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

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

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

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

**5. What is generator comprehension?**

**SOLUTIONS**

1. *Some of the new features added in Python 3.8 version include:*

* *The walrus operator (:=)*
* *Positional-only parameters*
* *The f-strings support "=" for self-documenting expressions and debugging*
* *The "breakpoint" function for debugging*
* *Performance improvements in the interpreter and built-in modules*

1. *Monkey patching in Python refers to the technique of modifying code at runtime by replacing or adding new methods or attributes to an object. It can be a powerful tool for debugging or adding functionality, but it can also make code harder to understand and maintain.*
2. *A shallow copy in Python creates a new object but shares the references to the original nested objects. A deep copy creates a new object and recursively copies all the nested objects as well. This means that changes made to the nested objects in a deep copy will not affect the original object, while changes in a shallow copy may.*
3. *In Python, the maximum possible length of an identifier is implementation-dependent, but it is usually limited to 255 characters. However, it is generally a good practice to keep identifiers short and descriptive.*
4. *A generator comprehension in Python is a concise way to create a generator object, which generates a sequence of values on-the-fly. It has a similar syntax to list comprehensions but uses parentheses instead of brackets. For example,* ***(x\*\*2 for x in range(10))*** *generates the sequence of squares of the numbers from 0 to 9.*