CoGrammar

Programming with Functions (Part One)

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

Data Science Lecture Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(FBV: Mutual Respect.)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Open Classes. You can submit these questions here: Open Class Questions

Data Science Lecture Housekeeping cont.

- For all **non-academic questions**, please submit a query: www.hyperiondev.com/support

- Report a **safeguarding** incident: www.hyperiondev.com/safeguardreporting

- We would love your **feedback** on lectures: Feedback on Lectures

Lecture Objectives

- Understanding both built-in and creating our own functions.
- Calling functions and understanding function scope.

CoGrammar

What are functions?

- ★ It is a reusable and organised block of code.
- ★ Sometimes it's called, a 'method'.
- ★ Similar to functions is maths - f(x) takes an input of x and produces an output.
- ★ Also useful for abstraction.
  - ○ Abstraction - the concept of defining complex functionality by using a single term. Great for defining high-level bits of functionality.

Functions in Python

★ Python does come with built-in functions bundled alongside it. For example :
  - print()
  - input()
  - Both of these are staple examples of built-in functions that come with python.

CoGrammar

More Python Functions

★ There are many more functions that we can use in Python and it does not stop with what is built-in.

★ We can use something called Pip (python package manager) to install various packages that contain modules.
  - Note : Some packages come preinstalled, such as the math module.

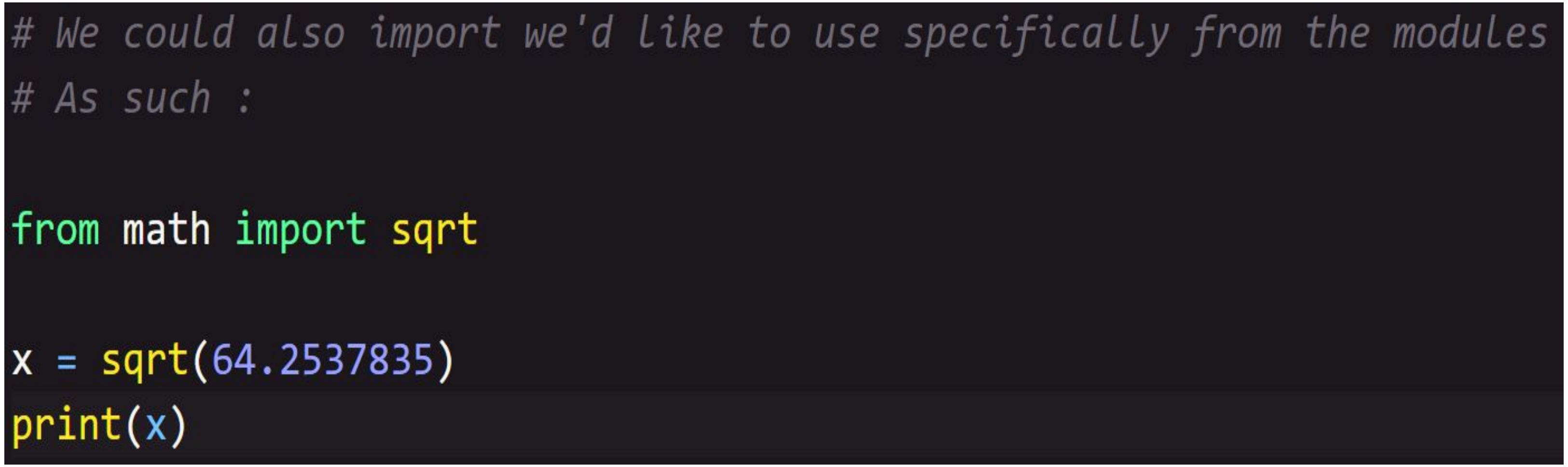
★ These modules can be imported into our scripts using the import statement.

Importing modules

```python
# Remember to always import your modules before you begin.
# It'd be awkward if you call a module that you have not referenced yet.

import math

x = math.sqrt(64.25673)
print(x)
```

CoGrammar

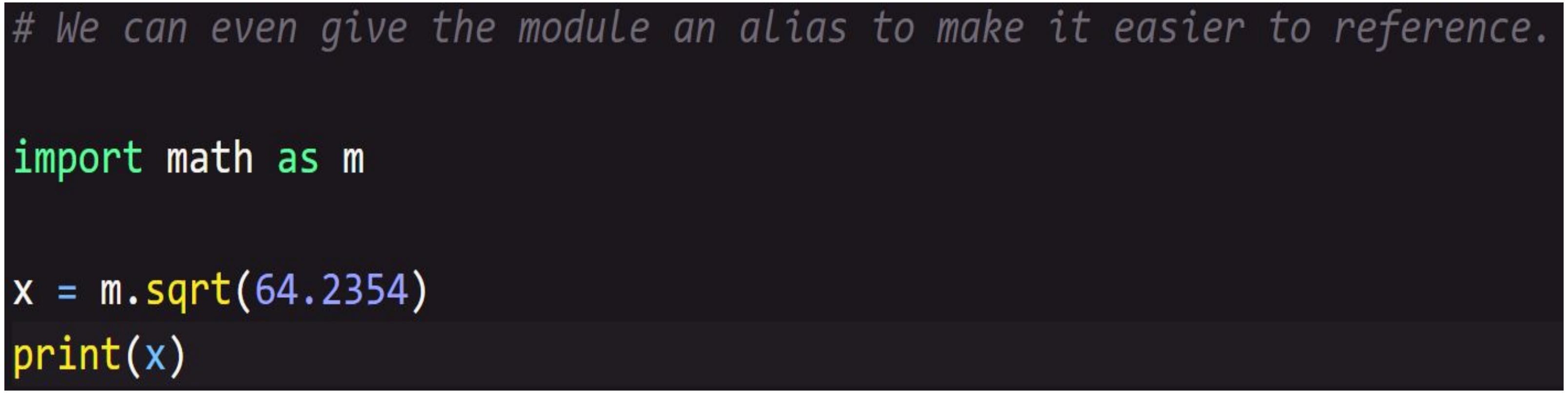
Importing Modules



```
# We could also import we'd like to use specifically from the modules
# As such :

from math import sqrt

x = sqrt(64.2537835)
print(x)
```

CoGrammar

Importing Modules



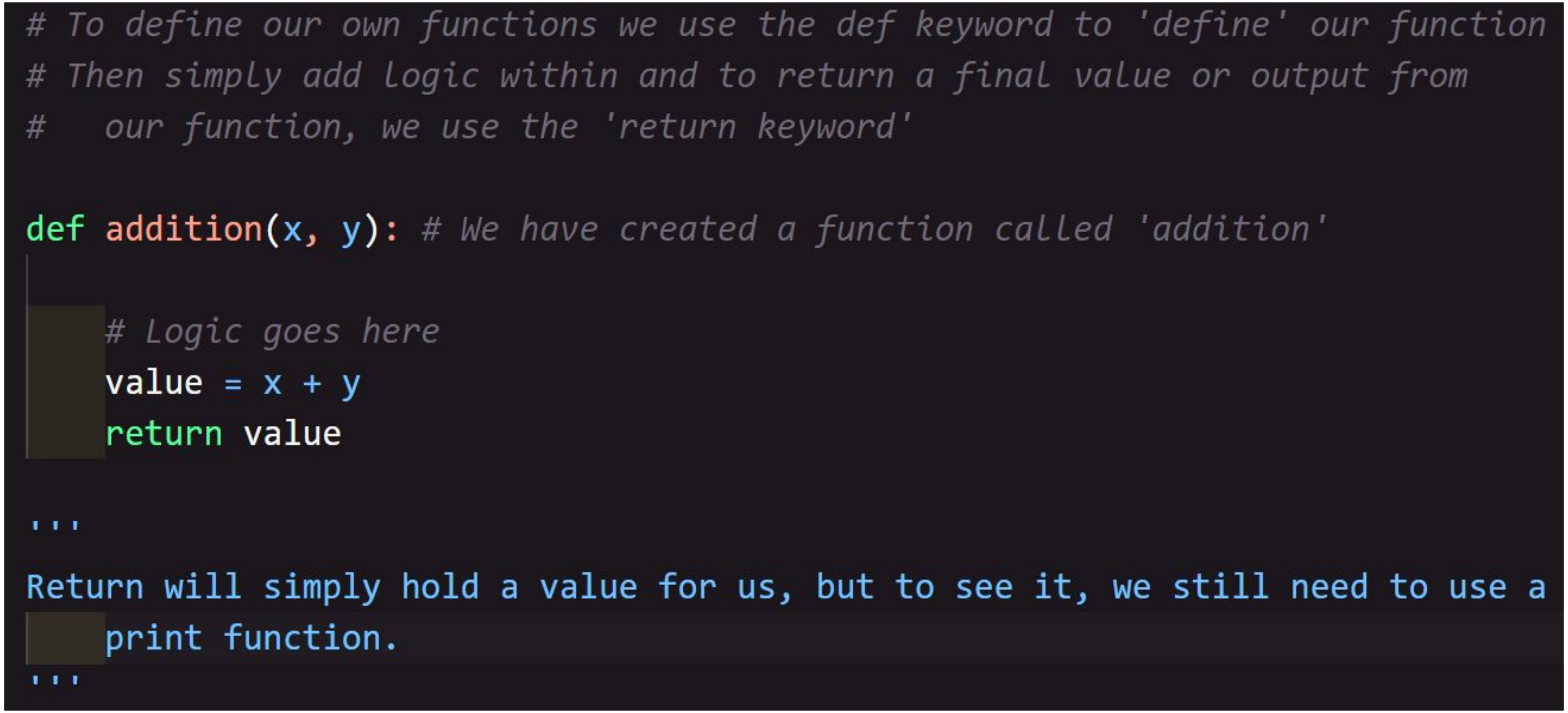
```python
# We can even give the module an alias to make it easier to reference.

import math as m

x = m.sqrt(64.2354)
print(x)
```

Let’s Breathe

Let’s take a small break before moving on to the next topic.

CoGrammar

Creating our own Functions



```python
# To define our own functions we use the def keyword to 'define' our function
# Then simply add logic within and to return a final value or output from
#   our function, we use the 'return keyword'

def addition(x, y): # We have created a function called 'addition'

    # Logic goes here
    value = x + y
    return value

'''
Return will simply hold a value for us, but to see it, we still need to use a
    print function.
'''
```

Important Terminology

★ **Function - A block of code that performs an action.**

★ **Method - A function defined or owned by an object. Not quite the same as functions but very similar for our purposes.**

★ **Parameters - The defined input of a function.**

★ **Arguments - The values passed to the parameters.**

Why Functions?

- ★ **Reusable code - Sometimes we'll need to do the same thing multiple times.**
- ★ **Error checking / validation - Makes this process easier, as the logic is placed in one place that is easy to find.**
- ★ **Dividing code up into manageable chunks - Makes our code easier to read and understand.**
- ★ **Rapid development - The same functionality does not need to be defined again.**

CoGrammar

Q & A SECTION

Please use this time to ask any questions relating to the topic, should you have any.

CoGrammar

Thank you for joining!