---

1. What is Python, and why is it popular?

Python is a high-level, interpreted programming language known for its clean syntax and readability.
It supports multiple programming paradigms, including procedural, object-oriented, and functional.
It is popular due to its simplicity, large community, and strong libraries for data science, web development, AI, and more.

---

2. What is an interpreter in Python?

An interpreter is a program that executes code line by line at runtime.
Python uses an interpreter instead of a compiler, which makes debugging easier.
This allows quick testing and immediate execution of Python scripts.

---

3. What are pre-defined keywords in Python?

Pre-defined keywords are special reserved words in Python with predefined meanings.
They are used to define the structure and syntax of the Python language.
Examples include if, else, while, for, def, and import.

---

4. Can keywords be used as variable names?

No, keywords cannot be used as variable names in Python.
They are reserved for specific programming purposes and cause errors if used improperly.
For example, using for = 5 will raise a syntax error

---

5. What is mutability in Python?

Mutability refers to the ability of an object to be changed after it is created.
Mutable objects like lists and dictionaries can be altered in place.
Immutable objects like strings, integers, and tuples cannot be changed after creation.

---

6. Why are lists mutable, but tuples are immutable?

Lists are mutable because they are meant for dynamic data that can change over time.
Tuples are immutable for use cases where data should remain constant and safe from modification.
This design allows tuples to be used as dictionary keys and in sets.

---

7. What is the difference between == and is operators in Python?

== checks if two variables have the same value.
is checks if two variables refer to the exact same object in memory.
For example, two lists with the same elements may be == but not is.

---

8. What are logical operators in Python?

Logical operators are used to combine conditional statements.
Python has three logical operators: and, or, and not.
They return Boolean values (True or False) based on the conditions.

---

9. What is type casting in Python?

Type casting means converting one data type to another.
For example, converting a string "10" to an integer using int("10").
It helps in handling user input, arithmetic, and data processing.

---

10. What is the difference between implicit and explicit type casting?

Implicit casting is done automatically by Python when combining compatible types.
Explicit casting requires the programmer to manually convert types using functions like int(), str().
Explicit casting gives more control but may raise errors if done incorrectly.

---

11. What is the purpose of conditional statements in Python?

Conditional statements help control the flow of the program based on certain conditions.
They allow the program to make decisions using if, elif, and else.
This makes programs dynamic and responsive to different inputs or states.

---

12. How does the elif statement work?

The elif keyword stands for "else if".
It allows checking multiple conditions after the initial if statement.
If the first condition is false, Python checks the elif blocks one by one.

---

13. What is the difference between for and while loops?

A for loop is used when you know the number of iterations in advance, often used to loop through sequences.
A while loop continues as long as a condition is true, regardless of how many times it runs.
Use for for definite loops, and while for indefinite ones.

---

14. Describe a scenario where a while loop is more suitable than a for loop

A while loop is more suitable when you don’t know how many times the loop will run.
For example, reading user input until they type "exit".
It keeps running until the condition is no longer true.