🐍 Session 15: Object-Oriented Programming (OOP)

🤹 Topic: Classes and Objects

🎯 Learning Goals

By the end of this session, you will: ✓ Understand what **classes** and **objects** are.
✓ Know how to use **attributes** and **methods**.
✓ Understand the difference between **instance variables** and **class variables**.
✓ Write your own fun OOP-based programs.

🧩 1. What is OOP?

Imagine you are **playing with LEGO blocks** 🧱.

- Each LEGO piece has some properties: color, size, shape.
- When you combine them, you create **objects** like a car, house, or robot.

In Python:

- **Class** = The **blueprint** (like the design of a LEGO house 🏠).
- **Object** = The **actual thing built** from the blueprint (your LEGO house).

👉 Example: "Car" is a **Class**.
👉 A red Tesla, a black BMW are **Objects** of the Car class.

📘 2. Defining a Class

A class is created using the `class` keyword.

```
# A very simple class
class Student:
    pass   # pass means "do nothing" (we will fill this later)

# Creating an object
s1 = Student()
print(s1)
```

```
class Student:
    # Constructor: runs automatically when an object is created
    def __init__(self, name, age):
        self.name = name    # instance variable
        self.age = age

    # Method
    def display(self):
        print(f"Name: {self.name}, Age: {self.age}")
```

⚡ 3. Attributes and Methods

Attributes = data (variables) Methods = behavior (functions)

💡 Fun analogy:

- Attribute → A person's name, age, height.
- Method → Things they can do: eat(), sleep(), study().

🏗️ 4. Instance vs Class Variables

- Instance Variable → Belongs to each object separately.
- Class Variable → Shared by all objects of the class.

```
# Creating objects
s1 = Student("Rahul", 20)
s2 = Student("Sonia", 21)
```

```
s1.display()
```

```
Name: Rahul, Age: 20
Name: Sonia, Age: 21
```

```
class Student:
    college = "ABC University"   # class variable (common for all)

    def __init__(self, name, roll):
        self.name = name    # instance variable
        self.roll = roll
```

```
s1 = Student("Rahul", 101)
s2 = Student("Sonia", 102)

print(s1.name, "studies at", s1.college)
print(s2.name, "studies at", s2.college)
```

🏗️ 4. Instance vs Class Variables

Instance Variable: Belongs to each object separately.

Class Variable: Shared across all objects of the class.

🔄 5. Example: Real-World Object

```
class Car:
    wheels = 4   # class variable

    def __init__(self, brand, color):
        self.brand = brand
        self.color = color

    def drive(self):
        print(f"{self.color} {self.brand} is driving 🚗")

c1 = Car("Tesla", "Red")
c2 = Car("BMW", "Black")

c1.drive()
c2.drive()
print("Wheels in each car:", c1.wheels)
```

📝 6. Practice Problems

- Q1. Create a class Book with attributes: title, author, price. Add a method to display book details.
- Q2. Create a class Rectangle with attributes: length and width. Write methods to calculate area and perimeter.
- Q3. Create a class BankAccount with:

Attributes: account holder name, balance

Methods: deposit(amount), withdraw(amount), display_balance()