

ENGG1003 - Lab 2

Brenton Schulz

1 Library Basics

This lab will be your first official introduction to using Python *libraries*.

A library is a collection of *functions* (blocks of code referenced by name with specific input, functionality, and output) which perform tasks beyond those “built in” to the Python language.

For example, Python has no built in way of calculating square roots so this mathematical operation is implemented in the `sqrt()` function from the `math` or `numpy` libraries.

Libraries can also contain data in addition to functions. For example, the `math` library (used in the task below) contains the numerical constant π , accessed by using the text `pi`.

Task 1: Math Library Example

Read through Section 1.3 of the textbook (https://link.springer.com/chapter/10.1007/978-3-030-16877-3_1#Sec9).

Be sure to observe the (intentional!) error which results from running the first code example

The full documentation for all `math` library functions can be read here: <https://docs.python.org/3/library/math.html>

2 Importing Library Functions

Python allows for several different ways to *import* libraries, each with their own *syntax* and behaviour.

Task 2: Importing Libraries

Read through Section 1.4 of the textbook https://link.springer.com/chapter/10.1007/978-3-030-16877-3_1#Sec10

Note the difference between:

- `from library import things`
- `from library import *`
- `import library`
- `import library as name`

Section 1.4.5 lists the libraries used by the textbook. Some of them (like `math`) are installed by default while others need to be explicitly installed.

3 Installing Libraries

Normally you would only install the libraries needed for a particular project. It is also best practice to limit the installation of a library to a *virtual environment*. The reasons can be complex, but you should be aware that libraries are often under constant development and their behaviour can change as updates are released. Containing each Python project inside a virtual environment means that each project can use different versions of each library. This is done so that code written in the past will continue to work as new updates are released because that project's virtual environment only contains libraries known to work with that code.

Task 3: Installing Libraries

Install the `numpy` and `matplotlib` libraries in your PyCharm project.

To do this, click on the “Terminal” tab (to the left of the “Python Console” tab) and execute the following commands:

- `pip install numpy`
- `pip install matplotlib`

Test that the installation was successful by running `import numpy` and `import matplotlib` in the Python console.