

3.9: Parameters Walkthrough

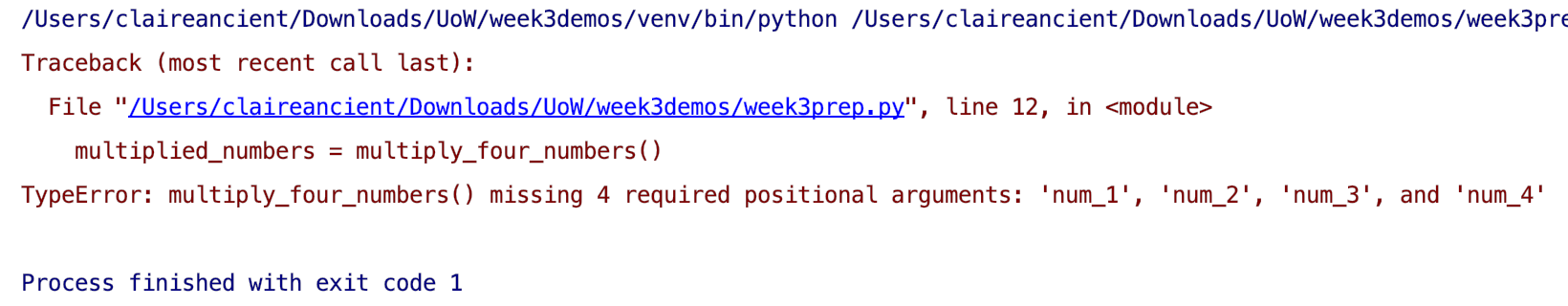
Whilst it is useful to be able to repeatedly run the same function, it is less beneficial if the results are always the same. This is where parameters come into play. Parameters allow us to pass values to a method for use. This allows us to run the same process each time, with various inputs and, therefore, different outputs.

Whilst the multiply\_four\_numbers() function is quite useful, it would be more beneficial to be able to multiply different numbers, not just 5, 6, 7, and 8. So let’s adapt our multiply\_four\_numbers() function to take two parameters. These two parameters will be what we are trying to multiply together.

def multiply\_four\_numbers(num\_1, num\_2, num\_3, num\_4):

return num\_1\*num\_2\* num\_3\*num\_4

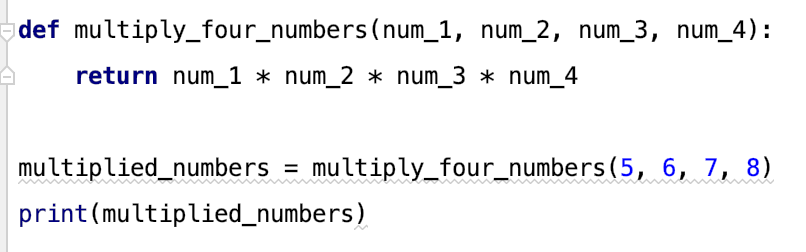
If we run the program now, without changing the code which calls the function, we will get the following error:



This error is caused by the fact that we were trying to call a function which requires four arguments. Arguments are the values which have been entered to be passed through parameters to the function. Let’s update the call for the multiply\_four\_numbers() function to include four arguments:

multiplied\_numbers = multiply\_four\_numbers(5, 6, 7, 8)

The completed function and its call should look as follows:



Now when we run the function, we get the following output:

