Introduction to Coding and Python

Hawken Coding Club
Learning Meeting No. 1

What is coding and what is it used for?

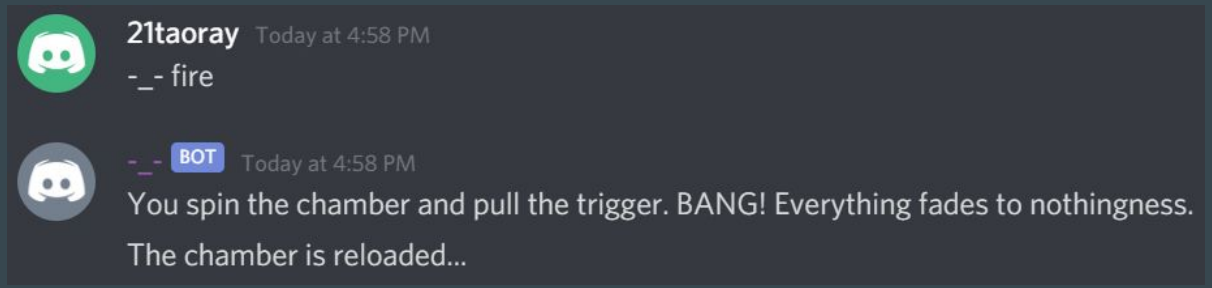
- Coding is giving a computer instructions to get it to do something that you want.
- Coding has a variety of uses, such as, making websites, making apps, managing data, and so much more.
- Coding can be used in your everyday lives or on a professional level.

Why are we teaching Python?

- Python is one of the most all around languages that is commonly used. It can be used for almost everything, which makes it an excellent choice to learn first.
- Python is one of the most forgiving languages. This means that even if you make a mistake, sometimes python will ignore the errors and still be able to run your code.
- Python also is a procedural language, this means that you can start coding right away without making classes or methods.

What can you build with Python?

- Websites (using Django or Flask)
- Artificial intelligence (using TensorFlow or PyTorch)
- A Discord bot! (using discord.py)




```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import os
from dotenv import load_dotenv
import logging

from discord import Embed, Status
from discord.ext import commands
from discord.ext.commands import Cog, Context

from types import ModuleType

# get token as environment variable
load_dotenv()
TOKEN = os.getenv('DISCORD_TOKEN')
PREFIX = os.getenv('COMMAND_PREFIX') + ' '
DEBUG_CHANNEL_ID = int(os.getenv('DEBUG_CHANNEL_ID'))

logging.basicConfig(level=logging.INFO)


class Bot(commands.Bot):
    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)

    async def on_connect(self):
        logging.info('Connected to Discord.')

    async def on_ready(self):
        logging.info(f'Logged in as {bot.user}.')

    def add_cog(self, cog: Cog) -> None:
        super().add_cog(cog)
        logging.info(f"Added cog: {cog.qualified_name}")
```

Other Languages

There are many programming languages.

Each has its own use and features.

Other Languages

- Java
- JavaScript
- C#

Java is a great all around language that has many uses such as app development and managing databases.

Java's biggest feature is *platform independence*. That means that a Java app can run on your computer, on the internet, and on Android phones.

Minecraft, the world's most popular video game, was originally programmed in java.

Other Languages

- Java
- JavaScript
- C#

JavaScript is a quick prototyping language that is often used to add some automation and flair to websites. Although many coders often think of Javascript as being strictly limited to websites, JavaScript has recently been expanded to all sorts of fields.

Other Languages

- Java
- JavaScript
- C#

C# is a compiled language which can be used to develop .exe applications, operating systems, and basic command line interfaces. It is one of the most generally used languages, but is slowly being phased out.

C# is used for the Unity game engine.

C# is backed by Microsoft, so it works well with their products.

Setup!

Where can you program?

- Usually, you need to install some software to run code.
- Code can be written in any text editor, like Notepad.
  Most programmers use a special code editor or an Integrated Development Environment (IDE).

To get going quickly, we'll use an online environment instead.

Jupyter Notebooks

- **Jupyter Notebook** is a tool that allows you to combine text and code.
- Jupyter Notebooks are used by professionals to share tutorials and code snippets. They are not used for larger programs.

We will use Jupyter Notebooks through **Google Colab**.

Google Colab is a free service by Google that hosts Jupyter Notebooks for you.

The notebooks are stored in Google Drive and run on Google's servers.

Setup

1. Go to this link: https://colab.research.google.com/
2. Click on “Google Drive”, then click “New Notebook”
3. (there is no step 3, you’re done)

Code can be typed right at the cursor.

To run your code, press Ctrl + Enter or click the play button.

We will demonstrate this now.

Hello World!

Learn to make your first program in python

The `print()` Function

- To print “text”, type the statement **`print("text")`**
- “Quotes” are required to output **text**, but nothing is required to output non-text items (integers and true or false)

- Run the code by clicking the play button to see what it outputs

Variables - 1. Basics 2. Types 3. Manipulation

Variables store data to be used later.

Think of variables like a labeled box.

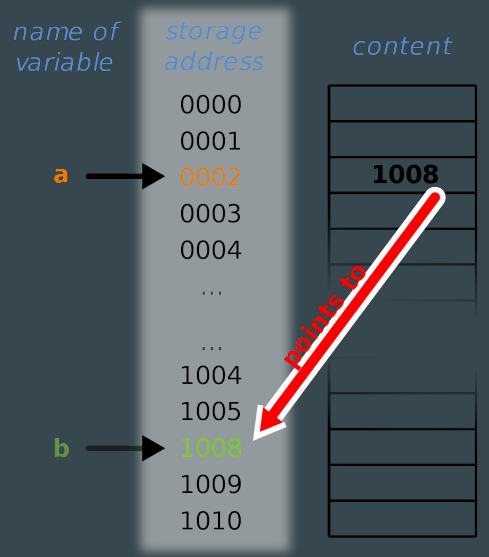
You can put an object in the box.

```
label = object
```

You can access the object using the box's label.

```
print(label)
```

For now, each box can only hold one thing at a time.



Variables - 1. Basics 2. Types 3. Manipulation

There are four types of objects. A variable can hold objects of any type.

`int`	integers (numbers)	`my_integer = 8`
`float`	floats (decimals)	`my_float = 8.55`
`str`	strings (characters/text)	`my_string = "Hello, world!"`
`bool`	booleans (`True` or `False`)	`my_boolean = True`

Variables - 1. Basics 2. Types 3. Manipulation

You can assign a variable to something new whenever you want.

You can even assign the variable to an object of a different type.

```
counter = 1

counter = “2”
```

However, the new value replaces the old value. The old value is lost.

`print(counter)` prints `2`

Combining Strings, Integers, and Variables

“String” + “Integer” + Variable

Example:

Input:

- `print(“Hello, “ + “ how are you?”)`
- `print(“I am “ + “15” + “ years old.”)`
- ```
  a = “Alex”
  print(“Hello, “ + a + “, it’s nice to meet you.”)
  ```

Output:

- Hello, how are you?
- I am 15 years old.
- Hello, Alex, it’s nice to meet you.
  ```

Math in Python

Python also has multiple math operators built-in:

- **Addition (+) ex.** `4 + 5`
- **Subtraction (-) ex.** `4 - 5`
- **Multiplication(*) ex.** `4 * 5`
- **Division(/) ex.** `4 / 2`
- **Exponents (**) ex.** `4 ** 2`

Try running this code:

```
x = 4 + 5

print(x)
```

There are even more math functions built into other packages of Python. We will learn this later.

YOUR FIRST CHALLENGE

Make two variables named **`age`** and **`DAYS_IN_YEAR`**.

Set **`age`** equal to your age in years and set **`DAYS_IN_YEAR`** equal to 365.

Make a third variable called **`total`** equal to **`age`** * **`DAYS_IN_YEAR`**.

Now print a string saying "Total Days:" followed by the variable **`total`**

You can print two things on the same line like this:

`print("here's a number", 5)` prints `here's a number 5`

The `input()` Function

- The `input` function is used to get user input.
- This input can be stored in a STRING.
- The `input` function also is used to print text before taking input.

```
name = input("Please enter some info: ")
```

`name` will now equal the user input.

Control Flow

Changing the order your program runs, dynamically.

What is control flow?

Python programs are executed one line at a time, from top to bottom.

However, you can change this order using special statements.

Today, we will learn about conditional statements, or *if statements*.

Later, we will learn about loops and functions.

Conditional Statements

An if statement only runs if the condition is true.

```
[9]  a = 5
     if a == 5:
       print("Yay! I get printed!")
     if a == 6:
       print("No! You won't see me at all!")
```

```
Yay! I get printed!
```

- if
- if-else
- if-elif-else

Conditional Statements

- if
- if-else
- if-elif-else

```
[10] a = 5
     if a == 6:
       print("I still don't get printed...")
     else:
       print("...but that means that I do!")
```

```
...but that means that I do!
```

Conditional Statements

- if
- if-else
- if-elif-else

You can add as many elif blocks as you want!

```
favorite_thing = input("What's your favorite thing?")
if favorite_thing == "C++":
  print("Ah, yes, fellow 133t hacker person.")
elif favorite_thing == "Python":
  print("Yay! Python is fun and powerful, too!")
else:
  print("Okay, glad you enjoy what you enjoy!")
```

```
What's your favorite thing?Python
Yay! Python is fun and powerful, too!
```

FINAL CHALLENGE

Ask the user `"What country do you live in? "`

Store their answer in a string called **country**.

If **country** equals U.S.A, then print “You are an American”

If **country** equals Canada then print “You are a Canadian”

If **country** equals Mexico then print “You are a Mexican”

Finally if neither of those were true print “You are not from North America”

Notes

Click here for notes on today's lesson:

https://colab.research.google.com/drive/1_dBQWNkcJGbHlgs7_eY0V85LcDVl6x0h?usp=sharing