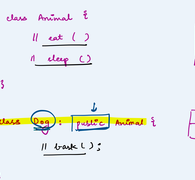
Inheritance

→ allows us to create a new class from an existing class.

→ The derived class inherits the features from the base class.



class Dog extends Animal {

}

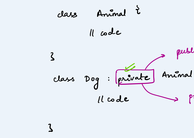
Inheritance

is - a relationship

→ A car is a vehicle

→ Orange is a fruit

→ A dog is an animal

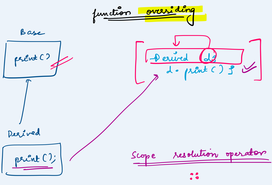


① public : just the way they are

② private : all members of base class becomes private in derived class

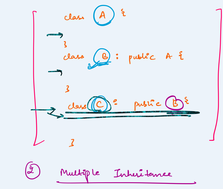
③ protected : public → protected
private → private

function overriding

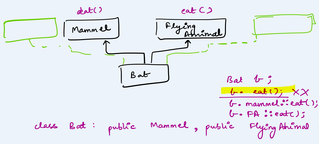


Types of Inheritance

→ ① Multi level



② Multiple Inheritance

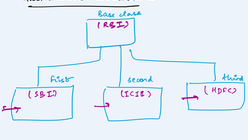


Bat b;
b.eat(); ✗✗
b.Mammal::eat();
b.FA::eat();

class Bat : public Mammal, public FlyingAnimal {

}

Hierarchical Inheritance



③ Hybrid

Friend function

→ allows to access member functions from outside the class.

add (classA obj A, class B obj B) {

return objA.valA + objB.valB;

}

Friend class

class friend

all the member functions of the friend class become friend functions

class classA {

friend class classB;

};

class classB {

[ ]

}

Virtual Functions

→ A virtual function is a member function in the base class that we except to redefine in derived class.