

# AP Computer Science A

## Lesson 2

### Classes And Objects

- Object
  - A specific **instance** of a class, representing a real-world entity with properties (data) and actions (methods).
  - Created using the **new** keyword, e.g., a specific car like "red Toyota" is an object of the Car class.
- Class
  - A template or **blueprint** that defines the **properties (fields)** and **actions (methods)** for creating objects.
  - Acts like a recipe for objects, e.g., the Car class defines what all cars have (like color) and do (like drive).
- Public
  - **An access modifier** that allows fields or methods to be accessed from **anywhere** in the program.
  - Used for parts of a class that other classes need to access, e.g., a public method to get a car's speed.
- Private
  - **An access modifier** that restricts fields or methods to be accessible **only within the same class**.
  - Protects data by hiding it from outside access, e.g., a private field like a car's serial number.
- Static
  - A **keyword** indicating a field or method belongs to the class **itself**, not to any specific object.
  - Shared across all objects of the class, e.g., a static counter for the total number of cars created.

- 
- Method
    - A **block of code** in a class that defines a specific action or behavior an object can perform.
    - Can take inputs (parameters), return outputs, or perform tasks, e.g., a drive() method for a car.
  - Method Overloading
    - Creating multiple methods with the **same name** but different parameter types or counts.
    - Allows flexibility, e.g., an add() method for integers and another for doubles.
  - Scope
    - The **region** of code where a variable or method is accessible.
    - Includes class scope (class-wide variables), method scope (variables inside a method), and block scope (variables inside a loop or if-statement).
  - Reference
    - A variable that stores the **memory address** of an object, not the object's actual data.
    - Used for complex types like String or custom objects; a null reference points to no object.
  - Null Reference
    - A reference variable that **does not** point to any object in memory.
    - Represented by null, e.g., String name = null; means name has no value yet.
  - this
    - A **keyword** referring to the **current** object of the class in which it is used.
    - Used to distinguish instance variables from parameters or local variables with the same name, e.g., this.name = name; in a constructor assigns the parameter name to the class's name field.

## Keywords

- import, class
- public, private, static, void, final
- new, this, return
- if, else, for, while, do
- int, double, boolean, char, null