FA26B Advanced Programming

Assignment 4 - Functions

Instructions

In your student folder on Teams, create a new folder named day_04. Submit your files to this folder. For each of the exercises, do the following:

- Create a py file and name it with the format: lastname_exercise#.py
- Add these three comments to your file (3 separate lines)
    - Your name
    - The exercise number
    - Description of what the program does
- Complete the exercise
- Upload the py files to your day_04 folder

Exercises

1. Write a function called make_shirt() that accepts a size and the text of a message that should be printed on the shirt. The function should print a sentence summarizing the size of the shirt and the message printed on it. Call the function once using positional arguments to make a shirt Call the function a second time using keyword arguments.

2. Modify the make_shirt() function so that shirts are large by default with a message that reads I love Python. Make a large shirt and a medium shirt with the default message, and a shirt of any size with a different message.

3. Write a function called make_album() that builds a dictionary describing a music album. The function should take in an artist name and an album title, and it should return a dictionary containing these two pieces of information. Use the function to make three dictionaries representing different albums. Print each return value to show that the dictionaries are storing the album information correctly.

4. Add an optional parameter to make_album() that allows you to store the number of tracks on an album If the calling line includes a value for the number of tracks, add that value to the album’s dictionary. Make at least one new function call that includes the number of tracks on an album.

5. Start with your program from Exercise 4. Write a while loop that allows users to enter an album’s artist and title. Once you have that information, call make_album() with the user’s input and print the dictionary that’s created. Be sure to include a quit value in the while loop.

6. Write a function that accepts a list of items a person wants on a sandwich. The function should have one parameter that collects as many items as the function call provides, and it should print a summary of the sandwich that is being ordered. Call the function three times, using a different number of arguments each time.

7. Write a function that stores information about a car in a dictionary. The function should always receive a manufacturer and a model name. It should then accept an arbitrary number of keyword arguments. Call the function with the required information and two other name-value pairs, such as a color or an optional feature. Return the dictionary. Your function should work for a call like this one:

   ```
   car = make_car('subaru', 'outback', color='blue', tow_package=True)
   ```