# Homework 3 — Baking Bonanza!

Due Saturday, September 18th at 9 pm

(50 points)

**Objectives:**

- Write a program that uses functions for organization and code reuse
- Learn how to incrementally develop and run programs
- Understand the scope of different variables
- Understand the structure and purpose of the `main` function
- Understand the use of parameters with functions

**Turn In:**

- `hw3_recipe.ipynb`

You should turn this file in on Canvas under the “Homework 3 - Baking Bonanza” link. The notebook should include your answers to the questions in both part 1 and part 2 of this homework.

## <u>Part 1 (5 points): Warm-up</u>

Write a program that helps a user to build a letter with a STRONGLY WORDED (uppercased) message.

The program should ask a user for the following inputs:

- Who are they sending the letter to?
- What should the greeting/salutation be?
- Body of the letter, i.e. the message that the user wants to emphasize.
- Signature/Sender’s name.

The program should then output a completed letter with the help of inputs taken.
In the completed output letter, the body of the letter should be UPPERCASED to denote the STRONGLY WORDED message.

You do not need to define your own functions for this question.

Pay attention to the formatting of your generated letter, you must replicate the output formatting as shown in the example below!
Example output (user input **<u>bold underlined</u>**):

```
Who are you writing to? Fred Astaire
What is your greeting? Cheerio
What is your message? I would like more dancing
What is your signature? Felix

Your letter:
Cheerio Fred Astaire,
I WOULD LIKE MORE DANCING
Felix
```

```
Who are you writing to? Dumbledore
What is your greeting? Good evening
What is your message? I would like a pair of woolly socks
What is your signature? Suchi

Your letter:
Good evening Dumbledore,
I WOULD LIKE A PAIR OF WOOLLY SOCKS
Suchi
```

## Part 2 (45 points): Baking Bonanza!

Basil wants to make cupcakes for their entire class, but the recipe does not make enough! We want to write a program to help calculate how much of each ingredient will be needed so that everyone in the class (for example, 37 students) can have a cupcake.

Write a program that lets a user (like Basil) enter how many cupcakes they want to make, then report how much of each ingredient they will need. Use **grams** to specify the amount of flour, sugar, and butter, **teaspoons** for vanilla and baking powder, and **number** of eggs for the eggs.

**For this part of the homework, the only code not contained within a function should be a single call to a `main()` function.**

The table below shows all the recipe ingredients to make 12 cupcakes on the left (link is included if you'd like to take a look). On the right the expected output by the program given the user wants to make 12 cupcakes (user input is shown in **underlined bold**) after doing the calculations is shown.

| | |
|---|---|
| Recipe:<br>Makes 12 cupcakes<br>- 100 grams flour<br>- 100 grams sugar<br>- 100 grams butter<br>- 2 eggs<br>- 1 teaspoon vanilla extract<br>- 1 teaspoon baking powder | `How many cupcakes do you want to make?` **12**<br>`100.0 grams flour`<br>`100.0 grams sugar`<br>`100.0 grams butter`<br>`2.0 eggs`<br>`1.0 teaspoons vanilla`<br>`1.0 teaspoons baking powder` |

Next, display the same information, but this time convert the amounts in grams to cups so the user can still make the recipe even if they've lost the scale! The table below shows the conversion factors on the left. On the right, the expected output by the program (given you're still making 12 cupcakes) is shown but now converted into units of cups.

| | |
|---|---|
| Conversion factors:<br>- 120 grams flour = 1 cup<br>- 200 grams sugar =  1 cup<br>- 240 grams butter = 1 cup | `Now, in cups!`<br>`0.83 cups flour`<br>`0.5 cups sugar`<br>`0.42 cups butter`<br>`2.0 eggs`<br>`1.0 teaspoons vanilla`<br>`1.0 teaspoons baking powder` |

**Expected Full Output:**

You should expect to exactly reproduce the output below. Notice that the numbers in the expected output are rounded to two decimal places. Use the `round(value, places)` function to help you.

Expected output (second example on next page, user input in **underlined bold**)

```
How many cupcakes do you want to make? 12
100.0 grams flour
100.0 grams sugar
100.0 grams butter
2.0 eggs
1.0 teaspoons vanilla
1.0 teaspoons baking powder

Now, in cups!
0.83 cups flour
0.5 cups sugar
0.42 cups butter
2.0 eggs
1.0 teaspoons vanilla
1.0 teaspoons baking powder
```

```
How many cupcakes do you want to make? 1
8.33 grams flour
8.33 grams sugar
8.33 grams butter
0.17 eggs
0.08 teaspoons vanilla
0.08 teaspoons baking powder

Now, in cups!
0.07 cups flour
0.04 cups sugar
0.03 cups butter
0.17 eggs
0.08 teaspoons vanilla
0.08 teaspoons baking powder
```

**Development:**

Follow each step given below then test your code before moving to the next step. At each step, it may help to add `print()` statements to make sure the code is working the way you expect, but these should be deleted  (or commented out) once you are done writing the program.

1) Write a function named `main()`, where you will ask the user how many cupcakes they want to make and calculate the "recipe_multiplier". For example, your "recipe_multiplier" will be 1 if the user is making 12 cupcakes, 2 if they are making 24, 0.0833333333 if they are making 1 cupcake, etc. The function itself won’t run unless you call it, so make sure to do that.
2) Write a function to calculate a single ingredient amount that is adjusted for the number of recipes (“recipe multiplier”) you are making. This function will require four arguments/parameters: the recipe multiplier, the original amount needed of the ingredient, the name of the ingredient and the unit the ingredient is measured in. (Hint: for eggs, the unit should just be an empty string “”). This function should print the new, adjusted value of the ingredient rounded to the nearest 100ths place.

3) Print out the needed recipe ingredients in grams by calling the function you defined in step 2. The calls should come from your `main()` function.
4) Write a function to calculate how many cups are in a given number of grams of an ingredient. This function should return this new, converted value, again rounded to the nearest 100ths place. This function takes two parameters, the amount of this ingredient needed in grams for the base recipe and the number of grams in a cup for this ingredient. For example, when making 12 cupcakes and calculating the amount of flour in cups, this function would be passed the values `100` and `120`.
5) Calculate the converted ingredient amounts for flour, sugar, and butter by calling your function from step #5 in your `main()` function. Don't forget to save the return values!
6) Print out the needed recipe ingredients in cups. Make sure to make use of the function you defined in step 2!

Make sure that your `main` function is structured like a “table of contents” rather than a waterfall.
Note that this is an example, we expect you to give your functions names that are meaningful to your program, not “chapter1”, etc!

| Table of contents (good! main has control, calls the functions, control returns to main) | Waterfall (bad! functions are strung together, the first calling the next, and so on) |
|---|---|
| `def chapter1():`<br>`    # code`<br><br>`def chapter2():`<br>`    # code`<br><br>`def chapter3():`<br>`    # code`<br><br>`def main():`<br>`    chapter1()`<br>`    chapter2()`<br>`    chapter3()` | `def chapter1():`<br>`    # code`<br>`    chapter2()`<br><br>`def chapter2():`<br>`    # code`<br>`    chapter3()`<br><br>`def chapter3():`<br>`    # code`<br><br>`def main():`<br>`    chapter1()` |

As a reference, our solution, including empty lines and comments, is about 65 lines.

**Comments (5 points, included in the 45 points associated with part 2)**
Your code must be commented. You must include a file comment, inline comments, and function comments. An example function comment is shown below:

```
def fahrenheit_to_celsius(fahrenheit):
    “””
    This function converts a temperature in fahrenheit to celsius and prints
    the equivalence.

    Parameters:
        fahrenheit: (int or float) degrees fahrenheit
    Return:
        celsius: (float) equivalent temperature in celsius
    “””
    celsius = (fahrenheit - 32) * (5/9)
    return celsius
```

**Bonus (3 points)**
Add a second recipe to your program! After printing out the cupcakes information, do the same things (ask the user for the amount, print the adjusted recipe, print the converted recipe) but with a different recipe. Note! If your original recipe uses cups instead of grams, the adjusted recipe should use cups and the converted recipe should use grams! Add your new recipe to the output of your original program, after the cupcakes recipe (don’t copy and paste everything again—that’s the whole benefit of functions!).

Consult a website such as: https://www.kingarthurflour.com/learn/ingredient-weight-chart.html to help you find any new conversions that you may need.