Microspectra Software Technologies Pvt. Ltd

**Python Batch 5, Day1 Assignment 1.**

Q1) How does a computer Program Works ?

Ans) A computer program is a collection of instructions to perform a specific task. A computer program is usually written by a computer programmer in a programming language. From the program in its human readable form of source code a compiler or assembler can derive a machine code( **a form consisting of instructions that the computer can directly execute**).

Alternatively a computer program may be executed with the help of an interpreter. Computer programming is the process of writing or editing source code(human readable programming language).

Q2) Can we compare natural vs Programming Language?

Ans) The differences in the grammar of high level programming languages (e.g. Java) and natural language (e.g. English) are as follows:

* Natural languages are used for communication between people and programming languages enable human to interact with machines.
* Programming languages need a high degree of expertise, completeness and precision because computer can not think outside the statement while in speaking, some minor errors are ignored.
* The programming language syntax is not based on natural language grammar.

Thus, the main difference in the grammar of high level language (Java) and natural language (English) is that natural language may be informal in speaking but not in written communication while the grammar of programming language follows specific syntax.

Q3) What is Compilation Vs Interpretation?

Ans) **interpretation** is reading or checking the code line by line and if any error is found you can't go further before correcting it. whereas **compilation** is reading and checking the whole code at once and reporting all errors found in it.

|  |  |  |
| --- | --- | --- |
| Sr no. | Compiled language | Interpreted language |
| 1 | A compiled language is a programming language whose implementations are typically compilers and not interpreters. | An interpreted language is a programming language whose implementations execute instructions directly and freely, without previously compiling a program into machine-language instructions. |
| 2 | In this language, once the program is compiled it is expressed in the instructions of the target machine. | While in this language, the instructions are not directly executed by the target machine. |
| 3 | There are at least two steps to get from source code to execution | There is only one steps to get from source code to execution. |
| 4 | In this language, compiled programs run faster than interpreted programs. | While in this language, interpreted programs can be modified while the program is running |
| 5 | In this language, compilation errors prevent the code from compiling. | In this languages, all the debugging occurs at run-time. |
| 6 | The code of compiled language can be executed directly by the computer’s CPU. | A program written in an interpreted language is not compiled, it is interpreted. |
| 7 | This language delivers better performance. | This languages delivers relatively slower performance. |
| 8 | Example of compiled language – C, C++, C#, CLEO, COBOL, etc | Example of Interpreted language – JavaScript, Perl, Python, BASIC, etc. |

Q4) What Does Interpreter actually Do ?

Ans) An interpreter is a computer program that is used to directly execute program instructions written using one of the many high-level programming languages. The interpreter transforms the high-level program into an intermediate language that it then executes the high-level source code and then performs the commands directly, which is done line by line or statement by statement.

The interpreter reads each statement of code and then converts or executes it directly. Since an interpreter reads and then executes code in a single process, it very useful for scripting and other small programs. As such, it is commonly installed on Web servers, which run a lot of executable scripts. It is also used during the development stage of a program to test small chunks of code one by one rather than having to compile the whole program every time.

Q5) What is Python ?

Ans) Python is an interpreted, object-oriented, high-level programming language with dynamic semantics.

Python’s simple and easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance.

It has fewer steps when compared to Java and C.It was founded in 1991 by developer Guido Van Rossum.It is used in many organizations as it supports multiple programming paradigms.It also performs automatic memory management.

**Advantages :**  
1) Presence of third-party modules  
2) Extensive support libraries(NumPy for numerical calculations, Pandas for data analytics etc)  
3) Open source and community development  
4) Easy to learn  
5) User-friendly data structures  
6) High-level language  
7) Dynamically typed language(No need to mention data type based on value assigned, it takes data type)  
8) Object-oriented language  
9) Portable and Interactive  
10) Portable across Operating systems

**Applications :**  
1) GUI based desktop applications(Games, Scientific Applications)  
2) Web frameworks and applications  
3) Enterprise and Business applications  
4) Operating Systems  
5) Language Development  
6) Prototyping

Q6) What are Characteristics of Python ?

Ans)

1. **EASY TO CODE**: Python is a very developer-friendly language which means that anyone and everyone can learn to code it in a couple of hours or days. As compared to other object-oriented programming languages like Java, C, C++, and C#, Python is one of the easiest to learn.
2. **OPEN SOURCE AND FREE:** Python is an open-source programming language which means that anyone can create and contribute to its development.
3. **SUPPORT FOR GUI:** GUI or Graphical User Interface is one of the key aspects of any programming language because it has the ability to add flair to code and make the results more visual. Python has support for a wide array of GUIs which can easily be imported to the interpreter, thus making this one of the most favourite languages for developers.
4. **OBJECT ORIENTED APPROACH:** One of the key aspects of Python is its object oriented approach. This basically means that Python recognizes the concept of class and object encapsulation thus allowing programs to be efficient in the long run.
5. **HIGH LEVEL LANGUAGE:** Python has been designed to be a high-level programming language, which means that when you code in Python you don’t need to be aware of the coding structure, architecture as well as memory management.
6. **INTEGRATED BY NATURE:** Python is an integrated language by nature. This means that the python interpreter executes codes one line at a time.
7. **HIGHLY PORTABLE:** Suppose you are running Python on Windows and you need to shift the same to either a Mac or a Linux system, then you can easily achieve the same in Python without having to worry about changing the code.
8. **EXTENSIVE ARRAYS OF LIBRARIES:** Out of the box, Python comes inbuilt with a large number of libraries that can be imported at any instance and be used in a specific program.
9. **Support for other languages:** Being coded in C, Python by default supports the execution of code written in other programming languages such as Java, C, and C#, thus making it one of the versatile in the industry.

Q8) What are different flavours of Python ?

Ans) 1. **C-Python**: C Python is the Python compiler implemented in C programming language. In this, Python code is internally converted into the **byte code** using standard C functions. Additionally, it is possible to run and execute programs written in C/C++ using CPython compiler.

2. **J-Python** : Earlier known as **JPython**. Jython is an implementation of the Python programming language designed to run on the Java platform. Jython is extremely useful because it provides the productivity features of a mature scripting language while running on a JVM.

3. **PyPy**: This is the implementation using Python language. PyPy often runs faster than CPython because PyPy is a just-in-time compiler while CPython is an interpreter.

4. **Iron Python**: IronPython is an open-source implementation of the Python programming language which is tightly integrated with the .NET Framework.

5. **Ruby Python**: RubyPython is a bridge between the Ruby and Python interpreters. It embeds a Python interpreter in the Ruby application’s process using FFI (Foreign Function Interface).

6. **Pythonxy**: Python(x,y) is a free scientific and engineering development software for numerical computations, data analysis and data visualization based on Python.

7. **Stackless Python**: Stackless Python is a Python programming language interpreter. In practice, Stackless Python uses the C stack, but the stack is cleared between function calls

8. **Anaconda Python**: Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing, that aims to simplify package management and deployment. Package versions are managed by the package management system conda.