Task 2: Data visualization

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load data
data = pd.read_csv('student_performance.csv')

# Visualization 1: Scatter plot
sns.scatterplot(data=data, x='study_time', y='final_grade')
plt.title('Final grade vs. study time')
plt.show()

# Visualization 2: Histogram
sns.histplot(data=data, x='final_grade')
plt.title('Distribution of final grades')
plt.show()

# Visualization 3: Bar chart
sns.barplot(data=data, x='gender', y='final_grade')
plt.title('Average final grade by gender')
plt.show()

# Visualization 4: Box plot
```

```
sns.boxplot(data=data, y='final_grade')
plt.title('Distribution of final grades')
plt.show()

# Visualization 5: Line plot
sns.boxplot(data=data, y='final_grade')
plt.title('Distribution of final grades')
plt.show()
```

Explanation of what analysis has become easier with each of the visualizations:

Scatter plot: The scatter plot allows us to see if there is any relationship between study time and final grade. We can use this visualization to identify any patterns or trends in the data.

Histogram: The histogram helps us understand the distribution of final grades. We can use this visualization to see if the grades are normally distributed, or if there are any outliers or skewedness in the data.

Bar chart: The bar chart shows us the average final grade by gender, allowing us to compare the performance of male and female students. We can use this visualization to identify any gender-based differences in performance.

Box plot: The box plot gives us a visual representation of the distribution of final grades. We can use this visualization to identify the median, quartiles, and outliers in the data.

Line plot: The line plot shows us how the final grade changes over time, by gender. We can use this visualization to identify any differences in performance between male and female students over time.

To create the folder structure for the first task, you can follow the following steps:

Create a new folder called project_name.

Inside the project_name folder, create a new folder called data.

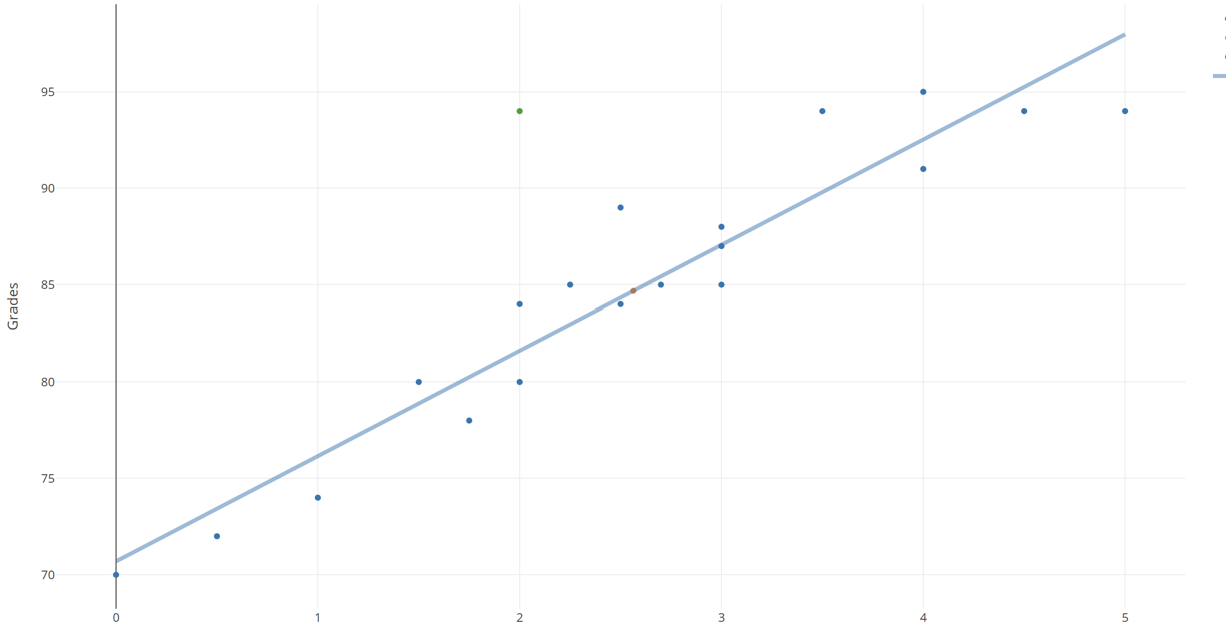
Inside the data folder, create three subfolders: raw_data, processed_data, and final_data.

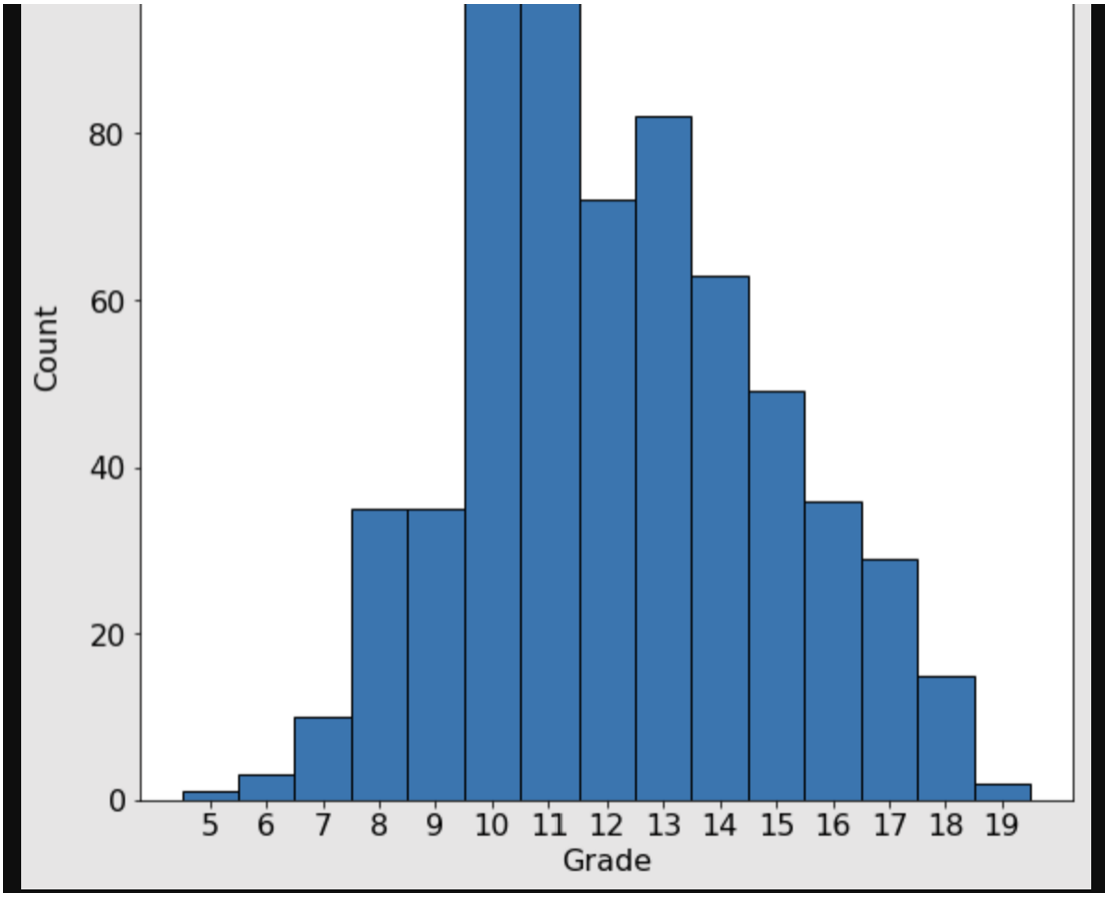
Inside the raw_data folder, place the raw data file (my_data.csv).

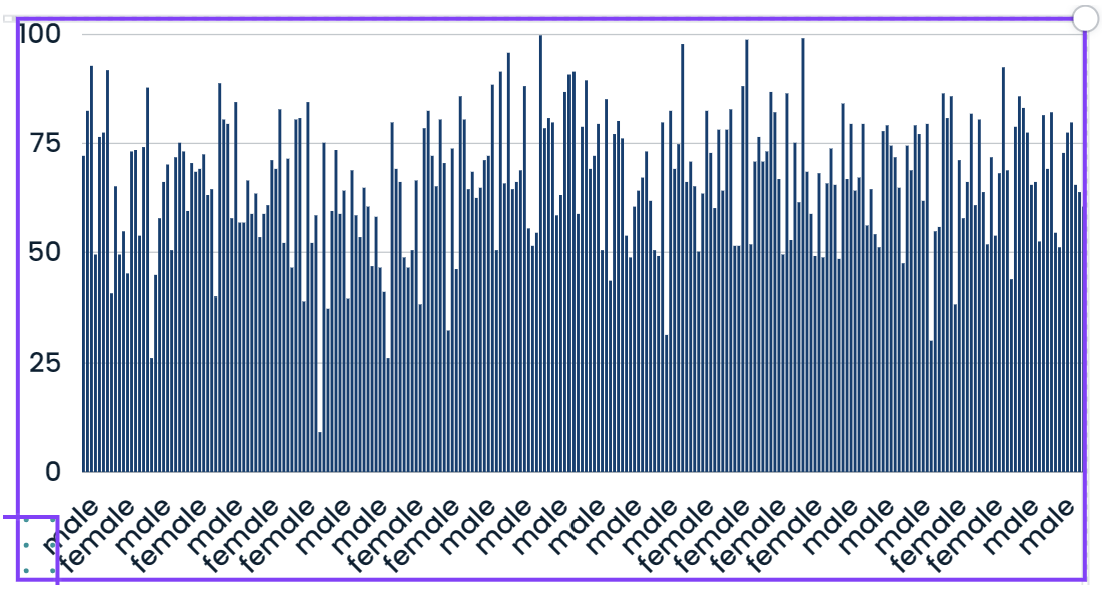
Inside the processed_data folder, create a new folder for each processing step. For example, if you preprocess the data using Python scripts, you can create a folder called preprocessing_python.

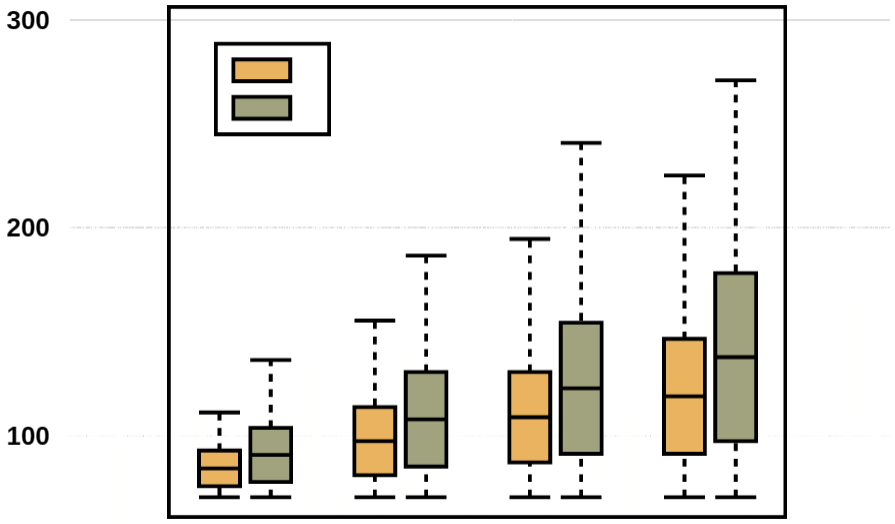
Inside each processing step folder, create a subfolder called code, where you can store all the scripts used for that step.

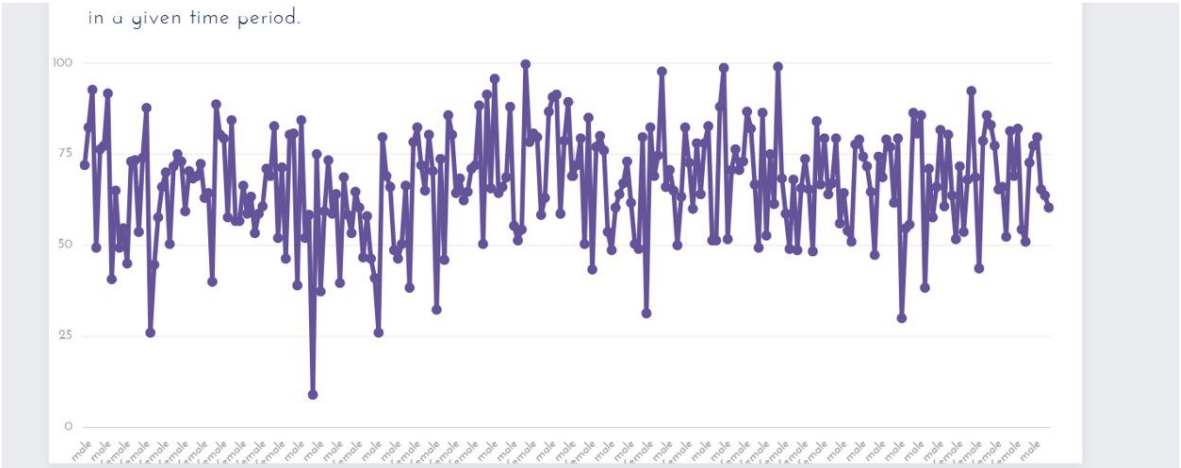
Inside each processing step folder, create a subfolder called output, where you can store the output generated by the scripts.


Grades
95
90
85
80
75
70
0
1
2
3
4
5


Count
80
60
40
20
0
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
Grade


100
75
50
25
0
male
female


300
200
100


in a given time period.
100
75
50
25
0
male
female