Department of Computer Science and Engineering
PES University, Bangalore, India

Lecture Notes
Python for Computational Problem Solving
UE23CS151A

Lecture #10

Programming Paradigms and Introduction to Python

By,
Prof. Sindhu R Pai,
Anchor, PCPS - 2023
Assistant Professor
Dept. of CSE, PESU

Verified by,
PCPS Team - 2023

Many Thanks to
Dr. Shylaja S S (Director, CCBD and CDSAML Research Centers, Former Chairperson, CSE, PES University)
Prof. Chitra G M (Asst. Prof, Dept. of CSE, PCPS Anchor – 2022)

Few points to think!

- **How do you communicate with the computer?**
- **How do you provide instructions to the computer to do our job?**
- **Which language do you use to give the instructions to computer?**

Introduction to Programming Language

Language is the primary mean of communication and human interaction for thousands of years. Similarly, we need a **language to communicate with computers**. It has hardware and software that need to communicate with each other. Your application is reacting to the mouse and keyboard or even the mic, it can read files from your disk storage and so on. But at the end of the day, the machine understands bits, 1s, and 0s, the combination of which creates meaning. It is obvious that we find it difficult to communicate to the computer in their language. Hence, we would use a language which makes it easier for us to express our ideas means the language which is closer to the way we perceive.

Programming language: Provides the necessary constructs to instruct the computer to do the tasks/jobs for us.

Program: A sequence of instructions used to "instruct the computer to perform various tasks".

Note: Programming is an art/creative activity.

Levels of Programming Languages:

- **Low Level**
  - Binary codes which CPU executes
  - Programmer's responsibility is more
  - Eg. Machine language
- **Middle Level**
  - Offers basic data structure and array definition, but the programmer should

take care of the operations.

- Eg. C and C++

- **High Level**
  - Programmer concentrates on the algorithm and programming itself
  - Eg. Java, Python, Pascal

How does a computer understand a program in a middle level or high level language?

We require a mechanism to translate a program written in a middle level or high-level language to a program in a machine language. These are called **translators.** Different translators are discussed in **Lecture #9.**

Paradigms of Programming Languages

- Imperative
- Structured Programming
- Procedural
- Declarative
- Functional
- Logical
- Object-Oriented

How many Programming Languages are there? And Which one to choose?

Many / Infinity …

Choose the right language based on the application

TIOBE Index:

The TIOBE Programming Community index is an indicator of the **popularity of programming languages.** The index is **updated once a month**. The ratings are based on the number of skilled engineers world-wide, courses and third-party vendors. There are 25 search engines that are used to calculate the TIOBE index. Popular search engines such as Google, Bing, Yahoo!, Wikipedia, Amazon, YouTube, and Baidu etc. It is important to note

that the TIOBE index is not about the *best* programming language or the language in which most lines of code have been written.

Refer to this link for more details: **TIOBE Index - TIOBE**

Average positions of top 5 languages for a period of 12 months is given below.

Programming Language	2023	2018	2013	2008	2003	1998	1993
Python	1	4	8	7	12	25	18
C	2	2	1	2	2	1	1
C++	3	3	4	4	3	2	2
Java	4	1	2	1	1	18	-
C#	5	5	5	8	9	-	-

Introduction to Python Language

High Level Language developed by **Guido van Rossum is published in Feb 1991**. It is named after **Monty Python circus Show**- a satirical show aired on BBC in the 60s and 70s. It is a simple yet powerful language well received by both the academia and the industry.

Why Python?

Life is all about selecting between the alternates. You must have debated before selecting Engineering over medicine, selecting college A over college B. I am sure you would have some criteria before choosing between the alternates – Your parents run a hospital, your uncle is a software specialist, your neighbors whom you do not like boast a lot about their child's achievement or you tossed a coin to decide.

It is equally same here too. We have umpteen number of languages to choose from. After lots of experiments with various languages, we have decided that you should start your journey in programming with Python. Recently Stanford also changed over to Python as the first language to be taught.

We prefer Python for a number of reasons. The most important reasons are:

- Simple to learn
- Easy to master
- Extensively used in the industry
- Extensively used in research

Features of Python:

- Uses an elegant syntax, making the programs you write **easier to read**.
- Is **an easy-to-use language** that makes it simple to get your program working.
- Comes with a large standard library that supports many common programming tasks such as connecting to **web servers, searching text with regular expressions, reading and modifying files**. Python Package index contains a more than 72000 packages. It includes **GUIs, test frameworks, automation and web scraping, scientific computing, text processing, image processing, graph generation**
- Python's interactive mode makes it easy to test short snippets of code. There's also a bundled development environment called IDLE.
- **Runs anywhere**, including Mac OS X, Windows, Linux and Unix.
- Is **free software in two senses**. It doesn't cost anything to download or use Python, or to include it in your application. Python can also be freely modified and re-distributed, because while the language is copyrighted it's available under an open-source license.
- A **variety of data structures(types) are available**: numbers (floating point, complex, and unlimited length long integers), strings (both ASCII and Unicode), lists, and dictionaries.
- The language supports object-oriented programming with **classes and multiple inheritance**. Also supports **raising and catching exceptions**, resulting in cleaner error handling.
- Code can be grouped into **modules and packages**.
- Contains advanced programming features such as **generators and list**

comprehensions.

- **Automatic memory management** feature of python frees you from having to manually allocate and free memory in your code.

- END -