# OMIS 30 - Fall 2019 - Project 1

## Logistics:

Assigned: Thursday, October 3, 2019
Due: Thursday, October 17, 2019 1159pm

## Objective:

Write a program that performs various functions in Python, based on the user input.

## The requirements are:

- Print out a menu of options, then query the user to select one.
- Based on the user's choice, calculate the following:
    - Calculate the first n Fibonacci numbers (where the user enters a number, n)
    - Print the numbers 1 to n, and print "Fizz" for mulitiples of 3, and "Buzz" for multiples of 5, and "FizzBuzz" for multiples of both (where the user enters a number, n)
    - Calculate the primes numbers up to n (where the user enters a number, n)
    - Create a list of random numbers, each random number being between -50 and 50, where the list is n long (where the user enters a number, n)

## You must use:

- print()
- input()
- If, elif, else
- A loop (for or while)
- comments
- Input validation check for all user inputs

## Collaboration:

You will work individually on the assignment.  You are allowed and encouraged to use Google extensively.

## Submission:

- Name your final file <your_username>_project1_fall2019.py (mine would look like dvrdoljak_project1_fall2019.py)
- Your name and SCU email must be at the top of the project as a comment
- Make sure it runs completely and correctly on your computer
- Submit it via Camino
- (We will run your program on our computer to test your answers)

Grading Rubric:

| Section | Grade | Criteria |
| --- | --- | --- |
| Each calculator | 40% (10% each) | Math formulas correct;<br>Returns correct answer |
| User prompts | 10% | Uses input;<br>Does >= 1 data validation;<br>Specificity of prompts |
| User prompt to input data | 15% | Uses input;<br>Uses if,elif,else;<br>Does >= 1 data validation in each |
| Ease of use | 10% | Prompts well defined;<br>Error handling done |
| Use of comments | 10% | Documentation of author & dates;<br>Explanation of steps |
| Readability | 10% | Use of whitespace;<br>Use of new lines;<br>Naming convention of variables |
| General & Submission | 5% | Code runs completely through w/o errors;<br>Submission named properly |

Bonuses:

- Make your program loop infinitely, until the user selects an option to quit (5%)
- Use functions for each of the calculations (5%)
- Instead of an if/else tree for the menu selection, use a dictionary where the keys are the user options, and the values are references to the functions for each (10%)