CSCI 141 - Spring 2019
Lab 6: Functions and Function Composition
Due Date: See Canvas

Introduction

By this time in the quarter you've seen functions in lecture, in lab, and from here on they will be required in the coding portions of the homework assignments, including A4 which is due this week.

In this lab, you'll write a program that calculates the total square footage in a home. Imagine you are working for a construction company that builds homes with only square or rectangular rooms. The houses vary in size, with as little as 1 room and up to 100s. **Your task is to write a program that will prompt the user to input the number of rooms and their dimensions, and which then calculates the total square footage of the home**. Two invocations of such a program are shown below.

```
What is the number of rooms? 3
What is shape of room 0? square
What is side length of the room? 45
What is shape of room 1? rectangle
What is length of the the room? 30
What is width of the room? 37
What is shape of room 2? rectangle
What is length of the the room? 22
What is width of the room? 17
Total square footage: 3509

>>>
```

```
What is the number of rooms? 4
What is shape of room 0? square
What is side length of the room? 10
What is shape of room 1? square
What is side length of the room? 15
What is shape of room 2? square
What is side length of the room? 26
What is shape of room 3? rectangle
What is length of the the room? 21
What is width of the room? 56
Total square footage: 2177

>>>
```

Figure 1: Two sample invocations

The figure on the left is the program running where a user specifies a house with 3 rooms – a square room and two rectangle rooms. The image on the right is the calculation for a house with 4 rooms – three square rooms and a rectangular room.

You'll create such a program from scratch. You'll be asked to follow good coding conventions – if you find yourself typing similar code over and over again, create a function for performing that repeated task (as was shown in lecture).

Look at the above program invocations, and notice the following:

- To calculate the size of a square room, you need only the dimension of one of the walls

- To calculate the size of a rectangular room, you need both dimensions, the width and length

- There are repeated print statements that are very similar, but not identical. However, they all begin with "What is the", and end with a question mark.

From these observations, your supervisor (boss) has decided that your program can have ONLY 3 functions:

1. `room_square_feet`, which takes a single argument to designate whether the room is a square or rectangular room. If the room is a square room, the function `room_square_feet`

invokes the `prompt_user` function once to retrieve the single dimension needed to calculate the size of the room. If the room is a rectangular room, the function `room_square_feet` invokes the `prompt_user` function twice to retrieve the two dimensions needed to calculate the room's size. The function `room_square_feet` returns the square footage of a room.

2. `main`, which prompts the user for the count of rooms and calls `room_square_feet` for each room, using the accumulator pattern (refer to Section 6.4 of the textbook) to keep a running total of the square footage of the house.

3. `prompt_user`, which receives two arguments and returns one value. The first parameter, of type `str`, is the text of the question that the user is prompted. In the examples above, "length of the room" and "width of the room" and "shape of room 1, square or rectangle" would be the three Strings that are provided as input on three separate occasions to the function `prompt_user`. The second argument designates the return type of the function. **Hint:** use type conversion function(s) to convert the user's input into the appropriate type of data to be returned. A diagram showing two example usages of the `prompt_user` function for two sample inputs is included below.
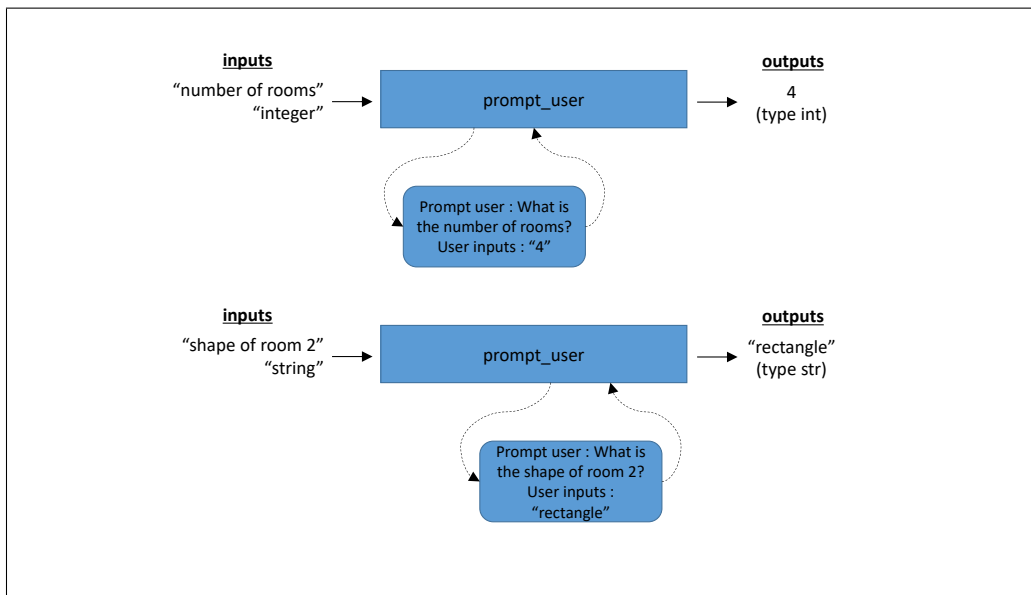
1 Setup

Create a `lab6` folder and in it create a Python file called `squarefootage.py`.

2 Example Program Flow

The steps below illustrate an example of the program's behavior when prompting for the information and calculating the square footage of a single room. Recall that your main program will do this once for each room in the house. Notice that `main` uses `prompt_user` twice, then calls `room_square_feet`, which itself calls `prompt_user` once or twice depending on the shape of the room.

Step 1: `prompt_user` is invoked with first and second arguments "shape of room 1" and "string" both of type string.

Step 2: `prompt_user` prompts the user with the question "What is the shape of room 1?"

Step 3: The user responds to the prompt and types "square"

Step 4: `prompt_user` returns the string "square", which is then passed to the function `room_square_feet`

Step 5: The function `room_square_feet` calls `prompt_user` with two arguments, "side length of square room" and "integer"

Step 6: The function `prompt_user` prompts the user with the question "What is the side length of square room"

Step 7: The user responds to the prompt and types "12"

Step 8: The function `prompt_user` returns the integer 12, which is saved in a variable in the function `room_square_feet`

Step 9: The function `room_square_feet` calculates the size of the room $(12 * 12)$ and returns the integer 144.

3   Game Plan and Guidelines

- All three functions are fairly simple, taken by themselves. You should be able to write each one in less than about 10 lines of code.

- Start by writing a header and specification for the `prompt_user` function based on the description above. Then implement the function and test it using the interactive shell. Test it for both integer and string output types.

- Write a header and specification for `room_square_feet`. Implement it, using your already-tested `prompt_user` function, and test its functionality in the interactive shell.

- Write the `main` function, noting that it takes no arguments and has no return value; when the total square footage is calculated, *print* the total.

- Finally, at the bottom of your program, call your `main` function in a main guard so that running the program calls the main function:

```
if __name__ == "__main__":
    main()
```

- As usual, you are **not** required to check for and handle input from badly behaved users: you may assume the user enters numbers when they're supposed to, and that they enter either `square` or `rectangle` for the shape of each room.

- We test your programs with automated scripts, so the prompts must be made in exactly the order shown in the sample outputs to receive full credit. The text of the prompts does not need to match exactly, but the order must match.

Submission

Submit `squarefootage.py` via Canvas.

Rubric

Your file is called `squarefootage.py` and has a comment specifying author, date, and purpose at the top.	1 points
Your program uses good, descriptive variable names	2
`main`, `room_square_feet`, and `prompt_user` have headers and docstrings consistent with the description in the lab handout	6
Each function is at most 10 lines of code (not counting comments)	6
`prompt_user` prompts the user and returns the correct type	4
`room_square_feet` uses `prompt_user` to prompt for the dimensions	2
`main` uses `prompt_user` to specify how many rooms	2
`main` calls `prompt_user` to prompt for shape of the room	2
`main` calls `room_square_feet` for each room and the return value is added to an accumulator variable	2
`main` function prints correct final square footage	3
Total	30 points