Object Orientation

Objectives

- Appreciate OOPs concepts

Table of Content

Feature: Object oriented programming

> *Object-oriented programming is a method of implementation in which programs are organized as cooperative collections of objects, each of which represents an instance of some class, and whose classes are all members of a hierarchy of classes united via inheritance relationships.*
> - Grady Booch

Terms

- Object
- Class
- Abstraction
- Encapsulation
- Inheritance

What is OO ?

- Been since 1970
- Implemented by languages : C++,Java,SmallTalk,Eiffel
- View things in the world as objects

Benefits Guaranteed

- Reusability & Extensibility
- Reduction in Maintenance cost
- Decreases development time
- Increases profit
- Easy to understand

Object

- A thing in a real world that can be either physical or conceptual. An object in object oriented programming can be physical or conceptual.
- Conceptual objects are entities that are not tangible in the way real-world physical objects are.
- Bulb is a physical object. While college is a conceptual object.
- Conceptual objects may not have a real world equivalent. For instance, a Stack object in a program.
- Obj has state and behavior.

What is the state and behavior of this bulb?

Attributes and Operations

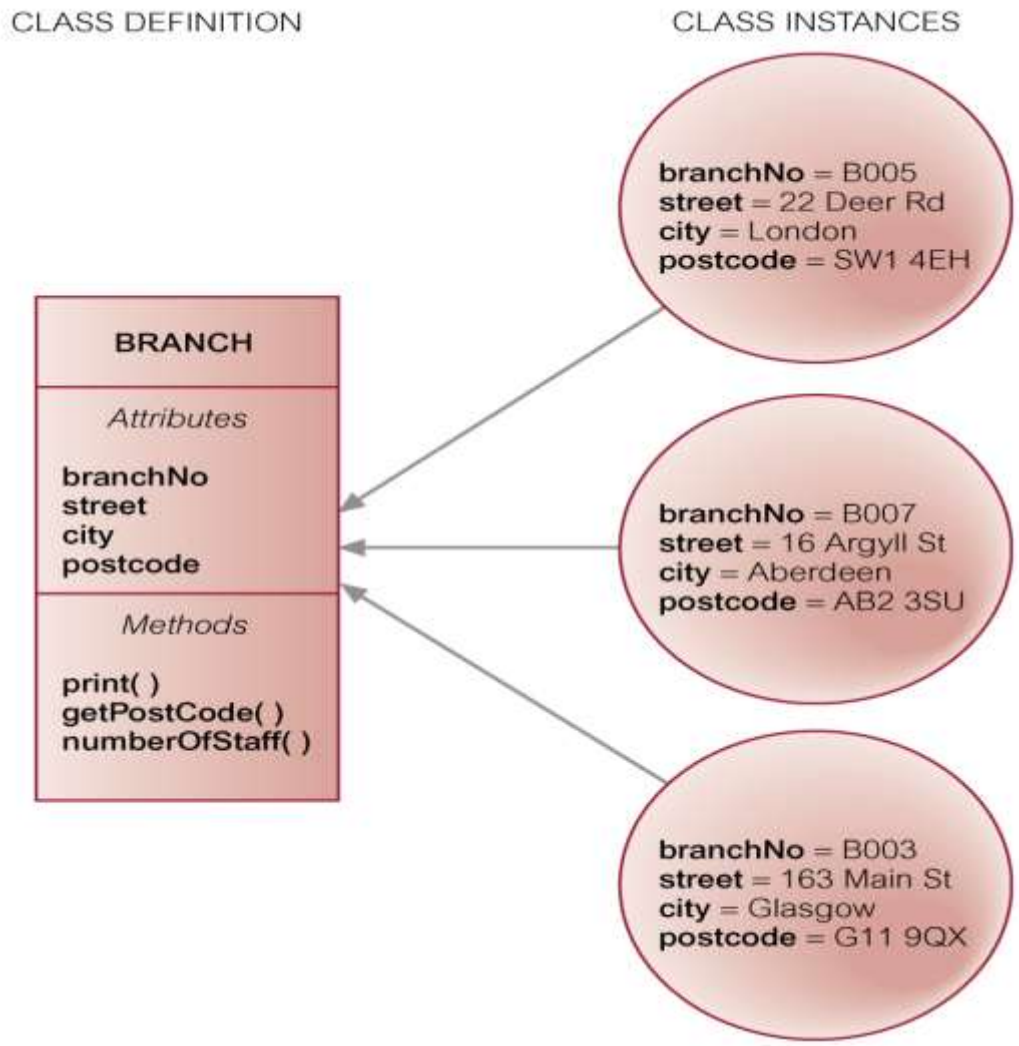
- The object's state is determined by the value of its properties or attributes.
- Properties or attributes → member variables or data members
- The object's behaviour is determined by the operations that it provides.
- Operations → member functions or methods

Class

- A class is a construct created in object-oriented programming languages that enables creation of objects.
- Also sometimes called blueprint or template or prototype from which objects are created.
- It defines members (variables and methods).
- A class is an **abstraction.**

Class: Blueprint for defining a set of similar objects.

Objects created in a class are called *instances*.



Putting it together

A bulb:

1. It's a real-world thing.
2. Can be switched on to generate light and switched off.
3. It has real features like the glass covering, filament and holder.
4. It also has conceptual features like power.

5. A bulb manufacturing factory produces many bulbs based on a basic description / pattern of what a bulb is.

object

methods

member variables

class

Abstraction

Abstraction denotes essential characteristics of an object that distinguish it from all other kinds of objects and thus provide crisply defined conceptual boundaries, relative to the perspective of the viewer.
-Grady Booch

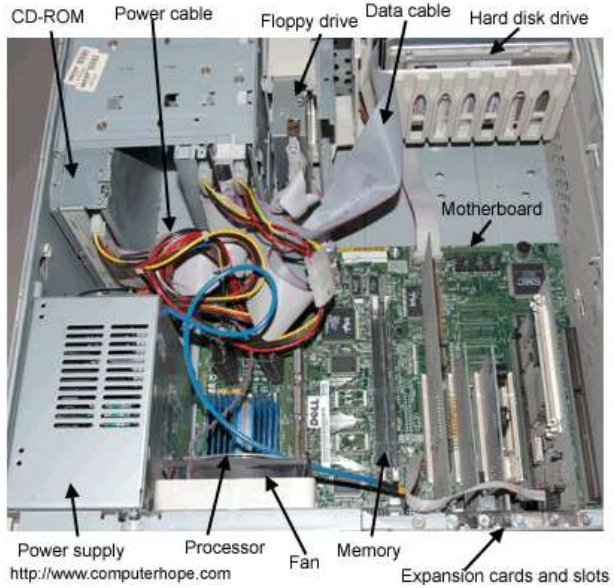
- Abstraction is the process of taking only a set of essential characteristics from something.
- Example
  - For a Doctor→ you are a Patient
    - Name, Age, Old medical records
  - For a Teacher→ you are a Student
    - Name, Roll Number/RegNo, Education background
  - For HR Staff→ you are _____
    - _____,_____,_____

You

Encapsulation

Would you like it if your CPU is given to you like this?

What are the problems if it were given to you like this?



Encapsulation is the process of compartmentalizing the elements of abstraction that constitute its structure and behavior; encapsulation serves to separate the contractual interface of an abstraction and its implementation.
- Grady Booch

- Encapsulation is binding data and operations that work on data together in a construct.
- Encapsulation involves Data and Implementation Hiding.

Class and Encapsulation

Student
• -roll: long • -name: String
• + display(): void • + read(): boolean

Rectangle
• -length: int • -width:int
• +area(): int

LinkedList
• -Node
• + addFirst(Node n) • + remove(Node n) • + add(Node n, int pos)

- : private
+ : public

Reuse in Object Oriented Language

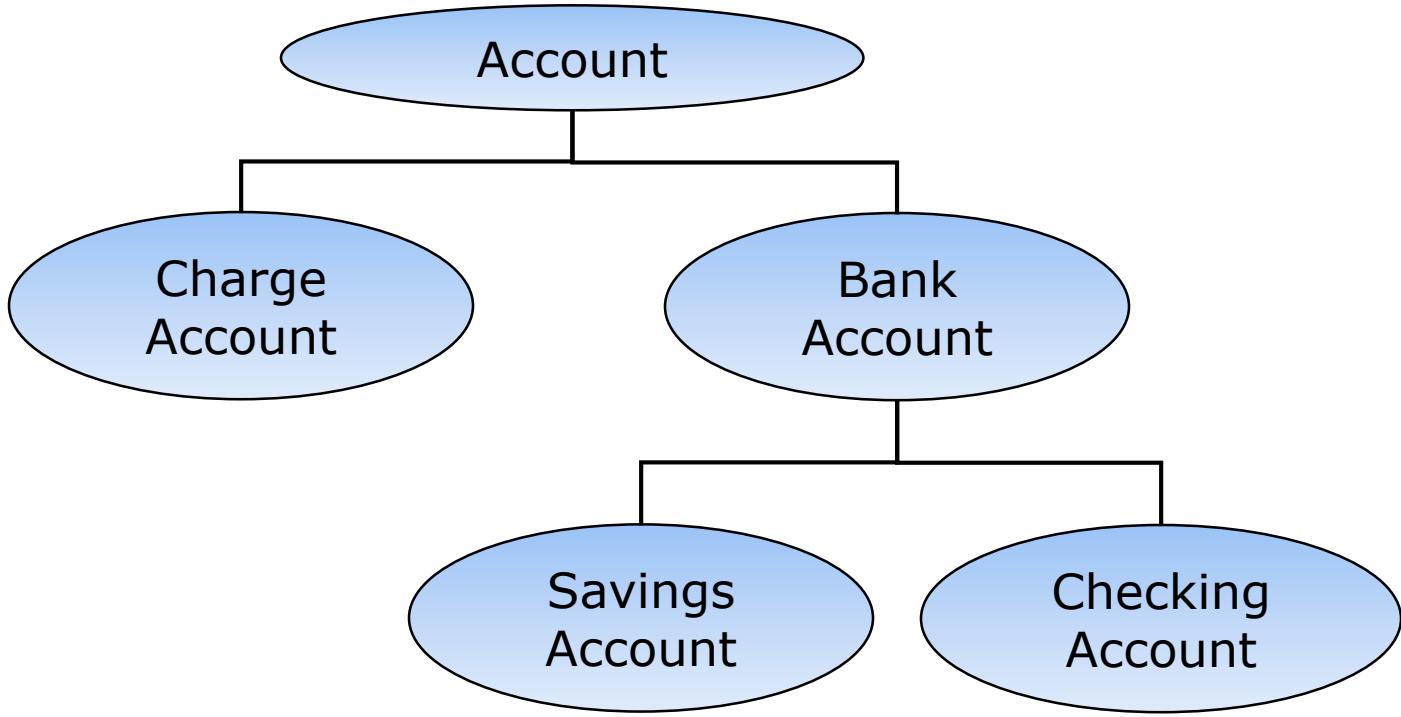
- Object Oriented Languages also implements reuse in the same way that we do in real life.
- Using
  - has-a
  - is-a
- Has-a or composition relationship is implemented by having a class having another class as its member, or rather an object having another object as its member.
  - Car has a Stereo
  - College has Teachers and Students
- Is-a is implemented through what we call ***inheritance*** relationship

Inheritance

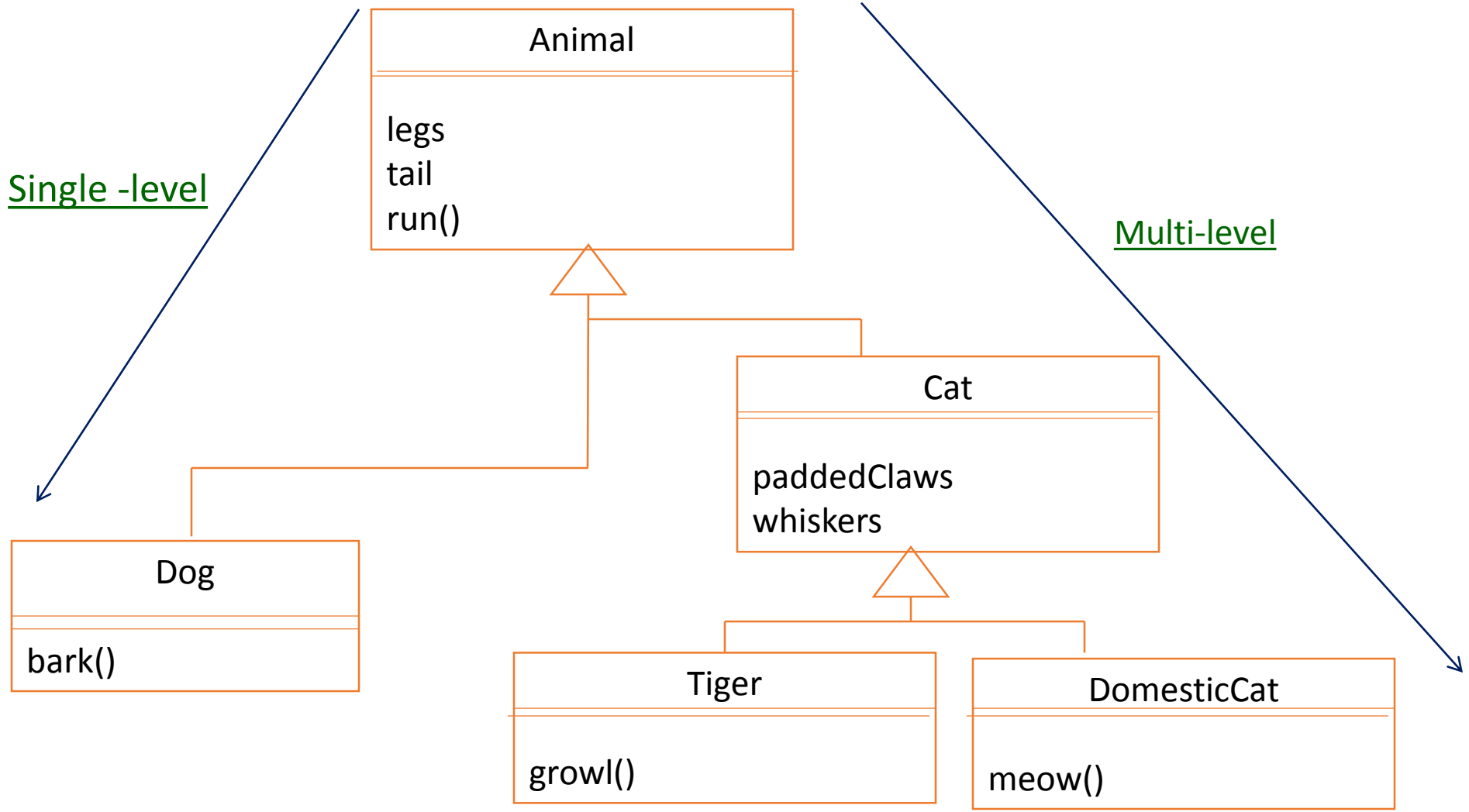
Inheritance defines relationship among classes, wherein one class share structure or behavior defined in one or more classes.

- Grady Booch

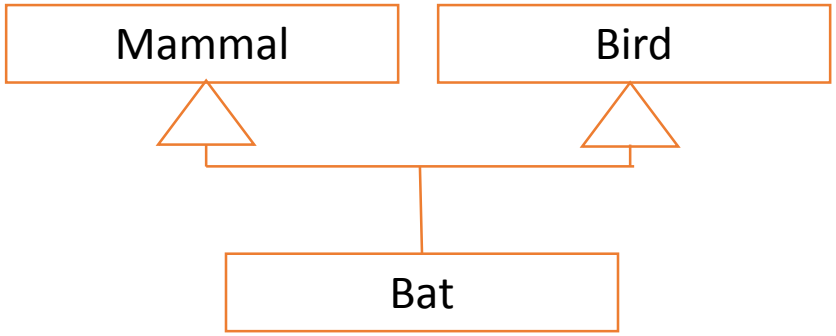
- Defines IS-A relationship between classes
- Cat IS-A Animal
- Car IS-A Vehicle
- Rose IS-A Flower



Inheritance hierarchy

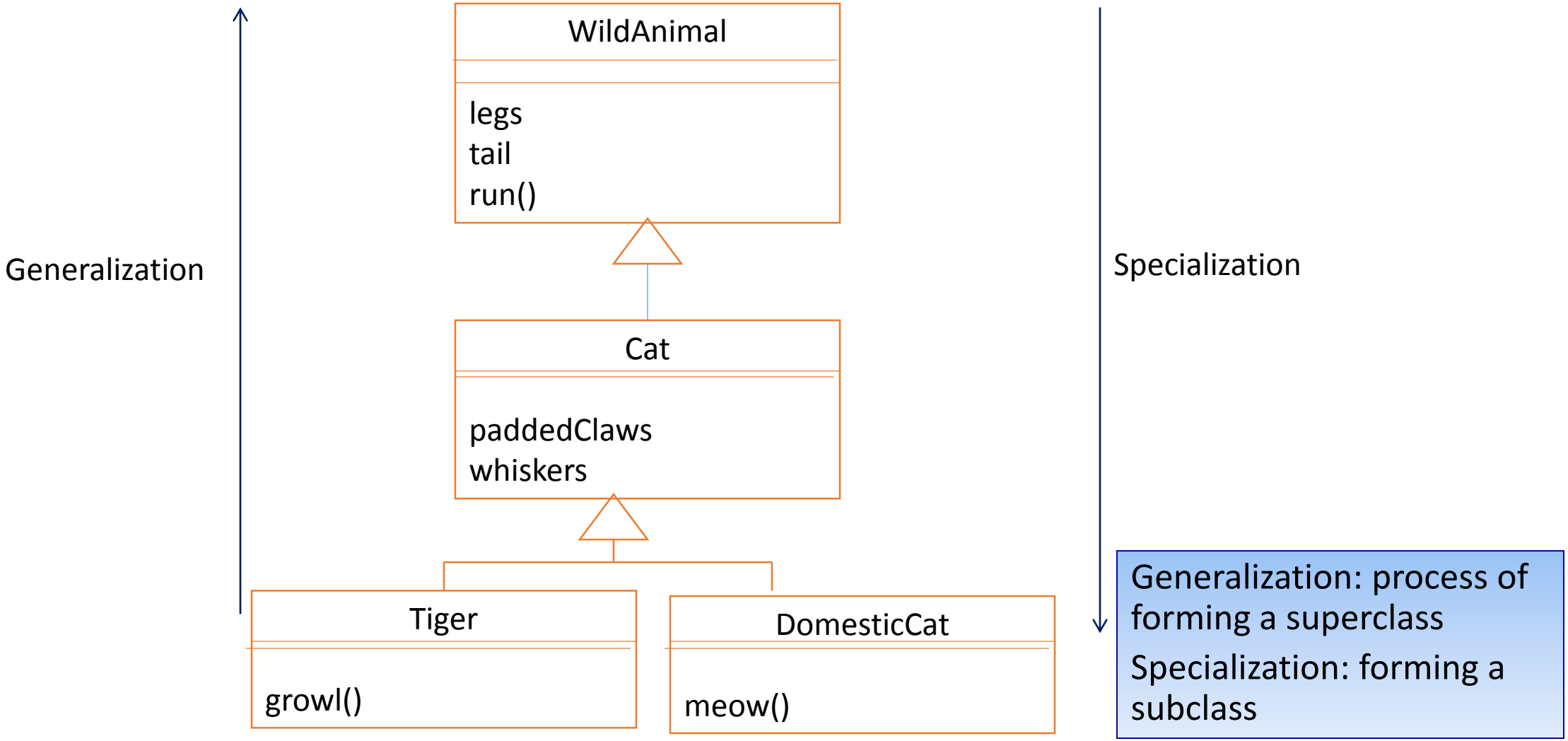


Multiple inheritance



- Many Object-orientated language does not support this type of inheritance.
- Java, C# are the examples of object-oriented language that does not support multiple inheritance through classes.

Generalization and Specialization



Generalization: process of forming a superclass

Specialization: forming a subclass

Activity: match the following

Super class

- Fruit
- Library
- Cat
- Bird

Subclass

- Parrot
- Music
- Tiger
- Books
- Apple
- Mango

Activity

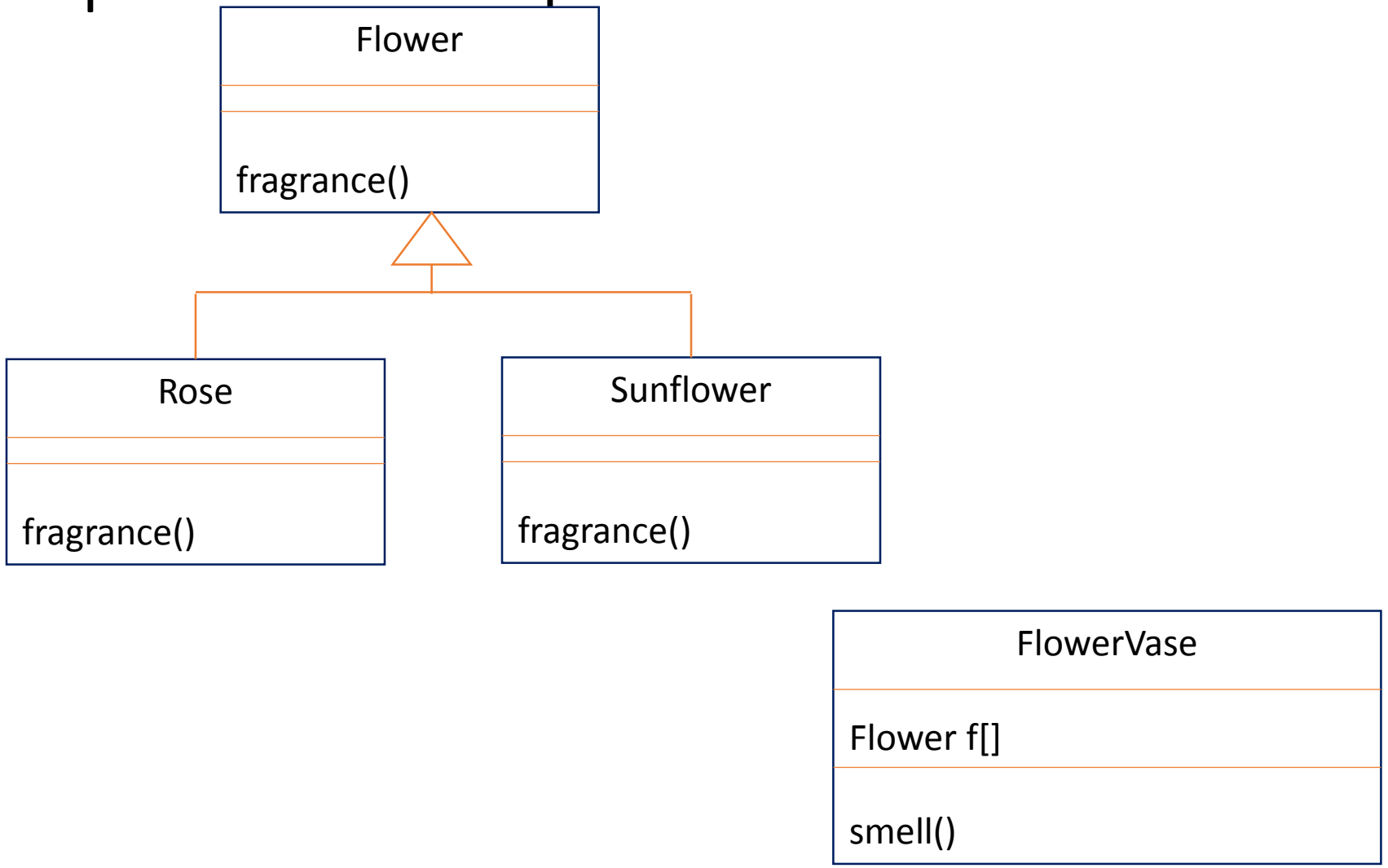
- Given the following classes. Can you form inheritance hierarchy?
  - Person
  - Teacher
  - Student
  - HOD
  - Typist
  - Clerk
- What are the common members shared ?

Polymorphism

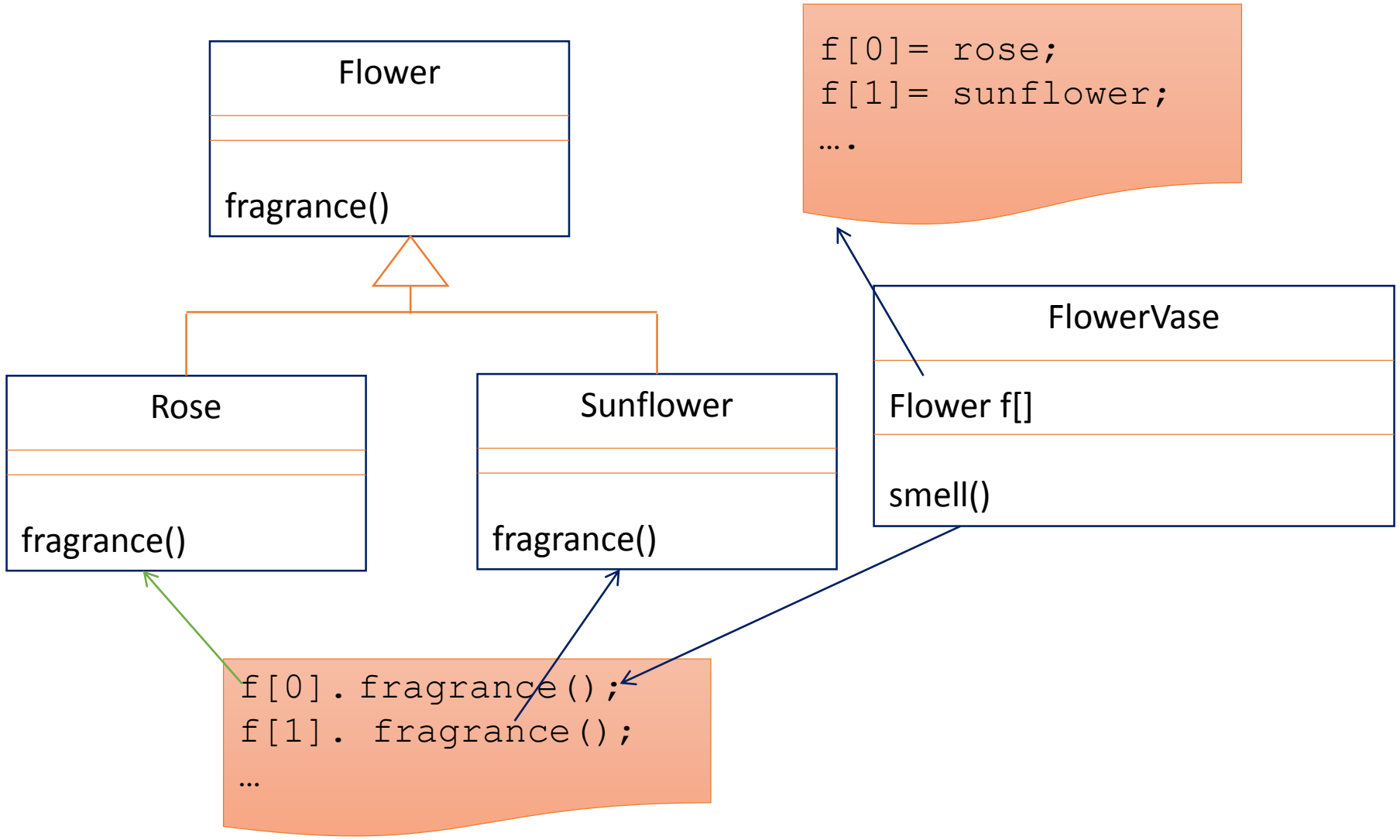
A concept in type theory, according to which a name (such as a variable declaration) may denote objects of many different classes that are related by some common superclass; thus, any object denoted by this name is able to respond to some common set of operations in different ways.

- Grady Booch

Polymorphism example



- You pick a rose from the vase and smell it?
- What fragrance do you expect?



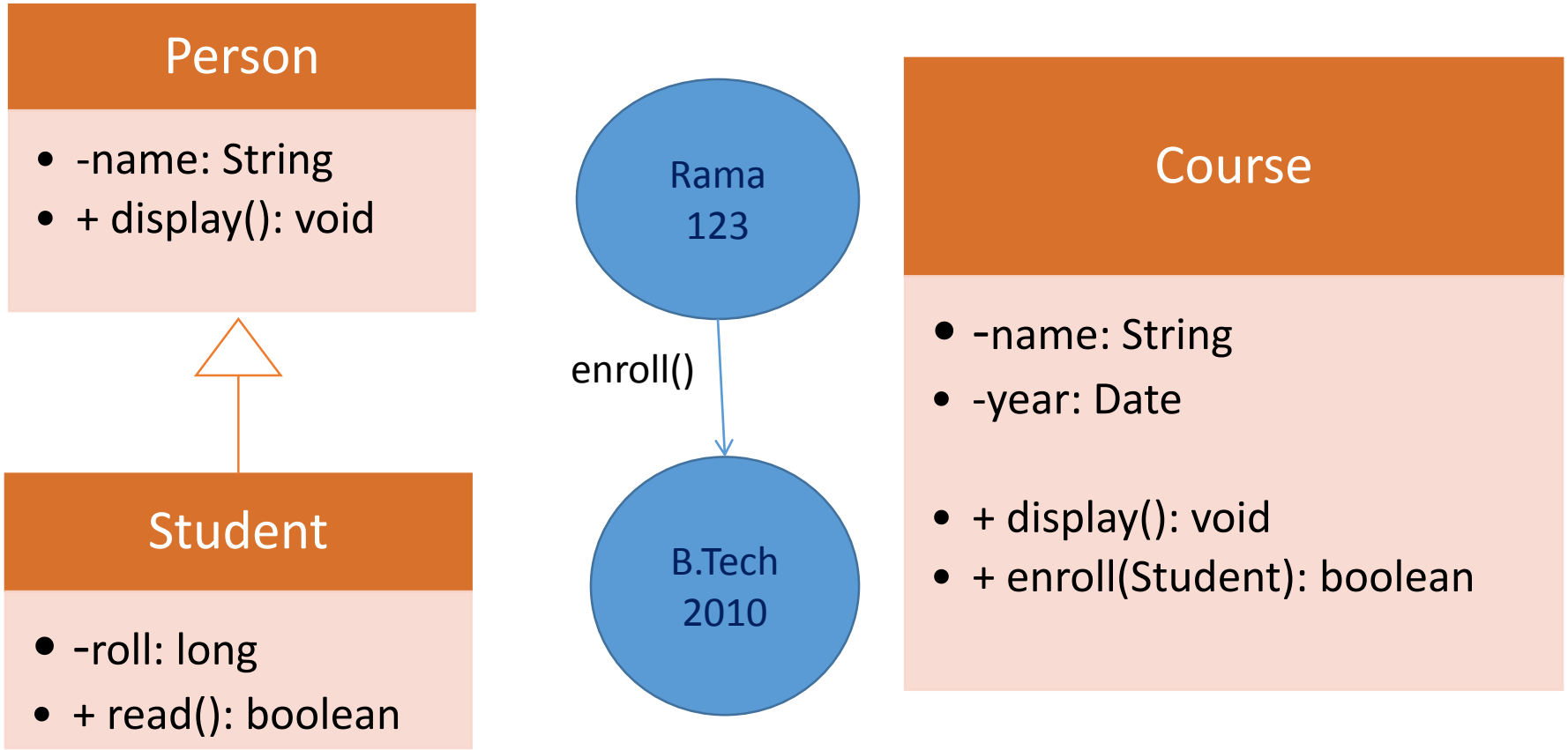
Exercise: Inheritance

- Among the classes that you identified, are there any classes that you can relate via inheritance.

(Hint: are there any classes which have common attributes/methods that can be moved up the hierarchy)

Revisiting definition

Object-oriented programming is a method of implementation in which programs are organized as cooperative collections of objects, each of which represents an instance of some class, and whose classes are all members of a hierarchy of classes united via inheritance relationships.



Summary

- Object-oriented programming is a method of implementation in which programs are organized as cooperative collections of objects.
- The object's state is determined by the value of its properties and its behavior is determined by the operations that it provides.
- Abstraction is the process of taking only a set of essential characteristics from something.
- Encapsulation is binding data and operations that work on data together in a construct.
- Inheritance defines relationship among classes, wherein one class share structure or behavior defined in one or more classes.
- Polymorphism is using a function in many forms. Poly means 'many', Morphism means 'forms'.