1. What is dynamic typing in Python?

Dynamic typing means **you don’t need to declare the data type of a variable**.
Python decides the type **at runtime**, and the same variable can change type.

```python
x = 10         # int
x = "hello"   # str
```

2. Difference between mutable and immutable data types

Type	Meaning	Examples
Mutable	Can be changed after creation	list, dict, set
Immutable	Cannot be changed after creation	int, float, str, tuple

```python
lst = [1, 2]
lst.append(3)   # allowed

s = "hi"
s[0] = "H"      # error (immutable)
```

3. What are local and global variables?

- **Local variable**: Defined inside a function, accessible only there
- **Global variable**: Defined outside a function, accessible everywhere

```python
x = 10  # global

def func():
    y = 5  # local
    print(y)
```

The **`return`** statement is used inside a function to **send a result back to the caller** and **end the function’s execution**.

5. What is list comprehension?

A **short and clean way** to create lists using a single line.

✅ Basic Syntax

python

```
new_list = [expression for item in iterable if condition]
```

```
squares = [x*x for x in range(5)]
```

Equivalent to:

python

```
squares = []
for x in range(5):
    squares.append(x*x)
```

🔷 List Comprehension with Condition

```python
even_numbers = [x for x in range(10) if x % 2 == 0]
```

Output:

```csharp
[0, 2, 4, 6, 8]
```

🔷 List Comprehension with `if-else`

```python
result = ["Even" if x % 2 == 0 else "Odd" for x in range(5)]
```

Output:

```css
['Even', 'Odd', 'Even', 'Odd', 'Even']
```

6. Difference between `break`, `continue`, and `pass`

Keyword	Use
`break`	Stops the loop completely
`continue`	Skips current iteration
`pass`	Does nothing (placeholder)

7. What is recursion? Give example

Recursion is when a **function calls itself** to solve a problem.

```
def factorial(n):
    if n == 1:
        return 1
    return n * factorial(n-1)
```

8. What is a lambda function?

A **lambda function** is a **small, anonymous (nameless) function** written in **one line** using the `lambda` keyword.

- Can take **any number of arguments**
- Must contain **only one expression**
- Automatically returns the result

9. Difference between `is` and ==

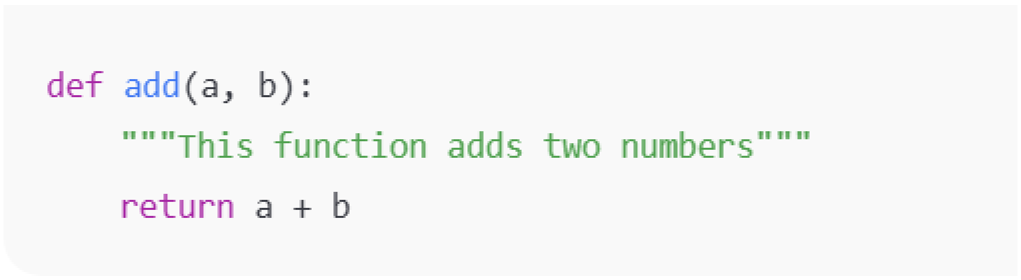
Operator	Checks
`==`	Value equality
`is`	Memory (object) identity

```
a = [1,2]
b = [1,2]

a == b    # True
a is b    # False
```

10. What are docstrings and why are they important?

Docstrings are **documentation strings** written inside functions/classes to explain their purpose



```python
def add(a, b):
    """This function adds two numbers"""
    return a + b
```