**Conceptual Questions:**

**1 . What is the purpose of using control flow statements like if, else, and elif in Python?**

* **Ans : Control flow statements like if, else, and elif in Python allow you to control the execution of your code based on certain conditions. They help you make decisions and execute different blocks of code depending on whether a condition is true or false.**

**Qno** **2 : How does Python determine which block of code to execute in an if-else statement?**

* **Ans :** **Python** **determine** **block of code to execute in an if-else** **statement .when the condition is true, the code within the “If” block is executed. If the condition is false, the code within the “Else” block is executed.**

**Qno** **3 : Explain the difference between the if-elif-else and nested if-else structures.**

* **Ans: The if-elif-else structure allows you to test multiple conditions sequentially. It checks each condition in order, and if one is true, it executes the corresponding block and skips the rest. If none are true, the else block is executed.On the other hand, nested if-else structures involve placing one if-else construct inside another. It provides a way to handle multiple scenarios in a more intricate manner.**
* **Ans : We can use logical** **operators to create more complex conditions in your if statements. Python supports three logical operators:** **and ,** **or , and** **not . and operator: Returns True if both conditions are true. or operator: Returns True if at least one of the conditions is true.**

**Qno 5 : Describe scenarios where nested if-else statements are preferred over if-elif-else structures.**

* **Ans : Nested if-else statements may be preferred over if-elif-else structures in scenarios where you need to check multiple conditions independently, and the conditions are not mutually exclusive. For example, if each condition requires a unique set of actions, using nested if-else statements can provide more flexibility in handling various cases.**

**Qno** **6 : How does Python handle multiple conditions in an if-elif-else** **ladder ?**

* **In Python, an if-elif-else ladder is processed sequentially. When the interpreter encounters such a structure, it evaluates the conditions one by one in order, from top to bottom. If a condition is true, the corresponding block of code is executed, and the rest of the ladder is skipped.**

**Qno** **7 : Why is it important to indent properly when using control flow statements in Python?**

* **Ans : Indentation is a very important concept of Python because without properly indenting the Python code, you will end up seeing Indentation Error and the code will not get** **compiled .**