13 Teach: Team Activity

Areas of Shapes Revisited

Instructions

Face-to-face students will complete this activity in class.

Online students should arrange for a one hour synchronous meeting to work through the activity together.

Overview

Earlier in the course, you completed a team activity to compute the areas of squares, rectangles, and circles. Please refer to 03 Teach: Team Activity for the details of that activity.

For this assignment, you are going to repeat the earlier calculations, but this time, you will make three functions, one to calculate each of the areas.

Assignment

Write functions to compute and return the areas of squares, rectangles, and circles. These functions **should not** display the values directly, but rather should return them, so they could be used in other parts of the program.

I. CORE REQUIREMENTS

Work through these core requirements step-by-step to complete the program. Please don't skip ahead and do the whole thing at once, because others on your team may benefit from building the program up slowly.

01. Write a function **`compute_area_square`** that accepts a single value as a parameter, and then computes the area and returns it.

    Below the function, write code to prompt the user for the side of a square and save it into a variable, then pass this variable to the function to compute the area. Finally, get the result back from the function and display it.

02. Repeat the previous step to write and test the functions **`compute_area_rectangle`** and **`compute_area_circle`**.

03. Write a loop to ask the user what kind of shape they have, then prompt for the length of a side, or sides, or radius, and then calls the appropriate function, and displays the result. The program should continue looping until the user enters "quit" for the shape.

II. STRETCH CHALLENGE

01. Recognize that you can compute the area of a square by passing the task along to a function that computes the areas of rectangles, by giving it the side of the square twice.

    Change your program so that the **`compute_area_square`** function doesn't compute the area directly, but instead calls the **`compute_area_rectangle`** to do the work. It should pass the square side length to it (twice) and then return the value that the **`compute_area_rectangle`** function computes.

02. Write a new function called **`compute_area`** that accepts a first parameter of **`shape`** that can be either "square" or "circle" and then a value for the length of the side or the radius depending on the context. The function should then determine the correct function to use, based on the first parameter, call it, and return the value.

    Change your program to use this function for the square and circle cases and verify that it works.

03. Add the ability for your new **`compute_area`** function to also compute the areas for rectangles. This means that you'll have to be able to accept calls like this: **`compute_area("circle", 5)`**, this: **`compute_area("square", 10)`**, and this **`compute_area("rectangle", 7, 8).`**

    In order to make this work correctly, you'll need to make use of a default value for the last parameter.

    Change your program to use this for all three calculations and verify that it works.

Sample Solution

When your program is finished, please view the sample solution for this program to compare it to your approach.

You should work to complete this team activity for the one hour period first, without looking at the sample solution. However, if you have worked on it for at least an hour and are still having problems, you may feel free to use the sample solution to help you finish your program.

- Sample solution (Core requirements): teach13_sample.py
- Sample solution (Stretch challenges): teach13_stretch_sample.py

Submission

When complete, please report your progress in the associated I-Learn quiz.

If you decided to do additional work on the program after your team activity, either by yourself or with others, feel free to include that additional work when you report on your progress in I-Learn.