OOP'S FEATURES

ALFA FARIDH SUNI

PEMROGRAMAN BERORIENTASI OBJEK

2020

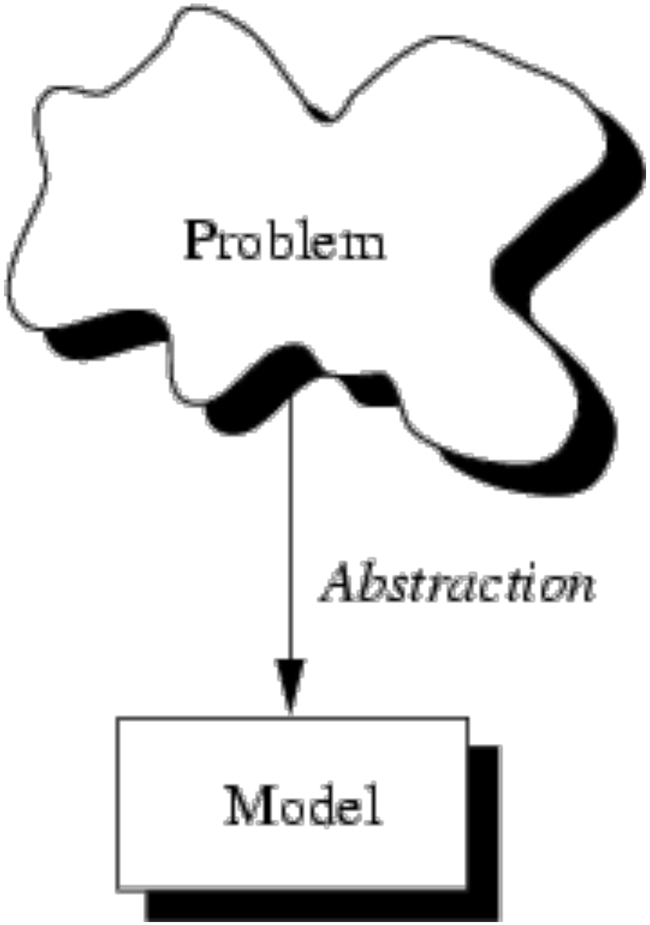
LET'S WATCH A VIDEO

SUMMARY FROM VIDEO

OOP Features: APIE

- Abstraction
- Polymorphism
- Inheritance
- Encapsulation

ABSTRACTION



ABSTRACTION

Abstraction: to understand the problem to separate necessary from unnecessary details.

Abstraction is the structuring of a nebulous problem into well-defined entities by defining their data and operations.

What we get after we do an abstraction:

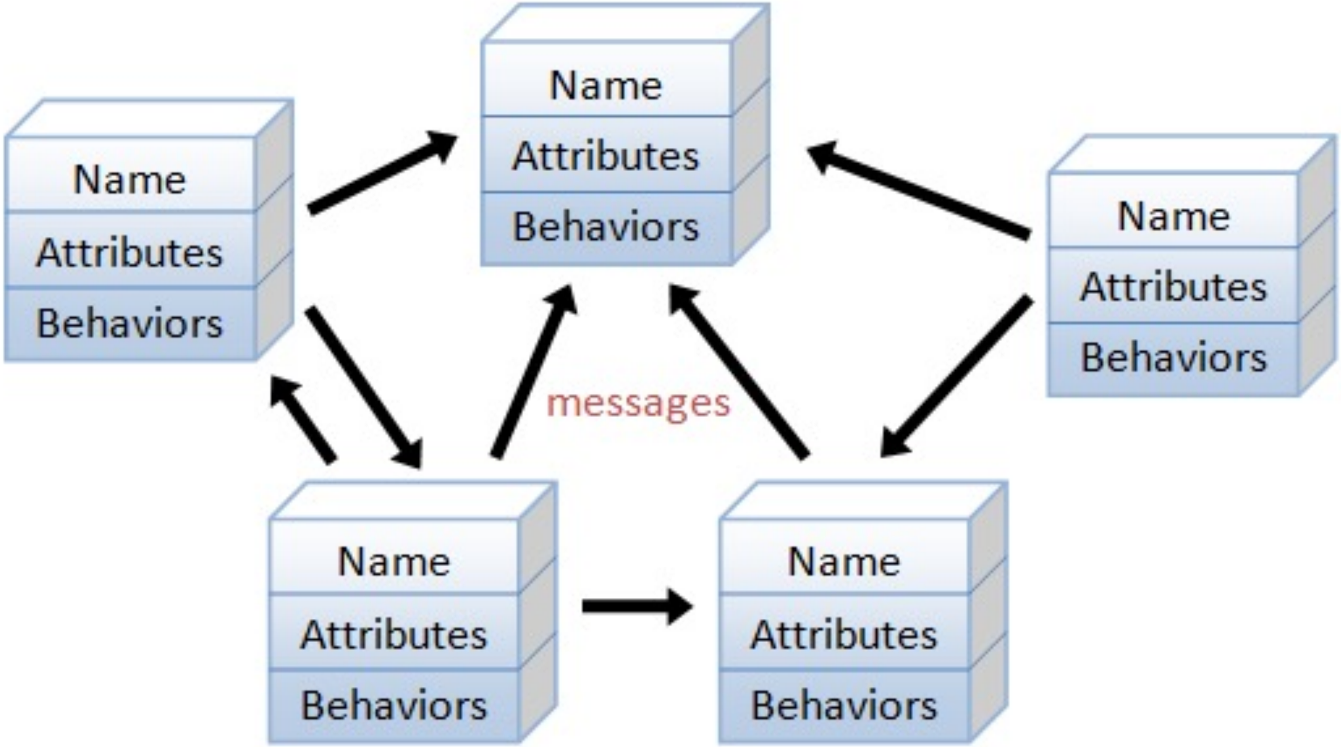
Model of the problem’s solution → Class

Well defined class consist of: data & operation (attributes & methods of class)

ENCAPSULATION

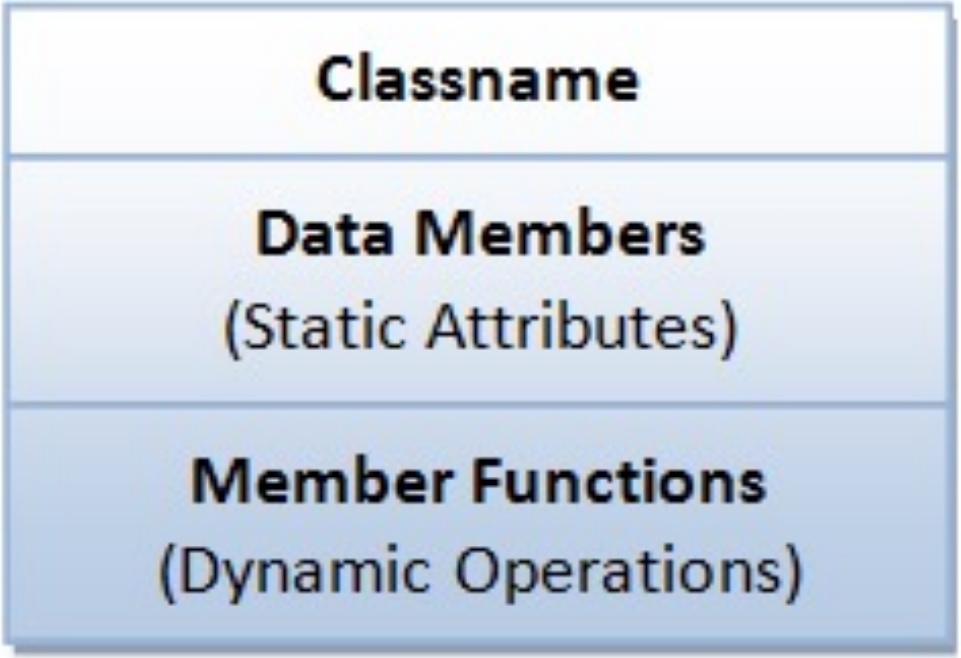
- The basic unit of OOP is a ***class***, which ***encapsulates*** both **the *static attributes* and *dynamic behaviors* within a "box"**, and specifies **the public interface** for using these boxes.
- **OOP combines the data structures and algorithms** of a software entity inside the same box.

ENCAPSULATION



An object-oriented program consists of many well-encapsulated objects and interacting with each other by sending messages

CLASS ENCAPSULATION



Classname

Data Members
(Static Attributes)

Member Functions
(Dynamic Operations)

A class is a 3-compartment box encapsulating data and functions

ACCESS MODIFIER

Type of access modifier:

1. **Public**: Accessible from any class
2. **Private**: Only accessible from within class
3. **Protected**: Accessible from subclass/child class (**we will discuss later in inheritance**)

INHERITANCE

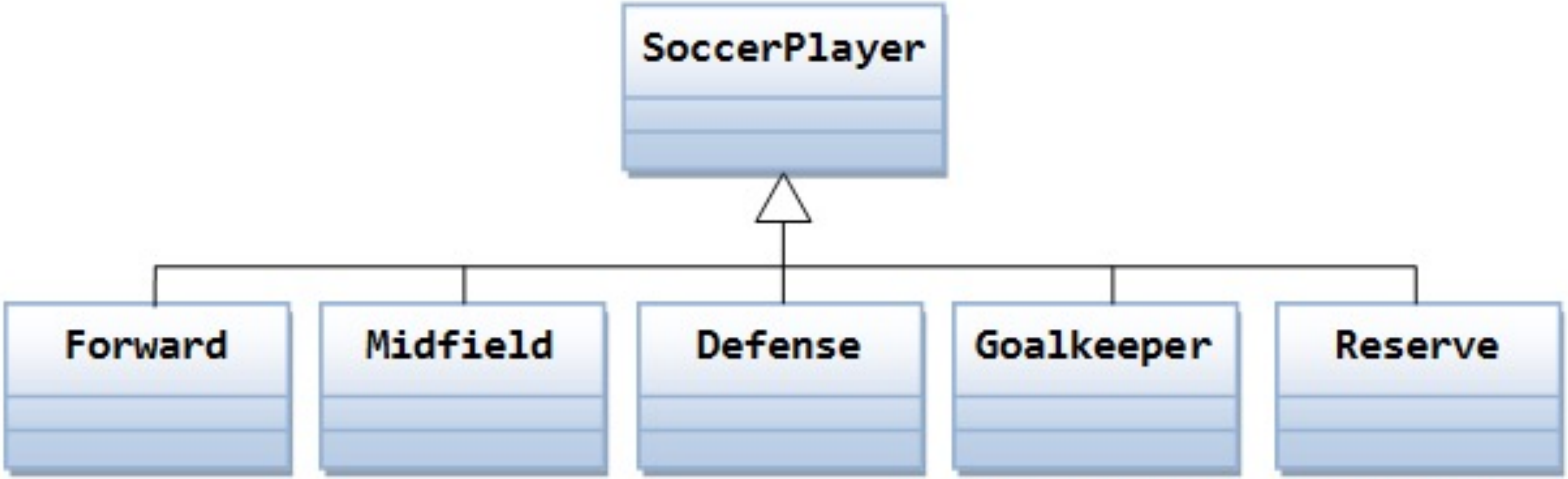
In OOP, we often organize classes in hierarchy to avoid duplication and reduce redundancy.

The classes in the lower hierarchy inherit all the variables (static attributes) and methods (dynamic behaviors) from the higher hierarchies.

A class in the lower hierarchy is called a subclass (or derived, child, extended class). A class in the upper hierarchy is called a superclass (or base, parent class).

By pulling out all the common variables and methods into the superclasses, and leave the specialized variables and methods in the subclasses, redundancy can be greatly reduced or eliminated as these common variables and methods do not need to be repeated in all the subclasses

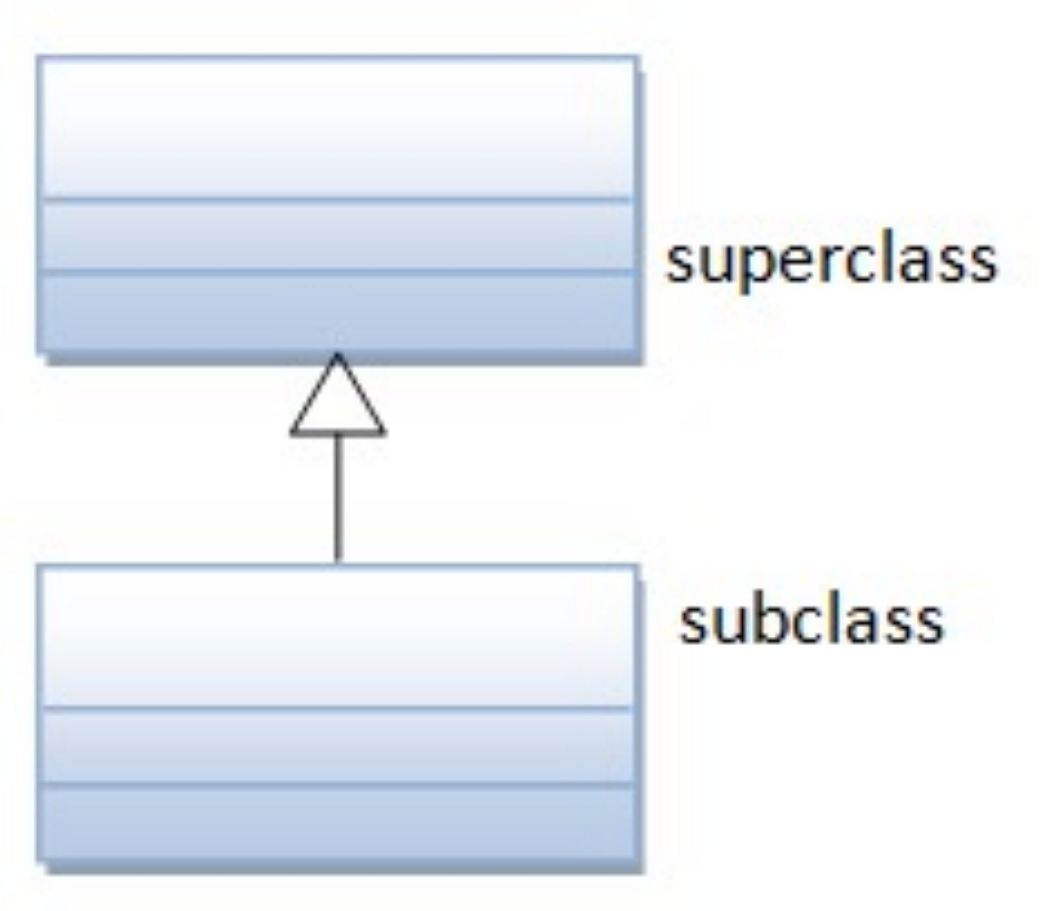
INHERITANCE



Can use hands

Not playing

SUPERCLASS & SUBCLASS



A subclass inherits all the variables and methods from its superclasses

INHERITANCE SYNTAX

```
// in C++
class SubclassName : inheritance-access-specifier SuperclassName {
   // your subclass code here
   . . . .
};

// in Java
class SubclassName extends SuperclassName {
   // your subclass code here
   . . . .
};
```

POLYMORPHISM

Poly : many,

Morphos: form

Object can take many form.

POLYMORPHISM (IN C++)

Polymorphism works on object pointers and references using so-called dynamic binding at run-time. It does not work on regular objects, which uses static binding during the compile-time.

We typically allocate object dynamically via the new operator and manipulate the return pointer in polymorphism. Recall that we can dynamically allocate objects for the Point and MovablePoint classes as follows:

SUBSTITUTION (IN C++)

A subclass instance inherits all the properties of the superclass, in the case of public-inheritance. It can do whatever a superclass instance can do. This is known as a "is-a" relationship. Hence, you can substitute a subclass instance to a superclass reference.

SUBSTITUTION (IN C++)

1. A subclass instance can be substituted for a superclass reference.
2. Once substituted, only the superclass' functions can be called, no the subclass'.
3. If the subclass overrides a superclass function. We wish to run the overridden version in the subclass, instead of the superclass' version (as in the previous example).

VIRTUAL FUNCTION (IN C++)

Virtual Functions: To implement polymorphism, we need to use the keyword virtual for functions that are meant to be polymorphic. In this case, if a superclass pointer is aiming at a subclass objects, and invoke a virtual function that is overridden by the subclass, the subclass version will be invoked, instead of the superclass version.