Variables and Operators

can code
communities

Variables

Variables are places we can store information.

You should give variables **short** but **meaningful** names.

In our variable names we can have letters, numbers, and underscores, although we can only start a variable's name with a letter or underscore.

Although, starting with an underscore or capital letter means certain things.

- We can see this in PEP-8 which we will talk about later.
- It's safest to start your variable names with **lowercase letters.**

Arithmetic Operators

+	Addition
-	Subtraction
*	Multiplication
/	Division
**	Exponents
//	Integer Division
%	Modulo/mod/remainder

Keep in mind

Order of operations: PEMDAS

- Parentheses
- Exponents
- Multiplication/Division and Mod, for Python
- Addition/Subtraction

Use parentheses for clarification if need

Example

Write some Python code that calculates the perimeter of a rectangle based on the length and width.

Exercise

Write some code that converts fahrenheit to celsius.

To convert fahrenheit to celsius, subtract 32, then multiply by 5/9.

C = (F - 32) * 5/9

Data Types

Bool (Boolean) - True or False

Int (Integer) - Negative and Positive Whole Numbers

Float (Floating Point) - Numbers with a decimal point

Strings - Text, surrounded by quotes

Lists - Group of Objects (Talk about later in course)

Dictionaries - Group of Objects with Keys (Talk about later in course)

There are many more data types.

Discussion: Why do we need different data types?

Floating Point Accuracy

Floats are sometimes not completely accurate, and you might see small errors in your code when using them.

This is because every number is represented in **binary** in your computer, and some floating point numbers are infinitely repeating fractions in binary.

For example, the number 0.1 is the fraction 1/10. In binary, this is:

```
0.0001100110011001100...
```

Since you must stop at a finite number of bits, the number gets approximated.

More info: https://docs.python.org/3/tutorial/floatingpoint.html

Scientific Notation

Python automatically displays values in scientific notation if they are bigger than `1*10^15` or smaller than `1*10^-4`.

You can also replace the `10` with an `e`

The number `123000000000000000 = 1.23*10^17`,
and Python prints it as `1.23e+17.`

Try running this line of code:
`print(0.000000001)`

What do you think the output will be?

PEP 8

We use PEP to make sure that our code is readable by others.

We also use it to standardise code so we can understand it in the future ourselves.

Some IDE's will auto complete for PEP or let you know when you are breaking the rules. The other way is by reading the style guide.

https://www.python.org/dev/peps/pep-0008

PEP-8 will apply to every topic we cover in this class.

PEP 8 Variable Names

https://www.python.org/dev/peps/pep-0008/#naming-conventions

In Python, we usually use `snake_case` to name variables (lowercase with an underscore separating each word)