Sudipta mukherjee

## Assignment (29th Jan 2023)

**Question No. 1: Who developed Python Programming Language?**

Guido van Rossum developed the Python programming language.

**Question No. 2: Which type of programming does Python support?**

Python is a multi-paradigm programming language, supporting the following programming paradigms:

1. Imperative Programming
2. Object-Oriented Programming
3. Functional Programming
4. Procedural Programming
5. Reflective Programming

**Question No. 3: Is Python case sensitive when dealing with identifiers?**

Yes, Python is case sensitive when dealing with identifiers (e.g. variable names, function names, etc.). For example, the names "my Variable" and "myvariable" would be considered two different names in Python.

**Question No. 4: What is the correct extension of the Python file?**

The correct extension of a Python file is ".py". This is the standard extension used for Python files and is recognized by the Python interpreter.

**Question No. 5: Is python code compiled or interpreted?**

Python code is interpreted, not compiled. This means that the code is executed line by line by the Python interpreter, rather than being translated into machine code and then executed like in compiled languages such as C or C++.

**Question No. 6: Name a few blocks of code used to define in Python language.**

In Python, there are a few blocks of code used to define code structure, they include:

1. Functions: defined using the "def" keyword.
2. Classes: defined using the "class" keyword.
3. Loops: defined using the "for" and "while" keywords.
4. Conditional statements: defined using the "if" keyword.
5. Exception handling: defined using the "try" and "except" keywords.

**Question No. 7: State a character used to give single-lined comments in Python.**

The character "#" is used to give single-line comments in Python. Everything to the right of the "#" on a line is ignored by the Python interpreter and is treated as a comment.

**Question No. 8: Mention functions which can help us to find the version of Python that we are currently working on.**

There are two functions in Python that can be used to determine the version of Python you are running:

1. sys.version: This function returns a string that contains the version number of the Python interpreter, along with other information such as the build number and the version of the underlying operating system.
2. sys.version\_info: This function returns a named tuple containing information about the version of Python, including the major version number, minor version number, micro version number, and release level.

**Question No. 9: Python supports the creation of anonymous functions at runtime, using a construct called** lambda.

**Question No. 10: What does PIP stands for in Python?**

PIP stands for "Pip Installs Packages" in Python. It is a package management system used to install and manage packages or modules that are not part of the standard library.

**Question No. 11: Mention a few built-in functions in Python.**

1. print(): used to display the output of your program.
2. len(): used to find the length of a sequence, such as a string, list, or tuple.
3. int(): used to convert a value to an integer.
4. float(): used to convert a value to a floating-point number.
5. str(): used to convert a value to a string.
6. sum(): used to add up the elements of a sequence.
7. min(): used to find the minimum value in a sequence.
8. max(): used to find the maximum value in a sequence.
9. sorted(): used to sort a sequence in ascending or descending order.
10. range(): used to generate a sequence of numbers.

**Question No. 12: What is the maximum possible length of an identifier in Python?**

In Python, there is no specified limit on the length of an identifier (variable name, function name, class name, etc.).

**Question No. 13: What are the benefits of using Python?**

1. Easy to learn and read: Python has a simple, easy-to-learn syntax that emphasizes readability and reduces the cost of program maintenance.
2. Large community: Python has a large and active community, which means that there is a wealth of resources and support available.
3. Versatile: Python can be used for a wide range of tasks, including web development, scientific computing, data analysis, artificial intelligence, and more.
4. Dynamic typing: In Python, you do not need to declare the type of a variable, and the type can change at runtime. This makes it easier to write and debug your code.
5. Built-in libraries: Python comes with a large number of built-in libraries and modules, making it easy to perform common tasks and reducing the need to write code from scratch.
6. Interoperable: Python can be used in conjunction with other languages, such as C and C++, to extend existing applications or integrate with existing systems.
7. High-level: Python is a high-level programming language, which means that it abstracts away many of the low-level details that you would need to handle in other languages, such as memory management.

**Question No. 14: How is memory managed in Python?**

In Python, memory management is handled by a built-in garbage collector, which automatically frees up memory that is no longer being used by the program. The garbage collector tracks all objects in the system and periodically frees objects that are no longer referenced by the program.

This means that, unlike in languages like C or C++, you do not need to manually allocate and free memory in Python. This makes it much easier to write and maintain your code, as you do not need to worry about memory leaks or other memory-related issues.

However, it's important to note that, like any garbage collector, the Python garbage collector can introduce some overhead and performance penalties, especially when dealing with very large datasets or real-time applications.

**Question No. 15: How to install Python on Windows and set path Variables?**

Here are the steps to install Python in Windows and set the PATH environment variable:

1. Download the Python installer for Windows from the official Python website (<https://www.python.org/downloads/>).
2. Run the installer and follow the prompts to install Python. Make sure to select the option to add Python to the PATH environment variable during the installation process.
3. Open the Start menu and search for "Environment Variables".
4. Click on "Edit the system environment variables".
5. Click on the "Environment Variables" button.
6. Scroll down the list of "System Variables" and find the "Path" variable.
7. Click on the "Edit" button to edit the Path variable.
8. Add the path to the Python executable to the Path variable. For example, if you installed Python in the default location, the path should be C:\PythonXX\, where XX is the version number of Python.
9. Restart any open command prompts or terminal windows for the changes to take effect.
10. To verify that Python is installed and the PATH variable is set correctly, open a command prompt and type python. You should see the Python prompt and be able to run Python commands.

**Question No. 16: Is indentation required in Python?**

Yes, indentation is required in Python. In Python, indentation is used to define blocks of code, such as the body of a function, loop, or conditional statement. Unlike many other programming languages, Python does not use curly braces or keywords like "begin" and "end" to delimit blocks of code.