CS 160 – Exploring Computer Science

Assignment 3: Arrays

Academic Integrity

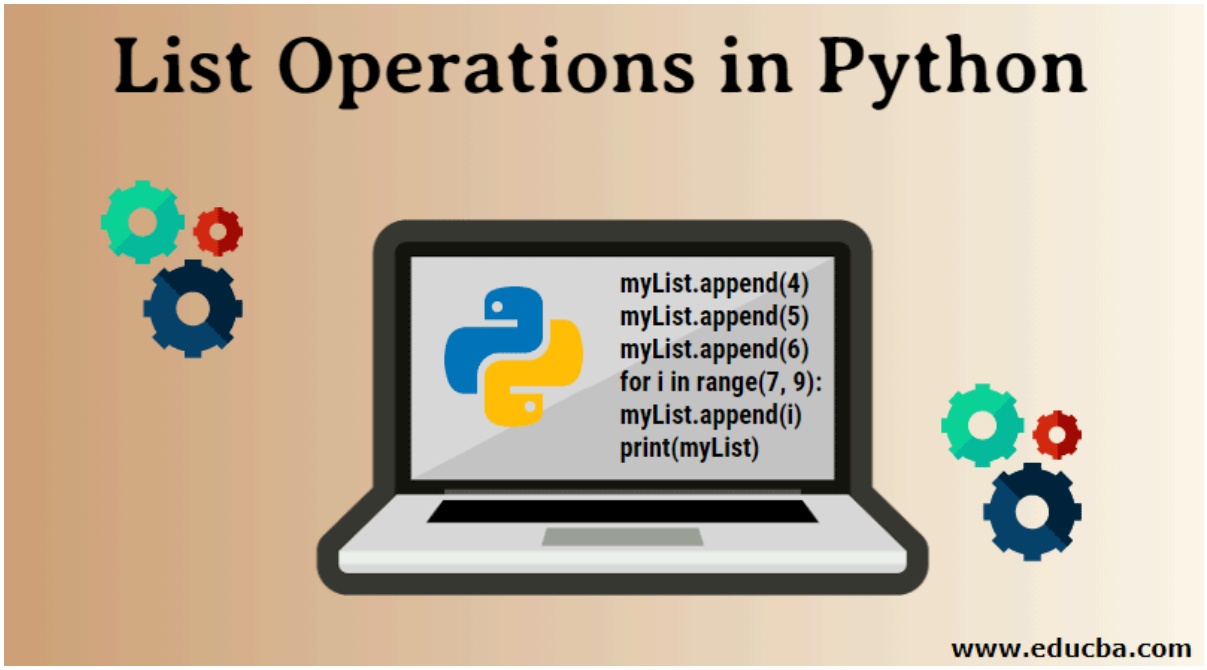
You may NOT, under any circumstances, begin a programming assignment by looking for completed code on StackOverflow or Chegg or any such website, which you can claim as your own. Please check out the Student Code of Conduct at PCC.

The only way to learn to code is to do it yourself. The assignments will be built from examples during the lectures, so ask for clarification during class if something seems confusing. If you start with code from another source and just change the variable names or other content to make it look original, you will receive a zero on the assignment.

I may ask you to explain your assignment verbally. If you cannot satisfactorily explain what your code does, and answer questions about why you wrote it in a particular way, then you should also expect a zero.

I understand that acts of academic dishonesty may be penalized to the full extent allowed by the Portland Community College Student Conduct Code, including receiving a failing grade for this assignment. I recognize that I am responsible for understanding the provisions of the PCC Student Conduct Code as they relate to this Course.

Your name here.



Purpose

For this assignment, you will create a more complex modular program that uses at least two parallel lists for storing related information, and has at least one loop that accesses the data in the lists. As long as your program satisfies the requirements listed below, you are free to design and write any type of program that you care to. You are encouraged to be creative, and pick something that has meaning for you, because you'll have more fun. Feel free to create a more complex version of the program you did in an earlier lab, as long as it meets all of the additional requirements below.

After completing this assignment you will be able to:

- Create programs in Python using simple lists.
- Access and process data in lists using loops in your program.
- Use parallel lists to hold related information of different types.

Task

- Before you get started
  - Look at this Assignment 03 Sample for an example and tutorial video.
- Open the Algorithmic Design Document, make a copy, and follow the steps to create your algorithm.
- You must express your algorithm as **pseudocode** or a **flowchart.**
- Print a welcome message for your program.
- Write an interactive application in Python that uses at least 2 parallel lists for storing related information in a meaningful manner. Some examples may include: student name and gpa, pet name and age, stock name and price, etc.
- The program must use loops to process the lists to calculate other information. For example, the student with the highest/average gpa, the average pet age, the highest/lowest stock price, etc.
- Here are some ideas to get you started:
  - Music Player Database - Get list of songs, artists from the user. Make a list and search for artist names and display the songs in the list.
  - Student scores - Have a list of student names and scores. Find the maximum and minimum scores and the class average.
- Algorithmic Design Document Requirements
  - Make a copy of the Algorithmic Design Document.
  - Start with a sample run to help with the design process.
  - Plan the logic steps for your algorithm. **You must answer the following questions with numbered steps and clear instructions to solve the problem.**
    - Identify all of the user input. What are the data types of the inputs? Define the input variables.
    - Describe the program output. What is displayed to the user? What are the data types of the output? Define the output variables.
    - What calculations do you need to do to transform inputs into outputs? List all formulas needed, if applicable. If there are no calculations needed, state there are no calculations for this algorithm.

        - Describe the process needed to transform inputs to outputs *using numbered steps*. Here is where you would use conditionals, loops, functions or array constructs (if applicable) and explain the process in transforming inputs into outputs.
        - Do not use any code or syntax here, just plain English sentences.
- ❖ Check the Criteria for Success section to make sure you have completed all requirements and to see how to submit your assignment.
- ❖ Check the Additional Support section for more resources to help you complete the assignment.

Criteria for Success

- ❖ Look at the Assignment Rubric. You will be graded on the following coding constructs:
    - ➢ lists
    - ➢ variables
    - ➢ input/output
    - ➢ input validation
    - ➢ mathematical computation
    - ➢ iteration/loops
- ❖ Include the required comment header (see style guide) and any explanatory comments in your code (always explain any mathematical calculations).
- ❖ Run your program and test with different user input. Do your prompts explain what the user should enter?
- ❖ Take a look at your user interface (UI) and user experience (UX). Did you provide a welcome message letting the user know what to expect? Is the UI nicely spaced with no typos?
- ❖ Your program must use a main() function and include the correct program comment header.
- ❖ Download your Algorithmic Design Document as a PDF (File -> Download -> PDF), rename it to `a03.pdf`, and upload it to the D2L assignment folder.
- ❖ Upload your `a03.py` Python source file to the D2L assignment by **Sunday**.
- ❖ Do your own work. Consult the syllabus for more information about academic integrity.

Additional Support

- ❑ Post a question for the instructor in the Ask Questions! area of the Course Lobby.
- ❑ Look at this Assignment 03 Sample for an example and tutorial video.
- ❑ Runestone Academy Resources:
    - ❑ Read through Section 10.1 from Chapter 10, Lists from this textbook. This section talks about *lists* which is a sequential collection of Python data values. Try some programs in the workspace provided.