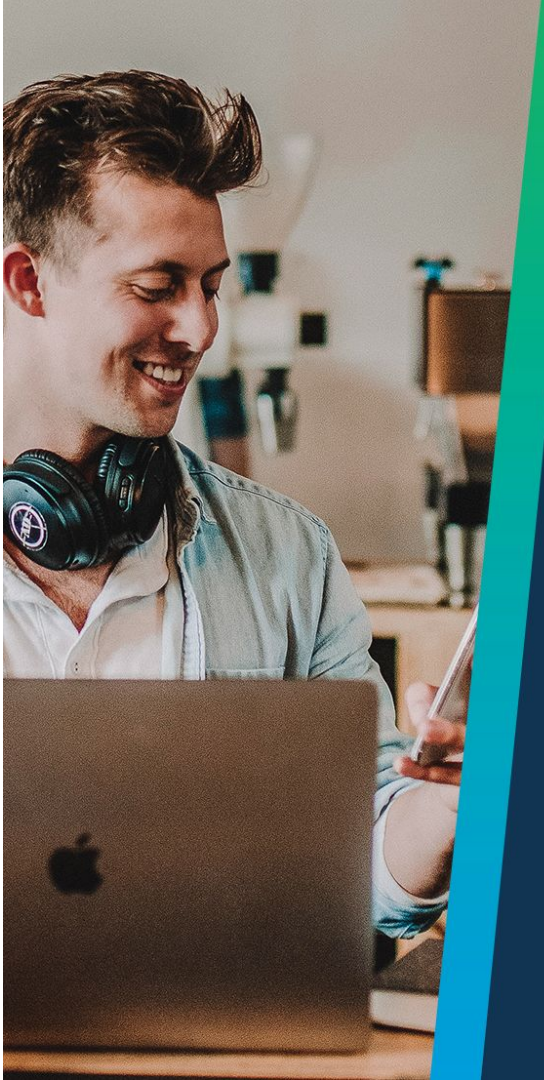

SE Bootcamp

Hyperiondev

Your First Computer Program, and Using Variables

Welcome

Your Lecturer for This Session

Yolandi Viljoen

Lecture Times

Weekdays: 7 pm

Weekends: 11 am

RULES AND REGULATIONS FOR LIVE LECTURES

- The use of disrespectful language is prohibited in the chat.
- Please refer all non-academic queries to support@hyperiondev.com
- You are more than welcome to ask questions in the chat, however, please keep them related to the topic being discussed.
- There is a Q/A session at the end of each session, should you wish to ask any follow-up questions.

Objectives

- Getting acquainted with Python, the powerful, easy-to-learn and popular programming language.

What Is Python?

Python is a widely used, high-level programming language, mainly utilised for general-purpose programming. Python is old, around 32 years old, however, to this day it is still being improved. It is a programming language that has been around long enough, making it a good learning investment.

It was originally created by a Guido van Rossum and first released on the 20th of February 1991. These days, Python is maintained by the Python Software Foundation.

Python Basics

We will be covering the following to become a bit more familiar with the basics of Python:

- ★ The print() function.
- ★ The input() function.
- ★ Variables and variable naming conventions.

The Print Function

- ★ The print() function is used when the output of the program needs to be displayed.
- ★ This is done by entering the print command with an argument, which creates a statement.
- ★ Think of it as such: command + argument = statement.

- ★ Example:

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

The Input Function

- ★ The input() function is a means to receive user input should that be required.
- ★ To achieve this we enter the input command along with the instructions for the user.
- ★ What happens then is that the program will be halted, until it receives input from the user.
- ★ Example:

```
name = input("Please enter your name : ")
```

Note that the variable name stores whatever the user entered into the input as a string. Storing and declaring variables does not produce an output.

Variables

- ★ Variables are a named storage location in memory for values to be stored, e.g. name = “Jimmy”.
- ★ All variables need a descriptive name.
- ★ The value is what the variable stores.
- ★ To create a variable, we first type the name, then equals sign (=), then the value. This is known as variable assignment.

More on Variables

- ★ Variables can be assigned to other variables.
- ★ In Python, the variable's value can be updated as the program runs.
- ★ Several variables can be assigned at the same time in one line.
- ★ Examples:

```
python_is_cool = 100
another_variable = python_is_cool
python_is_cool = 1000
multiple, variables, assigned = 5, 10, 15
```

Variable Naming

- ★ Selecting a good name for your variables is key to making your programs easier to understand.

- ★ Example: a variable tracking a player's health points in a game could be effectively named. health_points, instead of something ambiguous or difficult to understand such as : hp or points.

Variable Naming Rules

- ★ It is vital that variables are given descriptive names that reference the value stored.
- ★ Here are a few rules to follow when naming variables:
    - Variables must start with a letter or underscore.
    - The remainder of the variable can consist of letters, numbers and underscores.
    - Variables are case sensitive, meaning that name and Name are treated as different variables.

Variable Naming Rules cont.

- ★ Keep in mind that Python keywords should not be used as a variable name.
- ★ A reserved word has a fixed meaning and cannot be redefined by the programmer.
- ★ For instance, you would not be able to name a variable print, since Python already recognises this as a keyword.

Variable Data Types

- ★ Data Type:
  - the type of value within a variable.
- ★ Python has several data types, however, we will look into the most common ones which are:
  - Integers
  - Floats
  - Strings
  - booleans

Python Syntax Rules

- ★ All programming languages have syntax rules.
- ★ Syntax means the “spelling and grammar rules” of a programming language.
- ★ Common syntax errors consist of:
  - Not closing the quotation marks (“”).
    - Remember that all quotation marks requires a closing one!
  - Another one is not closing the brackets.
  - Finally, remember that Python code is case sensitive, meaning that print is not the same as Print.
  - Syntax errors will prevent your program from running and display an error.

Hyperiondev

Q & A Section

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

Hyperiondev

Thank You for Joining Us