Homework 1 - Functions & Expressions

CS 1301 - Intro to Computing - Spring 2021

Important

- Due Date: **Tuesday, January 26th, 11:59 PM**.
- This is an individual assignment. High-level collaboration is encouraged, **but your submission must be uniquely yours.**
- Resources:
  - TA Helpdesk
  - Email TA's or use class Piazza
  - How to Think Like a Computer Scientist
  - CS 1301 YouTube Channel
- Comment out or delete all function calls. Only import statements, global variables, and comments are okay to be outside of your functions.
- **Read the entire document before starting this assignment.**

The goal of this homework is for you to practice and understand how to write functions and evaluate expressions. The homework will consist of 5 functions for you to implement. You have been given a `HW01.py` skeleton file to fill out. However, below you will find more detailed information to complete your assignment. Read it thoroughly before you begin.

Hidden Test Cases: In an effort to encourage debugging and writing robust code, we will be including hidden test cases on Gradescope for some functions. You will not be able to see the input or output to these cases. Below is an example output from a failed hidden test case:

```
Test failed: False is not true
```

Written by Assata Quinichett (aquinichett@gatech.edu) & Fareeda Kasim (kasimfareeda@gatech.edu)

Helpful Information To Know

String Formatting

A concept that will be very helpful for this homework is string formatting. String formatting allows you to manipulate strings using variables so that string values can change based on whatever information is stored in the variables. To explore this concept, let's look at an example where a user inputs a name and age, and the code prints out the corresponding information:

```
name = input("What is your name?")
age = input("How old are you?")
print("Your name is {} and you are {} years old!".format(name, age))
```

Anywhere in a string, you can put `{}` to indicate a placeholder for a variable. After the end quotation marks of the string, you write `.format()`, and inside the parentheses will be the variables that you want to include. The variables inside the parentheses must be in the order that you want them to be included in the string.

Rounding Numbers

Python has a built-in function that allows you to round numbers. For example:

```
>>> rounded_number = round(3.1415926, 4)
>>> print(rounded_number)
3.1416
```

Inside the parentheses of the `round()` function, put the number you want to round, followed by a comma and the number of decimal places you want to round the number to.

Bake

Function Name: bake()
Parameters: N/A
Returns: None
Description: To make some extra money this semester, you decide to start a baking service. You can make cakes in 100 minutes, cupcakes in 70 minutes, and cookies in 45 minutes. Write a function that asks the user how many cakes, cupcakes, and cookies they want you to make that day. Print a response with how much time it will take to make their order in hours and minutes. **You may assume all inputs will be integers.**

```
>>> bake()
How many cakes do you want? 2
How many cupcakes do you want? 5
How many cookies do you want? 4
It will take 12 hours and 10 minutes to make 2 cakes, 5 cupcakes, and 4 cookies.
```

```
>>> bake()
How many cakes do you want? 1
How many cupcakes do you want? 3
How many cookies do you want? 12
It will take 14 hours and 10 minutes to make 1 cakes, 3 cupcakes, and 12 cookies.
```

Cake Volume

Function Name: cakeVolume()
Parameters: N/A
Returns: None
Description: Your customers often request cakes of different sizes, so you calculate the volume of each cake you make. Write a function that asks the user what the radius and height of the cake is, and then calculate the volume of the cake with the given dimensions. Print the volume of the cake. **The inputs can be floats. Round your answer to two decimal points.**

Note: Use `3.14` as the value for pi.
Note: The volume of a cylinder is `V = (pi)(r^2)(h)`, where r is radius and h is height.

```
>>> cakeVolume()
What is the radius of the cake? 3
What is the height of the cake? 4
The volume of the cake is 113.04.
```

```
>>> cakeVolume()
What is the radius of the cake? 5
What is the height of the cake? 8
The volume of the cake is 628.0.
```

Celebrate

Function Name: celebrate()
Parameters: N/A
Returns: None
Description: Since you are earning money from your baking service, you decide to surprise your friends with food for a fun celebration! You can buy pizza for $14, pasta for $10, and burgers for $7. Write a function that asks the user how many of each food item they would like, and the percentage they want to tip the delivery person. Print the total price you pay, and the amount you tipped. **All inputs will be positive integers. Round all outputs to two decimal places.**

```
>>> celebrate()
How many pizzas do you want? 3
How many orders of pasta do you want? 4
How many burgers do you want? 2
What percent would you like to tip? 10
You paid $105.6 and tipped $9.6.
```

```
>>> celebrate()
How many pizzas do you want? 7
How many orders of pasta do you want? 3
How many burgers do you want? 15
What percent would you like to tip? 20
You paid $279.6 and tipped $46.6.
```

Bookstore

Function Name: bookstore()
Parameters: N/A
Returns: None
Description: This new year, you've decided to clean up your space and tidy up. While cleaning, you found a pile of books you borrowed from the bookstore last year, and decided to finally return them. However, the bookstore has a very strict policy regarding borrowed books. After 14 days, there's a $0.25 fine per day for each borrowed book. Write a function that asks the user how many days they've had their book for and tells the user how much they need to

pay for the book they borrowed based on the number of days they've had it **All inputs will be positive integers.**

Note: All inputs will be greater than or equal to 14

```
>>> bookstore()
How many days have you had this book for? 234
Your total amount due is $55.
```

```
>>> bookstore()
How many days have you had this book for? 15
Your total amount due is $0.25.
```

Monthly Allowance

Function Name: monthlyAllowance()
Parameters: N/A
Returns: None
Description: It's the end of the month and you've finally received your monthly allowance. You want to know how much money you'll have left to spend after setting aside some percentage of your allowance and then paying your monthly subscription fees. Your monthly Netflix and Spotify subscription fees are $5.50 and $3.99 respectively. Write a function that asks the user what their monthly allowance is and what percentage of their allowance they want to save. Print how much money they'll have left over after taking into account savings and subscription fees. **All inputs will be positive integers. Round all outputs to two decimal places.**

```
>>> monthlyAllowance()
How much is your allowance this month? 1500
What percentage do you want to save? 70
You have $440.51 left after savings and fees.
```

```
>>> monthlyAllowance()
How much is your allowance this month? 350
What percentage do you want to save? 60
You have $130.51 left after savings and fees.
```

Grading Rubric

Function	Points
bake()	20
cakeVolume()	20
celebrate()	20
bookstore()	20
monthlyAllowance()	20
Total	**100**

Provided

The `HW01.py` skeleton file has been provided to you. This is the file you will edit and implement. All instructions for what the functions should do are in this skeleton and this document.

Submission Process

For this homework, we will be using Gradescope for submissions and automatic grading. When you submit your `HW01.py` file to the appropriate assignment on Gradescope, the autograder will run automatically. The grade you see on Gradescope will be the grade you get, unless your grading TA sees signs of you trying to defeat the system in your code. You can re-submit this assignment an unlimited number of times until the deadline; just click the "Re-submit" button at the lower right-hand corner of Gradescope. You do not need to submit your `HW01.py` on Canvas.