Object Oriented Programming
Introduction

CS(217) Object Oriented Programming

Abeeda Akram

Procedural or Structured Programming

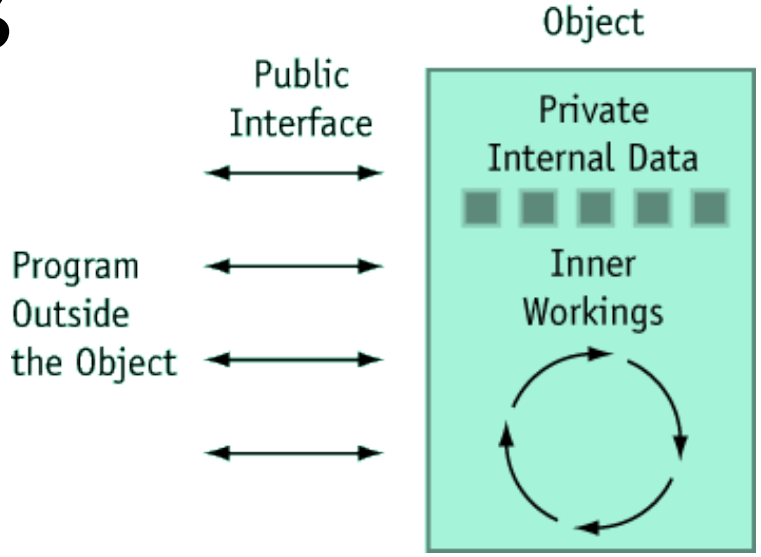
- Centered around procedures (**functions**)
  - Take data as parameters
  - Process data
  - Return results
- Provide procedural abstraction
  - Hide low level implementation details from user
  - Examples:
    - **Search** (Linear or Binary)
    - **Sorting** (Bubble, Selection , Even-Odd, Insertion)
    - **Square Root Calculation**
    - **Prime Numbers**

What's Wrong with Procedural Programming?

- **Complex** programs
- Programs are difficult to **modify** and **extend**
- Examples Management Systems
  - Student
  - Health Care
  - Inventory
  - Library

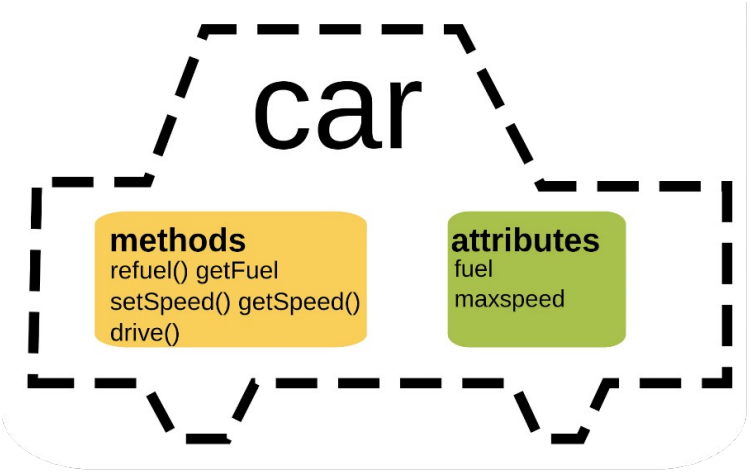
Object Oriented Programming

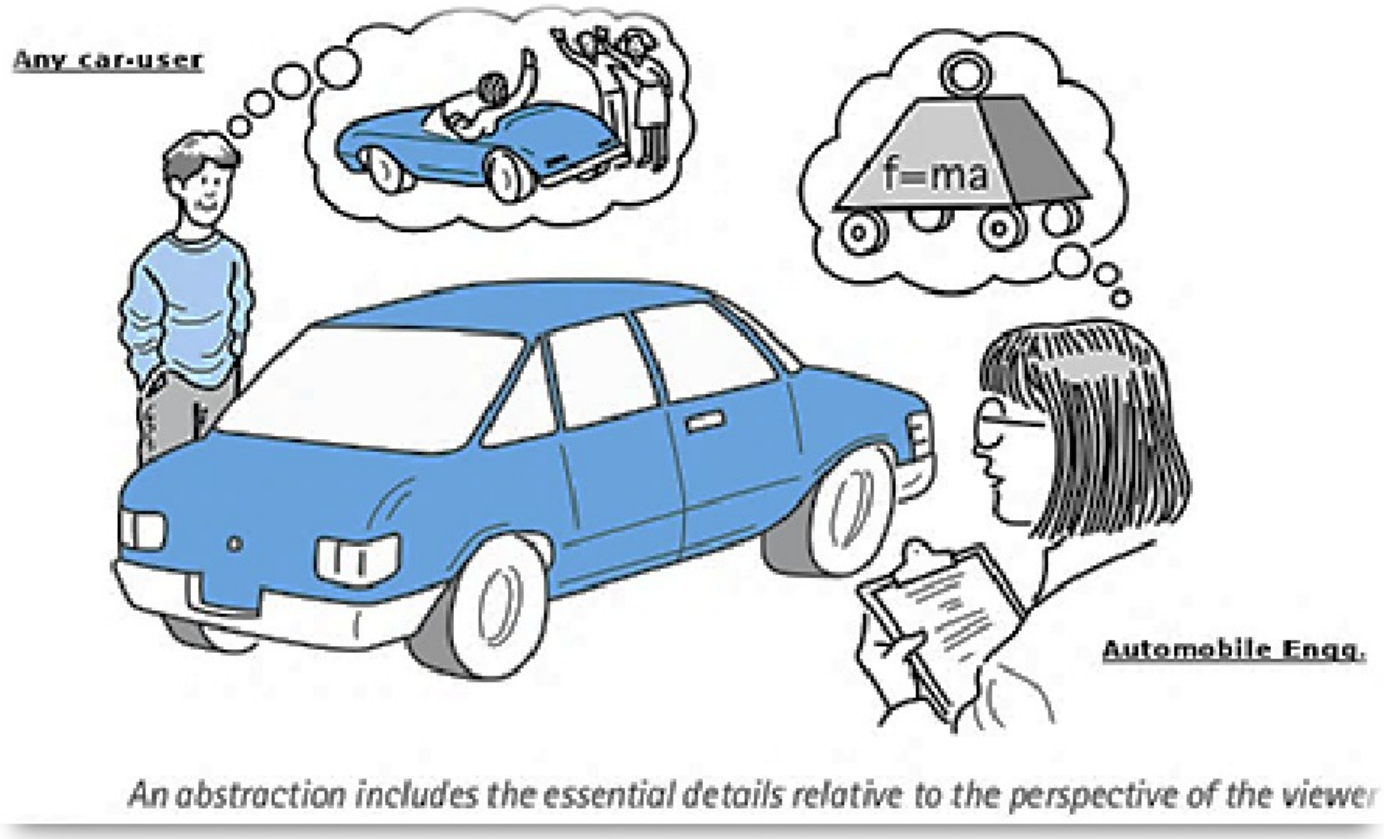
- Centered around Objects
- An Object has combined
  1. Data storage
  2. Functions
- **Provides Encapsulation**
  - Information and implementation hiding
- **Provides Abstraction**
  - Data abstraction
    - **Hide low level storage details**
  - Procedural abstraction
    - **Hide implementation of an action and low level functional details**



Object Oriented Programming

- **User is concerned with only working**
  - **What it takes?**
  - **What it does?**
  - **What it returns?**
- **Hide How it actually do that?**





Object Oriented Programming Use

- Commonly used to create user defined abstract data types (**ADTs**)
  - Which are very specific Or very general in purpose

1. **Improve built in C++ data types**
   - Arrays (with boundary checks),
   - String (with additional Functionalities)
2. **Create missing datatypes from C++** as if they were built-in data types.
   - Matrices (Simple, Triangular, Sparce)
   - Sets
3. **Creating objects that perform commonly needed tasks**
   - Input validation
   - Screen output in a graphical user interface
   - Voice processing

Object Oriented Programming Use

Data types **created for a specific application**.

For example,

- **Student Management System**
  1. Student
  2. Course
- **Hospital Management System**
  1. Doctor
  2. Patient
  3. Appointments