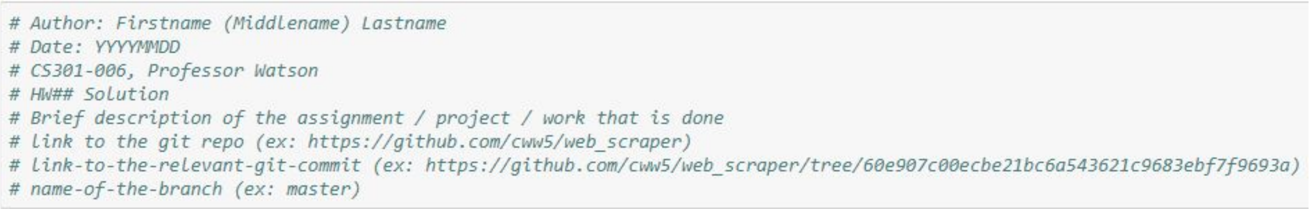
CS 301 HW 02

Total possible score: 20 points

Problem 1) 5 points for successfully handing in the proper submission requirements (assuming you have downloaded everything properly, and include the required header for the file as described below, you should receive full points.)

Please utilize Module 02 in Canvas to download the Anaconda distribution to your computer. If you aren't experienced with Git-related Version Control, I suggest downloading GitHub Desktop (it's what I use), otherwise feel free to use the Git command line, or whatever you're comfortable with. Assuming you submit the homework with the below specified format, you will get full points for this problem.

NOTE: From this point forward, unless specified, all homework submissions must be in a .ipynb (Jupyter Notebook) file, **and** you must upload it to your GitHub account, **and** it must include the following comments in the first cell:



```
# Author: Firstname (Middlename) Lastname
# Date: YYYYMMDD
# CS301-006, Professor Watson
# HW## Solution
# Brief description of the assignment / project / work that is done
# link to the git repo (ex: https://github.com/cww5/web_scraper)
# link-to-the-relevant-git-commit (ex: https://github.com/cww5/web_scraper/tree/60e907c00ecbe21bc6a543621c9683ebf7f9693a)
# name-of-the-branch (ex: master)
```

Problem 2) 5 test cases, 1 point for each passed test case. (5 points total)

Using Python in the Jup.Notebook, write a function called *q_summary_V1* which takes as input:

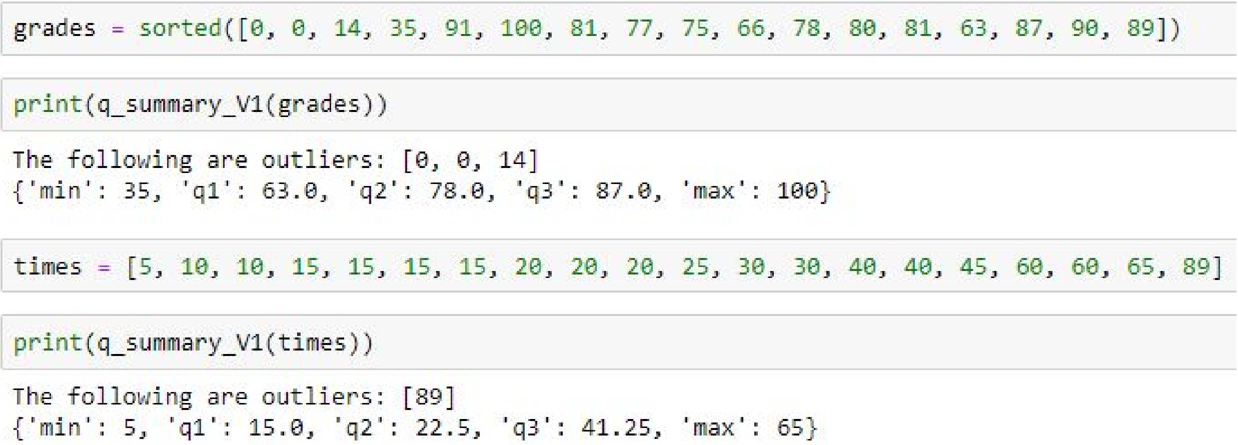
numbers - a list of numbers

The function *q_summary_V1* should return a dictionary containing the min, Q1, Q2, Q3, and max. Feel free to utilize the numpy function for quantile instead of implementing the formulas to calculate index positions that we reviewed in class. You will not lose credit for using numpy or for using default python, however, your answers should be correct either way.

q_summary_V1 should also warn the user if there are any outliers by printing them to the screen.

The following screenshot shows an example function call:

These are the two test cases that you should pass to gain 2/5 points as well.



```
grades = sorted([0, 0, 14, 35, 91, 100, 81, 77, 75, 66, 78, 80, 81, 63, 87, 90, 89])
```

```
print(q_summary_V1(grades))
```

```
The following are outliers: [0, 0, 14]
{'min': 35, 'q1': 63.0, 'q2': 78.0, 'q3': 87.0, 'max': 100}
```

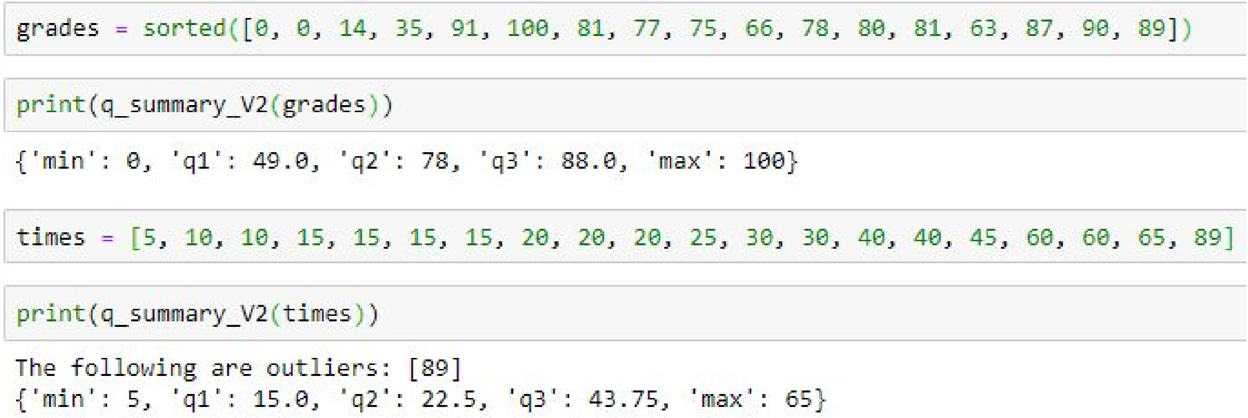
```
times = [5, 10, 10, 15, 15, 15, 15, 20, 20, 20, 25, 30, 30, 40, 40, 45, 60, 60, 65, 89]
```

```
print(q_summary_V1(times))
```

```
The following are outliers: [89]
{'min': 5, 'q1': 15.0, 'q2': 22.5, 'q3': 41.25, 'max': 65}
```

Problem 3) 5 test cases, 1 point for each passed test case. (5 points total)
Using the sample of travel times data set from class, the answers using numpy are incorrect. In fact, there are many ways to calculate via interpolation. Please refer to this link, which offers an explanation as to why there are many ways to retrieve the index. In this class, we will use the formula provided for consistency. To see the difference, please alter your code from Problem 2 and make a new function called *q_summary_V2* which should perform exactly the same thing as *q_summary_V1*, but via the method from class.

The following are the test cases you will need to pass in order to gain 2/5 points.



```
grades = sorted([0, 0, 14, 35, 91, 100, 81, 77, 75, 66, 78, 80, 81, 63, 87, 90, 89])
```

```
print(q_summary_V2(grades))
```

```
{'min': 0, 'q1': 49.0, 'q2': 78, 'q3': 88.0, 'max': 100}
```

```
times = [5, 10, 10, 15, 15, 15, 15, 20, 20, 20, 25, 30, 30, 40, 40, 45, 60, 60, 65, 89]
```

```
print(q_summary_V2(times))
```

```
The following are outliers: [89]
{'min': 5, 'q1': 15.0, 'q2': 22.5, 'q3': 43.75, 'max': 65}
```

Problem 4) 5 points for completing the survey
Please complete the weekly survey available here.
This is the link: https://forms.gle/sS76YiGEnxWAoCMP9

This survey is for you to submit your opinions about the topics covered each week. It is also feedback that I will use to improve the course moving forward.