In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(color_codes=True)
```

In [2]:

```
# We will work with a simple dataset that contains details of wine quality
# Task 1
# Load and study the data
```

In [3]:

```
# Read File
study = pd.read_csv(r"C:\Desktop\DataAnalytics\UnifiedMentor\Wine Quality D
ataset.csv")
```

In [4]:

```
#Take a look at the data
study.head()
```

Out[4]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcoh
0	7.0	0.27	0.36	20.7	0.045	45.0	170.0	1.0010	3.00	0.45	8
1	6.3	0.30	0.34	1.6	0.049	14.0	132.0	0.9940	3.30	0.49	9
2	8.1	0.28	0.40	6.9	0.050	30.0	97.0	0.9951	3.26	0.44	10
3	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9
4	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9

In [5]:

```
# Get dimensions of dataframe

study.shape
```

Out[5]:

```
(4898, 12)
```

In [6]:

```
# get the row names

study.index
```

Out[6]:

```
RangeIndex(start=0, stop=4898, step=1)
```

In [7]:

```
# Get the columns
study.columns
```

Out[7]:

```
Index(['fixed acidity', 'volatile acidity', 'citric acid', 'residual suga
r',
       'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'densit
y',
       'pH', 'sulphates', 'alcohol', 'quality'],
      dtype='object')
```

In [8]:

```
# Basic Info

study.info()
```

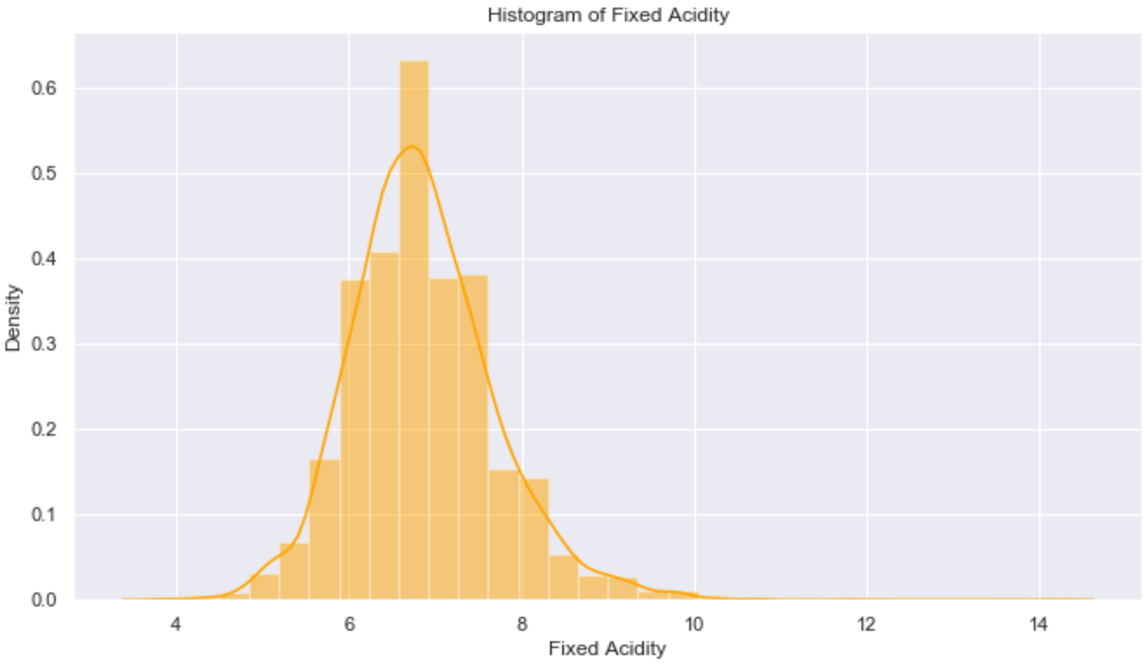
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4898 entries, 0 to 4897
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   fixed acidity         4898 non-null   float64
 1   volatile acidity      4898 non-null   float64
 2   citric acid           4898 non-null   float64
 3   residual sugar        4898 non-null   float64
 4   chlorides             4898 non-null   float64
 5   free sulfur dioxide   4898 non-null   float64
 6   total sulfur dioxide  4898 non-null   float64
 7   density               4898 non-null   float64
 8   pH                    4898 non-null   float64
 9   sulphates             4898 non-null   float64
 10  alcohol               4898 non-null   float64
 11  quality               4898 non-null   int64  
dtypes: float64(11), int64(1)
memory usage: 459.3 KB
```

In [9]:

```
# Observations from Task 1
#There are 4898 rows and 12 columns in the data
#Each row contains the details of the type of acids present in white wine a
nd the quality
#Features - Different acids and their quality
```

In [10]:
```
# Task 2 -
# View the distributions of various features in the dataset and calculate the central tendency
# Create histogram of fixed acidity features

plt.figure(figsize=(11, 6))
sns.distplot(study['fixed acidity'], color='orange', hist_kws={'edgecolor': 'linen', 'alpha': 0.5}, bins=30)
plt.title("Histogram of Fixed Acidity")
plt.xlabel('Fixed Acidity')
plt.ylabel('Density')
plt.show()
```



In [11]:
```
# Calculate mean
round(study['fixed acidity'].mean(),2)
```

Out[11]:
```
6.85
```

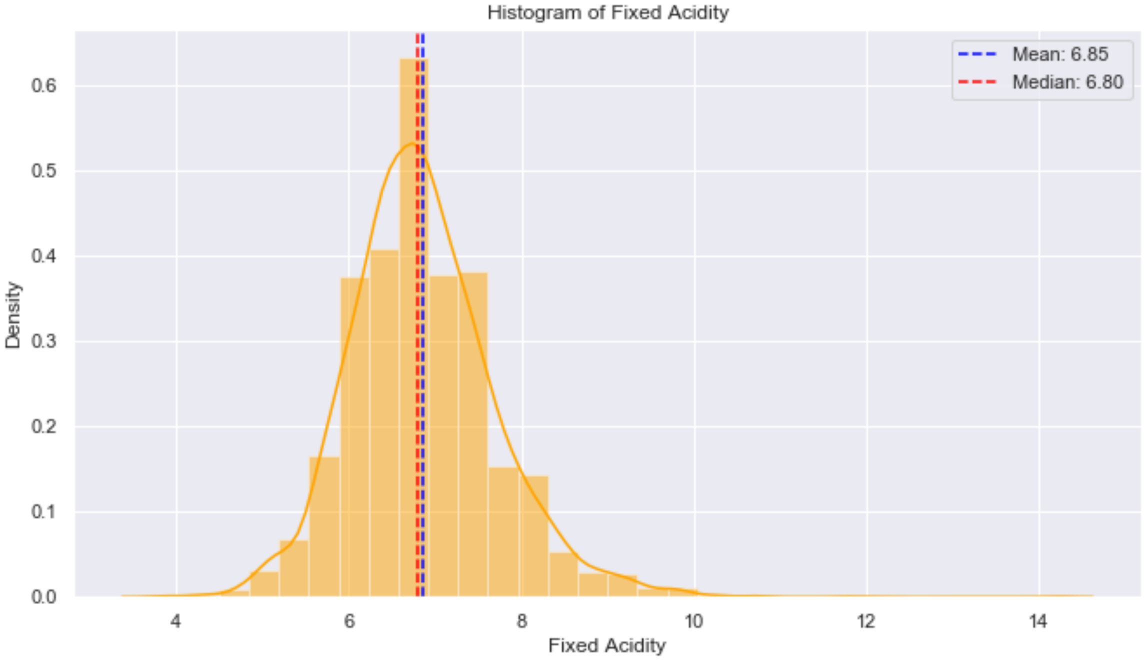
In [12]:
```
# calculate median
study['fixed acidity'].median()
```

Out[12]:
```
6.8
```

In [13]:

```
# Histogram with mean and median
plt.figure(figsize=(11, 6))
sns.distplot(study['fixed acidity'], color='orange', hist_kws={'edgecolor': 'linen', 'alpha': 0.5}, bins=30)
plt.title("Histogram of Fixed Acidity")
plt.xlabel('Fixed Acidity')
plt.ylabel('Density')
mean_val = round(study['fixed acidity'].mean(),2)
median_val = study['fixed acidity'].median()
plt.axvline(mean_val, color='blue', linestyle='--', label=f'Mean: {mean_val:.2f}')
plt.axvline(median_val, color='red', linestyle='--', label=f'Median: {median_val:.2f}')

plt.legend()
plt.show()
```

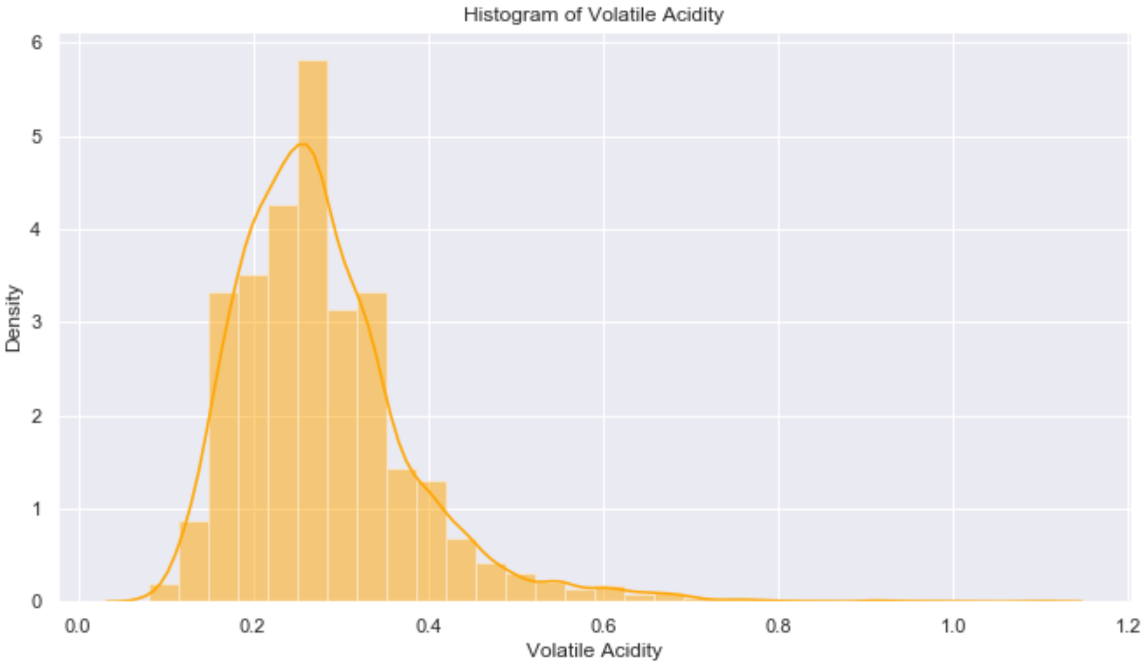


In [14]:

```
# Observations
# We can see that mean and median are clear representative of the data.
# Mean and median are very close to each other. So we are taking mean as the measure of central tendency
```

In [15]:

```
# Volatile Acidity Features
plt.figure(figsize=(11, 6))
sns.distplot(study['volatile acidity'], color='orange', hist_kws={'edgecolor': 'linen', 'alpha': 0.5}, bins=30)
plt.title("Histogram of Volatile Acidity")
plt.xlabel('Volatile Acidity')
plt.ylabel('Density')
plt.show()
```



In [16]:

```
#The above plot shows normal distribution. often in bell curve
# we can calculate skewness using skew function
study['volatile acidity'].skew()
```

Out[16]:

```
1.5769795029952025
```

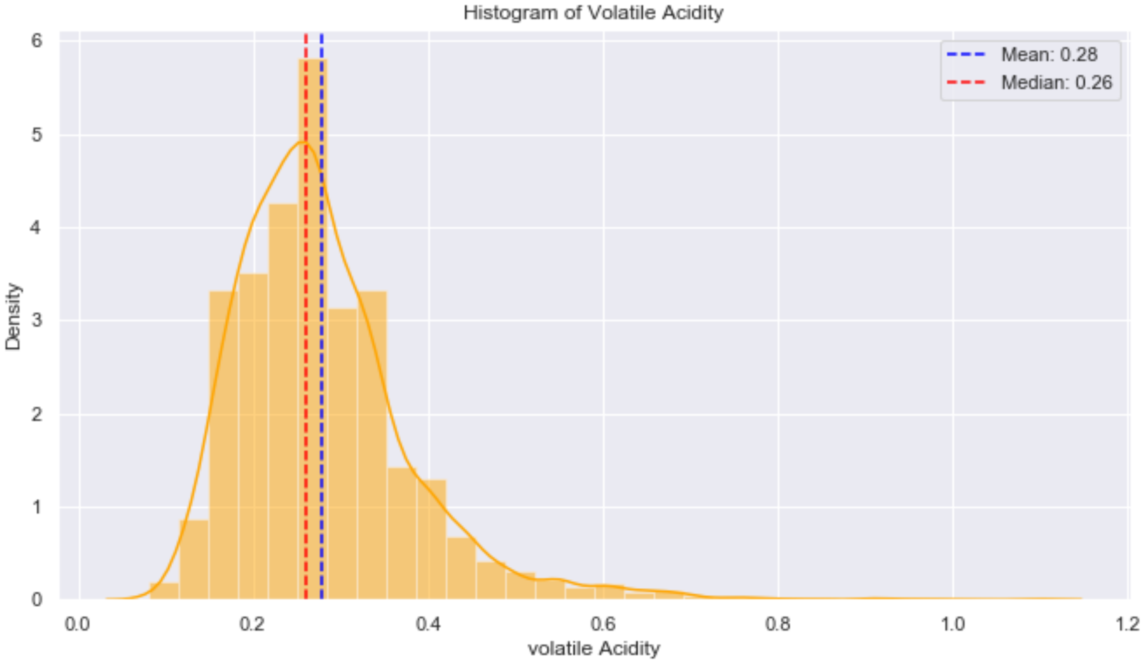
In [17]:

```
# We can clearly see that the skewness value is grater than 1. Hence it is positvely skewed
```

In [18]:

```
# Create histogram with mean and median
plt.figure(figsize=(11, 6))
sns.distplot(study['volatile acidity'], color='orange', hist_kws={'edgecolor': 'linen', 'alpha': 0.5}, bins=30)
plt.title("Histogram of Volatile Acidity")
plt.xlabel('volatile Acidity')
plt.ylabel('Density')
mean_val = study['volatile acidity'].mean()
median_val = study['volatile acidity'].median()
plt.axvline(mean_val, color='blue', linestyle='--', label=f'Mean: {mean_val:.2f}')
plt.axvline(median_val, color='red', linestyle='--', label=f'Median: {median_val:.2f}')

plt.legend()
plt.show()
```



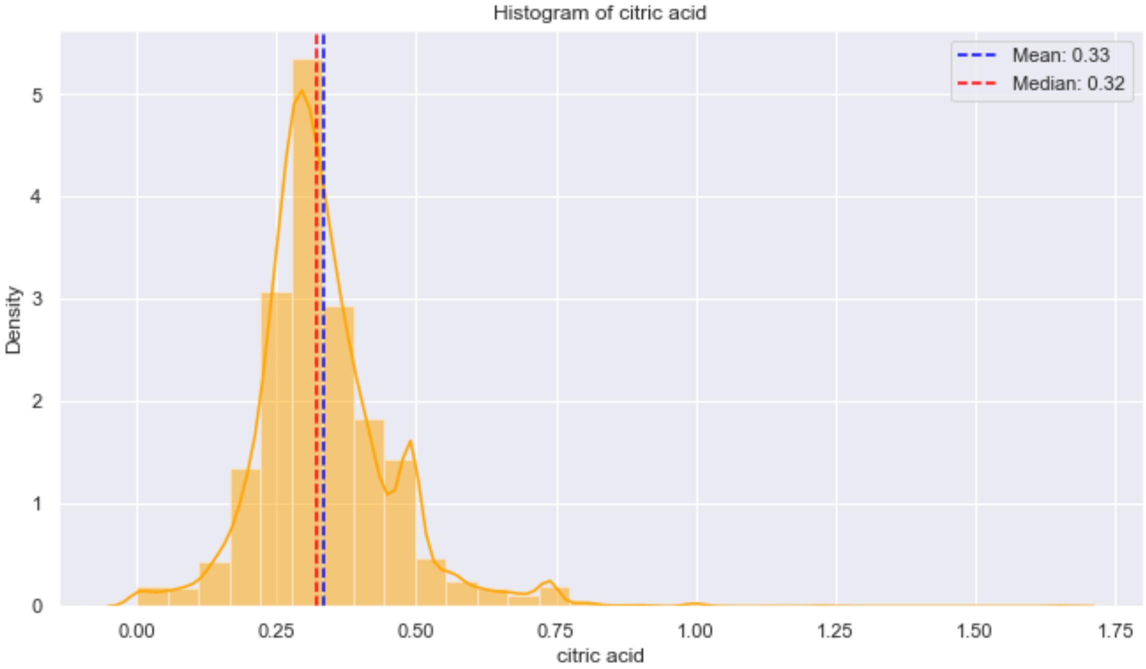
In [19]:

```
#Observations
#The mean and median are close. We can choose the mean as the central tendency
```

In [20]:

```
# Citic Acid
plt.figure(figsize=(11, 6))
sns.distplot(study['citric acid'], color='orange', hist_kws={'edgecolor': 'linen', 'alpha': 0.5}, bins=30)
plt.title("Histogram of citric acid")
plt.xlabel('citric acid')
plt.ylabel('Density')
mean_val = study['citric acid'].mean()
median_val = study['citric acid'].median()
plt.axvline(mean_val, color='blue', linestyle='--', label=f'Mean: {mean_val:.2f}')
plt.axvline(median_val, color='red', linestyle='--', label=f'Median: {median_val:.2f}')

plt.legend()
plt.show()
```

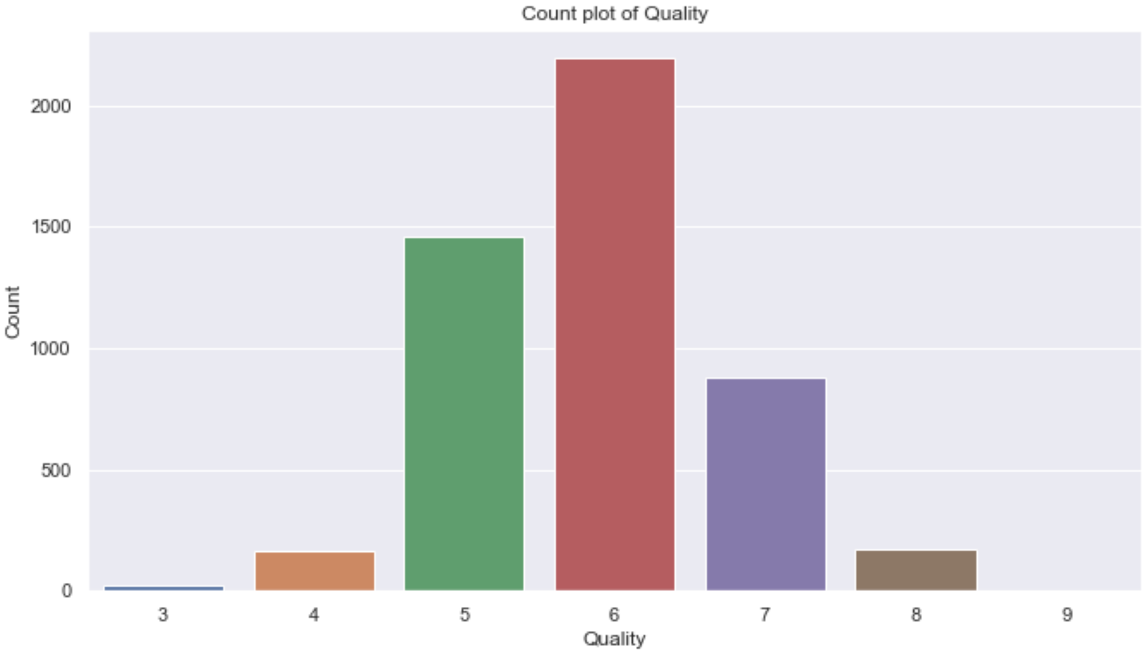


In [21]:

```
# Observation
# The mean and median are close
```

In [22]:

```
# Create count plot of the quality feature
plt.figure(figsize = (11,6))
sns.countplot(study['quality'])
plt.title("Count plot of Quality")
plt.xlabel('Quality')
plt.ylabel('Count')
plt.show()
```



In [23]:

```
# Observation
# It is clear from the count plot that 6 is the highest count of quality, w
here as 9 is negligible
```

In [25]:

```
# count the number of accurances of different categories of the quality
study['quality'].value_counts()
```

Out[25]:

```
6    2198
5    1457
7     880
8     175
4     163
3      20
9       5
Name: quality, dtype: int64
```

In [26]:

```
# mode
study['quality'].value_counts().index[0]
```

Out[26]:

```
6
```

In [27]:

```
#Observation of task 2
```

In [29]:

```
# Task 3
# We will now create a panda series

rep_acid = pd.Series(index = ['fixed acidity','volatile acidity','citric acid','quality'],
                     data = [study['fixed acidity'].mean(),
                             study['volatile acidity'].mean(),
                             study['citric acid'].mean(),
                             study['quality'].value_counts().index[0]])
rep_acid
```

Out[29]:

```
fixed acidity       6.854788
volatile acidity    0.278241
citric acid         0.334192
quality             6.000000
dtype: float64
```

In [30]:

```
# Observations
#The mean value of fixed acidity is 6.8,
#The mean value of volatile acidity is 0.2,
#The mean value of citric acid is 0.33,
# count of quality is 6
```

In [ ]:

```
#Final Conclusion
#From the given data, we can use simple visualization to get a sense of how data are distributed.
#We can use various measures of central tendency such as mean, median, mode to represent a group of observations.
# The type of central tendency measures to use depends on the type and distribution of the data.
```