**📌 1. What is match in Python?**

* Introduced in **Python 3.10**, match is similar to switch-case in other languages (like C++/Java).
* It allows **pattern matching**: matching a value or structure and executing corresponding code.

**📌 2. Syntax**

match variable:

case pattern1:

# code block

case pattern2:

# code block

case \_:

# default block (like "else")

**📌 3. Basic Example**

command = "start"

match command:

case "start":

print("System starting...")

case "stop":

print("System stopping...")

case \_:

print("Unknown command")

**Output:** System starting...

* \_ is a **wildcard** (matches anything) → acts like default.

**📌 4. Multiple Matches**

You can match **multiple values** in a single case:

status = "success"

match status:

case "success" | "ok":

print("Operation succeeded")

case "error" | "failed":

print("Operation failed")

**📌 5. Matching with Variables (Binding)**

You can bind matched values to a variable:

data = ("error", 404)

match data:

case ("error", code):

print(f"Error code: {code}")

case \_:

print("Unknown")

**Output:** Error code: 404

**📌 6. Using Conditions (if) in Cases (Guards)**

number = 10

match number:

case n if n > 0:

print("Positive")

case n if n < 0:

print("Negative")

case 0:

print("Zero")

**📌 7. Matching Data Structures**

match works with **lists, tuples, and dictionaries**:

point = (0, 5)

match point:

case (0, y):

print(f"Point on Y-axis at {y}")

case (x, 0):

print(f"Point on X-axis at {x}")

case (x, y):

print(f"Point is at ({x}, {y})")

**Output:** Point on Y-axis at 5

**📌 8. Nested Matching**

data = {"user": {"name": "Alice", "age": 25}}

match data:

case {"user": {"name": name, "age": age}}:

print(f"User {name} is {age} years old")

case \_:

print("No user data")

**📌 9. Key Points**

1. match is more powerful than switch because it can match **structures and patterns**, not just values.
2. \_ is a wildcard; it ignores values (like default).
3. Can use | (OR) for multiple matches.
4. Guards (if) allow conditions inside cases.

**📌 10. Example: Calculator Using match**

operation = "+"

a, b = 5, 3

match operation:

case "+":

print(a + b)

case "-":

print(a - b)

case "\*":

print(a \* b)

case "/":

print(a / b)

case \_:

print("Invalid operation")

**⚡ When to Use?**

* When you have many conditions based on the **value/structure of a variable**.
* When working with structured data like tuples, dicts, lists.