# Homework 6 (83 points)

For this assignment, you will be uploading a zip folder containing your responses in red font to Questions 1-5 and your three scripts for Questions 6-8. Some important submission notes:

1. Save your folder as <last\_name>.zip, where you substitute <last\_name> with your last name (e.g., bowers.zip).
2. Make sure you use .zip extension in your uploaded submission, .docx for the Word document or .pdf if you save it as a PDF, and .py extension for your three Python scripts. Submissions that do not have these extensions will not be accepted.
3. Code that does not compile will be given 0 points.

## Question 1 (6 points)

Describe the difference between positional and keyword arguments. Provide an example of each (i.e., an example not from the lecture slides or textbook).

in positional arguments the parameters are received in the order that they are given. In Keyword arguments the variable name and value are given so the order does not matter.

Ex: def company\_name(name1,name2):

print(name1+" "+name2)

company\_name("Tesla","GM")

company\_name(name2="GM",name1="Tesla")

both of the function calls print out the same thing even though the parameters where entered differently.

Question 2 (4 points)

What is a default parameter value? Provide an example (i.e., an example not from the. lecture slides or textbook).

A default parameter is a default value in a function in the case that no parameter is entered.

Ex: def company\_name(name1="Tesla",name2="Gm"):

print(name1+" "+name2)

company\_name()

## Question 3 (4 points)

What is a return value? Write a function that returns some sort of value and paste it here.

A return value is a variable or data structure that is returned after a function is called.

Ex: def company\_name(name1,name2):

companys = [name1,name2]

companys\_str = print(companys)

return companys\_str

company\_name("tesla","gm")

## Question 4 (4 points)

How is passing a list different than passing a single variable, like a string? Be specific on what happens when the parameter is changed inside the function and when the variable is used outside of the function.

Lists can be modified by the function, but single variables cannot. If a variable is changed inside of the function it does not affect the variable outside of the function. If a list is modified inside of a function it will be modified outside of the function as well.

## Question 5 (10 points)

What do each of the following statements do? Be specific.

* import random imports the random module
* import random as rand imports the random module under the alias rand
* from random import randint imports the randint function from the random module
* from random import randint as rand imports the randint function as the alias rand from the random module
* from random import \* imports all functions form the random module

## Question 6 (10 points)

Write a Python script (not used in the textbook or lecture slides) that defines a function with an arbitrary number of arguments. Your script must:

* Call that function twice
* Use different arguments for each function call
* Use a different number of arguments for each function call
* Somehow access or modify each of the arguments inside your function
* Be commented
* 5 points extra credit: use a combination of positional and arbitrary arguments

## Question 7 (10 points)

Write a Python script (not used in the textbook or lecture slides) that uses an arbitrary number of keyword arguments. Your script must:

* Call that function twice
* Use different keyword arguments for each function call
* Use a different number of keyword arguments for each function call
* Be commented
* Somehow access or modify each of the keyword arguments inside your function

## Question 8 (35 points)

HINT: look at the class slides from 11-9 that created the Dog class.

Write a Python script (all in one file, with comments) that:

* (3 points) Creates your own class
* (6 points) Has an init method that receives three parameters and creates three attributes (i.e., class variables)
* (6 points) Has three functions that describe the behavior of your class
* Outside of the class definition:
  + (8 points) Creates two class instances that use completely different attributes
  + (6 points) Print each attribute from both instances to the screen (i.e., six print statements)
  + (6 points) Calls each of the functions in the class for both instances (i.e., six function calls)