Problem Statement: Analyzing Air Quality Index (AQI) Trends in a City
Tasks to Perform:

1. Import the "City_Air_Quality.csv" dataset.
2. Explore the dataset to understand its structure and content.
3. Identify the relevant variables for visualizing AQI trends, such as date, pollutant levels, and AQI values.
4. Create line plots or time series plots to visualize the overall AQI trend over time.
5. Plot individual pollutant levels (e.g., PM2.5, PM10, CO) on separate line plots to visualize their trends over time.
6. Use bar plots or stacked bar plots to compare the AQI values across different dates or time periods.
7. Create box plots or violin plots to analyze the distribution of AQI values for different pollutant categories.
8. Use scatter plots or bubble charts to explore the relationship between AQI values and pollutant levels.
9. Customize the visualizations by adding labels, titles, legends, and appropriate color schemes.

In [50]:
```
# 1. Import dataset.
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

df = pd.read_csv('City_Air_Quality_Extended.csv')

df.head()
```

Out[50]:

	date	time	PM2.5	PM10	CO	AQI
0	2023-01-01	08:00:00	52.483571	81.958453	0.493077	68.037513
1	2023-01-01	20:00:00	49.857307	70.833513	0.441528	110.957795
2	2023-01-02	08:00:00	54.335286	85.316747	0.495035	143.912860
3	2023-01-02	20:00:00	59.259382	64.824840	0.465702	77.871483
4	2023-01-03	08:00:00	51.019618	92.757905	0.481942	110.631344

In [51]:
```
# 2. Explore the dataset to understand its structure and content.
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 730 entries, 0 to 729
Data columns (total 6 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   date    730 non-null    object 
 1   time    730 non-null    object 
 2   PM2.5   730 non-null    float64
 3   PM10    730 non-null    float64
 4   CO      730 non-null    float64
 5   AQI     730 non-null    float64
dtypes: float64(4), object(2)
memory usage: 34.3+ KB
```

In [52]:
```
df.describe()
```

Out[52]:

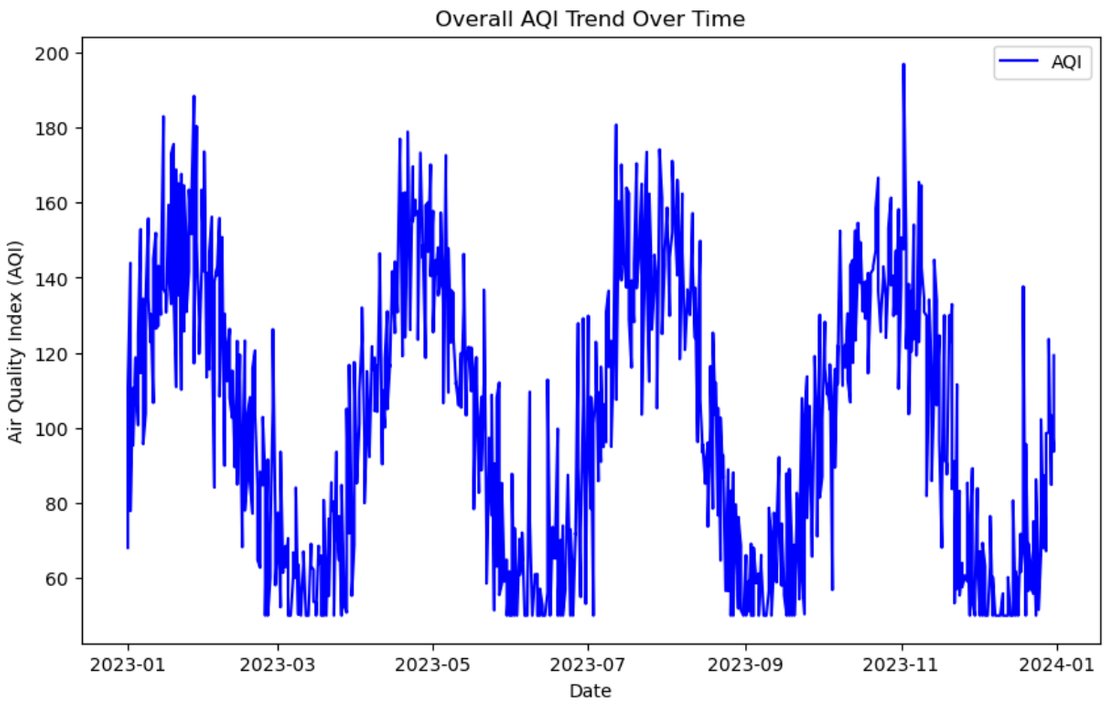
	PM2.5	PM10	CO	AQI
count	730.000000	730.000000	730.000000	730.000000
mean	50.535522	84.574138	0.537460	101.941368
std	14.523314	23.834302	0.142204	37.396928
min	20.000000	30.000000	0.195327	50.000000
25%	38.035410	64.827707	0.423133	65.992751
50%	51.773356	86.056276	0.564756	103.858187
75%	63.213634	104.905700	0.654935	131.963502
max	80.626134	133.917000	0.837871	196.924786

In [53]:
```
# 3. Identify the relevant variables for visualizing AQI trends, such as date, poll

print("Columns in the dataset:", df.columns)

df.isnull().sum()
```

```
Columns in the dataset: Index(['date', 'time', 'PM2.5', 'PM10', 'CO', 'AQI'], dtype
='object')
```

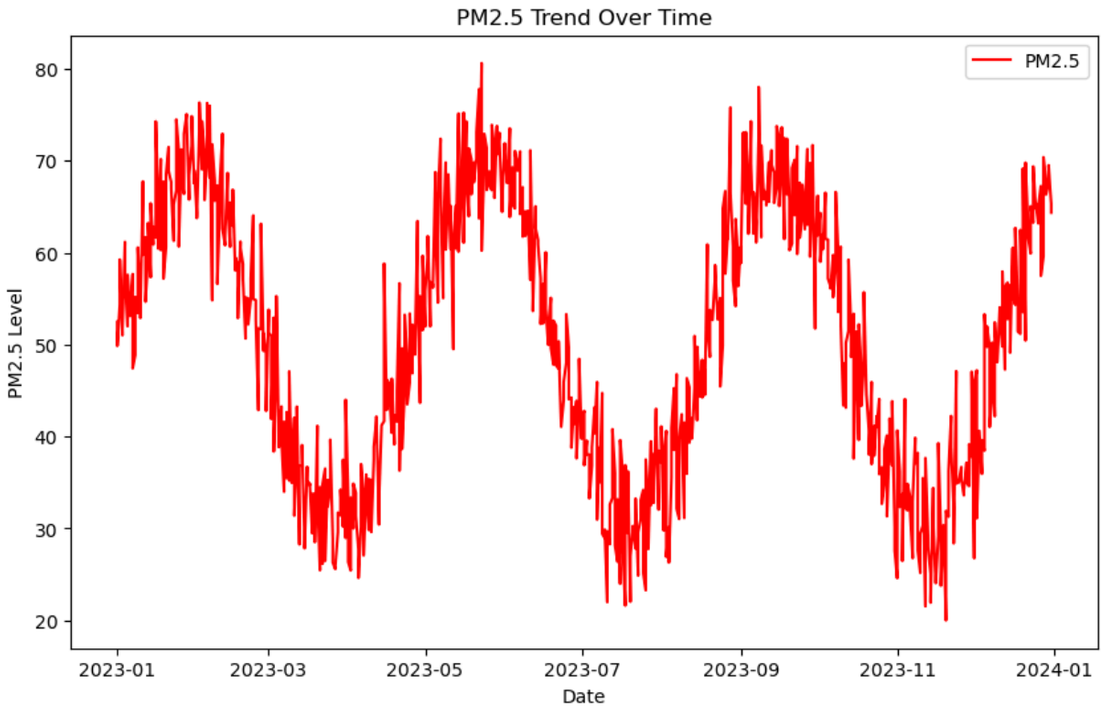
Out[53]:
```
date     0
time     0
PM2.5    0
PM10     0
CO       0
AQI      0
dtype: int64
```

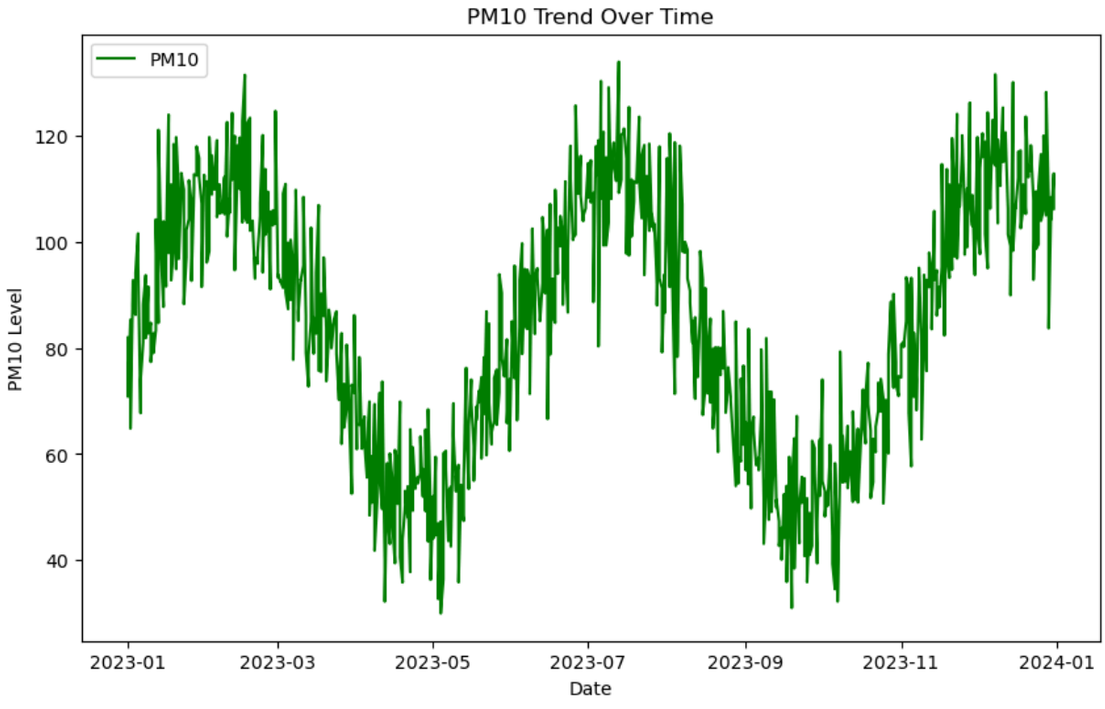
In [54]:
```
# 4. Create line plots or time series plots to visualize the overall AQI trend over

df['date'] = pd.to_datetime(df['date'])

plt.figure(figsize=(10, 6))
plt.plot(df['date'], df['AQI'], label='AQI', color='blue')
plt.xlabel('Date')
plt.ylabel('Air Quality Index (AQI)')
plt.title('Overall AQI Trend Over Time')
plt.legend()
plt.show()
```

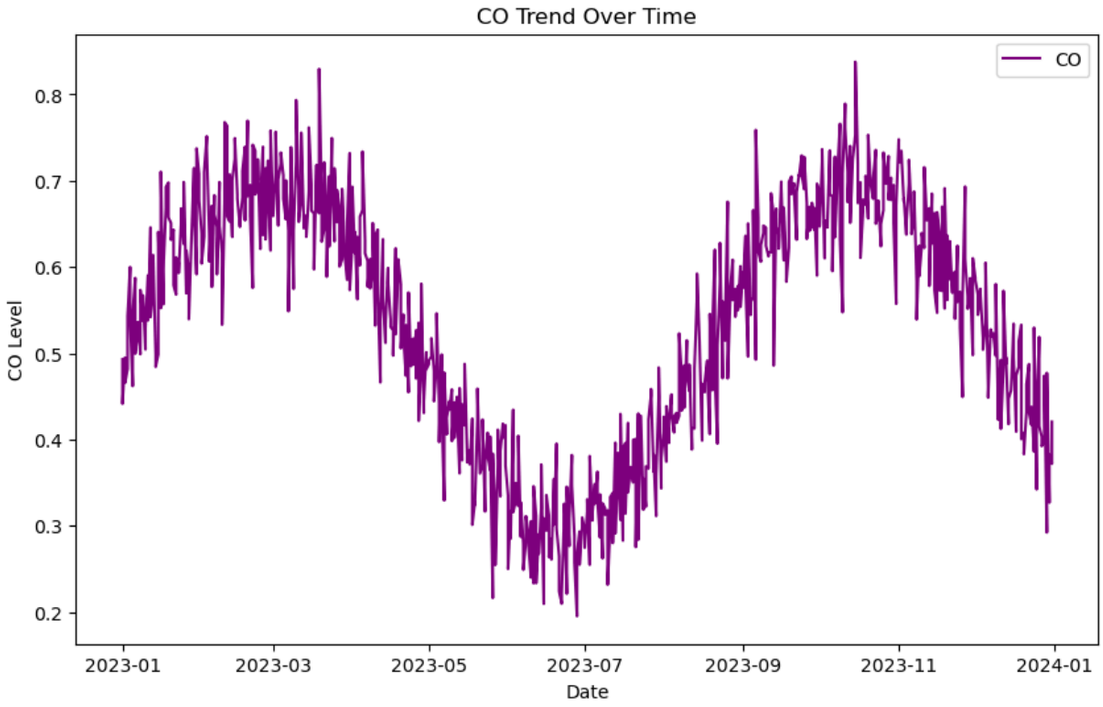


In [55]:
```
# 5. Plot individual pollutant levels (e.g., PM2.5, PM10, CO) on separate line plot

# PM2.5 Trend
```

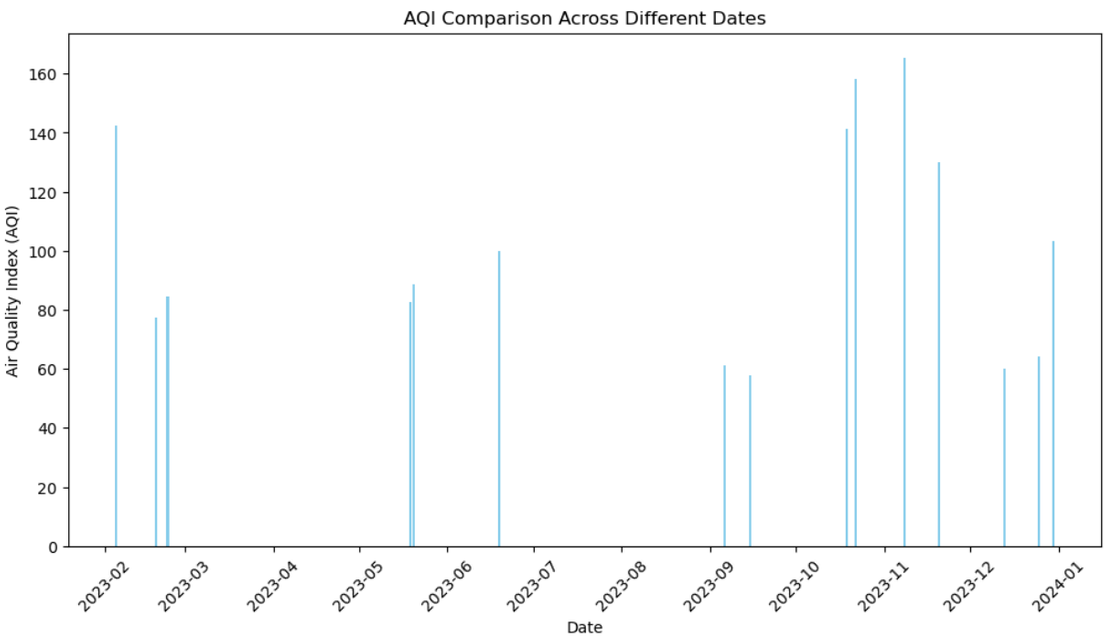
```
plt.figure(figsize=(10, 6))
plt.plot(df['date'], df['PM2.5'], label='PM2.5', color='red')
plt.xlabel('Date')
plt.ylabel('PM2.5 Level')
plt.title('PM2.5 Trend Over Time')
plt.legend()
plt.show()

# PM10 Trend
plt.figure(figsize=(10, 6))
plt.plot(df['date'], df['PM10'], label='PM10', color='green')
plt.xlabel('Date')
plt.ylabel('PM10 Level')
plt.title('PM10 Trend Over Time')
plt.legend()
plt.show()

# CO Trend
plt.figure(figsize=(10, 6))
plt.plot(df['date'], df['CO'], label='CO', color='purple')
plt.xlabel('Date')
plt.ylabel('CO Level')
plt.title('CO Trend Over Time')
plt.legend()
plt.show()
```



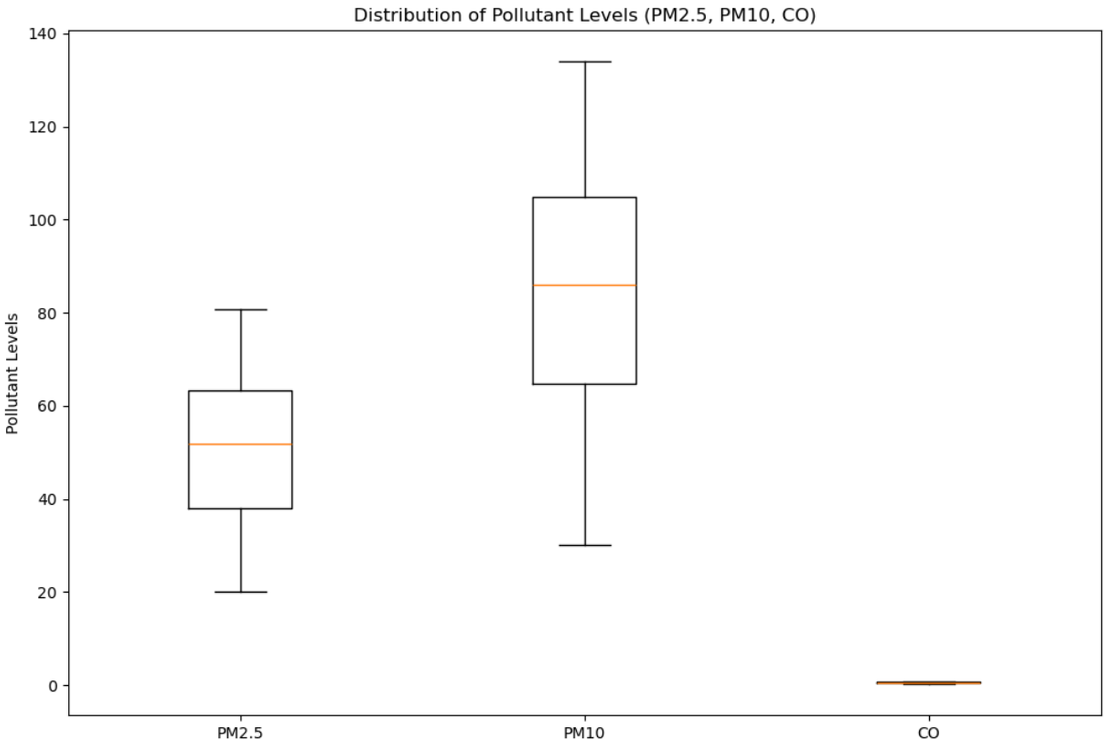


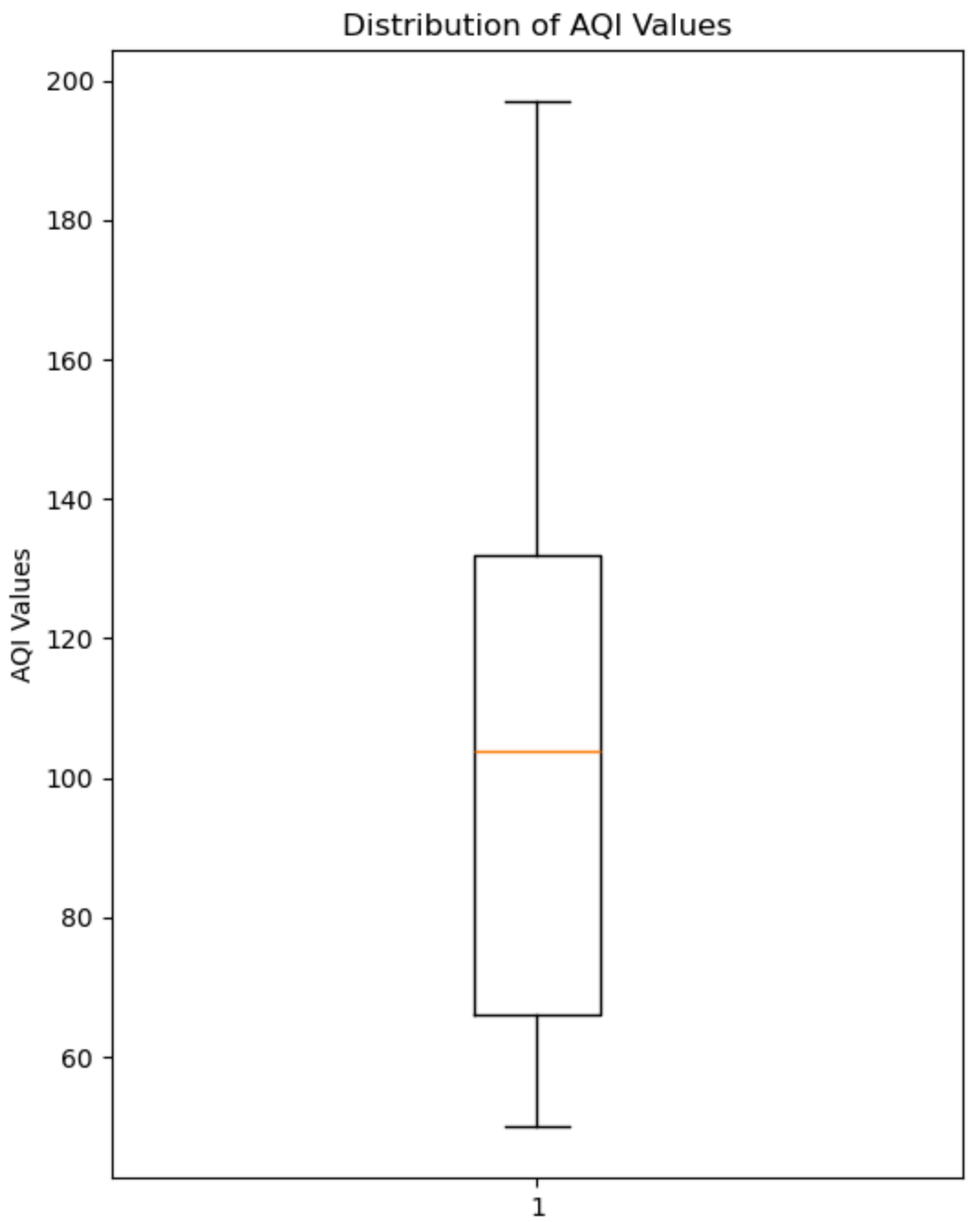


In [56]:
```
# 6. Use bar plots or stacked bar plots to compare the AQI values across different

sampled_df = df.sample(15)
plt.figure(figsize=(12, 6))
plt.bar(sampled_df['date'], sampled_df['AQI'], color='skyblue')
plt.xlabel('Date')
plt.ylabel('Air Quality Index (AQI)')
plt.title('AQI Comparison Across Different Dates')
plt.xticks(rotation=45)
plt.show()
```



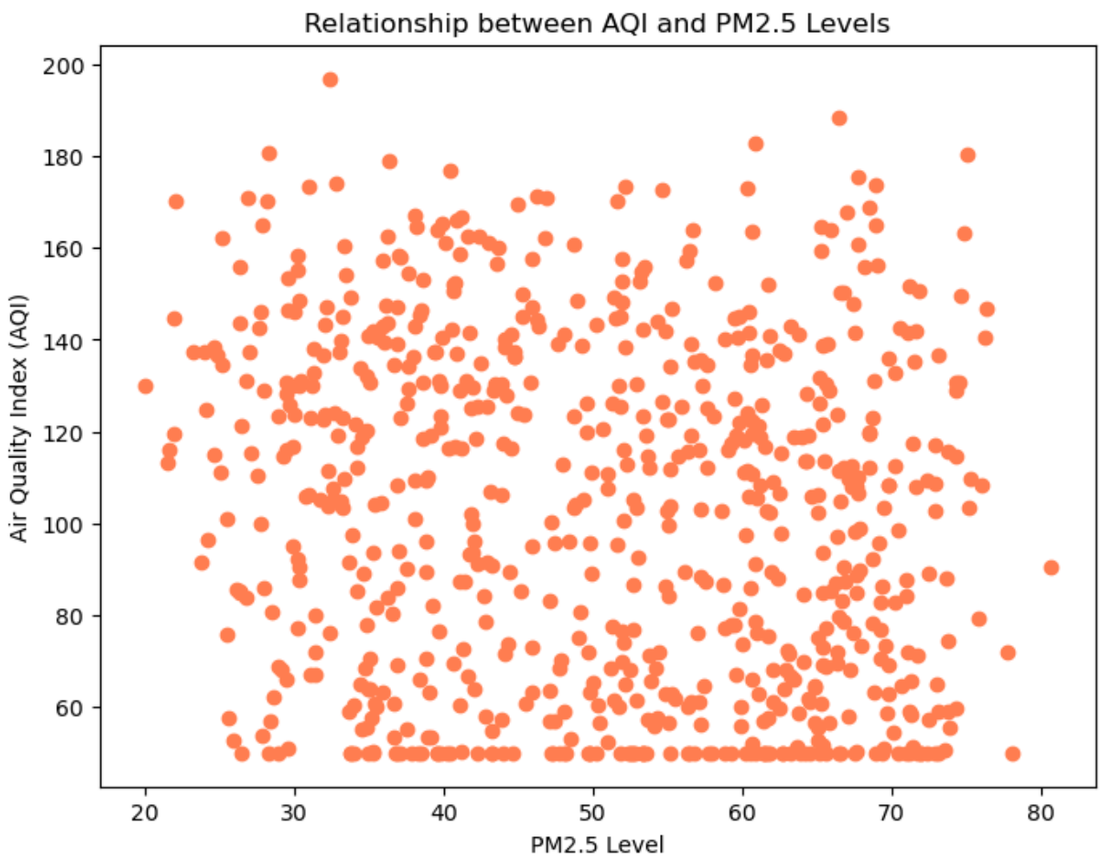
In [57]:
```
# 7. Create box plots or violin plots to analyze the distribution of AQI values for

# Box plot for each pollutant
plt.figure(figsize=(12, 8))
plt.boxplot([df['PM2.5'], df['PM10'], df['CO']], labels=['PM2.5', 'PM10', 'CO'])
plt.ylabel('Pollutant Levels')
plt.title('Distribution of Pollutant Levels (PM2.5, PM10, CO)')
plt.show()

# Box plot for AQI values
plt.figure(figsize=(6, 8))
plt.boxplot(df['AQI'])
plt.ylabel('AQI Values')
plt.title('Distribution of AQI Values')
plt.show()
```

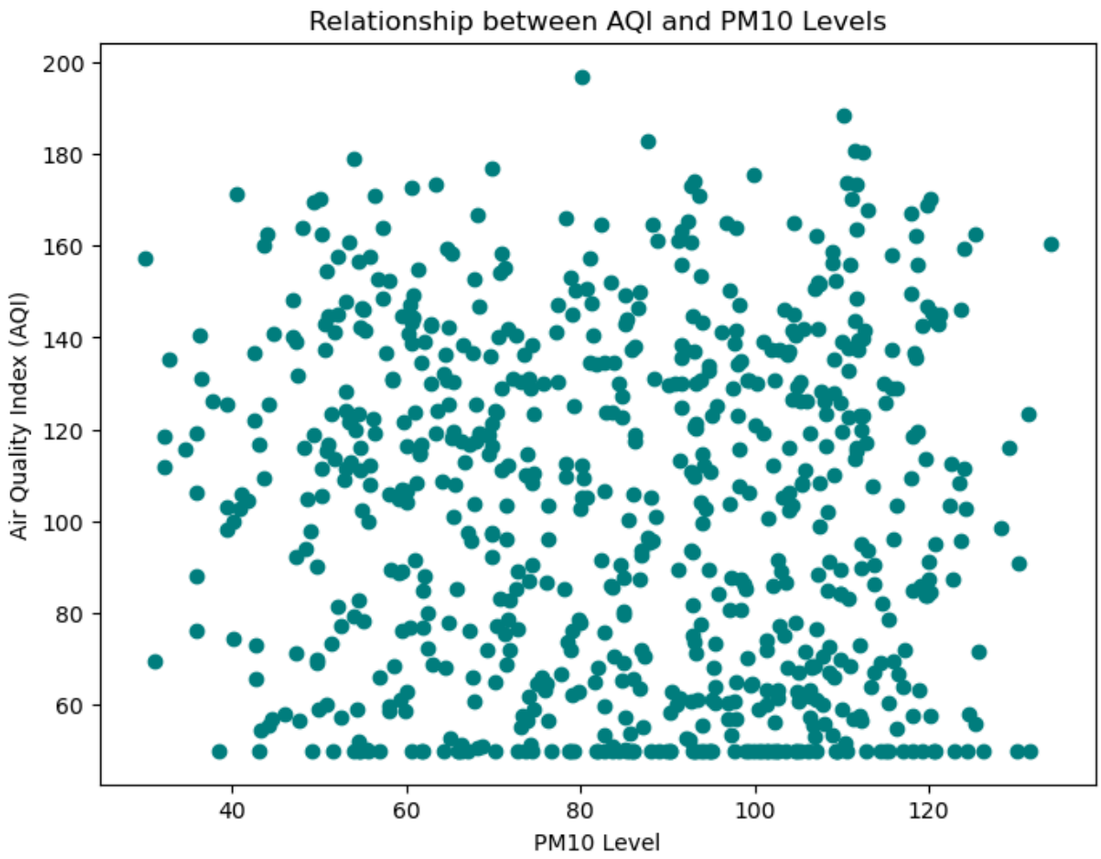


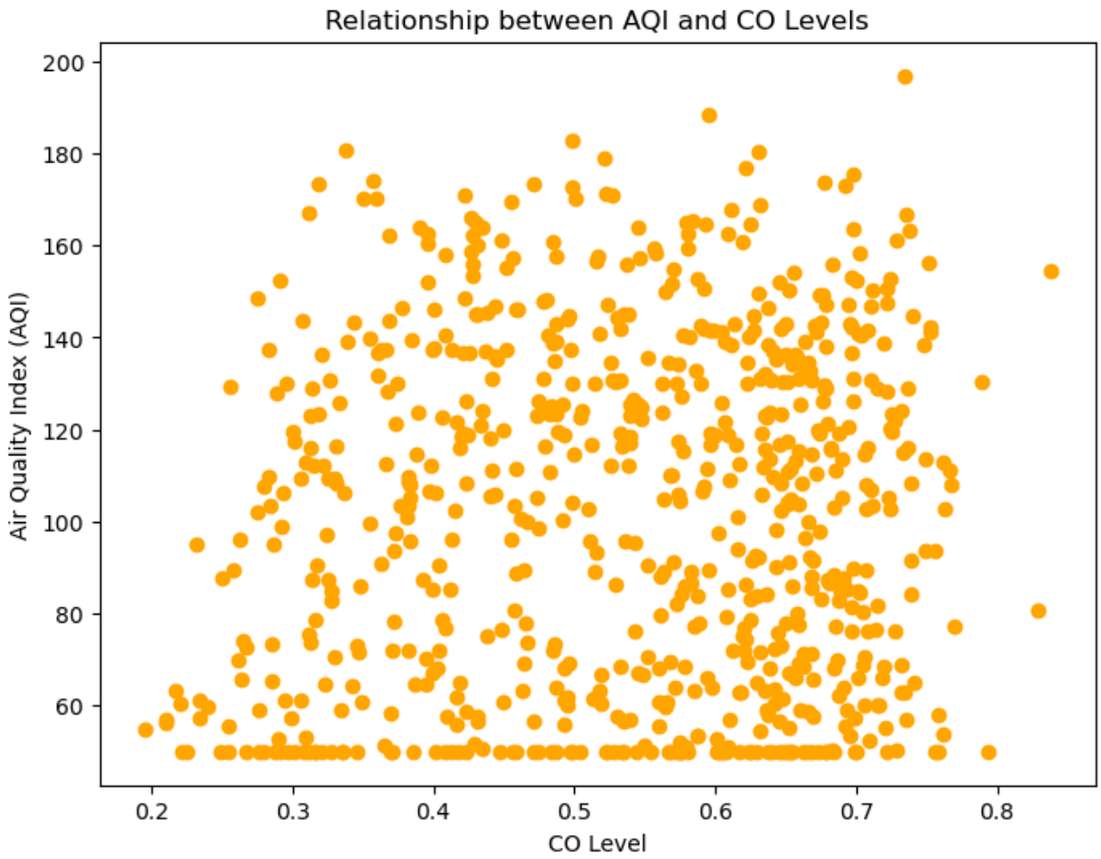


In [58]:
```
# 8. Use scatter plots or bubble charts to explore the relationship between AQI val

# Scatter plot: AQI vs PM2.5
plt.figure(figsize=(8, 6))
plt.scatter(df['PM2.5'], df['AQI'], color='coral')
plt.xlabel('PM2.5 Level')
plt.ylabel('Air Quality Index (AQI)')
```

```
plt.title('Relationship between AQI and PM2.5 Levels')
plt.show()

# Scatter plot: AQI vs PM10
plt.figure(figsize=(8, 6))
plt.scatter(df['PM10'], df['AQI'], color='teal')
plt.xlabel('PM10 Level')
plt.ylabel('Air Quality Index (AQI)')
plt.title('Relationship between AQI and PM10 Levels')
plt.show()

# Scatter plot: AQI vs CO
plt.figure(figsize=(8, 6))
plt.scatter(df['CO'], df['AQI'], color='orange')
plt.xlabel('CO Level')
plt.ylabel('Air Quality Index (AQI)')
plt.title('Relationship between AQI and CO Levels')
plt.show()
```

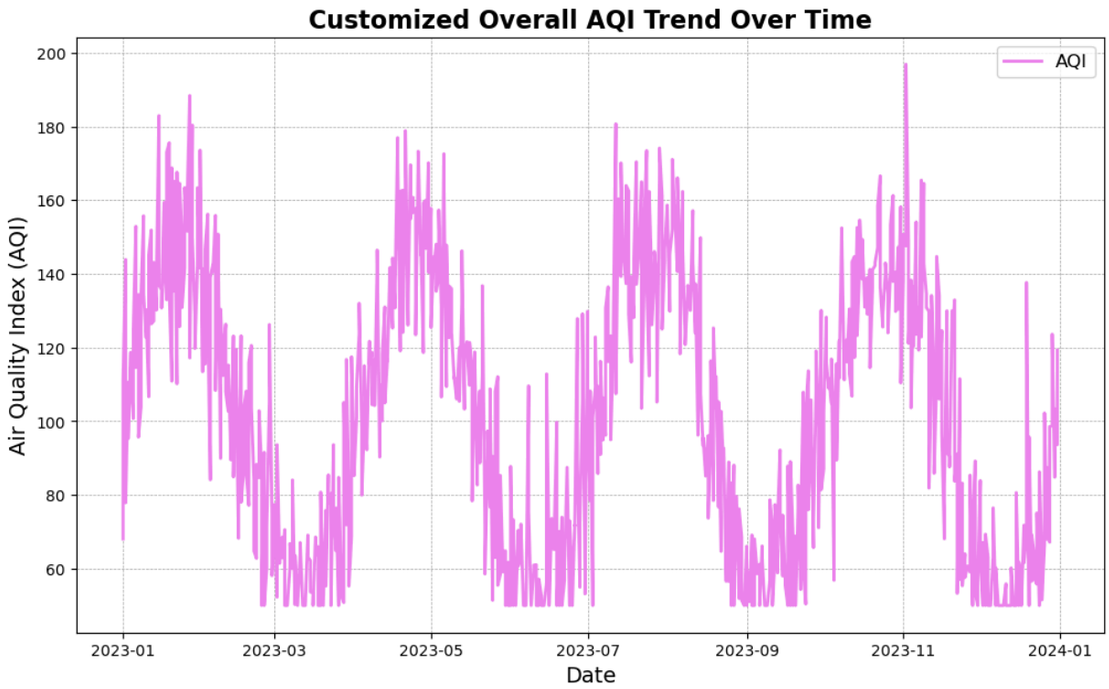


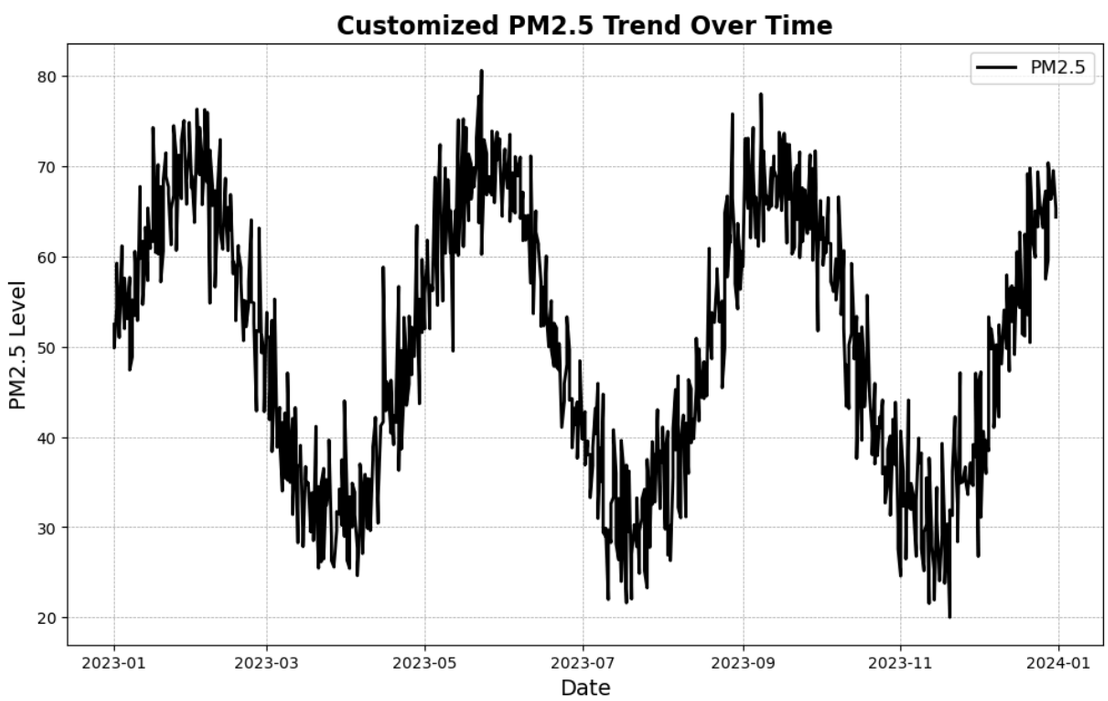




In [59]:
```
# 9. Customize the visualizations by adding labels, titles, legends, and appropriat
# schemes.
# AQI Trend with customized appearance
plt.figure(figsize=(12, 7))
plt.plot(df['date'], df['AQI'], label='AQI', color='violet', linewidth=2, linestyle
plt.xlabel('Date', fontsize=14)
plt.ylabel('Air Quality Index (AQI)', fontsize=14)
plt.title('Customized Overall AQI Trend Over Time', fontsize=16, fontweight='bold')
plt.legend(fontsize=12)
plt.grid(color='gray', linestyle='--', linewidth=0.5, alpha=0.7)
plt.show()

# PM2.5 Trend
plt.figure(figsize=(12, 7))
plt.plot(df['date'], df['PM2.5'], label='PM2.5', color='black', linewidth=2, linest
```

```
plt.xlabel('Date', fontsize=14)
plt.ylabel('PM2.5 Level', fontsize=14)
plt.title('Customized PM2.5 Trend Over Time', fontsize=16, fontweight='bold')
plt.legend(fontsize=12)
plt.grid(color='gray', linestyle='--', linewidth=0.5, alpha=0.7)
plt.show()
```





In [60]:
```
# Done
```