Assignment No –01

Q-1 Who developed Python Programming Language?

Ans: Python was developed by Guido van Rossum in the late 1980s while he was working at the National Research Institute for Mathematics and Computer Science (CWI) in the Netherlands. He named it after the comedy troupe Monty Python, and the language's playful and easy-to-learn nature reflects this inspiration. Guido van Rossum remained the primary author and maintainer of Python until 2018, when he retired from his position at Dropbox. Since then, the Python community has continued to develop and maintain the language, with a core development team overseeing its evolution.

Q-2 Which type of programming does python support?

Ans: Python supports several programming paradigms, including:

(i)Procedural Programming: Python supports procedural programming where code is organized in procedures/functions that operate on data.

(ii)Object-Oriented Programming (OOP): Python supports OOP, which emphasizes the use of objects that encapsulate data and functionality, allowing for code reuse and modular design.

(iii)Functional Programming: Python also supports functional programming, which emphasizes the use of pure functions and immutable data structures.

(iv)Aspect-oriented Programming (AOP): Python has some support for AOP, which allows developers to separate cross-cutting concerns (such as logging or error handling) from the main logic of the program.

(v)Scripting: Python is often used as a scripting language, allowing developers to write short, concise programs that can be quickly executed.

(vi)Other paradigms: Python also has some support for concurrent programming, event-driven programming, and other paradigms.

Q-3. Is Python case sensitive when dealing with identifiers?

Ans: Yes, Python is case sensitive when dealing with identifiers such as variable names, function names, and class names. This means that identifiers that differ only in capitalization are considered to be different identifiers

Q-4. What is the correct extension of the Python file?

Ans: The ".py" extension is not only a convention but it is also recognized by most text editors and development environments as indicating that the file contains Python code. This can be useful for syntax highlighting, code completion, and other features that are specific to Python.

It's worth noting that there are some other file extensions that you might come across in the context of Python development. For example, ".pyc" files are compiled Python files, and ".pyd" files are dynamic link libraries that are used on Windows systems. However, for most Python development, you'll be working with ".py" files.

Q-5. Is python code compiled or interpreted?

Ans: Python code is typically interpreted, although it can also be compiled.

When you run a Python script, the source code is translated into bytecode by the Python interpreter. This bytecode is then executed by the interpreter, which interprets each instruction and carries out the corresponding operations.

However, there are also ways to compile Python code into platform-specific machine code that can be executed without the need for an interpreter. One such method is using the built-in compile () function in Python, which converts Python source code into bytecode or machine code.Another way to compile Python code is by using tools such as PyInstaller or cx_Freeze, which package your Python code and its dependencies into a standalone executable that can be run without the need for a Python interpreter.In general, interpreting Python code provides the advantage of

faster development and easier debugging, as changes to the code can be made and tested immediately without the need for recompilation. On the other hand, compiling Python code can provide the advantage of faster execution and the ability to distribute your application as a standalone executable.

Q-6 Name a few blocks of code used to define in Python programming language?

Ans: Here are a few examples of code blocks that are commonly used in Python:

(i) Conditional statements (if/else): used to execute different code based on a condition

(ii) Loops (for/while): used to repeat code multiple times

(iii) Functions: used to group together a set of instructions and give them a name

(iv) Classes: used to define custom data types and their methods

(v) Exception handling (try/except): used to handle errors that may occur during code execution

Q-7. State a character used give single- line comments in Python?

Ans : In Python, the hash symbol (#) is used to create single-line comments. Any text that appears after the hash symbol in a line of code is ignored by the Python interpreter

```
# This is a single-line comment in Python

print("Hello, World!")  # This line also has a comment
```

In the code above, the first line is a single-line comment, and the second line is a print statement that also has a comment. The Python interpreter will ignore the text after the hash symbol in both cases.

Q-8 Mention functions which can help us to find the version of python that we are currently working on?

Ans: 'sys.version' - This function returns a string containing the version number of Python and additional information such as the build date and compiler used.

'sys.version_info' - This function returns a tuple containing the major, minor, and micro version numbers of Python.

'platform.python_version()' - This function returns the version number of Python as a string.

'platform.python_version_tuple()' - This function returns a tuple containing the major, minor, and micro version numbers of Python.

You can use any of these functions to determine the version of Python that you are currently working with.

Q-9 Python supports the creations of anonymous functions at runtime, using a construct called ….

Ans: Python supports the creation of anonymous functions at runtime using a construct called "lambda" functions. Lambda functions are small, anonymous functions that can be created on the fly and do not require a specific name to be defined. They are often used for simple, one-line operations that do not require a full function definition.

Lambda functions can take any number of arguments but can only have a single expression as their body. They are typically used as arguments to higher-order functions that expect a function object as input, or to create a list of functions in a concise and readable way.

Q-10 What does pip stand for python?

Ans: PIP stands for "Pip Installs Packages" (formerly "Pip Installs Python"). It is a package management system used to install and manage software packages written in

Python. PIP allows you to easily install, upgrade, and remove Python packages and their dependencies.

Q-11 Mention a few built-in functions in python?

Ans: Here are a few commonly used built-in functions in Python:

print (): Outputs text to the console or terminal.

len(): Returns the length of an object (number of items in a list, characters in a string, etc.).

range (): Generates a sequence of numbers.

input (): Allows the user to input text from the console or terminal.

type (): Returns the data type of an object.

str (): Converts an object into a string.

int (): Converts a string or other type into an integer.

float (): Converts a string or other type into a floating-point number.

list (): Creates a list from an iterable object.

dict(): Creates a dictionary from key-value pairs or an iterable object.

These are just a few examples of the many built-in functions available in Python.

Q-12 What is the maximum possible length of an identifier in Python?

Ans: In Python, the maximum possible length of an identifier (variable name, function name, etc.) is not explicitly defined. However, the practical limit is determined by the maximum size of a string, which is platform-dependent

Q-13 What are the benefits of using Python?

Ans: Python is a high-level, interpreted programming language with a simple and easy-to-learn syntax. It is a versatile language that can be used for a wide variety of applications, from web development to scientific computing to data analysis and machine learning. Some of the key benefits of using Python are:

Easy to learn and use: Python has a simple syntax and is easy to read and write, making it an ideal language for beginners.

Large and active community: Python has a large and active community of developers, which means there are many libraries, frameworks, and tools available to simplify development tasks and reduce development time.

Cross-platform compatibility: Python code can run on multiple platforms including Windows, Mac, and Linux, making it a popular choice for cross-platform development.

Rich libraries and frameworks: Python has a vast collection of libraries and frameworks for a variety of applications, including web development (Django, Flask), scientific computing (NumPy, SciPy), machine learning (TensorFlow, PyTorch), and data analysis (Pandas).

High productivity: Python's simple and readable syntax, combined with its vast collection of libraries and frameworks, makes it possible to develop complex applications quickly and with minimal code.

Open-source and free: Python is an open-source language and is free to use, distribute, and modify.

Extensible and versatile: Python is an extensible language, which means that it can be easily extended using C, C++, and other languages. It is also a versatile language that can be used for a wide range of applications, from web development to scientific computing to data analysis and machine learning.

Q-14 How is memory managed in Python?

Ans: In Python, memory is managed automatically through a process called "garbage collection". Garbage collection is the process by which the interpreter automatically frees up memory that is no longer being used by the program. This means that you don't have to manually allocate and free memory like you would in languages like C or C++.

Python uses a reference counting system to keep track of when an object is no longer being used. Every object in Python has a reference count, which is the number of references to the object. When an object's reference count goes to zero, the object is no longer being used and its memory can be freed.

In addition to reference counting, Python also has a garbage collector that periodically checks for objects that are no longer being used and frees their memory. The garbage collector is able to detect free circular references, where two or more objects reference each other but are no longer used by the program.

Python also uses an optimization called "object pooling" to reduce the amount of memory allocated for frequently used objects. For example, small integers and strings are pooled so that they are reused whenever possible, rather than being allocated new memory each time they are created.

Overall, Python's memory management system is designed to be automatic and transparent to the programmer, allowing them to focus on writing their code without having to worry about memory allocation and deallocation.

Q-15 How to Install Python on windows and set path variables?

Ans: Here are the steps to install Python on Windows and set path variables:

Download Python: Go to the official Python website **(https://www.python.org/downloads/)** and download the latest version of Python for Windows.

Run the installer: Once you have downloaded the Python installer, run it and follow the instructions to install Python on your system. Be sure to select the option to add Python to your PATH during the installation process.

Verify installation: After the installation is complete, open the Command Prompt and type "python". If Python is installed correctly, you should see the Python version number and a Python prompt (>>>) appear.

Set path variables: If Python is not recognized by the Command Prompt, you may need to add it to your system's PATH environment variable. Here are the steps to set the path variable:

a. Open the Control Panel and go to System and Security > System > Advanced system settings.

b. Click on the "Environment Variables" button.

c. Under "System Variables", scroll down and select the "Path" variable, then click on the "Edit" button.

d. Add the path to your Python installation directory (e.g. "C:\Python39") to the list of paths, separating each path with a semicolon.

e. Click "OK" to save the changes.

Verify path variables: To verify that the path variable has been set correctly, open a new Command Prompt and type "python". If Python is now recognized, you should see the Python version number and a Python prompt (>>>) appear.

Q-16 Is indentation required in python ?

Ans : Yes, indentation is required in Python. In Python, indentation is used to indicate a block of code. This is different from many other programming languages where curly braces or other delimiters are used to indicate code blocks.

Python uses whitespace (spaces or tabs) to indicate the level of indentation. The standard convention is to use four spaces for each level of indentation.