1. What are the new features added in Python 3.8 version?

* Assignment expressions:

Assignment expressions are a new feature in Python 3.8 that allow you to assign values to variables in a more concise and expressive way. For example, you can now write a = b = 1 instead of `a = 1; b = 1`.

* Positional-only parameters:

Positional-only parameters are a new feature in Python 3.8 that allow you to define parameters that must be passed by position and cannot be used as keyword arguments. This can be helpful for preventing errors and making your code more readable.

* f-strings:

F-strings are a new feature in Python 3.8 that allow you to embed expressions in strings in a more concise and expressive way. For example, you can now write f"The answer is {42}" instead of "The answer is %d" % 42.

* Walrus operator:

The walrus operator, :=, is a new feature in Python 3.8 that allows you to assign values to variables in an expression. For example, you can now write a := b = 1 instead of `a = 1; b = 1`.

* PEP 587:

PEP 587 is a new feature in Python 3.8 that allows you to configure the Python Initialization providing finer control on the whole configuration and better error reporting.

* New syntax warnings:

Python 3.8 adds new syntax warnings for some constructs that are deprecated or will be removed in future versions of Python. These warnings can help you to identify potential problems in your code and make it more future-proof.

* Performance improvements:

Python 3.8 includes a number of performance improvements that can make your code run faster. These improvements include a new tokenizer, a faster garbage collector, and improved performance for a variety of built-in functions.

* Type system improvements:

Python 3.8 includes a number of type system improvements that can make your code more robust and easier to analyze. These improvements include a new type annotation syntax, a new type checker, and improved type inference.

* Documentation improvements:

Python 3.8 includes a number of documentation improvements that can make it easier to learn and use Python. These improvements include a new online documentation site, a new tutorial, and improved documentation for a variety of modules and functions.

1. What is monkey patching in Python?

Monkey patching in python refers to modifying or updating a piece of code or class or any module at the runtime. In simple words, we can change the behavior or working of a class/ module at the runtime without changing the whole python code.

1. What is the difference between a shallow copy and deep copy?

In Shallow copy, a copy of the original object is stored and only the reference address is finally copied. In Deep copy, the copy of the original object and the repetitive copies both are stored.

1. What is the maximum possible length of an identifier?

The maximum possible length of an identifier in the Python language is 79 characters. Moreover, the PEP-8 prevents you from breaking the limit of 79 characters while creating identifiers.

1. What is generator comprehension?

A generator comprehension is a single-line specification for defining a generator in Python. It is absolutely essential to learn this syntax in order to write simple and readable code.