Introduction to OOP: Classes, Objects, Attributes, and Methods (1 hour) 🏫

What's OOP?

Imagine everything as objects — a car 🚗, a book 📚, or even a cute little puppy 🐶. OOP helps us model complex systems by treating real-world entities as objects within the program. We assign them **classes** (their blueprint), **attributes** (their characteristics), and **methods** (their actions)!

Key Concepts:

- **Class**: The blueprint or prototype, like a recipe 🍲! Defines the structure for creating objects.
- **Object**: A specific instance of a class. Think of it as a cake made from that recipe 🎂.
- **Attributes**: Characteristics or properties of an object, like the color of a car.
- **Methods**: Actions or behaviors that the object can perform (e.g., `drive()` for a car).

Example: Create Your First Class!

```
# Defining a class
class Car:
    color = "red"  # Attribute

    # Method
    def drive(self):
        print("The car is driving 🚗")

# Creating an object
my_car = Car()
print(my_car.color)
my_car.drive()
```

DOWNLOADED FROM CODING WITH TAWFIQ