DAEN 500- DL1 – Data Analytics Fundamentals

Fall 2020 Final Examination Exercise

11/24 – 12/05/2020

Final Submission Deadline: NLT 11:59PM (EST). Saturday, Dec 5, 2010

*Failure to submit ON TIME will result in DAEN COURSE FAILURE*

Name:\_\_\_\_\_\_\_\_Amy Vaughan\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ GMU G#\_\_\_ G01287247\_\_

Student Signature (Honor Certification): \_\_\_\_\_

This exam is **OPEN BOOK/OPEN NOTES**. You may consult any of the course texts, and the various reference materials recommended in the syllabus. ***The exam of course IS NOT “Open Web”,*** especially in that you may NOT utilize expert “help” sites such as Stack Overflow, or other programming help or collaboration sites.

HONOR CODE CERTIFICATION

**Your signature above declares that you have followed the conditions of this exam, and that the work is yours alone**. **Specifically:**

This must be your own work, authored and completed by you. As stated earlier, this is an “open source exam” – allowing books, notes or courseware, as well as *general* expert advice gained PRIOR to exam. YOU MAY NOT, HOWEVER, SEED OR USE ANY ADVICE ON HOW TO SOLVE THE QUESTION OR ANY CODE WRITTEN BY ANY OTHER INDIVIDUAL. *Any violation will result in an immediate failure in the exam and for the course, as well as referral to the GMU Honor Committee for determination of any other appropriate disciplinary consequences.*

*NOTE: Your* ***submission*** *of any responses, files, programs, etc. in response to the DAEN500 final exam instructions, will also be your personal certification of your full compliance with the spirit and letter of the* ***GMU Honor Code*** *standards for take home and/or in-class exams.*

Additionally, you are restricted from discussing the substance of the questions on this exam with any other individual, until after you have submitted your final response for grading. The completed exam -- with your answers embedded in this docx document (add extra pages as necessary) should be submitted following instructions contained in the Final Exam Instructions BB site. If you have any trouble submitting and have extra parts of the answers you have trouble appending to this document, you may simply submit additional pages separately (the exam submission site is set for multiple submissions, just in case). Make certain all are submitted PRIOR TO THE DEADLINE!

 FINAL EXAM PROBLEMS

COMPLETE ALL & INSERT ANSWERS BELOW QUESTIONS

# Problem 1: Python Programming Problem (15 Points Total)

* **Design and implement a Python program that is based on the following requirements: a) program will find all numbers which are divisible by 7 but are not a multiple of 5; and b) numbers between 2000 and 3200.**
* **INSERT (cut&paste) your Python code in space below and *then insert a screen shot in space below, showing code, your successful run, input and output.***

NOTE of alternative for help: To help test your code, you also may use a Python “programming window” found in the. **Zybooks Section 35 Additional Material**.

results = []

number = 2000

while number <= 3200:

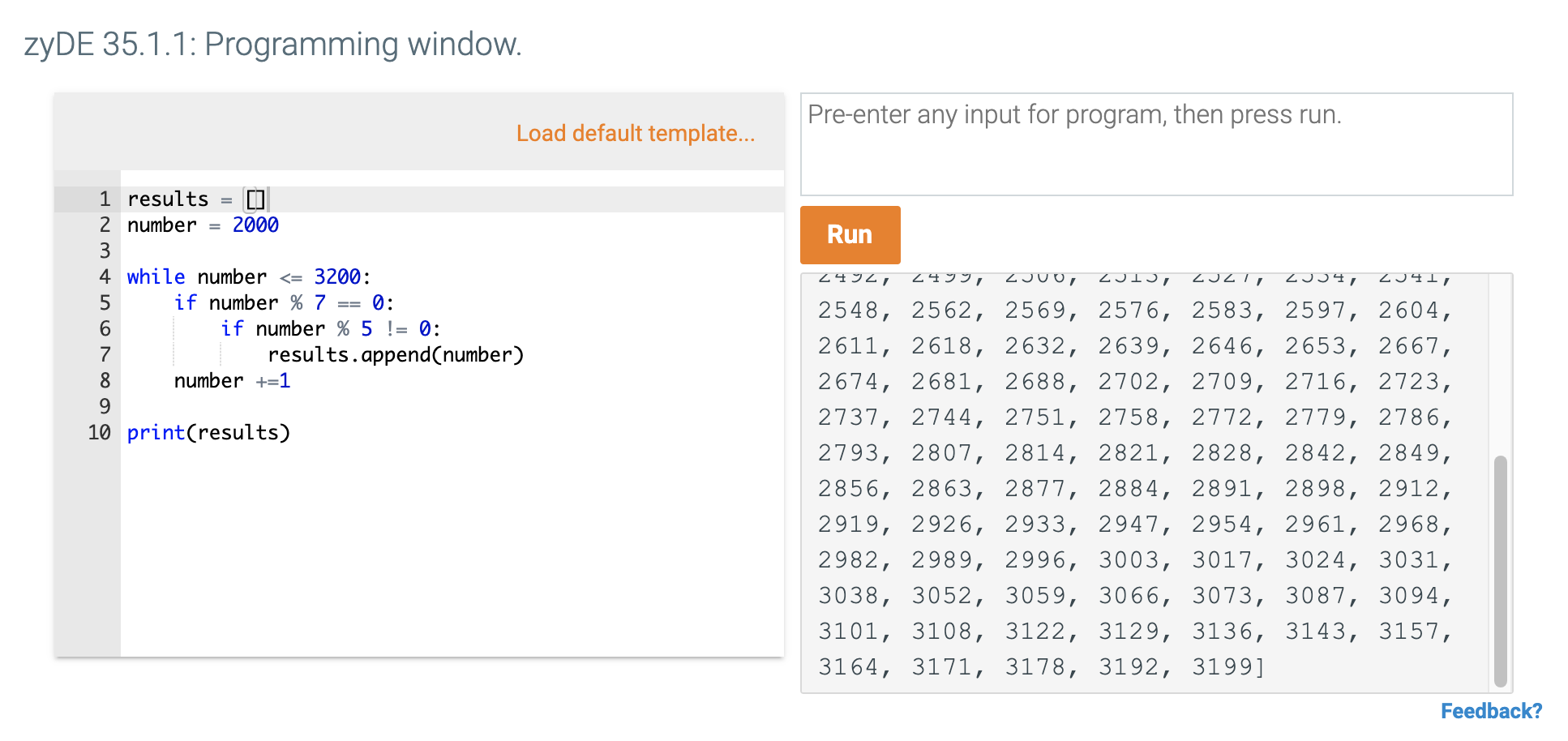
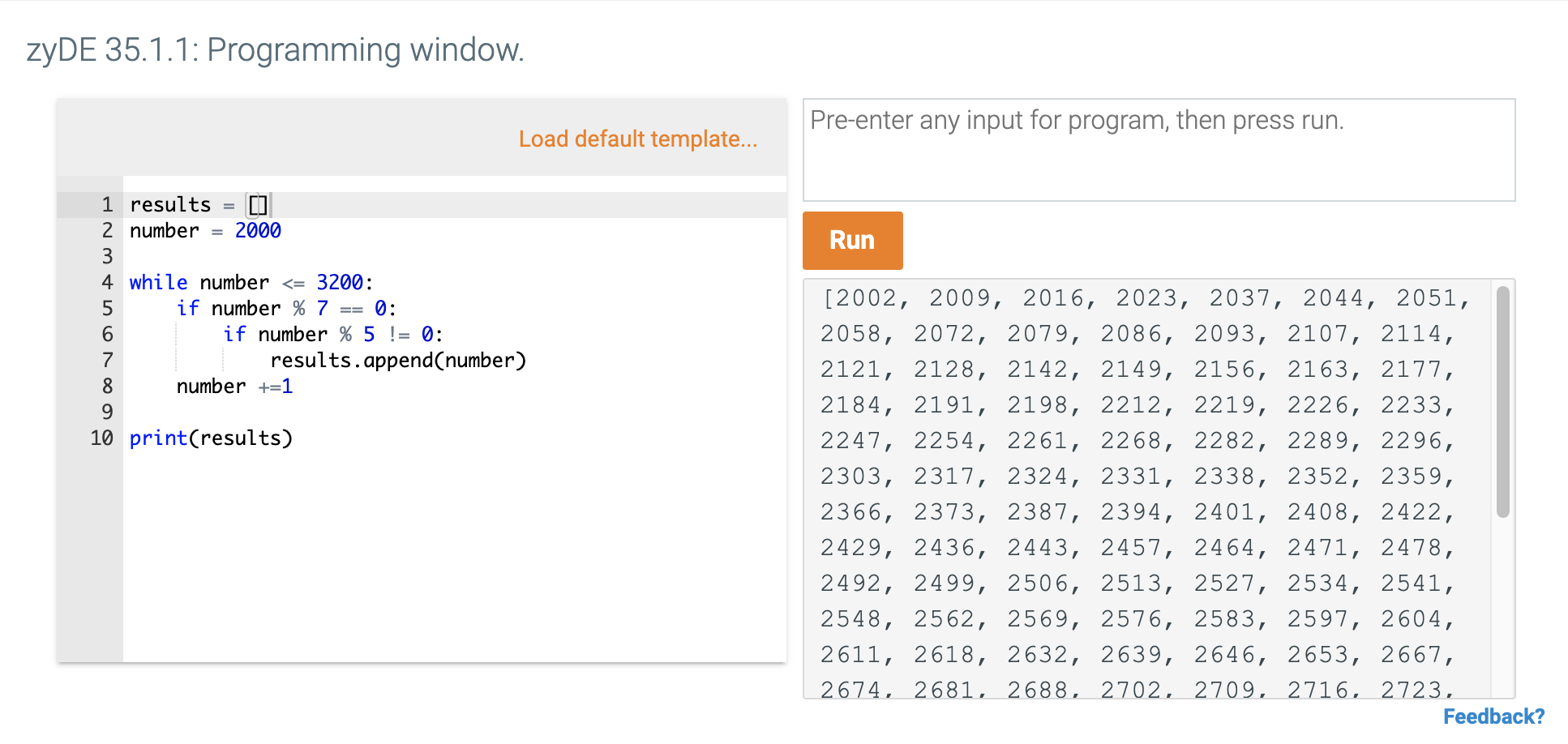
if number % 7 == 0:

if number % 5 != 0:

results.append(number)

number +=1

print(results)



# Problem 2: Python Programming Problem

# (15 Points Total)

* **Design and implement a Python program that is based on the following requirements:**

**a) define a class which has *at least two* methods**

* + **Method 1 – getString: to get a string from console input; and,**
  + **Method 2 - printString: to print the string in upper case.**

**b) demonstrate code works using three different test input strings**

* ***INSERT* *code below* and *INSERT* a screen shot of the program and successfully run output that *includes test input for input strings (test strings must include (a) all upper case, (b) all lower case, and (c) mix of upper and lower case).***

class SillyString:

def \_\_init\_\_(self):

self.string = None

def getString(self, inputString):

self.string = inputString

def printString(self):

if self.string is not None:

print(self.string.upper())

if self.string is None:

print("No string to print, use getString to add one\n")

if \_\_name\_\_ == "\_\_main\_\_":

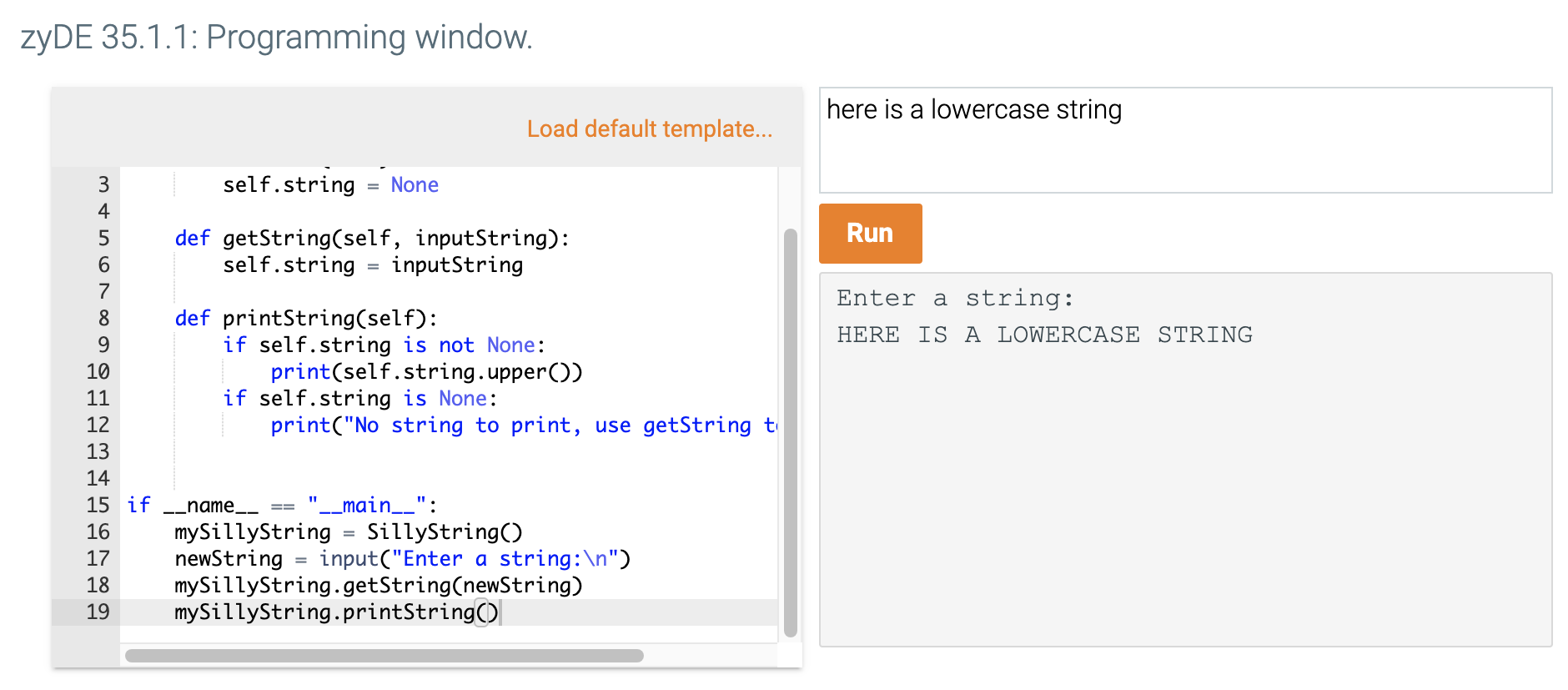
mySillyString = SillyString()

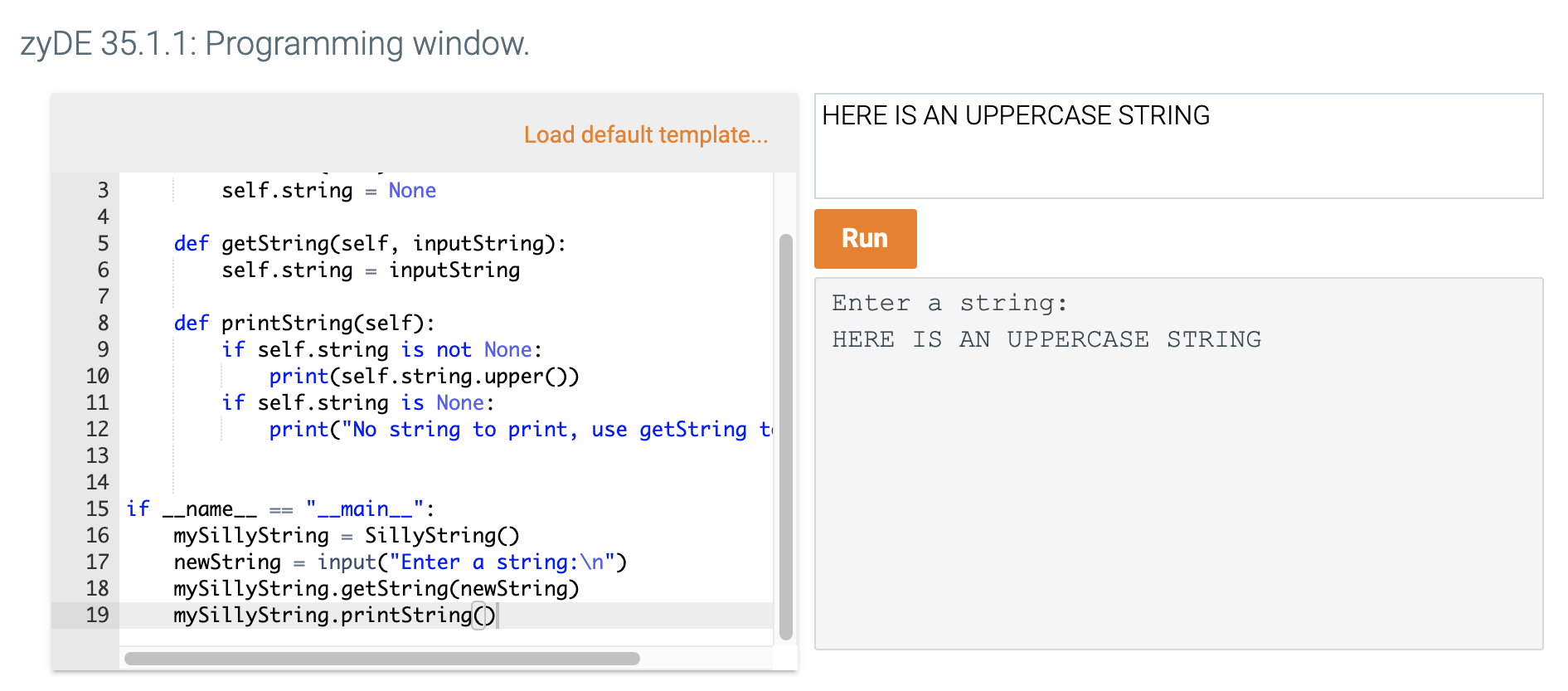
newString = input("Enter a string:\n")

mySillyString.getString(newString)

mySillyString.printString()









# Problem 3: R Programming Problem

# (20 Points Total)

* **Perform the following problems using R:**
  + Create a vector of courses (e.g., MATH 101) you have taken previously. Make sure you have at least 8 courses. Name the vector myCourses
  + Get the length of the vector myCourses
  + Get the first two courses from myCourses
  + Get the 3rd and 4th courses from myCourses
  + Sort myCourses using a method
  + Sort myCourse in the reverse direction
* *INSERT* *code below* and *INSERT* a screen shot of the program and successfully run output.

myCourses <- c('LING 101', 'LING 220', 'LING 250', 'LING 300', 'ANTH 101', 'HUM 110', 'REL 200', 'REL 220')

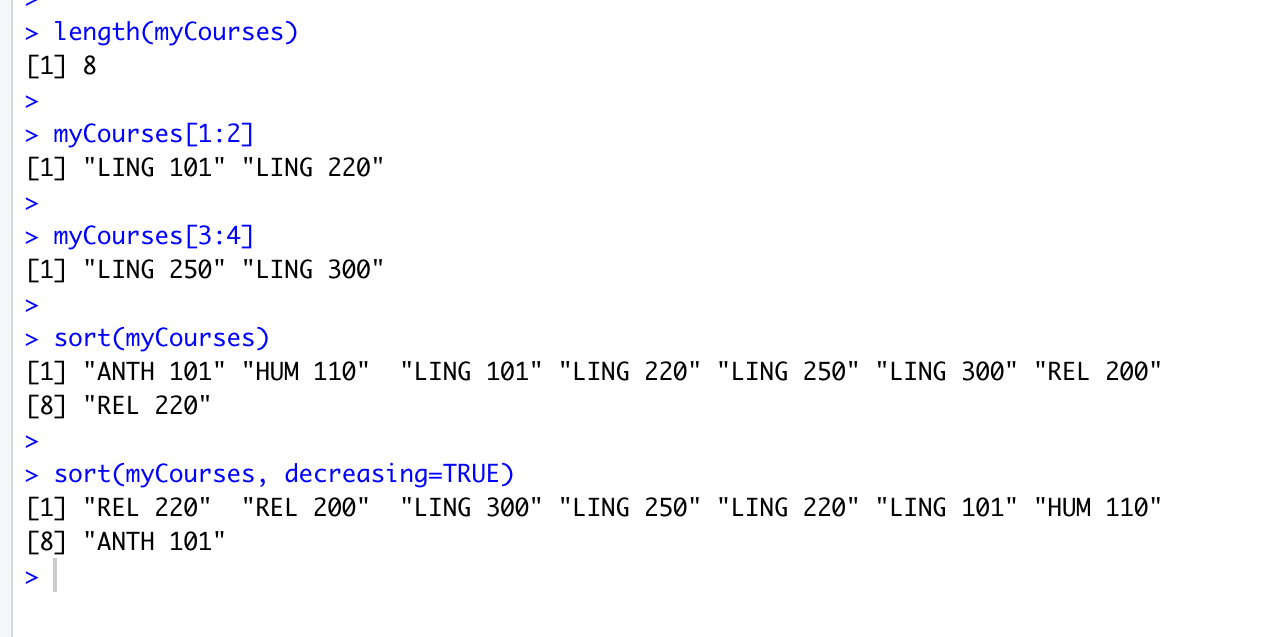
length(myCourses)

myCourses[1:2]

myCourses[3:4]

sort(myCourses)

sort(myCourses, decreasing=TRUE)





# Problem 4: Principal Component Analysis

# (25 points)

**Provide a description of the following:**

1. What is a component – Provide a description (5 points)

In Principal Component Analysis, data with numerous dimensions or variables are reduced so that the data is represented with fewer variables. The resulting variables, created by combining weighted versions of the old variables, are the components.

1. Principal Component Analysis – Provide a description.(5 points)

Principal Component Analysis is a method of simplifying complex data with numerous dimensions by reducing the variables to a smaller number of variables (i.e., 3). This allows data points which previously had too many dimensions for a human to visualize to be graphed in a more understandable way (though the nature/interpretation of the dimensions created is not easily explainable). Often, the transformed data shows different groupings or clusters which could not be easily visualized in the original data.

1. **Provide an specific example of Principal Component Analysis(15 points)**

Stylometry is the practice of analyzing text to identify stylistic components that are believed to be unique to specific authors, most commonly to determine whether an unknown document can be attributed to a specific author based on other known works by the same author. The most popular method of doing this involves performing frequency counts of words that appear in the documents, then treating each word as a variable. While traditionally this method used the top 100 or 200 words, more recent studies can involve the top 2000 words. This means that you could be dealing with 2000 variables per document. While it’s easy for a computer to compare those 2000 variables per document and produce the desired output, it is not really feasible to visualize this comparison. Principal Component Analysis can be used to collapse those 2000+ variables into a mere 2 or 3 components, which allows documents to be visualized in a way where their proximity can be interpreted as a measure of their similarity. Thus, when looking at a corpus of two or three authors, you can see their works forming clusters in different areas of a PCA graph, and decide which author a questioned document is most similar to based on its location in this graph.

# Problem 5: Multiple vs. Logistic

# (30 points)

# Describe: What is difference between Multiple Regression and Logistic Regression? What circumstances might determine which to use? (10 points)

# These two types of regression vary based on the type of equation used to calculate them, as well as the number of variables that are used. For example, a multiple regression is a linear regression (y = ax + b) which has more parameters (y = ax + cx + dx + b where a, c, and d are the additional parameters). A logistic regression is calculated with a completely different function (y = (e^(ax+b))/(1 + e^(ax+b)) + E, where E is error). A multiple linear regression function might be used in a problem where the data still have a linear relationship to each other, but there are multiple factors in determining the response variable y. As an example, when predicting the ultimate sale price of an item for auction, multiple factors, such as the age of the item, its condition, its color, and so on, might contribute. Ultimately, the response variable y is a continuous number. A logistic regression might be used where the response variable y is a discrete or binary option (such as true/false). For example, a logistic regression might be used when trying to predict whether a student will pass a class based on the time they spend studying, because passing a class is a binary response variable (they either pass or they don’t).

# Demonstrate: Using any data, and any tool set you’ve learned about, show differences (20 points)

# SUGGESTION: may be solved using RapidMiner, or other toolsets, BOTH TO ANALYZE AND TO VISUALIZE REGRESSION DIFFERENCES.

Step 1: Perform a quick search of the [UCIS public data archive](https://archive.ics.uci.edu/), a well-curated site which you already have seen as part of your introductory RapidMiner training.

Step 2: Pick a dataset you find interesting, input dataset into regression tools you’ve chosen.

Step 3: Run regression, .and use visualizations to demonstrate the conceptual answers you provided for 5.(a).

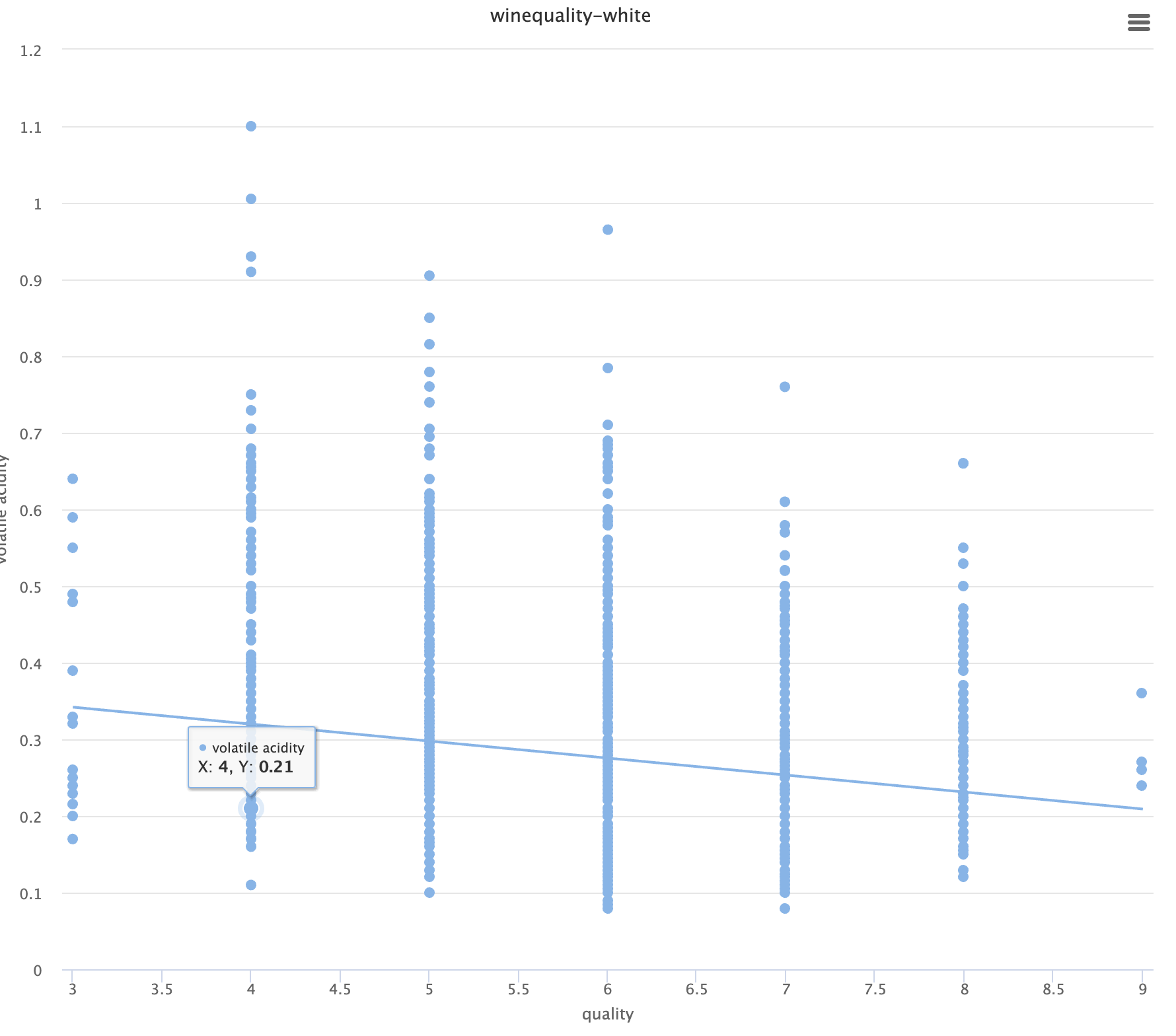
Dataset: Wine Quality

I chose the ‘Wine Quality’ dataset, which was a study of Portuguese wines and their chemical properties. Since I am a fan of this specific type of Portuguese wine, and particularly its white variety, I thought I would investigate if there are any particular attributes of the wine that are correlated with the quality score using linear regression. I am also a homebrewer, so I was hoping to learn something about making better quality wines at home. While the quality variable here is not technically a continuous variable, since the quality is judged on a scale of 1-10 and scores in between integers would still be logical, a linear regression still seems appropriate.

Here you can see the preliminary results of my linear regression in RapidMiner:



In a finding that may prove wine enthusiasts are correct that understanding wine is an art, many variables made statistically significant contributions to this regression analysis. Only two variables were likely to play no role: total sulfur dioxide (which is the infamous ‘sulfites’ that are used to preserve wine quality and make sure that no nasty bugs make it in) and chlorides (described as ‘the amount of salt in the wine’). Already, these are interesting results, because sulfites are popularly thought of as being ‘bad’. Meanwhile, while I think we can all agree that salty wine would be bad, but this doesn’t seem to bear out statistically speaking for this varietal. Most of the variables have a positive relationship to quality, while two have a strong negative relationship: density and volatile acidity. Volatile acidity represents the level of acetic acid in wine, so the reason why that relationship may exist seems clear – nobody wants to drink vinegar. In the visualization below, we can see that the highest quality wines have the lowest average volatile acidity.



# For our other heavily weighted attribute, density, the possible cause is not as clear.

# 

# Density had a strong negative effect on quality, so our linear regression line is a downward slope. The color values are set by the alcohol content of the wine (cool colors being low alcohol and hotter colors being higher), because I was curious to see if alcohol played a role here. Alcohol is inversely correlated with density, because the density of a wine is in part determined by how much sugar remains after fermentation. However, the alcohol content (alcohol by volume or ABV) does not have the same feature weight, and when we look at a graph of alcohol we can see its impact on quality is not pronounced, even though the highest rated wines have a higher average ABV:

# 

# Further research tells us ABV is not the only contributor to density, so there must be other variables that are unaccounted for in this particular study. Density could be an interesting direction for future research for my data-driven homemade wine.