

OOP (OBJECT - ORIENTED PROGRAMMING) - CLASSES

OOP - OBJECT ORIENTED PROGRAMMING

- Is more than a way of coding – it is a way of thinking
- Everything is an **object**
 - Those objects have **methods** (things they can do)
 - Interact with other objects



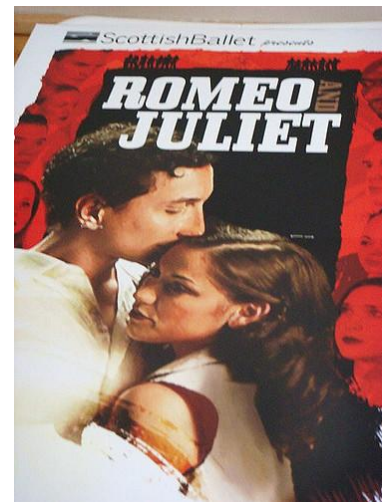
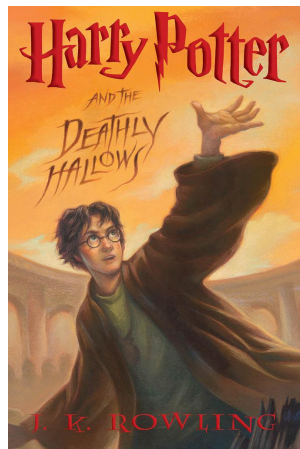
OOP

- Code is based on these interactions
- NOT inherently based on a logical series of steps



OOP - OBJECTS

- Objects are items with similar properties
 - Ex: Book object = “Romeo & Juliet”, “Watchmen”, “Harry Potter”

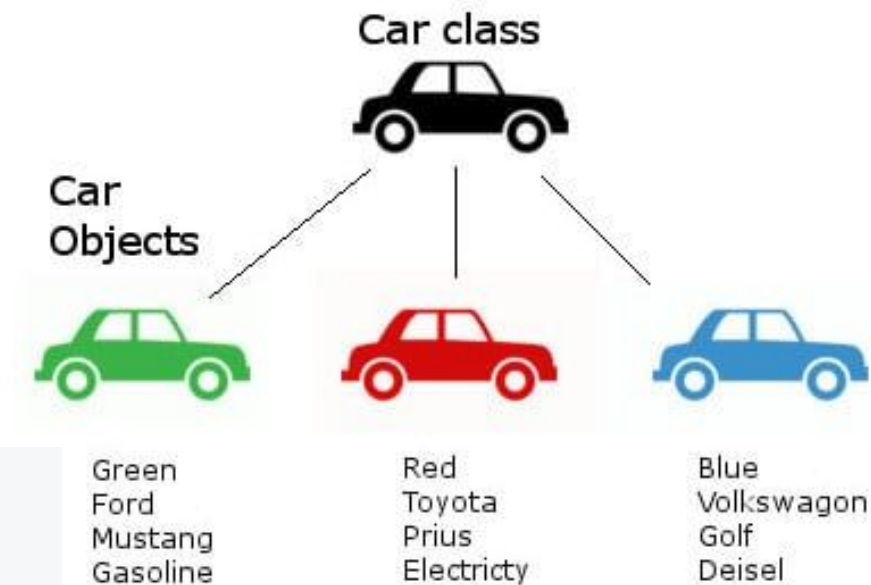
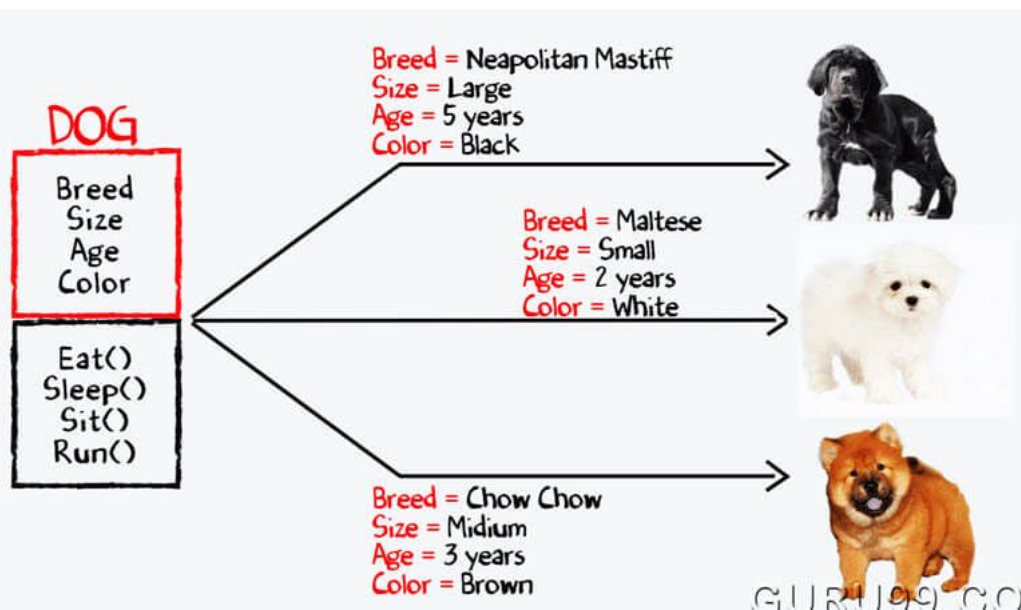


OOP - CLASSES



- In C++ (and most other OOP languages), objects are defined in code by **classes**
- Classes provide a framework to create the variables to represent objects
 - There is a “Book” class that generally represents all books

OBJECTS & CLASSES



CLASS VARIABLES

- Each class has certain variables
 - Belong to EVERY object that belongs to that class
 - Each object has different *instances* of each variable
 - (This is how objects are differentiated)



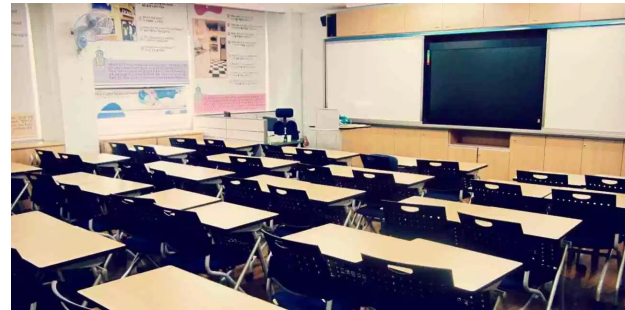
CLASS METHODS

- Each class has certain methods
 - Each method is something that an object can do
 - ALL objects have these class methods

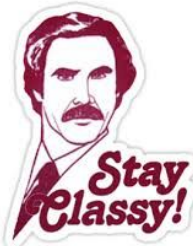
```
Eat()  
Sleep()  
Sit()  
Run()
```


CLASS VARIABLES

- Using the “classroom” class on the board, make class variables && methods for each class (student, teacher, computer)
 - Work in groups of 2-3
 - Write them down (will eventually be submitted as a group grade)



OOP - CLASS ABSTRACTION



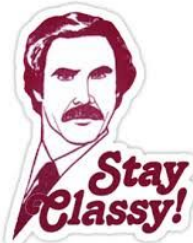
- Abstraction: Amount of data visible to the user of an object
 - Full abstraction = ALL data is visible
 - Limited abstraction = SOME data is visible
 - No abstraction = NO data is visible

OOP - CLASS ABSTRACTION



- Classes: Limited abstraction
 - Can define variables and methods as either **private** or **public**
 - Private → only visible to the object itself
 - Public → visible to everyone

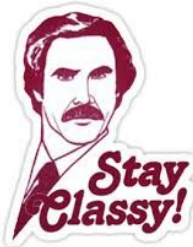
OOP - CLASSES



- Define `public` variables / methods under the *public* keyword
- Same for *private* variables / methods

```
class movie {  
    public:  
        string getActors();  
        string getActress();  
        void setRelease();  
        double getRevenue();  
    private:  
        string actor;  
        string name;  
        int release;  
        double revenue;  
};
```

OOP- CLASSES



- Standards: Make variables private, and make **methods** public
 - Allows methods to be the **interface** between the object variables and the code



ACTIVITY 2.0



- Create a set of classes (with public methods and private variables) that define the structure of a restaurant
 - Choose the restaurant & setup → this is entirely up to you
 - Will be submitted as a homework grade
 - Graded on ability to define variables and methods that describe a class