# Lecture 4: Functions and modular code

Dr Benjamin J. Morgan[1] and Dr Andrew R. McCluskey[1,2]

[1] *Department of Chemistry, University of Bath, email: b.j.morgan@bath.ac.uk*
[2] *Diamond Light Source, email: andrew.mccluskey@diamond.ac.uk*

July 29, 2019

## Aim

This lecture will introduce how to simplify your code by writing functions and how to reuse code easily in many different places.

## 1 Functions

Occasionally, there is a particular section of your code that you would like to reuse over-and-over, without having to write the code over and over (as previously mentioned, programmers are lazy). For this, we make use of *functions*, the use of which will be familiar. For example, we have used the `print()` function already in previous weeks, and the NumPy library contains a wide variety of functions, some of which were introduced last week. However, this week we shall see how it is possible to write our own functions in Python.

The general syntax for *defining* a function in Python is as follows,

```
# Defining a function

def my_function(argument_1, argument_2):
    result = argument_1 + argument_2
    return result
```

Above, we defined a function named `my_function` which took two *arguments*, added them together to produce a result, which was *returned*. Once defined it is possible to use this function in our code as follows,

```
# Using our function

a = 1
b = 2
c = my_function(a, b)

print(c)
```