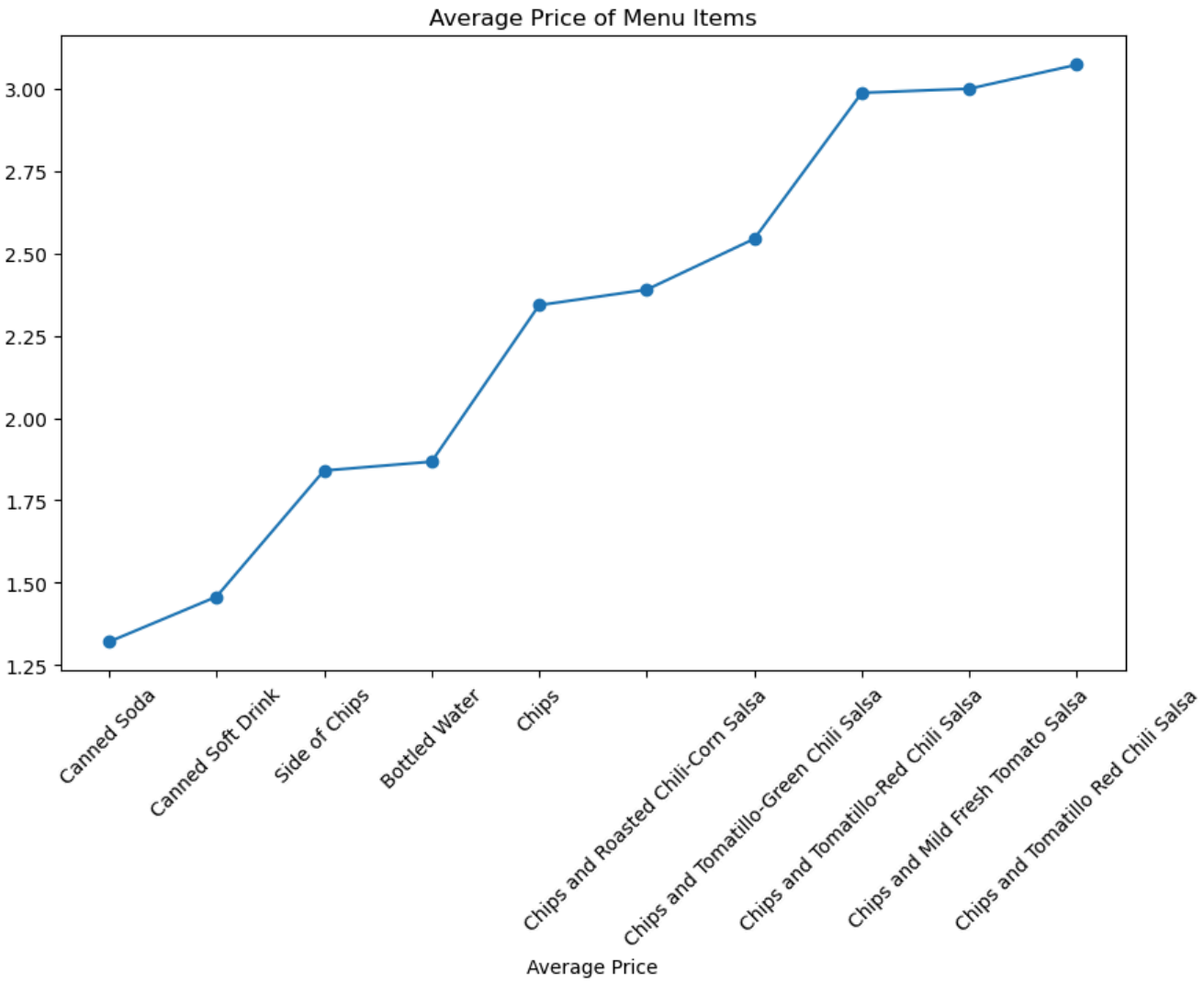
2-2. 주문 id 'order_id' 별 평균 가격을 막대 그래프로 시각화하세요.

In [15]:
```
# 데이터셋의 'order_id'를 기준으로 'item_price'의 평균을 구해서 변수에 저장
# 'item_price'에는 '$'가 포함되어 있는 문자열 값이 저장되어 있으므로 숫자값 변환 과정이 있어
df_copy = df.copy()
df_copy['item_price_numeric'] = df_copy['item_price'].str.replace('$', '').astype(float)
average_price = df_copy.groupby('order_id')['item_price_numeric'].mean()

# figure size (10, 6)으로 설정, bar 그래프 그리기
plt.figure(figsize=(10, 6))
plt.bar(average_price.index, average_price.values)

# title은 'Average Price per Order'로 설정
plt.title('Average Price per Order')

# xlabel과 ylabel은 'Order ID', 'Average Price'로 설정
plt.xlabel('Order ID')
plt.ylabel('Average Price')

# 그래프 출력
plt.show()
```



2-3. 상위 10개 메뉴의 평균 가격을 선 그래프를 그려 비교하세요.

In [17]:
```
# 데이터셋의 'item_name'를 기준으로 'item_price'의 평균을 구하고 정렬한뒤 상위 10개 메뉴를 반
average_item_price = df_copy.groupby('item_name')['item_price_numeric'].mean() \
                                    .sort_values().head(10)

# figure size (10, 6)으로 설정, marker를 'o'로 설정하고 bar 그래프 그리기
plt.figure(figsize=(10, 6))
plt.plot(average_item_price.index, average_item_price.values, marker='o')

# title은 'Average Price of Menu Items'로 설정
plt.title('Average Price of Menu Items')

# xlabel은 'Average Price'로 설정
plt.xlabel('Average Price')

# x축 항목 45도 기울이기
plt.xticks(rotation=45)

# 그래프 출력
plt.show()
```



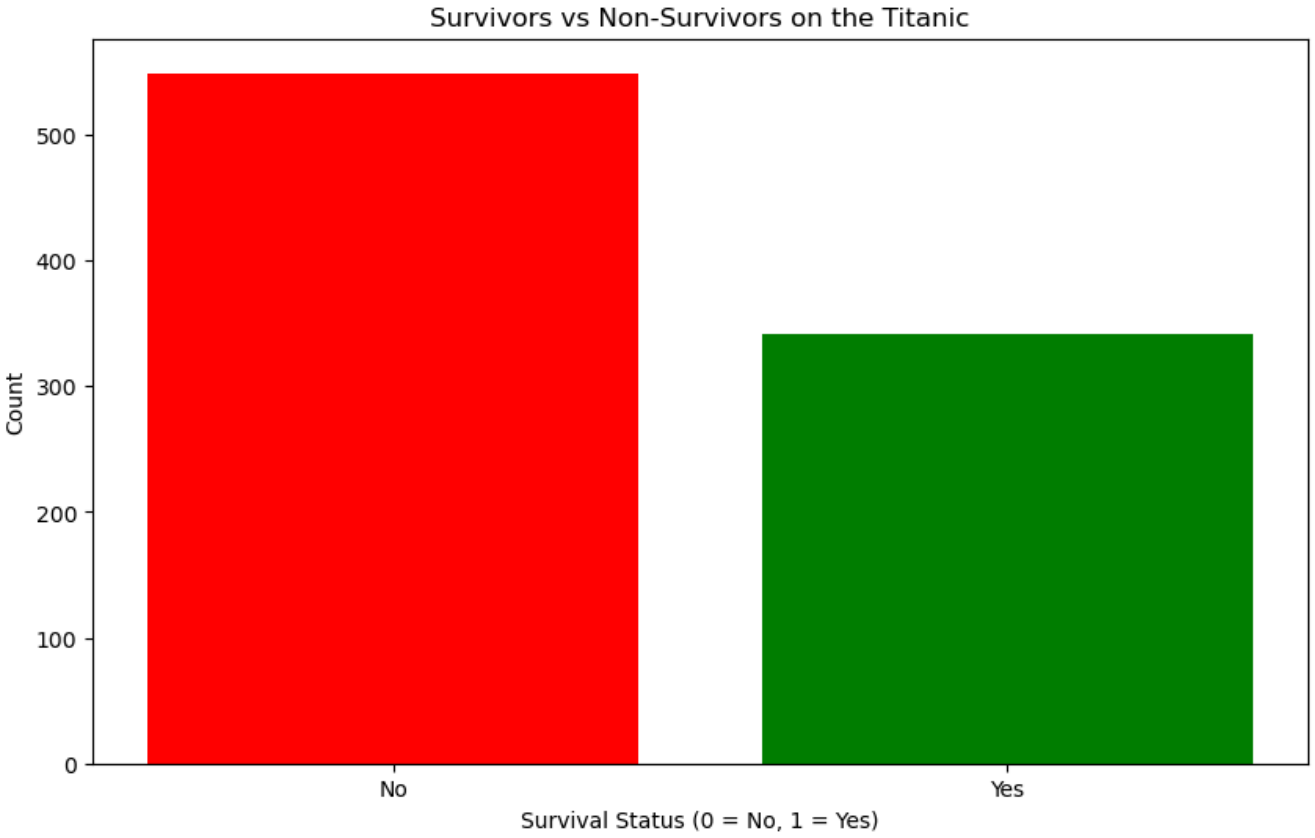
3. 타이타닉 승객 정보 데이터로 아래 문제에 맞게 코드를 작성하여 그래프를 그리세요.

In [19]:
```
import pandas as pd
import matplotlib.pyplot as plt

# 주어진 데이터 생성
df = pd.read_csv('https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic
```

3-1. 타이타닉 호의 생존자 수와 사망자 수를 막대그래프를 그려 비교하세요.

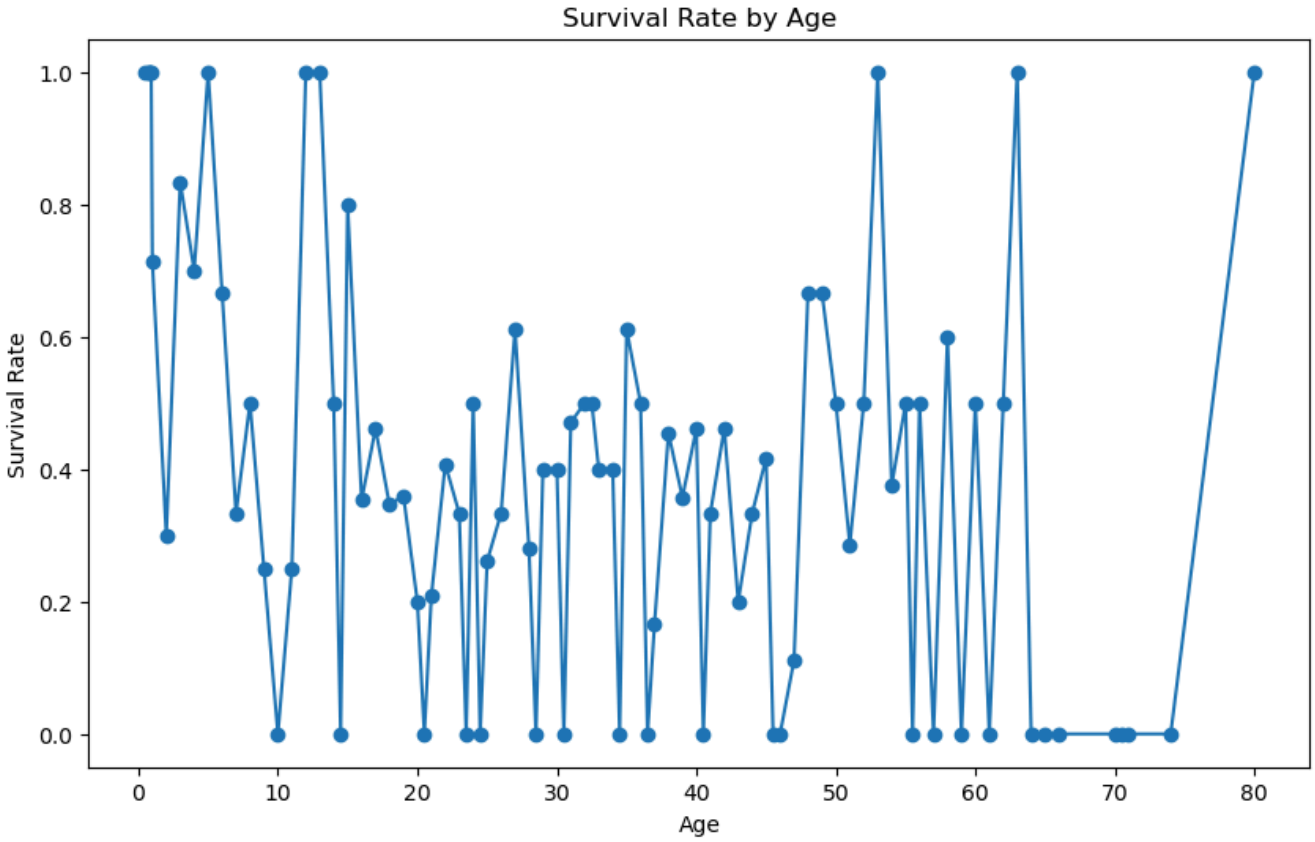
In [21]:
```
# 데이터의 'Survived' 속성의 0, 1 개수를 세서 생존자와 사망자의 수를 구하고 변수에 저장
# pandas.DataFrame.value_counts 활용, 다른 방법 사용 가능
survival_counts = df['Survived'].value_counts()

# figure size (10, 6)으로 설정, 색을 사망자는 'red', 생존자는 'green'으로 설정하여 bar 그래
plt.figure(figsize=(10, 6))
plt.bar(survival_counts.index, survival_counts.values, color=['red', 'green'])

# title은 'Survivors vs Non-Survivors on the Titanic'로 설정
plt.title('Survivors vs Non-Survivors on the Titanic')

# xlabel과 ylabel은 'Survival Status (0 = No, 1 = Yes)', 'Count'로 설정
plt.xlabel('Survival Status (0 = No, 1 = Yes)')
plt.ylabel('Count')

# x축 항목 이름을 [0, 1]에서 ['No', 'Yes']로 변경
plt.xticks(ticks=[0, 1], labels=['No', 'Yes'])
```

```
# 그래프 출력
plt.show()
```



3-2. 나이에 따른 생존 비율 변화를 선 그래프로 나타내세요.

In [23]:
```
# 'Age'를 기준으로 'Survived'의 평균을 계산하여 변수에 저장
# 0과 1로 구성된 'Survived'의 평균은 생존 비율을 나타냄
age_survival = df.groupby('Age')['Survived'].mean()

# figure size (10, 6)으로 설정
plt.figure(figsize=(10, 6))

# marker를 'o'로 설정하여 선 그래프 그리기
plt.plot(age_survival.index, age_survival.values, marker='o')

# title은 'Survival Rate by Age'로 설정
plt.title('Survival Rate by Age')

# xlabel과 ylabel은 'Age', 'Survival Rate'로 설정
plt.xlabel('Age')
plt.ylabel('Survival Rate')

# 그래프 출력
plt.show()
```

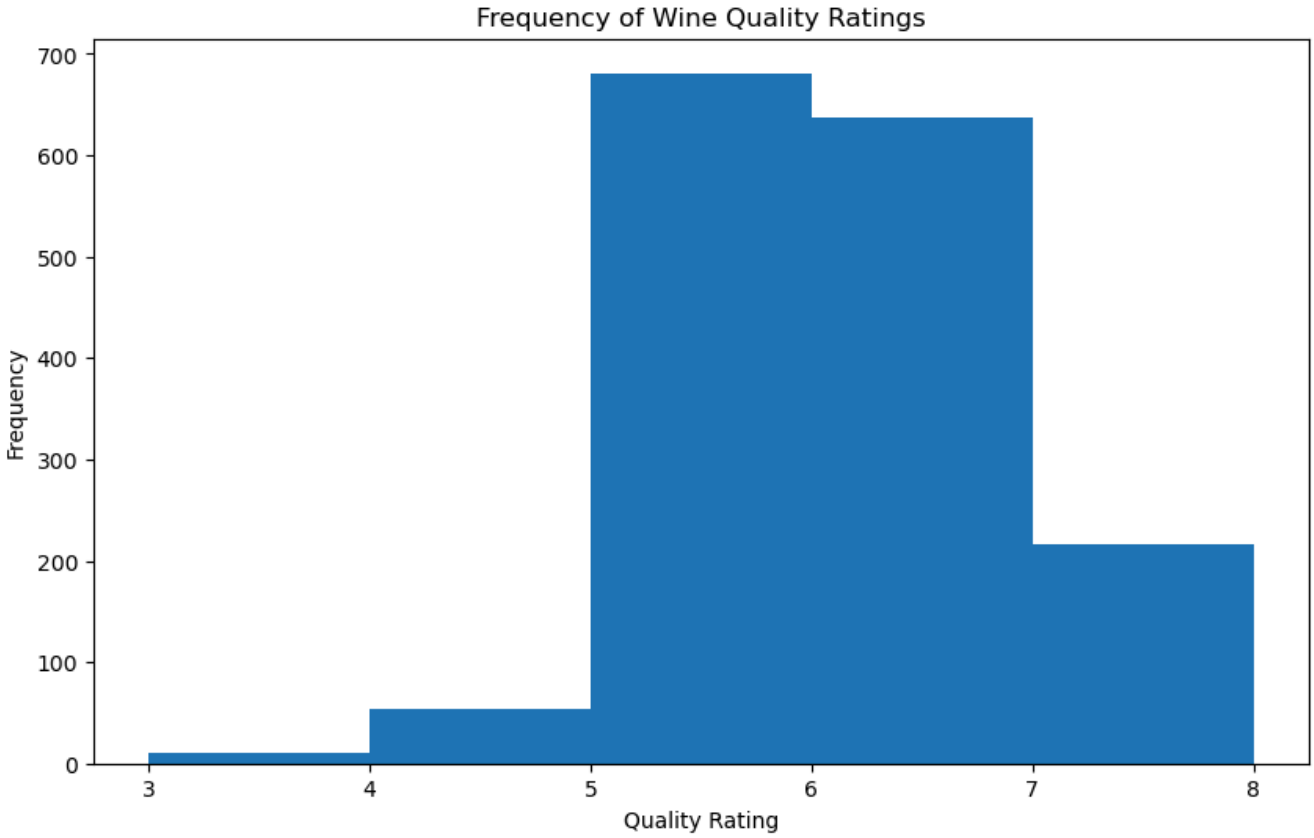


4. 와인 품질 데이터로 아래 문제에 맞게 코드를 작성하여 그래프를 그리세요.

In [25]:
```
import pandas as pd
import matplotlib.pyplot as plt

# 주어진 데이터 생성
df = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/wi
```

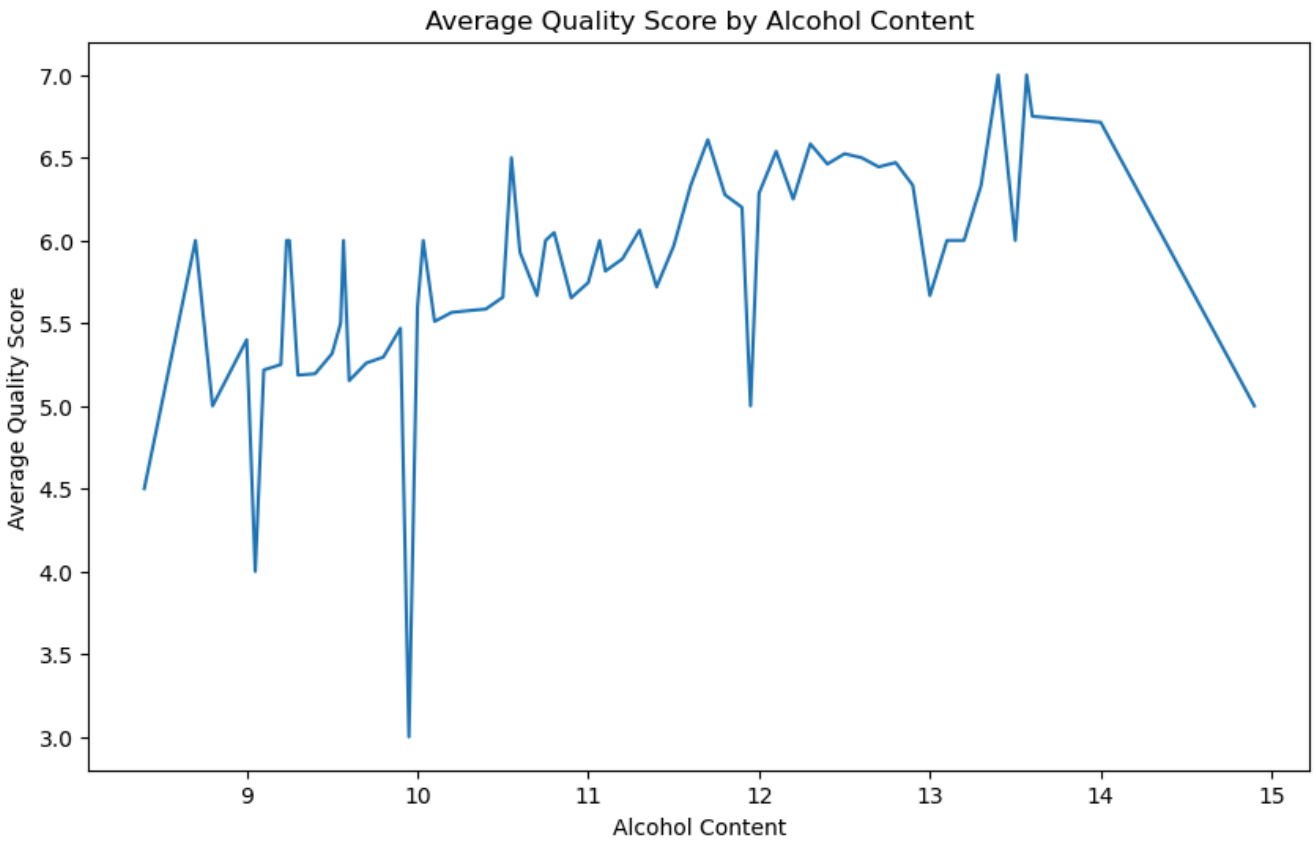
4-1. 히스토그램 그래프를 그려 와인 품질 분포를 나타내세요.

In [27]:
```
# figure size (10, 6)으로 설정
plt.figure(figsize=(10, 6))

# 데이터셋에 와인 퀄리티가 어떻게 분포하는지 히스토그램 그래프 그리기
# bins 는 5
plt.hist(df['quality'], bins=5)

# title은 'Frequency of Wine Quality Ratings'로 설정
plt.title('Frequency of Wine Quality Ratings')

# xlabel과 ylabel은 'Quality Rating', 'Frequency'로 설정
plt.xlabel('Quality Rating')
plt.ylabel('Frequency')

# 그래프 출력
plt.show()
```



4-2. 알코올 도수에 따른 평균 품질 점수를 상자 그림으로 그려보세요.

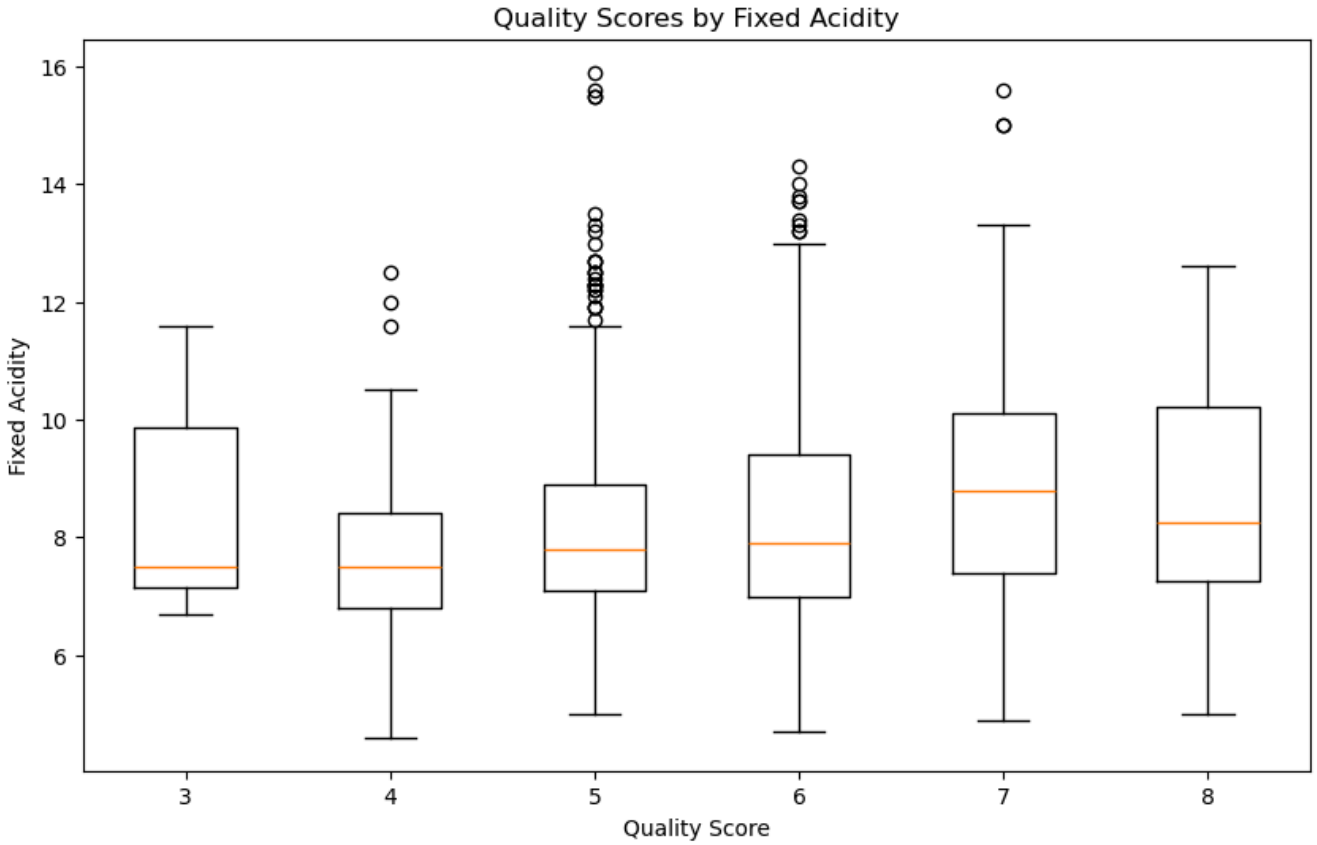
In [29]:
```
# 'alcohol' 속성을 기준으로 'quality'의 평균을 계산하여 변수에 저장
average_quality_by_alcohol = df.groupby('alcohol')['quality'].mean()

# figure size (10, 6)으로 설정
plt.figure(figsize=(10, 6))

# 선 그래프 그리기
plt.plot(average_quality_by_alcohol.index, average_quality_by_alcohol.values)

# title은 'Average Quality Score by Alcohol Content'로 설정
plt.title('Average Quality Score by Alcohol Content')

# xlabel과 ylabel은 'Alcohol Content', 'Average Quality Score'로 설정
plt.xlabel('Alcohol Content')
plt.ylabel('Average Quality Score')

# 그래프 출력
plt.show()
```



4-3. 품질에 따른 산도 분포를 상자 그림(box plot)으로 표현하세요.

In [31]:

```
# 각 품질 점수 (quality(3, 4, 5, 6, 7, 8)) 별로 나누어 산도 분포 ('fixed acidity') 저장
# 2차원 배열 형태로 만들 것 ex) [[1, 2, 3], [1.1, 2.2, 3.3], [5, 6, 7, 8], [6.5], [3.2, 4.5
quality3 = df[df['quality'] == 3]['fixed acidity'].values
quality4 = df[df['quality'] == 4]['fixed acidity'].values
quality5 = df[df['quality'] == 5]['fixed acidity'].values
quality6 = df[df['quality'] == 6]['fixed acidity'].values
quality7 = df[df['quality'] == 7]['fixed acidity'].values
quality8 = df[df['quality'] == 8]['fixed acidity'].values

acidity_by_quality = [quality3, quality4, quality5, quality6, quality7, quality8]

quality_levels = [3, 4, 5, 6, 7, 8]  # 사용할 산도 범위
#acidity_by_quality = [df[df['quality'] == level]['fixed acidity'].values for level in qual

# figure size (10, 6)으로 설정
plt.figure(figsize=(10, 6))

# x축 label을 3, 4, 5, 6, 7, 8로 설정하여 상자 그림(box plot) 그리기
plt.boxplot(acidity_by_quality, labels=quality_levels)

# title은 'Quality Scores by Fixed Acidity'로 설정
plt.title('Quality Scores by Fixed Acidity')

# xlabel과 ylabel은 ('Quality Score', 'Fixed Acidity'로 설정
plt.xlabel('Quality Score')
plt.ylabel('Fixed Acidity')

# 그래프 출력
plt.show()
```


Quality Scores by Fixed Acidity
Fixed Acidity
16
14
12
10
8
6
3
4
5
6
7
8
Quality Score