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

Solution - Python 3.8, released on October 14, 2019, introduced several new features and improvements. Here are some notable additions in Python 3.8:
Assignment Expressions (the "walrus operator"): This feature introduced the := operator, allowing assignment within expressions. It enables the assignment of a value to a variable as part of a larger expression, reducing the need for multiple lines of code.
Positional-Only Parameters: Functions can now specify parameters that can only be passed by position and not by keyword. This allows developers to define functions with a clearer parameter usage and enforce a specific calling convention.
The "f-strings" = specifier: Python 3.8 introduced the = specifier in f-strings, which displays the value being formatted alongside the expression. This helps with debugging and understanding the evaluated value.
The math.prod() function: A new math.prod() function was added to the math module, simplifying the calculation of products of a sequence of numbers.
The "typed_ast" module: A new module named typed_ast was introduced, providing a Python parser that produces typed Abstract Syntax Trees (ASTs), enabling better static analysis and type checking of Python code.
The "statistics" mode() improvements: The statistics module now includes improvements to the mode() function, making it easier to find the most common value(s) in a dataset.
The "math.isqrt()" function: The math module introduced the isqrt() function, which calculates the integer square root of a non-negative integer, returning the largest integer less than or equal to the square root.

2. **What is monkey patching in Python?**

Solution - Monkey patching in Python refers to the practice of modifying or extending the behavior of an existing module, class, or object at runtime. It allows you to add, replace, or modify attributes, methods, or functions of an existing object without altering its original source code.

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

Solution - In Python, the concepts of shallow copy and deep copy refer to different ways of creating copies or clones of objects.

Shallow Copy:

A shallow copy creates a new object that references the original object's memory address. In other words, it creates a new object, but the content of the new object is still connected to the original object. Any changes made to the original object will also be reflected in the shallow copy.

Deep Copy:

A deep copy creates a completely independent copy of an object, including all nested objects. It creates a new object and recursively copies all the objects referenced by the original object. Any changes made to the original object will not affect the deep copy.

4. **What is the maximum possible length of an identifier?**

Solution - An identifier can have a maximum length of 79 characters in Python**.**

5. **What is generator comprehension?**

Solution - Generator comprehension, also known as generator expression, is a concise way to create a generator in Python. It allows you to define a generator using a similar syntax to list comprehensions, but with parentheses instead of square brackets.