1. ***Data Load and Preprocessing: Lance.bae@icpc.foundation***
    a. Three datasets, representing ACT scores for the years 2019-2020, 2020-2021, and 2021-2022. three phases: Before COVID-19, during COVID-19, and after COVID-19. Performed data cleaning and non-numeric values and missing data
    b. The variables in the dataset included: District, District Name, Subgroup, Valid Tests, Participation Rate, Average English Score, Average Math Score, Average Reading Score, Average Science Score, Average Composite Score, Number Scoring 21 or Higher, Percent Scoring 21 or Higher, Number Scoring Below 19, and Percent Scoring Below 19.

2. ***Basic Statistics:*** calculated basic statistics such as mean, median, and standard deviation for 'Average Score', 'Participation Rates', 'Number Scoring 21 or Higher', and 'Number Scoring Below 19'.

e.g 2022 average composite score

```
In [54]: ##2022
         # Convert the 'Average Composite Score' column to numeric, forcing non-numeric values to NaN
         df_2022['Average Composite Score'] = pd.to_numeric(df_2022['Average Composite Score'], errors='coerce')

         # calculate the mean
         avg_composite_2022 = df_2022.groupby('District Name')['Average Composite Score'].mean()
         avg_composite_2022
```

```
Out[54]: District Name
         Achievement School District    13.325000
         Alcoa                          17.150000
         Alvin C York Institute         17.766667
         Anderson County                17.100000
         Arlington                      19.240000
                                           ...
         Weakley County                 17.300000
         West Carroll Sp Dist           15.933333
         White County                   17.525000
         Williamson County              20.540000
         Wilson County                  17.320000
         Name: Average Composite Score, Length: 130, dtype: float64
```

e.g Find districts with the most improved average composite scores from 2021 to 2022:

```
In [64]: #Find districts with most improved average composite scores from 2021 to 2022:
         top_districts2022_2021 = improvement2022_2021.nlargest(10)
         top_districts2022_2021
```

```
Out[64]: District Name
         Germantown           3.825000
         Fentress County      3.600000
         Williamson County    2.680000
         Gibson Co Sp Dist    2.285000
         Dyersburg            1.750000
         Cumberland County    1.685000
         Hawkins County       1.650000
         Lincoln County       1.625000
         Greeneville          1.541667
         Arlington            1.500000
         Name: Average Composite Score, dtype: float64
```

Participation difference

E.g Morgan County had the largest participation rate during the year 2020-2021 compared to other counties.

```
In [72]: participation_difference

Out[72]: District Name
         Achievement School District   -28.75
         Alcoa                           0.00
         Alvin C York Institute         -3.00
         Anderson County                 0.35
         Arlington                      -1.75
                                         ...
         Weakley County                 -3.50
         West Carroll Sp Dist            1.00
         White County                   -0.25
         Williamson County              -0.55
         Wilson County                   2.60
         Name: Participation Rate, Length: 133, dtype: float64

In [74]: top_districts2020_2021 = participation_difference.nlargest(10)
         top_districts2020_2021

Out[74]: District Name
         Morgan County        12.000000
         Robertson County      7.450000
         Madison County        7.200000
         Stewart County        5.250000
         Richard City          4.750000
         Germantown            4.550000
         Maury County          4.200000
         Montgomery County     3.900000
         Cocke County          3.500000
         Grundy County         3.333333
         Name: Participation Rate, dtype: float64
```

3. Correlation Analysis: The relationship between 'Participation Rates' and 'Average Scores'. It was found that there was a positive correlation, indicating that as participation rates increased, the average score also tended to increase.

```
#  calculate the correlation
correlation = df_2022['Participation Rate'].corr(df_2022['Average Composite Score'])

print(f"The correlation between participation rates and average scores in 2022 is: {correlation}")
```

```
The correlation between participation rates and average scores in 2022 is: 0.3393436065686978
```

```
In [76]: #2021
df_2021['Participation Rate'] = pd.to_numeric(df_2021['Participation Rate'], errors='coerce')
df_2021['Average Composite Score'] = pd.to_numeric(df_2021['Average Composite Score'], errors='coerce')

# calculate the correlation
correlation = df_2021['Participation Rate'].corr(df_2021['Average Composite Score'])

print(f"The correlation between participation rates and average scores in 2021 is: {correlation}")
```

```
The correlation between participation rates and average scores in 2021 is: 0.13410406077603307
```

```
In [77]: #2020
df_2020['Participation Rate'] = pd.to_numeric(df_2020['Participation Rate'], errors='coerce')
df_2020['Average Composite Score'] = pd.to_numeric(df_2020['Average Composite Score'], errors='coerce')

# calculate the correlation
correlation = df_2020['Participation Rate'].corr(df_2020['Average Composite Score'])

print(f"The correlation between participation rates and average scores in 2020 is: {correlation}")
```

```
The correlation between participation rates and average scores in 2020 is: 0.24475107046851555
```

Notably, the correlation was lowest in the year 2020-2021, during COVID-19, indicating that during this time, the participation rate had a weaker relationship with the scores.

Performance Analysis: analyzed the performance of students based on 'Number Scoring 21 or Higher' and 'Number Scoring Below 19'.calculated descriptive statistics and visualized the score distribution.
E.g 2022

```
# Histogram for Number Scoring 21 or Higher
plt.figure(figsize=(10,6))
plt.hist(df_2022['Number Scoring 21 or Higher'].dropna(), bins=30, alpha=0.5, label='Number Scoring 21 or Higher')
plt.hist(df_2022['Number Scoring Below 19'].dropna(), bins=30, alpha=0.5, label='Number Scoring Below 19')
plt.legend(loc='upper right')
plt.title('Distribution of Scores')
plt.xlabel('Score')
plt.ylabel('Frequency')
plt.show()
```