OBJECT ORIENTED PROGRAMMING WITH PYTHON

Second Class

1st Semester

- Problem Solving
- What is a program?
- Debugging
- types of programming Languages
- Python
- Why Python
- What we can do with Python

PROBLEM SOLVING

- The single most important skill for a computer scientist is **problem solving**. Problem solving means the ability to formulate problems, think creatively about solutions, and express a solution clearly and accurately. As it turns out, the process of learning to program is an excellent opportunity to practice problem-solving skills.

WHAT IS A PROGRAM?

- A **program** is a sequence of instructions that specifies how to perform a computation. The computation might be something **mathematical**, such as solving a **system of equations**, but it can also be a **symbolic computation**, such as searching and replacing text in a document.

WHAT IS A PROGRAM?

- **Input**: Get data from the keyboard, a file, or some other device.
- **Output**: Display data on the screen or send data to a file or other device
- **Math:** Perform basic mathematical operations like addition and multiplication
- **Conditional execution**: Check for certain conditions and execute the appropriate sequence of statements.
- **Repetition**: Perform some action repeatedly, usually with some variation

DEBUGGING

- Programming is a complex process, and because it is done by human beings, it often leads to errors.
- Programming errors are called **bugs** and the process of tracking them down and correcting them is called **debugging.**

- Three kinds of errors can occur in a program:
  - **Syntax errors**,
  - **Runtime errors**,
  - **Semantic errors**.

DEBUGGING

Syntax Error

- **Syntax** refers to the structure of a program and the rules about that structure.
- If the program contains a structural error or did not match the rules then there is **syntax error**.
- If there is a single syntax error anywhere in your program, Python will display an error message and quit, and you will not be able to run your program.

Runtime Errors

- The second type of error is a **runtime error**, so called because the error does not appear until you run the program. These errors are also called **exceptions.**

DEBUGGING

Semantic Error

- If there is a semantic error in your program, it will run successfully, in the sense that the computer will not generate any error messages, but it will not do the right thing
- The meaning of the program (its semantics) is wrong. Identifying semantic errors can be tricky because it requires you to work backward by looking at the output of the program and trying to figure out what it is doing.

EXPERIMENTAL DEBUGGING

- One of the most important skills you will acquire is debugging
- debugging is one of the most intellectually rich, challenging, and interesting parts of programming.
- debugging is like detective work. You are confronted with clues, and you have to infer the processes and events that led to the results you see.

TYPES OF PROGRAMMING LANGUAGES

- There are two types of programming Languages **low level languages** and **High level languages.**

- **Low Level Language**
- The language that only machine can understand, they do not need an interpreter or compiler, thus, they are closer to the computer Hardware. Two common types of low-level programming languages are:

  - **Machine language** and,
  - **Assembly language.**

TYPES OF PROGRAMMING LANGUAGES

- **High Level Language**

  The language that only human can understand, which are easy to write, read, and edit. They need an interpreter or compiler for the machine to understand them. (like Java, Python, JavaScript, C++, C, C#, and others).

- **There are two programming paradigms (style of programming or the way you program).**

  - Procedural Programming

  uses a list of instructions to tell the computer what to do step-by-step In, here the program consists of data and modules/procedures that operate on the data. The two are treated as separate entities (C, C++, Fortran, Pascal).

  - Object Oriented Programming

  is an approach to problem-solving where all computations are carried out using objects. An **object** is a component of a program that knows how to perform certain actions and how to interact with other elements of the program (Java, C++, C#, Python)

TYPES OF PROGRAMMING LANGUAGES

High Level Language

- Object Oriented Programming (OOP)
- Procedural Programming (PoP)

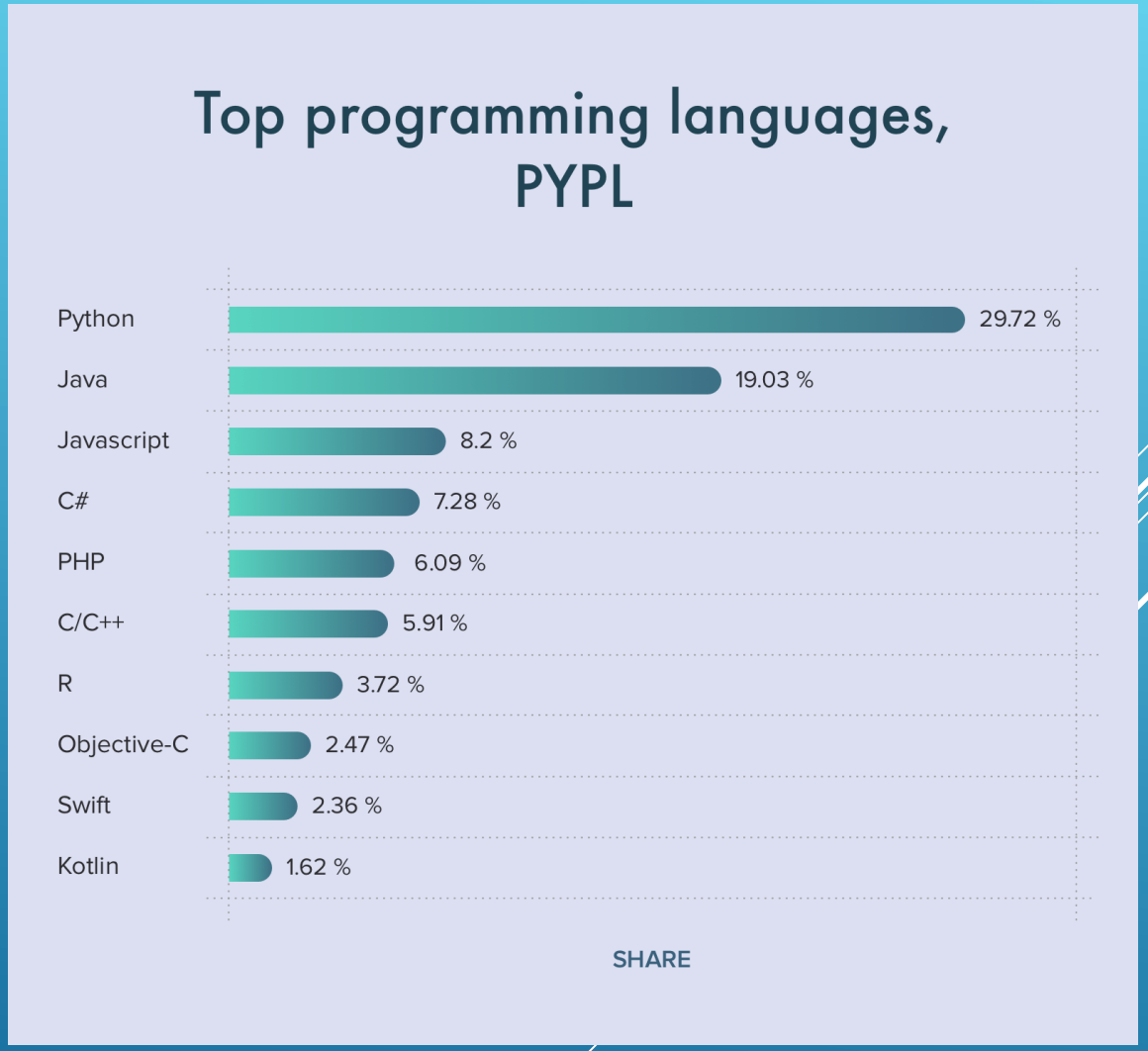
Low Level Language

- Assembly Language
- Machine Language

Hardware

PYTHON

- Python is a **high level**, **interpreted** language which has **easy** syntax, it is much easier than other programming languages and helps you create beautiful applications with less effort and much more ease.



WHY PYTHON?

Why choose Python? Here are some of the features that make Python a great option.

Popular	• Python has been growing in popularity over the last few years, Python developers are sought after and paid well.
Open source(Free)	• free to install, use, and distribute, even for commercial purposes
Simple	• Python code has a simple and clear structure that is easy to learn and easy to read

WHAT WE CAN DO WITH PYTHON!

Web Development	• Python has very good support for web development with its frameworks like **Django**, **Flask**, and others.
Building Games	• Python supports developing games. Its **Pygame** library is highly useful. It supports art, music, sound, video, and multimedia projects to be built with it
Data Science	• Python is well suited for data manipulation, analysis, and implementing complex algorithms.libraries like NumPy, scipy, scikit-learn, etc
Mobile Application	• In Python there are packages you can use to create mobile applications, like Kivy, PyQt, or even Beeware's Toga library.

Apps that use Python

YouTube

Instagram

Dropbox