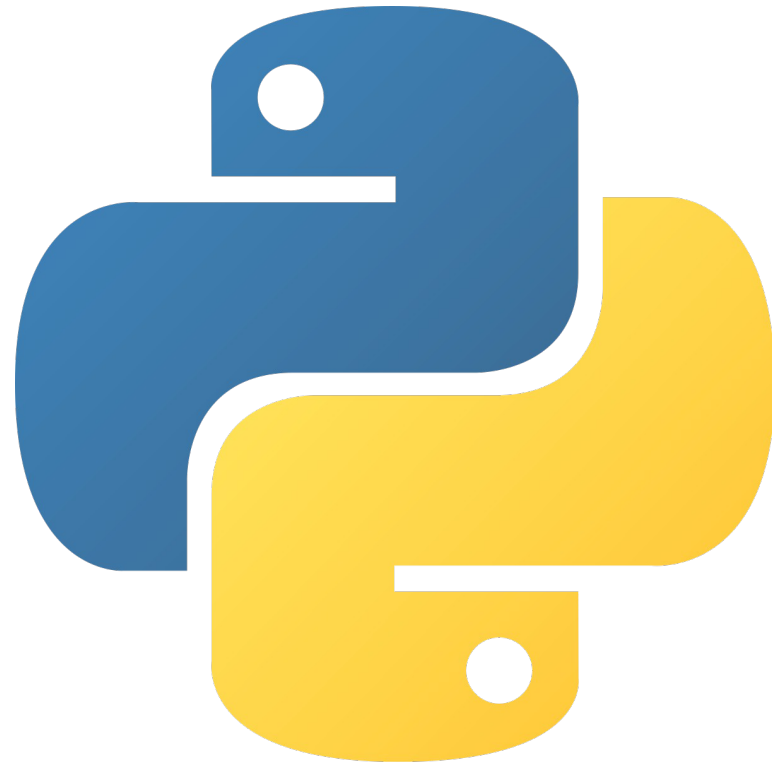# HackerFrogs Afterschool
Python Programming Basics: Intro

Class:
Programming (Python)

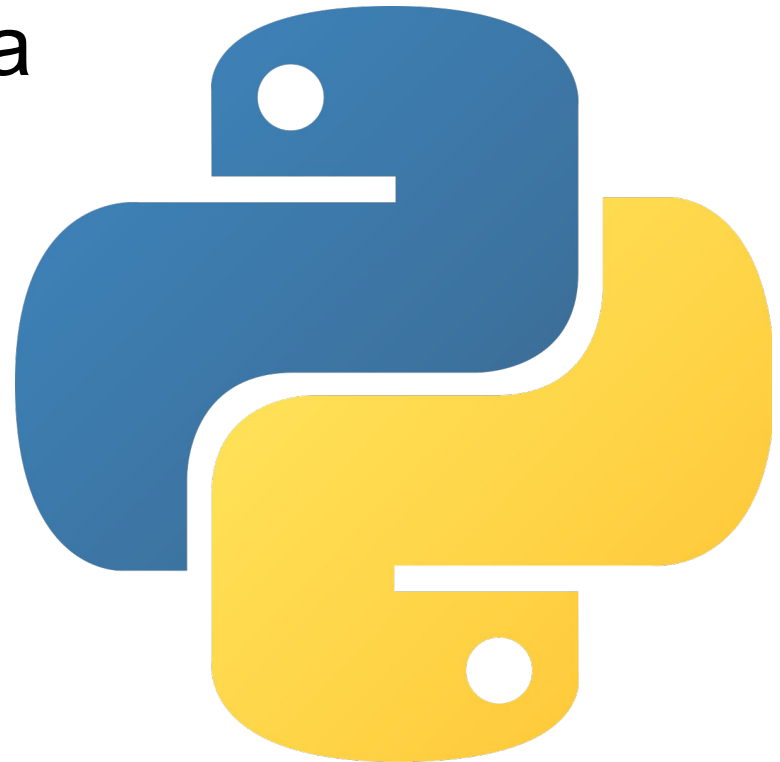Workshop Number:
AS-PRO-PY-00

Document Version:
1.2

Special Requirements:
None

Welcome to HackerFrogs Afterschool!

HackerFrogs Afterschool is a cybersecurity program for learning beginner cybersecurity skills across a wide variety of subjects.

This is the intro to Python programming.

What is Programming?

According to Wikipedia, computer programming is the process of performing a particular computation (or more generally, accomplishing a specific computing result), usually by designing and building an executable computer program.

What is Programming?

Put in broader terms, programming is the process of writing instructions for a computer to execute.

What is Programming?

The instructions can result in output as simple as printing the answer to a simple math equation or a word to the screen, or as complex as the latest video editing software, games, websites, or countless other pieces of software used in our daily lives.
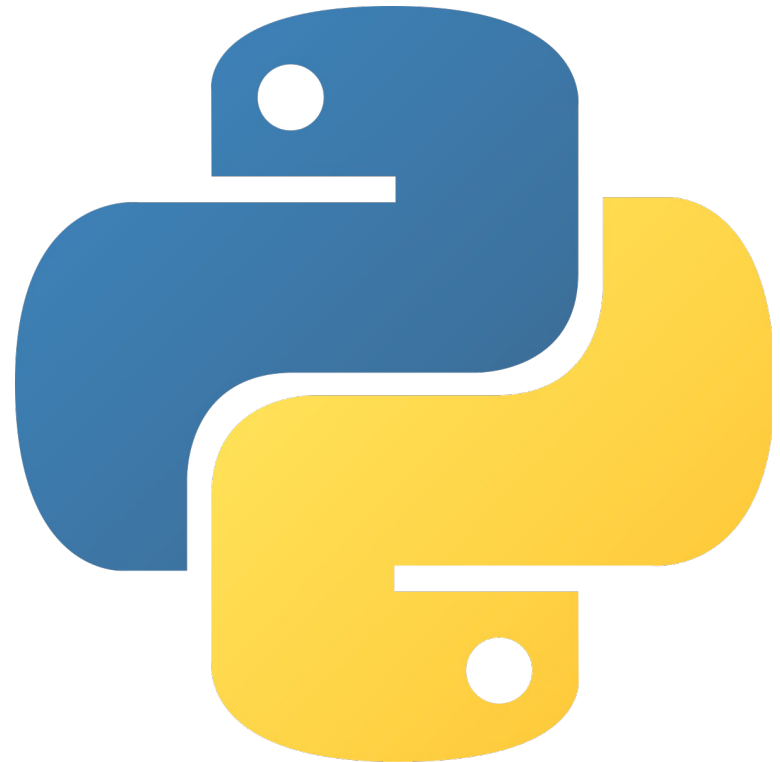
What is Programming?

Many different programming languages are used for writing software, and each programming language has its own advantages and disadvantages for developing different types of software.

What is Programming?

The programming language we will learn to use in these workshops is called Python.

Why Learn Python?

There are a lot of programming languages out there, so why are we learning Python?

We'll go over three reasons:

Why Learn Python?

1) Python is a very popular language with an active online community, so there's a lot of support for students.

Why Learn Python?

2) Python uses a syntax that is much closer to written English than other programming languages, so it's easier to read and understand

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

Why Learn Python?

3) Python automates some tedious operations common to other programming languages, such as memory management.
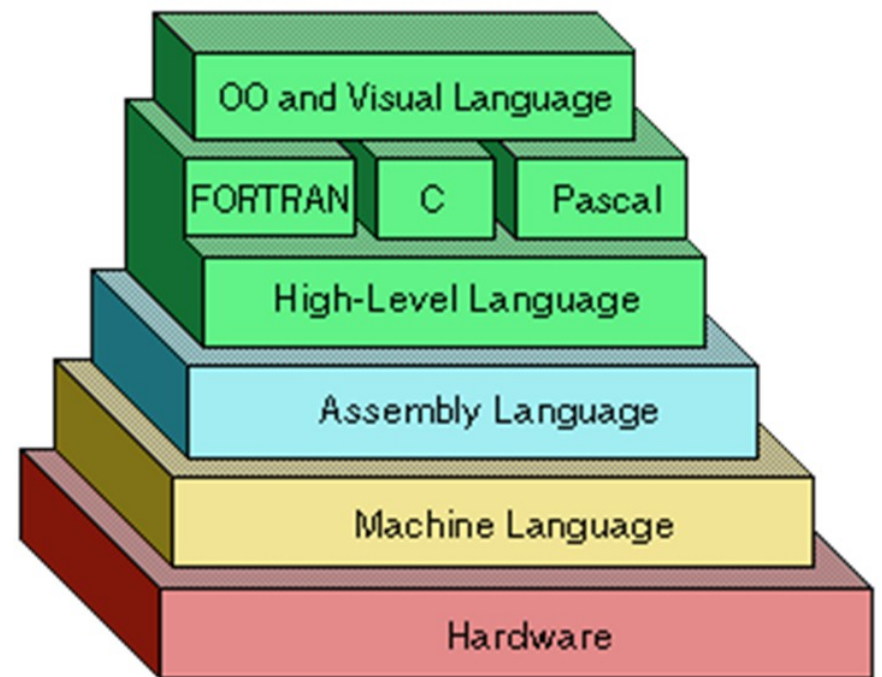
What is Python? An Overview

Python is a **high-level**, **interpreted**, **general-purpose** programming language. What does this mean?
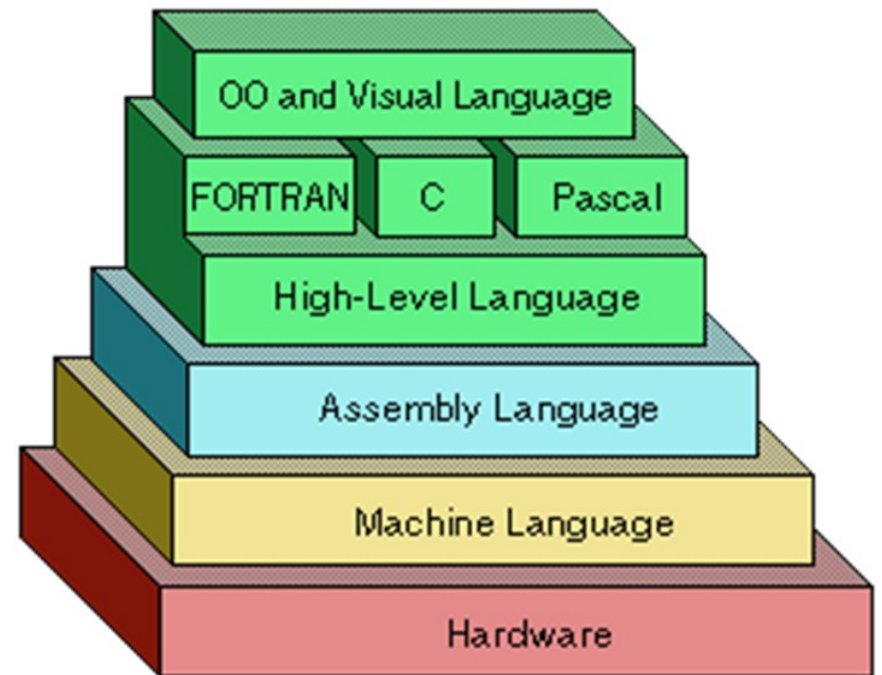
High-Level Language

Programming languages are classified as either low-level or high-level, depending on how abstract the language is from the pure machine code that computer processors use.
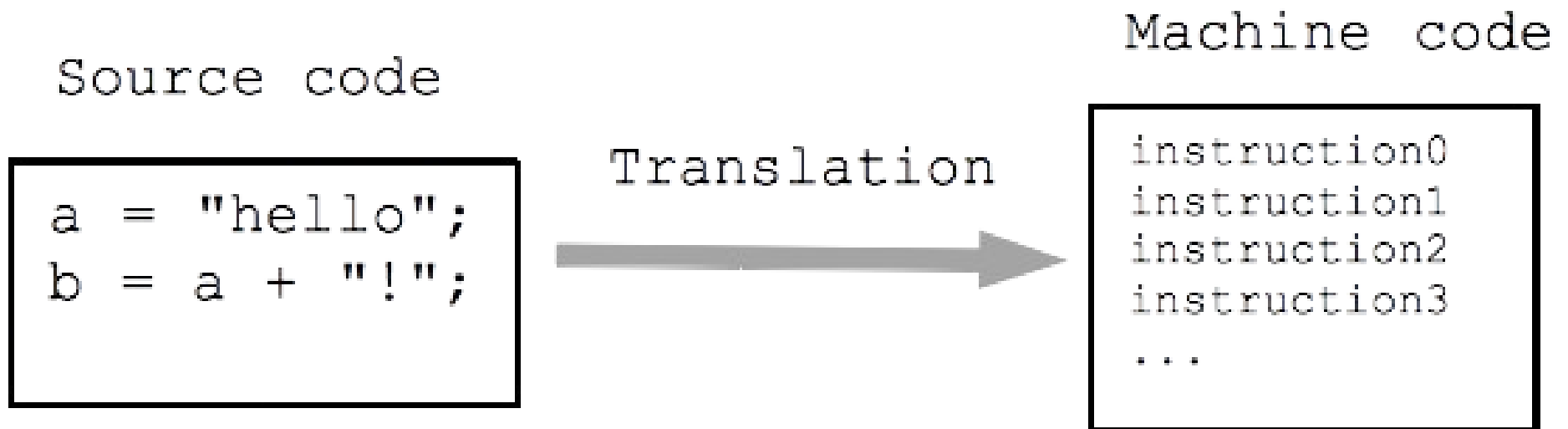
High-Level Language

Programming in a high-level language has two major effects:

High-Level Language

First, abstract syntax allows easier understanding for people who need to read code written in high-level languages and write code using the same.

Source code

```
a = "hello";
b = a + "!";
```

Translation

Machine code

```
instruction0
instruction1
instruction2
instruction3
...
```

High-Level Language

Second, due to its high level of abstraction from machine language, processes using high-level languages are relatively slower than the same processes performed by low-level languages.

High-Level Language

In other words, high-level programming language code is easier to write and read, but programs created with that code are slower than programs created with low-level languages.
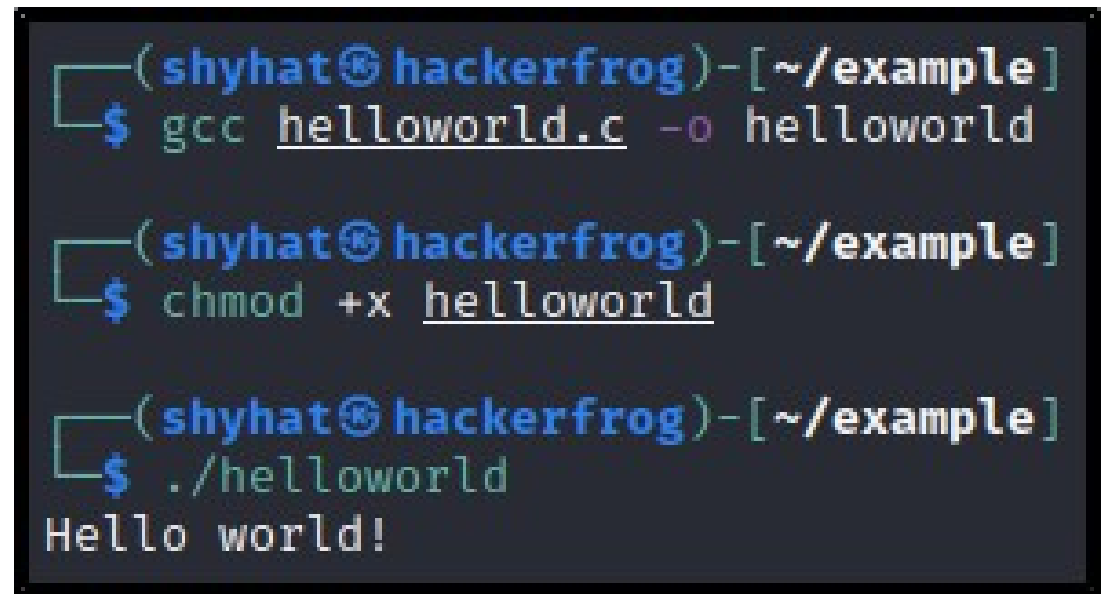
Interpreted Language

Interpreted languages are programming languages that do not require compilation before instructions are run.

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

Interpreted Language

As opposed to compiled languages, where programming code must be converted into machine code by a compilation program before it can be run.

General-Purpose Language

General-purpose programming languages are programming languages designed to be able to create a wide range of application types.

General-Purpose Language

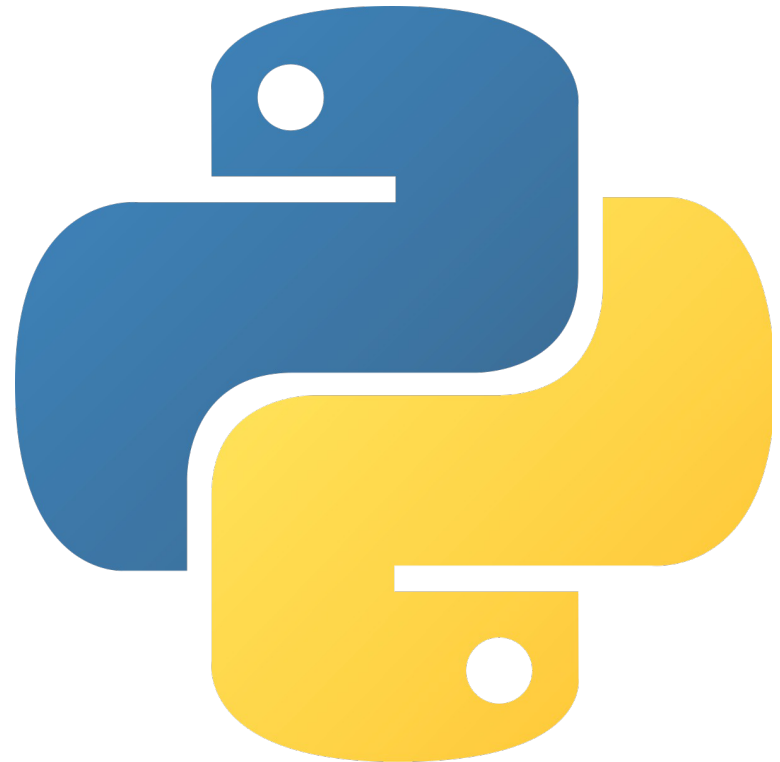As opposed to domain-specific programming languages, which typically are used to operate in specialized environments.

What's Next?

In the next HackerFrogs Afterschool programming workshop, we'll start learning Python with the **learnpython.org** website.

Until Next Time, HackerFrogs!