**What is Programming and Python?**

* Programming is a way for us to tell computers what to do.
* Computer is a very dumb machine and it only does what we tell it to do in programming lang
* Hence we learn programming and tell computers to do what we are very slow at - computation.
* If I ask you to calculate 5+6, you will immediately say 11.
* How about 23453453 X 56456?
* You will start searching for a calculator or jump to a new tab to calculate the same.

**What is Python?**

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

* web development (server-side),
* software development,
* mathematics,
* system scripting.

Features of Python

* Python is simple and easy to understand.
* It is Interpreted and platform-independent which makes debugging very easy.
* Python is an open-source programming language.
* Python provides very big library support. Some of the popular libraries include NumPy, Tensorflow, Selenium, OpenCV, etc.
* It is possible to integrate other programming languages within python

What can Python do?

Python can be used on a server to create web applications.

Python can be used alongside software to create workflows.

Python can connect to database systems. It can also read and modify files.

Python can be used to handle big data and perform complex mathematics.

Python can be used for rapid prototyping, or for production-ready software development.

Syntax of python:-

File Handling :-

Text File

Text files store data in ASCII or UNICODE format, which means each character is represented by a code point that occupies 1 byte of space. For example, the string "123.45" would indeed occupy 6 bytes of space.

Here's a breakdown of the space occupied by each character:

- "1" - 1 byte

- "2" - 1 byte

- "3" - 1 byte

- "." - 1 byte

- "4" - 1 byte

- "5" - 1 byte

Total: 6 bytes

Text files are human-readable and understandable, making them easy to work with. Each line in a text file is terminated by an End-Of-Line (EOL) character, which is typically:

- \n (Line Feed) on Unix-based systems (Linux, macOS)

- \r\n (Carriage Return + Line Feed) on Windows systems

The EOL character indicates the end of a line and allows text editors and programs to properly display and process the file contents.

Binary Files:

Binary files store data in the same format as it is stored in memory, which means that no translation or formatting is required when reading or writing data. This results in faster read and write times compared to text files.

Here are some key characteristics of binary files:

1. Faster read/write times: Binary files are faster to read and write because no translation is required.

2. Same format as memory: Binary files store data in the same format as it is stored in memory.

3. No translation: No translation or formatting is required when reading or writing data.

4. Not human-readable: Binary files are not readable by humans, as they contain raw binary data.

5. Platform-dependent: Binary files can be platform-dependent, meaning they may not be compatible across different systems or architectures.

Common examples of binary files include:

1. Executable files (.exe, .bin)

2. Image files (.jpg, .png)

3. Audio files (.mp3, .wav)

4. Video files (.mp4, .avi)

5. Database files (.db, .dat)

In contrast, text files store data in a human-readable format, with each character represented by a specific code (e.g., ASCII). Text files are slower to read and write than binary files but are more portable and human-readable

Class:

A class is a blueprint or a template that defines the properties and behaviors of an object. It's a design pattern or a template that defines the characteristics of an object.

Syntax:

Class Person:

name=”Dina”

Occupation=

Object:

An object is an instance of a class, it's a concrete entity that has its own set of attributes (data) and methods (functions) that describe and define its behavior.