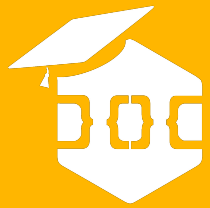

SOC presents

Programming with

Python for Beginners

Why should I bother?

- Save time
- Make cool stuff
- Employability + Degree
- It's fun (maybe?)

Goals:

- **Appreciate basic programming concepts and techniques**
- Understand and write programs in Python
- Become proficient enough to independently learn further material
- Have a little bit of fun!

Goals:

But more importantly, we will teach you how to think like a programmer!

How will I learn?

- We will teach you the syntax, concepts and features of the language.
- We will give you the opportunity to have a go and test your knowledge during the sessions, and ask our volunteers questions.
- You should practice in your own time! :)

What will I need?

- A laptop
- Python 3.x (preferably 3.6) installed on your laptop
- A text editor e.g. Atom (free), Visual Studio Code (free - suggested), Sublime text (not free), etc.
- A little bit of patience and dedication :D

1

Lecture 1

The Basics

What is Python?

"Python is a dynamically typed, interpreted, high-level and general-purpose programming language" - Wikipedia

Your first program

```
print("Hello World!")
```

Running your first program

MacOS/Linux

- Terminal
- ‘pwd’ to show current folder
- ‘cd’ to change directory
- ‘ls’ to show contents of folder

Windows

- <--- Powershell or cmd
- <--- Same
- <--- Same
- <--- ‘dir’

- ‘python3’ to run the program

Try:

- Printing out numbers
- Doing some maths
- Printing out multiple lines
- Printing out your name, then your age in years/months/days/hours.....

Variables

```
x = 5
print(x)
```

Variables


```
x = 5
print(x)
```

Try:

- Assigning and printing out multiple variables
- What happens if you don't assign x but try to print it anyways?
- Re-assigning a variable

Messing up variables

```
a = 3
b = "Some text"
c = a + b
```

Types

Every variable has a type! These include:

- Numbers (Integers, Floats (real numbers), etc)
- Strings
- Boolean (True, False)
- Tuples, Lists, Dictionaries
- Classes (Coming soon!)

Types

Every variable has a type! These include:

- Numbers (Integers, Floats (real numbers), etc)
- Strings
- Boolean (True, False)
- Tuples, Lists, Dictionaries
- Classes (Coming soon!)

They don't always work well with each other!

Getting Types to work together

```
a = '1'
b = 2
c = int(a) + b
```

Getting Types to work together

```
int(a)
float(a)
str(a)
```

These functions *try* to do the conversions

Maths

```
x = 5
print("x is currently:", x)
x = x + 5
print("x is now:", x)
```

Maths

```
a = 3
b = 2
c = a + b
c = a - b
c = a * b
c = a / b
c = a ** b
```

Try:

- Just having a play around
- Dividing by Zero
- Add two strings together

Getting User Input

```
print("Hello, what's your name?")
x = input()
print("Nice to meet you", x)
```

Getting User Input

```
print("Please enter a number")
x = input()
print(x * 2)
```

Getting User Input

```
print("Please enter a number")
x = input()
print(x * 2)
```

What went wrong?

Why?

How do we fix this?

Getting User Input

```
x = input("Please enter a number:")
print(int(x) * 2)
```

Comparison and Boolean operators

- a > b
- a < b
- a >= b
- a <= b
- These are comparison operators for *a and b of (almost) any type.*

- 'a and b'
- 'a or b'
- 'not a'
- These are boolean operators - *they only work on a, b booleans*

- Both comparison and boolean operators produce a result of boolean type

Try (in the interpreter):

- ‘abc’ > ‘cba’
- ‘abc’ > ‘Cba’
- ‘Hello’ > ‘Hello, World!’
- “Hello” and “World”
- “Hello” or “World”
- False or 5
- Have a play around!
- Can you figure out the rules?

Control Flow

```
x = int(input("Please enter a number:"))
if x > 10:
    print("Your number is bigger than 10")
```

Control Flow

```
x = int(input("Please enter a number:"))
if x > 10:
    print("Your number is bigger than 10")
elif x < 10:
    print("Your number is smaller than 10")
```

Control Flow

```
x = int(input("Please enter a number:"))
if x > 10:
    print("Your number is bigger than 10")
elif x < 10:
    print("Your number is smaller than 10")
else:
    print("Your number is 10")
```

Challenge #1:

- Write a simple program that “checks” the user’s response.
- For example, have the program call out (print) “Marco”, and the user must respond. If the response is exactly “Polo”, then the program responds with something nice. If the response is incorrect, tells the user to go away.
- BONUS: Ignore upper-case/lower-case errors, so allow ‘polo’, ‘POLO’, ‘pOlO’.....etc