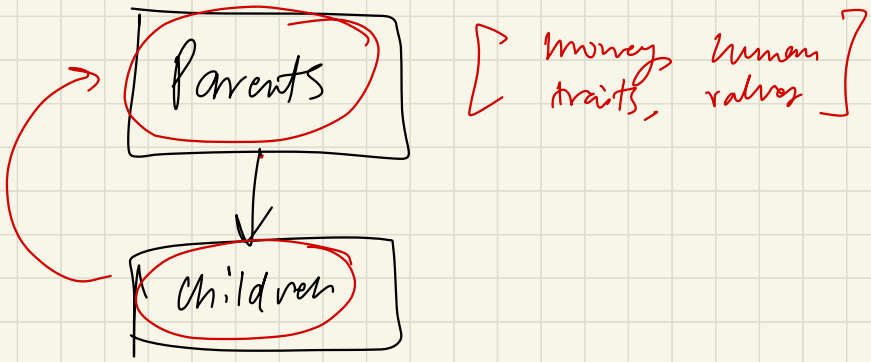
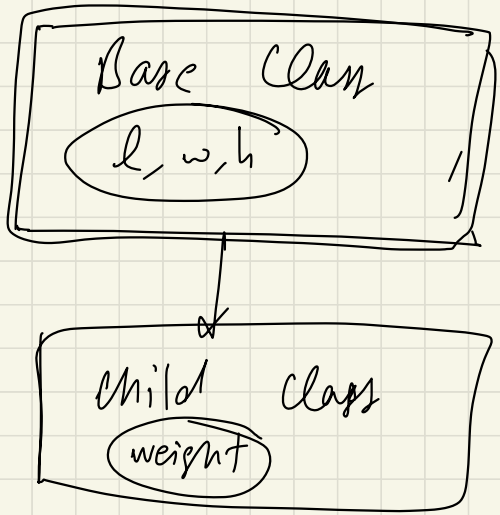
① Inheritance



In OOP

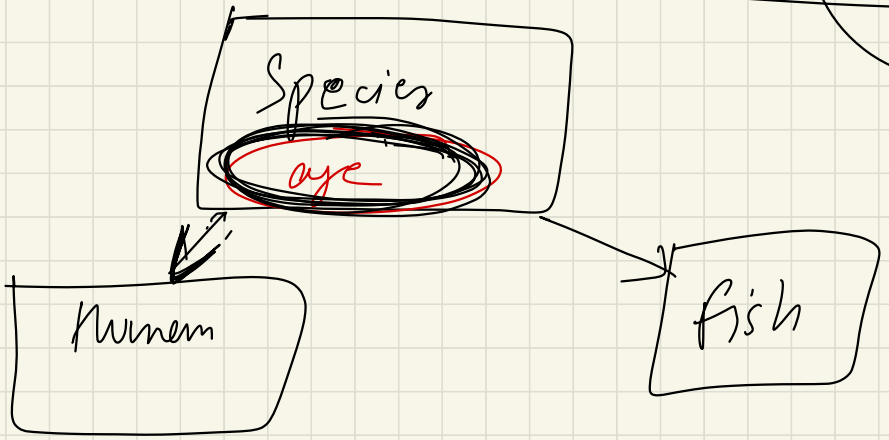


Child class is inheriting properties from base class.

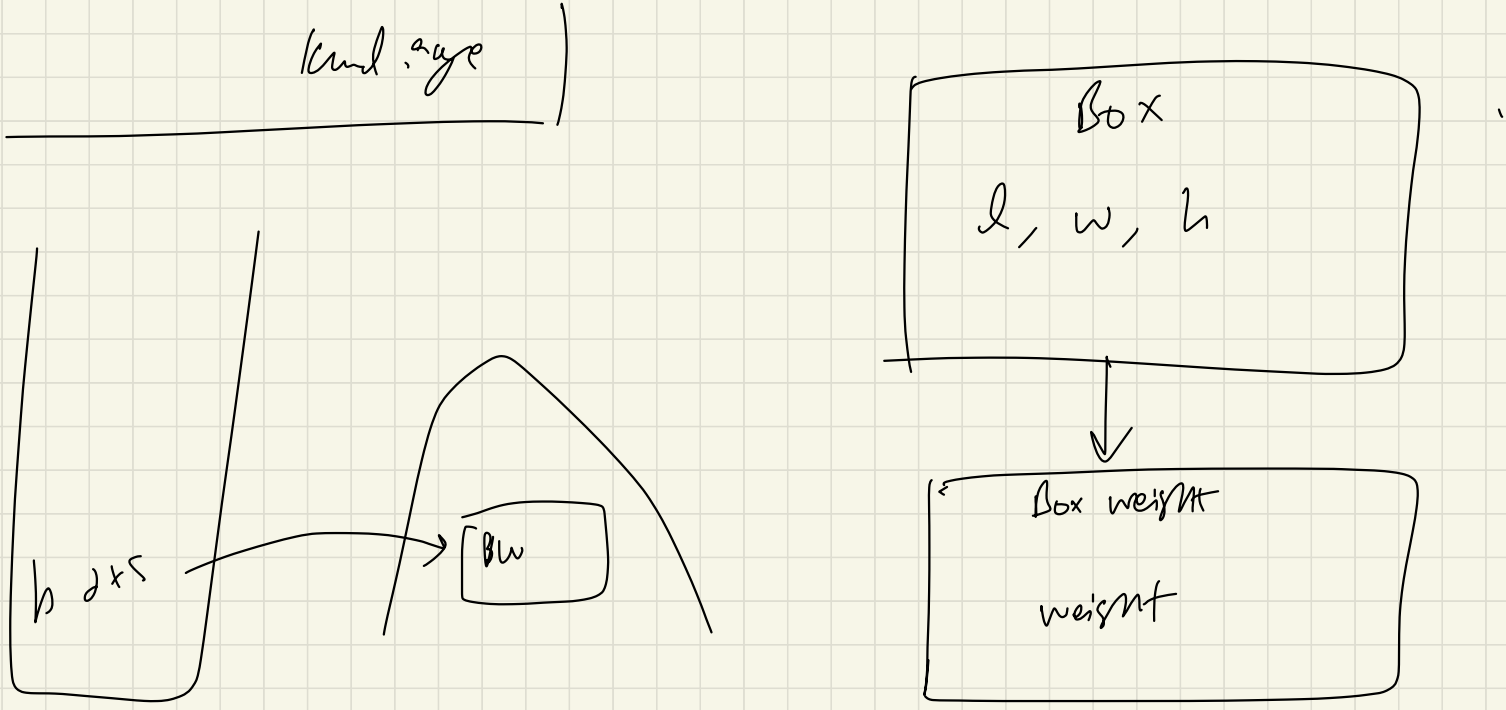
```
class Child extends Base {
    int weight;
}
```

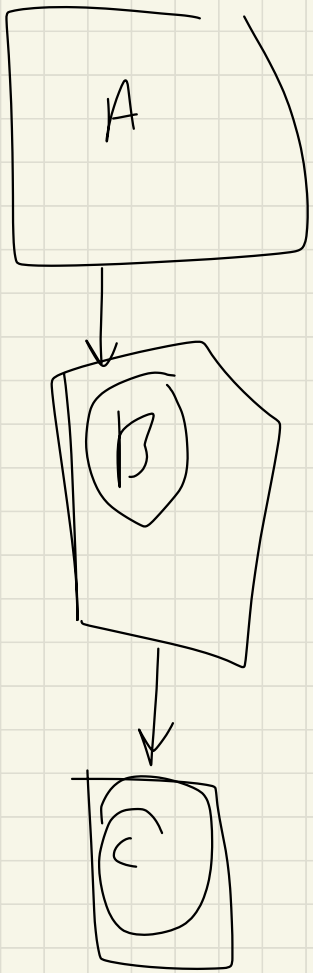
```
Child child = new Child();
child.l,
child.w,
child.h
```

initialize parent class variables also.



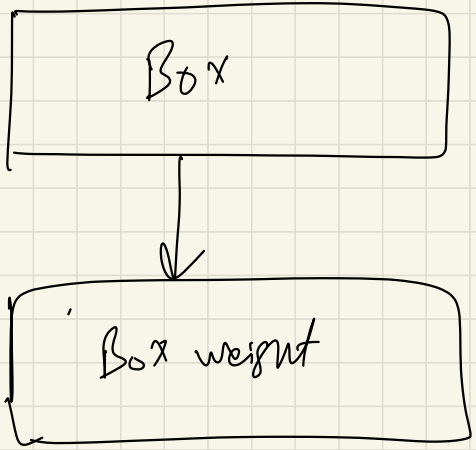
Human kind = new Human();


kind: age
Box
l, w, h
Box weight
weight
h dxs
Bw

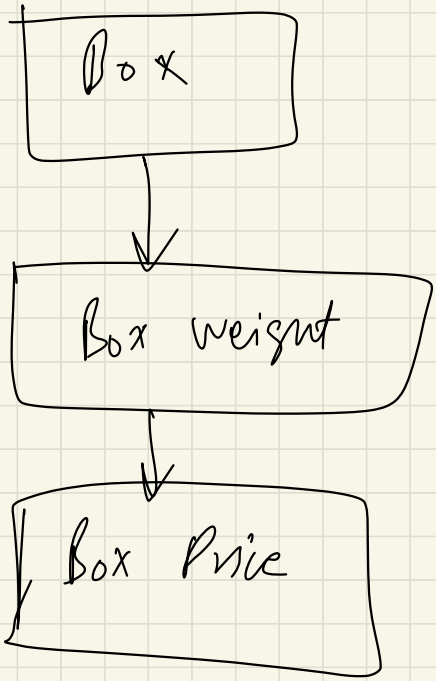

A
B
C

Types of inheritance :

(1) Single Inheritance : One class extends another class

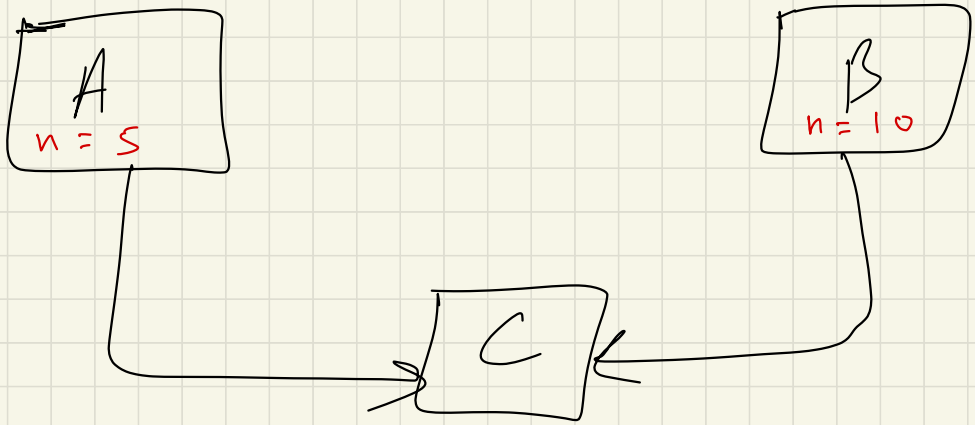


(2) multilevel Inheritance :



(3) Multiple Inheritance:

One class extends more than 1 class. Not allowed in Java. (we will do this in interfaces)

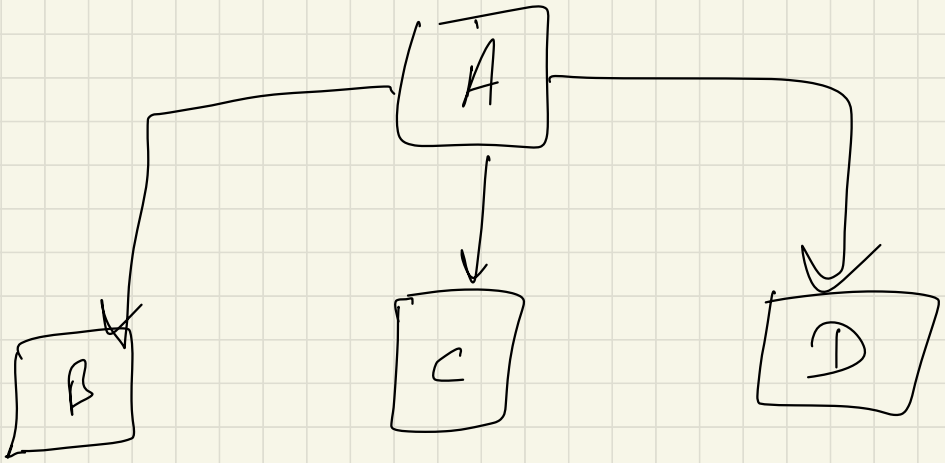


C obj = new C();

c.n // ? i.e. not in java.

(4) Hierarchial Inheritance :

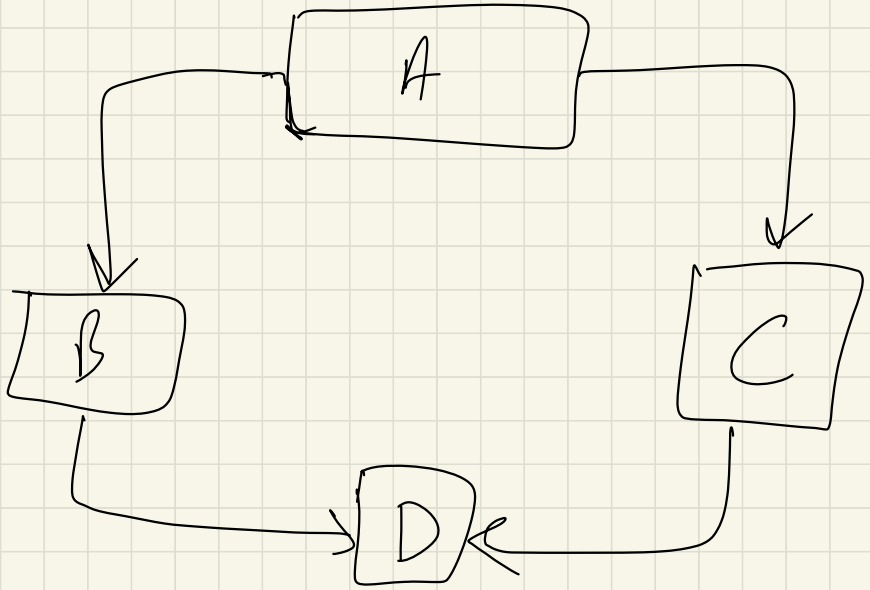
One class is inherited by many classes.



(5) Hybrid Inheritance:

Combination of single and multiple inheritance.

Not in java, (Achieve inter java feature).

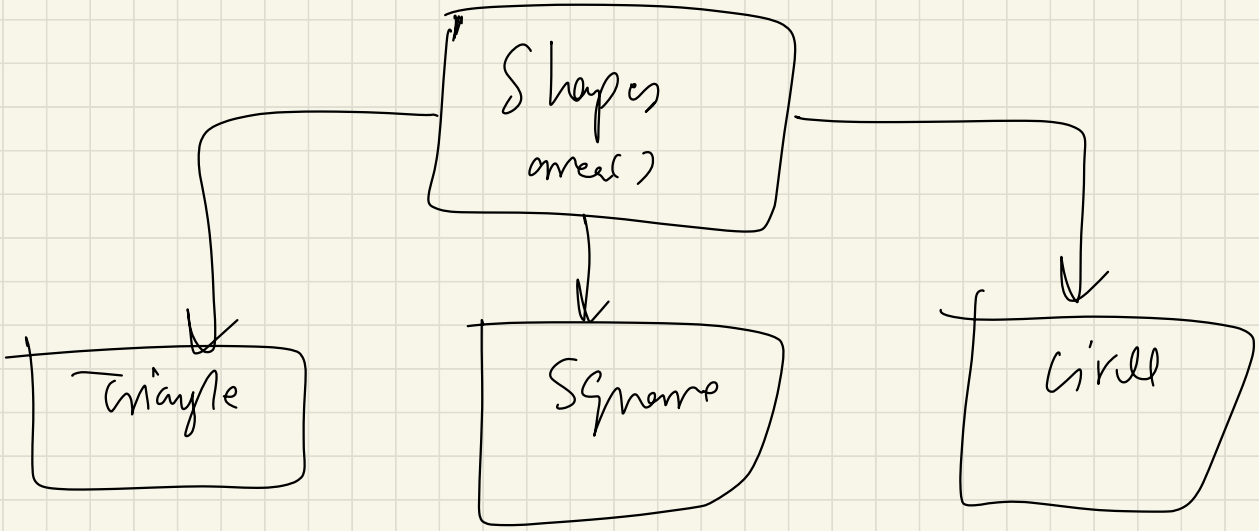


Poly morphism :

Poly Morphism

Many

ways to represent



Types of Poly morphism:

(1) Compile Time / Static Poly morphism.

Achieved via method Overloading.

Same name but types, arguments, return types, ordering can be different.

Ex: multiple constructors.

```
A a = new A();
A a2 = new A(3, 4);
```

(2) Runtime / Dynamic Polymorphism:

Achieved by method overriding.

✓ Parent obj = new Child();

Here, which method will be called depends on

This is known as Upcasting.

How Overriding works.

How Java determines this?

Dynamic method dispatch

Encapsulation:

Wrapping up the implementation of the data members & methods in a class.

Abstraction : Hiding unnecessary details & showing valuable information.

for in depth → check video & notes.