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

Ans:

a. **The Walrus Operator:** This operator (:=) is used for inline assignment expressions. It allows you to assign values to variables within larger expressions.

b. **Positional-Only Parameters**: Function parameters can now be designated as positional-only, meaning that they can only be supplied by position and not by keyword.

c. **f-strings now support the = specifier:** This allows you to specify a formatted string that will be used as the default value if the expression being formatted is None.

d. **The typing module has added Literal:** This allows you to specify a type that can only be one of several literal values.

e. **Improvements to the typing module:** Type hints now support protocols and structural subtyping.

f. **Parallel filesystem cache for compiled bytecode files:** Python 3.8 includes a new feature that allows it to store compiled bytecode files in a parallel filesystem cache, speeding up the import process.

g. **Debug Info for F-strings:** In Python 3.8, you can now enable debug information for f-strings using the -Xf option, which will display a warning if the format string is not a valid f-string.

h. **Performance Improvements:** Python 3.8 includes several performance improvements, including faster function calls, optimized dictionary access, and more efficient string formatting.

1. What is monkey patching in Python?

Ans: Monkey patching is a technique in Python where you dynamically modify a class or module at runtime by replacing, adding or removing attributes, methods or functions from it. It allows us to change the behavior of a module or class without modifying the original source code.

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

Ans: The difference between a shallow copy and a deep copy is that a shallow copy creates a new object that shares the contents of the original object, while a deep copy creates a new object with its own independent contents.

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

Ans: The maximum possible length of an identifier depends on the programming language being used.

For example, in Java, the maximum length of an identifier is 63 characters. In Python, the maximum length of an identifier is virtually unlimited, although it is generally recommended to keep identifiers relatively short for readability and maintainability purposes.

In general, most programming languages have a practical limit on the length of identifiers, usually ranging from around 30 to 80 characters. However, it is generally good practice to keep identifiers as short and meaningful as possible, to make the code easier to read and understand.

1. What is generator comprehension?

Ans: Generator comprehension is a concise way to create a generator in Python. It is similar to list comprehension, but instead of creating a list, it generates a sequence of values one at a time as required.

Generator comprehension is defined using parentheses instead of square brackets.

The key difference is that the generator comprehension does not create a list in memory. Instead, it generates each value on-the-fly as it is requested. This can be much more memory-efficient when dealing with large sequences of values.