# Lab: Functions

**Note:** This lab is adapted from S. Linge and H. P. Langtangen (2020). Licensed under the terms of the [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/deed.en) (https://creativecommons.org/licenses/by/4.0/deed.en).

## Background

A function is a block of code that does a specific task after calling it. You can pass data as parameters to it, and it can return a value.

Lab questions are designed to assess your comprehension of the course materials covered in this unit, therefore questions should be solved using **only** the information provided in the course materials up to the end of Unit 3-1. Use of materials outside of this will result in 0 on the lab question.

## Instructions

Use Python IDE to create a solution for the scenario presented in each question.

#### Functions for circumference and area of a circle

Write a program that takes a radius of a circle as input from the user and then computes the circumference and area of the circle. Implement the functions circumference() and area() that each have radius as a parameter. Write a main() function that calls these functions and produces a report as shown in the sample run. Note: Use math.pi.

Sample Run: (user input in bold underline)

Give the radius of a circle: **5**  
Circumference: 31.416, Area: 78.540

#### Simple calculator

Take a copy of the simple calculator (with looping) from Unit 2. Change the program to use functions. You will need to write the following functions:

menu() – prints the menu and retrieves a menu option from the user. This function should return the menu option once a valid choice is entered, i.e., loop until a valid menu option is entered.

add() – receives two numbers as arguments (addends) and returns the sum

subtract() - receives two numbers as arguments (minuend and subtrahend) and returns the difference

multiply() – receives two numbers as arguments (multiplicand and multiplier) and returns the product

divide() - receives two numbers as arguments (dividend and divisor) and returns message ‘Cannot divide by 0’ if is divisor is 0, otherwise returns the quotient

main() – controls the flow of the program

Sample Run: (user input in bold underline)

Simple Calculator

1. Add

2. Subtract

3. Multiply

4. Divide

0. Exit

Enter menu option: **4**

Enter first number: **5.1**

Enter second number: **1.7**

5.1 / 1.7 = 3.0

Simple Calculator

1. Add

2. Subtract

3. Multiply

4. Divide

0. Exit

Enter menu option: **4**

Enter first number: **5.1**

Enter second number: **0**

5.1 / 0.0 = Cannot divide by 0

Simple Calculator

1. Add

2. Subtract

3. Multiply

4. Divide

0. Exit

Enter menu option: **1**

Enter first number: **5.1**

Enter second number: **1.7**

5.1 + 1.7 = 6.8

Simple Calculator

1. Add

2. Subtract

3. Multiply

4. Divide

0. Exit

Enter menu option: **2**

Enter first number: **5.1**

Enter second number: **1.7**

5.1 - 1.7 = 3.3999999999999995

Simple Calculator

1. Add

2. Subtract

3. Multiply

4. Divide

0. Exit

Enter menu option: **3**

Enter first number: **5.1**

Enter second number: **1.7**

5.1 \* 1.7 = 8.67

Simple Calculator

1. Add

2. Subtract

3. Multiply

4. Divide

0. Exit

Enter menu option: **7**

Enter menu option: **f**

Enter menu option: **0**

Calculator app closed

**Submission**

For each question, submit your Python source code (.py file) and your test results (.txt file containing output from the Terminal Window after running your Python program) to the Brightspace submission folder for this lab assignment.

**Reference**

Linge, S. and Langtangen, H. P. (2020). Programming for computations – Python: A gentle introduction to numerical simulations with Python 3.6. (2nd ed.). Springer Open. ([CC BY-SA 4.0](https://creativecommons.org/licenses/by/4.0/deed.en)). Retrieved from https://library.oapen.org/viewer/web/viewer.html?file=/bitstream/handle/20.500.12657/23103/1007055.pdf