Portfolio Assignment 3

ENDG 233 Fall 2021

Department of Electrical and Software Engineering
Schulich School of Engineering
Due: November 26, 2021 at 11:59 PM

The purpose of these assignments is to build a portfolio of successful programs that could be used to demonstrate your understanding of coding and Python. This assignment is worth 10% of your final grade.

Portfolio Assignment #3 – Learning Outcomes

- Create and manipulate arrays using the numpy module
- Accept and validate user input
- Process data according to specifications
- Use dictionaries to store and search for values
- Manipulate and execute numpy array computations
- Plot data using the matplotlib module
- Create an object instance using a given class
- Use instance methods for a given class

Program Specifications

The City of Calgary publishes data on various demographics, include school enrollment. You are being asked to design a terminal-based application for computing, printing, and plotting statistical information based on given input. Enrollment data for a subset of public high schools is provided for your use via three csv files. An Excel file is also provided to give you the names and reference codes of each school.

Your application must meet the following design specifications:

Stage 1: Data Importing

- Create three separate numpy arrays using the provided high school enrollment data.
- Print the data of each array to the terminal.

Stage 2: User Input and School Stats

- Prompt the user to enter either the name or numerical code of a school. If the name or code do not exist within the given data, print “You must enter a valid school name or code.” and allow the user to retry their input.
- Use the given `School` class to print the school name and school code
- Use the numpy arrays to calculate and print the answers for the following school-specific statistics, using the given output as a template.
    - Mean enrollment for Grade 10 across all years (rounded down to the nearest integer)
    - Mean enrollment for Grade 11 across all years (rounded down to the nearest integer)

- Mean enrollment for Grade 12 across all years (rounded down to the nearest integer)
- Total number of students who graduated from 2019 – 2021 (provided in integer format)

Stage 3: Data Plotting

- Use matplotlib to create a plot of grade enrollment by year. Your formatting should match the format of the given example.
  - 2021 enrollment should be plotted using blue dots
  - 2020 enrollment should be plotted using green dots
  - 2019 enrollment should be plotted using red dots
  - y-axis should be the number of students (range will depend on the school)
  - x-axis should be only the three grade levels (hint: take a look at the plt.xticks method)
  - Title of the plot is “Grade Enrollment by Year”
  - Plot must include a legend

Bonus: Data Plotting

- You may attempt an optional bonus plotting task.
- Use matplotlib to create subplots of each grade’s enrollment by year. Your formatting should match the format of the given example.
  - Grade 10 enrollment should be plotted at the top using a yellow dotted line
  - Grade 11 enrollment should be plotted in the middle using a magenta dotted line
  - Grade 12 enrollment should be plotted at the bottom using a cyan dotted line
  - y-axis should be the number of students (range will depend on the school)
  - x-axis should be only the three given years (hint: take a look at the plt.xticks method)
  - Title of the plot is “Enrollment by Grade”
  - Each subplot must include a legend

Additional Specifications

- You may not modify the given code. Add your code in the designated areas within the given starter file.
- Your code should include and use at least three numpy arrays
- Your code should include and use at least one dictionary
- Your code must follow the conventions discussed so far in the course (names_with_underscores, ClassNames, four spaces for indentations, spaces between variables/operators, comments throughout, etc.)
- Any user-defined functions must contain docstring documentation. Main functions do not need a docstring but should be well-commented.
- Any user-defined functions may not access global variables.
- Your final output should match the given examples as closely as possible (see included screenshots)
- You may only use built-in Python functions that support compound data structures, user entry, or casting (such as len(), input() or int()). The only modules you may import are numpy, matplotlib, and math.
- Your code will be run by the TAs as your end user using VS Code and a minimum of Python 3.9.

- FAQs about the assignment will be answered on the D2L discussion boards. Please check the boards for any clarifications before submitting.
- The grading rubric is also included in this handout.

Assignment Tasks

- Make sure to watch the video lessons for Weeks 1 – 11 and review the corresponding active learning content.
- You do not need to submit a flowchart, but it is highly recommended that you start with a flowchart to plan your logic.
- Open VSCode and start a new terminal. Make sure that your virtual environment is activated. If your module imports cannot be resolved, check that your virtual environment is selected as the Python interpreter.
- `school_data.py` is provided as a starting point. Fill in the header with your own information and write your program in this file. There are hints provided in the comments. Do not modify the given code.
- Remember to test your program execution via the terminal.
    - For example, use `python school_data.py` or the corresponding command for your operating system.
- Take a screenshot of your successful program execution that shows:
    a) Your virtual environment is active when running the program
    b) Your program successfully computes the required functionality (see provided output examples)
- Submit the following items to the Assignment 3 D2L dropbox:
    - Your final 'school_data.py' file with your name in the comment header (do not change the file name)
    - Your execution screenshot as a PNG file (e.g. `00121343_assign3_output.png`)

Assignment 3 Rubric (36 marks + optional bonus 4 marks, 10% of overall grade)
Your code must successfully run to be given full marks. Code that does not execute may be given partial marks for some criterion listed below.

Commenting and Syntax (5 marks):

- (1) Your name must be included in the file header
- (2) Comments must be included throughout the code to explain the functionality. Any user-defined functions are fully documented using docstrings (including summary, parameters, and return values)
- (1) All variables and functions have clear and useful names that use lowercase words separated by an underscore, all classes have CamelCase names
- (1) Code is clearly indented and spaces are included between variables and operators
- One mark will be deducted for each error or missing component, up to a maximum of 5 marks

Code Structure and Semantics (5 marks):

- (3) Solution contains and uses at least three numpy arrays
- (2) Solution contains and uses at least one dictionary
- One mark will be deducted for each error or missing component, up to a maximum of 5 marks

User Interface and Functionality (12 marks):

- (1) User is given clear guidance on how to enter the input values
- (2) Program allows the user to search for a school via name or school code
- (1) Program checks that the provide input is valid and notifies user for re-entry if invalid
- (1) Program runs continuously until the user provides valid input
- (1) The program prints the data contents of each numpy array
- (2) The `School` class is used to print the name and code of the input school
- (1) All statistical output is rounded down to the nearest integer
- (3) Plot formatting matches the given example
- One mark will be deducted for each error or missing component, up to a maximum of 12 marks

Execution and Calculations (14 marks):

- Example test cases are provided in the screenshot below, but you may provide your own in your screenshots
- (5) Screenshot of successful execution is shown
    - Your screenshots should include all successful functionality
        - 1) printing the numpy array data
        - 2) returning the correct results when a school name is input (including a plot)
        - 3) returning the correct results when a school code is input (including a plot)
        - 4) handling incorrect input
    - Your screenshot must also show that your virtual environment is active
    - You only need to show your printing of the array data once in the screenshots even though it should print with each run of your program
    - You may provide your plots as screenshots or saved in PNG format, but they must be clearly readable
- (9) Your program will be executed to test the following cases:
    - Entering a school name that is included in the provided dataset (4 marks)

        - Accuracy of school name, school code, grade 10 mean enrollment, grade 11 mean enrollment, grade 12 mean enrollment, total graduation, data accuracy of plot, formatting accuracy of plot
    - Entering a school code that is included in the provided dataset (4 marks)
        - Accuracy of school name, school code, grade 10 mean enrollment, grade 11 mean enrollment, grade 12 mean enrollment, total graduation, data accuracy of plot, formatting accuracy of plot
    - Entering an invalid input that is not included in the provided data (1 mark)
    - All students will have their code tested with the same input values
- 0.5 – 1 mark(s) will be deducted for each error or missing component, up to a maximum of 14 marks

Optional Bonus (4 marks):

- To receive the bonus marks, you must do the following
- The above screenshots should include successful execution of the bonus plot for both the school code and school name execution (1 mark)
- Your bonus plot will be graded for accuracy in both the data values and the formatting for both the school code and school name tests (3 marks, 1 for each subplot)
- One mark will be deducted for each error or missing component, up to a maximum of 4 marks

Assignment 3 Example Output

Your terminal input/output and plots should match the given formatting as closely as possible.

Test Run #1: Invalid entry, then valid school name provided.

Terminal:

```
(venv) C:\Users\eamarasc\Dropbox\Teaching\ENDG 233\Fall 2021\Portfolio\Assignment 3>python school_data_key.py
ENDG 233 School Enrollment Statistics

Array data for 2020 - 2021:
 [[1224.  455.  436.  437.]
 [1679.  360.  352.  437.]
 [9626.   12.   21.   34.]
 [9806.  137.  143.  142.]
 [9813.  378.  361.  438.]
 [9815.  565.  555.  629.]
 [9816.  690.  727.  734.]
 [9823.  543.  401.  459.]
 [9825.  290.  234.  315.]
 [9826.  491.  662.  680.]
 [9829.  420.  465.  434.]
 [9830.   34.   59.   64.]
 [9836.  532.  449.  431.]
 [9847.  402.  382.  364.]
 [9850.  482.  251.  293.]
 [9856.  128.  102.  144.]
 [9857.  739.  746.  709.]
 [9858.  540.  463.  503.]
 [9860.  452.  482.  475.]
 [9865.  537.  533.  559.]]
Array data for 2019 - 2020:
 [[1224.  463.  481.  556.]
 [1679.  355.  430.  455.]
 [9626.   13.   23.   52.]
 [9806.  146.  146.  127.]
 [9813.  383.  397.  441.]
 [9815.  556.  615.  530.]
 [9816.  723.  724.  798.]
 [9823.  391.  391.  409.]
 [9825.  251.  234.  322.]
 [9826.  674.  692.  629.]
 [9829.  476.  447.  507.]
 [9830.   61.   56.   69.]
 [9836.  458.  434.  424.]
 [9847.  391.  381.  254.]
 [9850.  241.  245.  299.]
 [9856.  102.   77.  146.]
 [9857.  749.  677.  744.]
 [9858.  488.  514.  503.]
 [9860.  474.  449.  531.]
 [9865.  535.  528.  553.]]
```

```
 Array data for 2018 - 2019:
  [[1224.  485.  540.  582.]
  [1679.  398.  423.  391.]
  [9626.   23.   25.   58.]
  [9806.  150.   83.  120.]
  [9813.  395.  402.  527.]
  [9815.  568.  426.  510.]
  [9816.  716.  688.  651.]
  [9823.  387.  388.  419.]
  [9825.  242.  204.  254.]
  [9826.  685.  576.  535.]
  [9829.  437.  473.  428.]
  [9830.   38.   45.   57.]
  [9836.  417.  391.  398.]
  [9847.  386.  238.  249.]
  [9850.  229.  250.  495.]
  [9856.   81.  112.  135.]
  [9857.  703.  705.  744.]
  [9858.  533.  496.  580.]
  [9860.  443.  438.  460.]
  [9865.  551.  456.  473.]]
 Please enter the high school name or school code: My Invalid School
 You must enter a valid school name or code.
 Please enter the high school name or school code: Centennial High School
```

```
Please enter the high school name or school code: My Invalid School
You must enter a valid school name or code.
Please enter the high school name or school code: Centennial High School

***Requested School Statistics***

School Name: Centennial High School, School Code: 1224
Mean enrollment for Grade 10:  467
Mean enrollment for Grade 11:  485
Mean enrollment for Grade 12:  525
Total number of students who graduated in the past three years:  1575

(venv) C:\Users\eamarasc\Dropbox\Teaching\ENDG 233\Fall 2021\Portfolio\Assignment 3>
```

Figure 1 (Required)

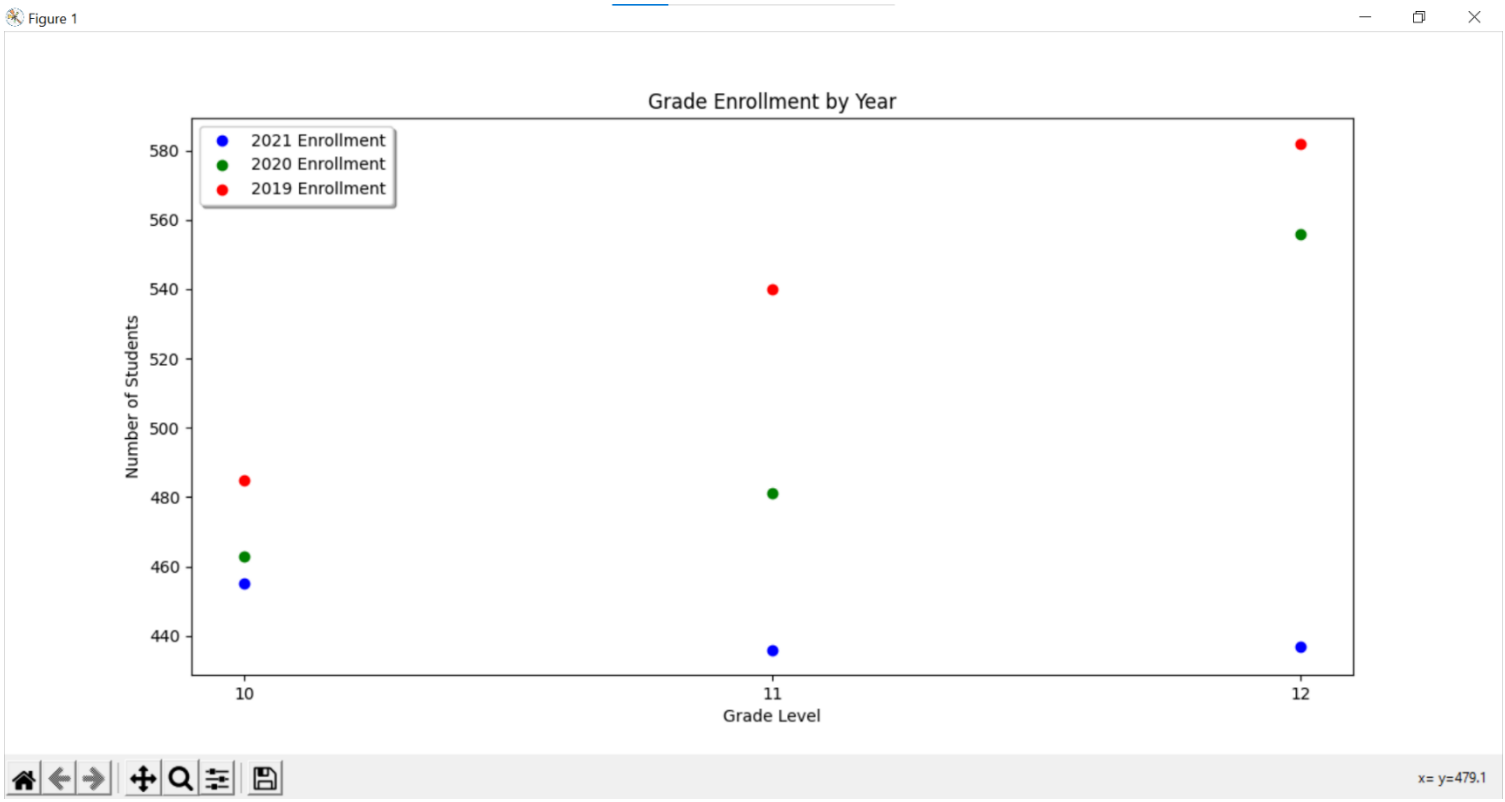
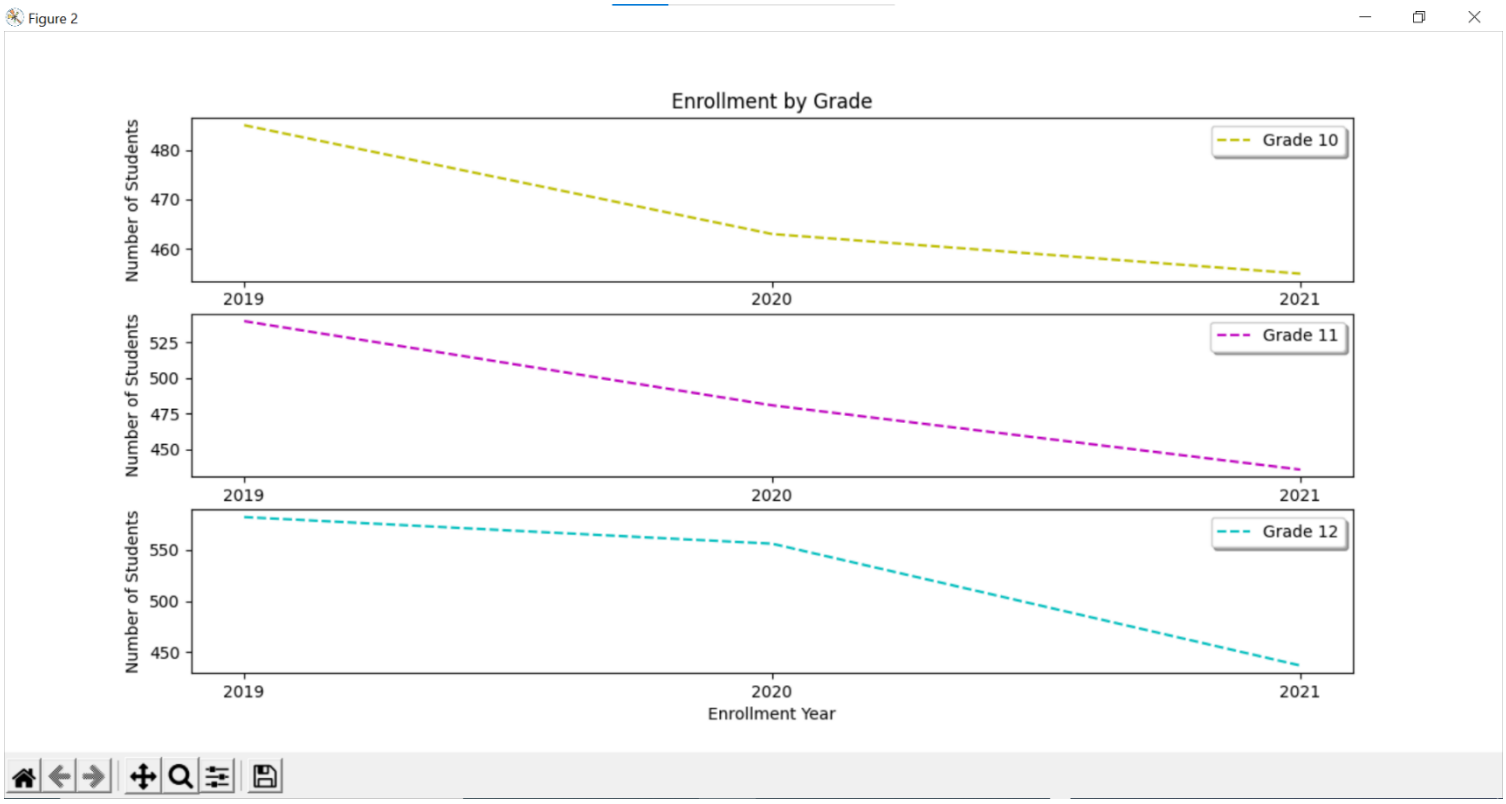


Figure 2 (Bonus)



Test Run #2: Invalid entry, then valid school code provided.

Terminal:

```
(venv) C:\Users\eamarasc\Dropbox\Teaching\ENDG 233\Fall 2021\Portfolio\Assignment 3>python school_data_key.py
ENDG 233 School Enrollment Statistics

Array data for 2020 - 2021:
 [[1224.  455.  436.  437.]
 [1679.  360.  352.  437.]
 [9626.   12.   21.   34.]
 [9806.  137.  143.  142.]
 [9813.  378.  361.  438.]
 [9815.  565.  555.  629.]
 [9816.  690.  727.  734.]
 [9823.  543.  401.  459.]
 [9825.  290.  234.  315.]
 [9826.  491.  662.  680.]
 [9829.  420.  465.  434.]
 [9830.   34.   59.   64.]
 [9836.  532.  449.  431.]
 [9847.  402.  382.  364.]
 [9850.  482.  251.  293.]
 [9856.  128.  102.  144.]
 [9857.  739.  746.  709.]
 [9858.  540.  463.  503.]
 [9860.  452.  482.  475.]
 [9865.  537.  533.  559.]]
```

```
 Array data for 2019 - 2020:
  [[1224.  463.  481.  556.]
  [1679.  355.  430.  455.]
  [9626.   13.   23.   52.]
  [9806.  146.  146.  127.]
  [9813.  383.  397.  441.]
  [9815.  556.  615.  530.]
  [9816.  723.  724.  798.]
  [9823.  391.  391.  409.]
  [9825.  251.  234.  322.]
  [9826.  674.  692.  629.]
  [9829.  476.  447.  507.]
  [9830.   61.   56.   69.]
  [9836.  458.  434.  424.]
  [9847.  391.  381.  254.]
  [9850.  241.  245.  299.]
  [9856.  102.   77.  146.]
  [9857.  749.  677.  744.]
  [9858.  488.  514.  503.]
  [9860.  474.  449.  531.]
  [9865.  535.  528.  553.]]
 Array data for 2018 - 2019:
  [[1224.  485.  540.  582.]
```

```
Array data for 2018 - 2019:
 [[1224.  485.  540.  582.]
 [1679.  398.  423.  391.]
 [9626.   23.   25.   58.]
 [9806.  150.   83.  120.]
 [9813.  395.  402.  527.]
 [9815.  568.  426.  510.]
 [9816.  716.  688.  651.]
 [9823.  387.  388.  419.]
 [9825.  242.  204.  254.]
 [9826.  685.  576.  535.]
 [9829.  437.  473.  428.]
 [9830.   38.   45.   57.]
 [9836.  417.  391.  398.]
 [9847.  386.  238.  249.]
 [9850.  229.  250.  495.]
 [9856.   81.  112.  135.]
 [9857.  703.  705.  744.]
 [9858.  533.  496.  580.]
 [9860.  443.  438.  460.]
 [9865.  551.  456.  473.]]
Please enter the high school name or school code: 0012
You must enter a valid school name or code.
Please enter the high school name or school code: 1679
```

```
Please enter the high school name or school code: 0012
You must enter a valid school name or code.
Please enter the high school name or school code: 1679

***Requested School Statistics***

School Name: Robert Thirsk School, School Code: 1679
Mean enrollment for Grade 10:  371
Mean enrollment for Grade 11:  401
Mean enrollment for Grade 12:  427
Total number of students who graduated in the past three years:  1283

(venv) C:\Users\eamarasc\Dropbox\Teaching\ENDG 233\Fall 2021\Portfolio\Assignment 3>
```

Figure 1 (Required)

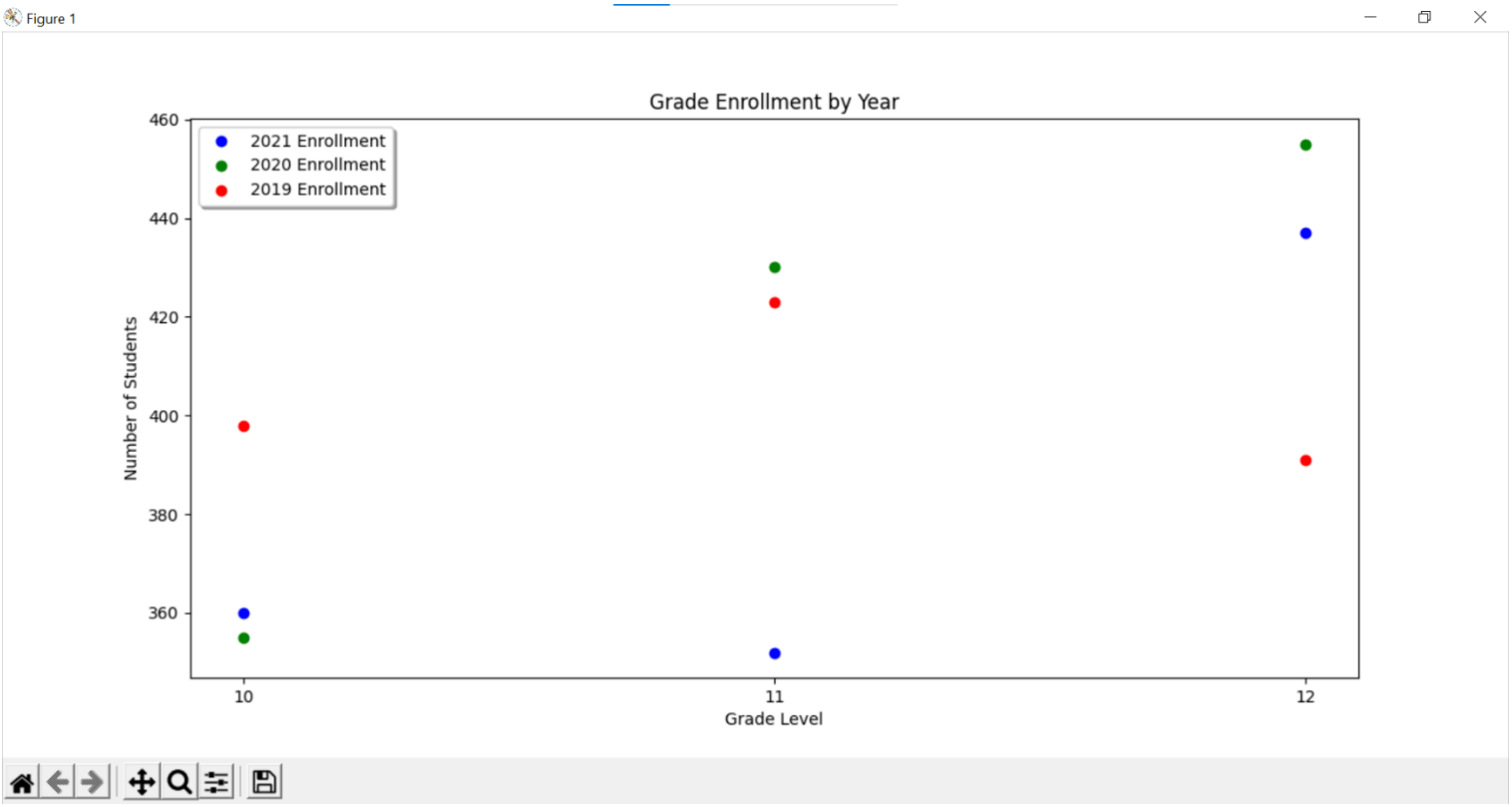


Figure 2 (Bonus)