Hirerachy
Association
- Dependent Object
                    Human has-a Heart
- Dependency Object

```
class Date{          class Employee{// Dependent
day                  id
month                string name
year                 salary
}                    Date doj // Association Dependency class

                     }
```

Composition

id	name	salary	doj			
				day	month	year
1	Anil	10000		1	1	2001
				int	int	int
int	string	double	Date			

Employee e;

Aggegration

cid	name	*dob
1	Anil	NULL
int	string	Date

Customer c1

cid	name	*dob
2	Mukesh	~~NULL~~ 0X300
int	string	Date

day
1
month
1
year
2001

0X300   new Date()

Customer c2

1,2,3-dob,4,5,6,7-dob,8,9-dob,10-dob
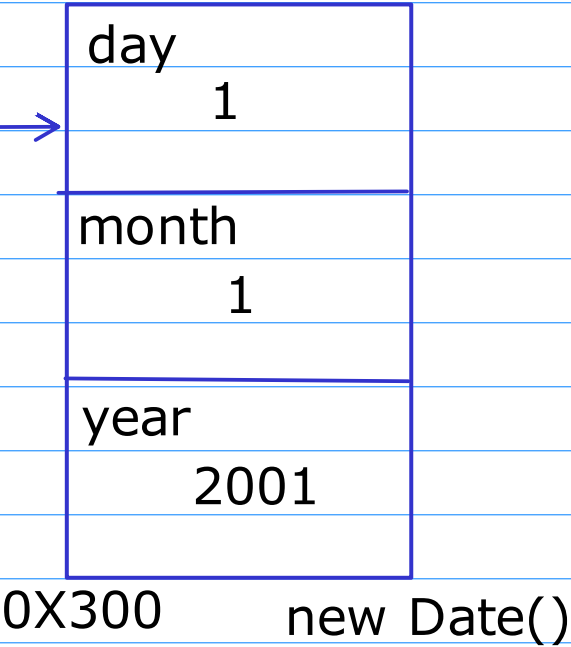
Inheritance

- is-a relationship

ElectornicDevice   Laptop is-a ElectronicDevice
Fruit   Mobile is-a ElectronicDevice
Shape
Parent -> Base   Apple is-a Fruit
     Orange is-a Fruit

Laptop,Mobile
Apple,Orange   Circle is-a Shape
Circle,Rectangle   Rectangle is-a Shape
Child -> Derived

Parent/Base

```
class Person{
string name;
string mobile;
string email;
}
```

child/Derived

```
class Employee : Person{
int id;
double salary;
}
```

child/Derived

```
class Student : Person{
int rollno;
double marks;
}
```
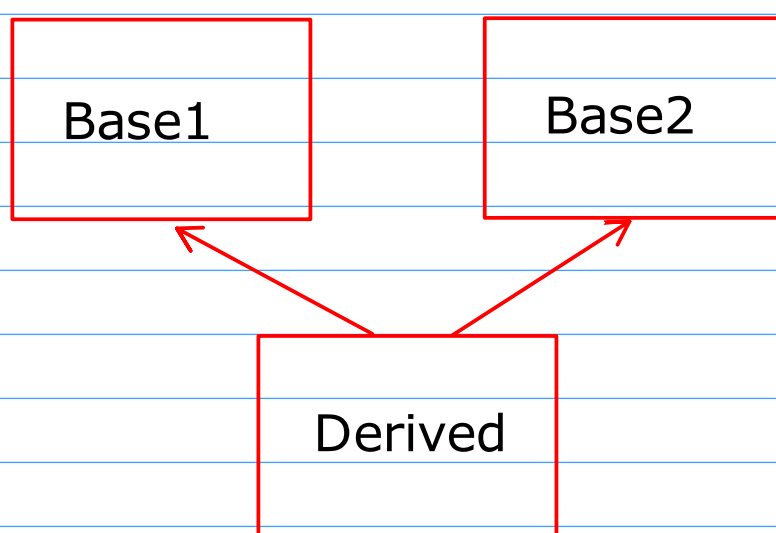
Employee is-a Person

- In inheritance the parent/Base class members are inherited inside the child/Dervied class
- Here the parent/Base class object is created within the Child/Derived class object
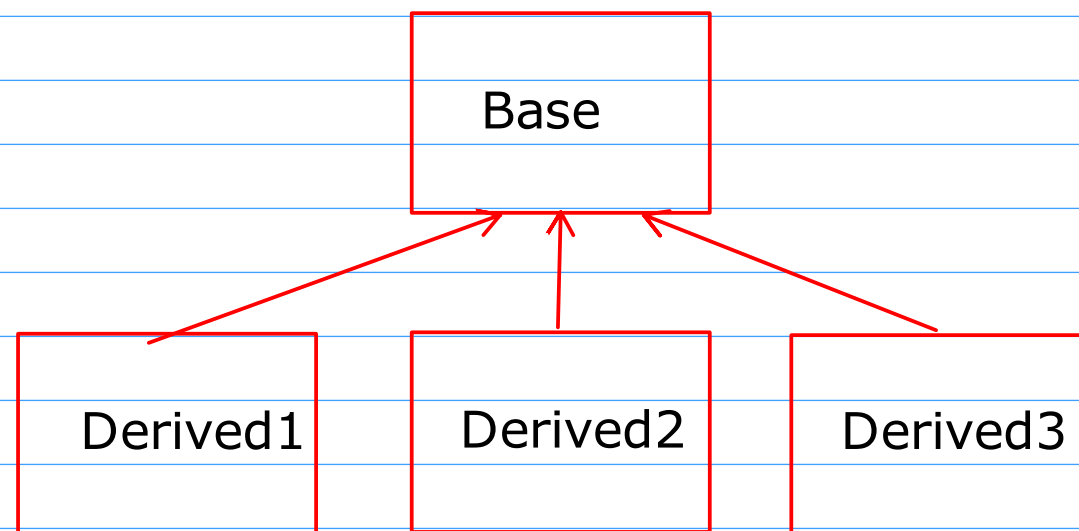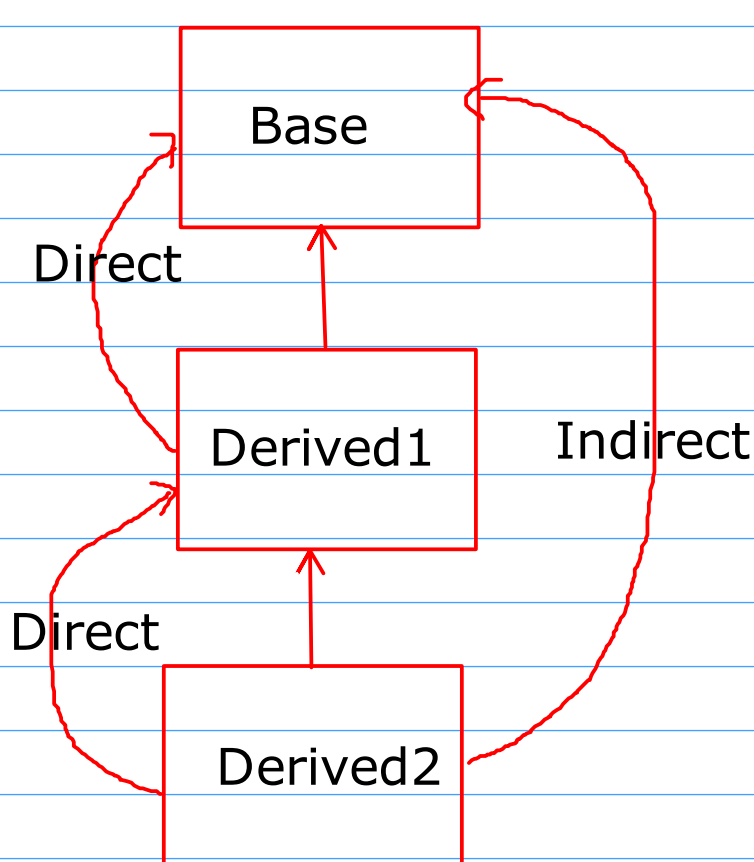- The object of parent/base class is created only once inside the Child/Derived class Object
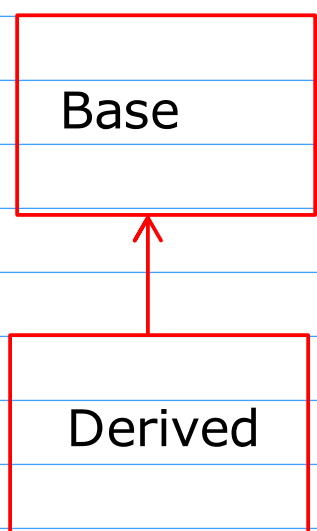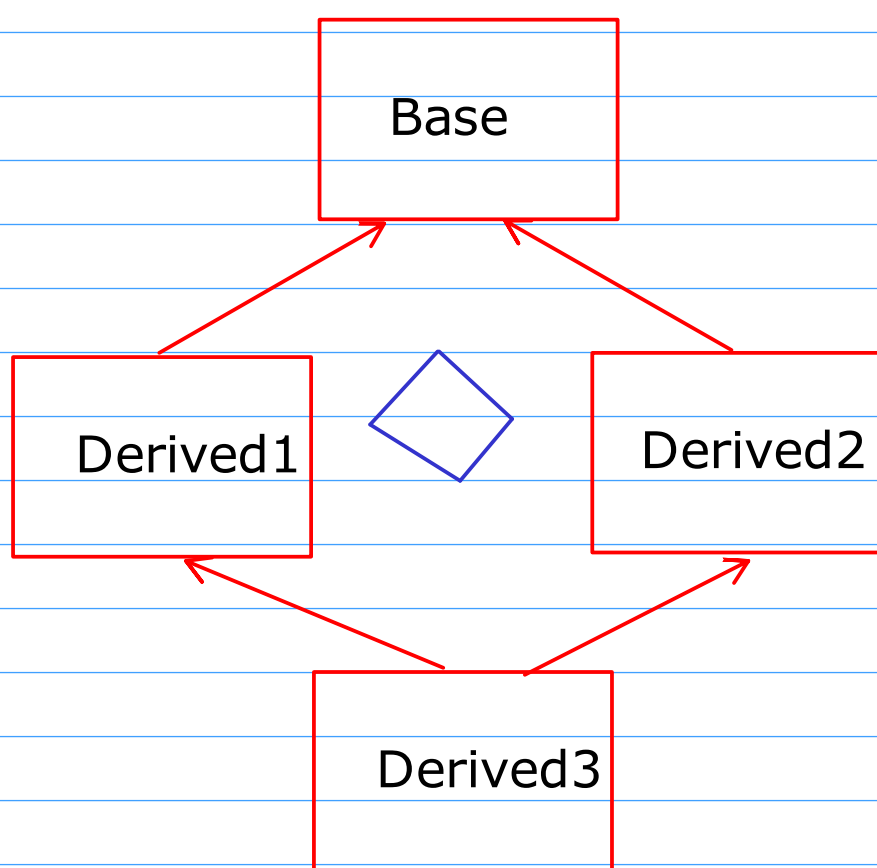
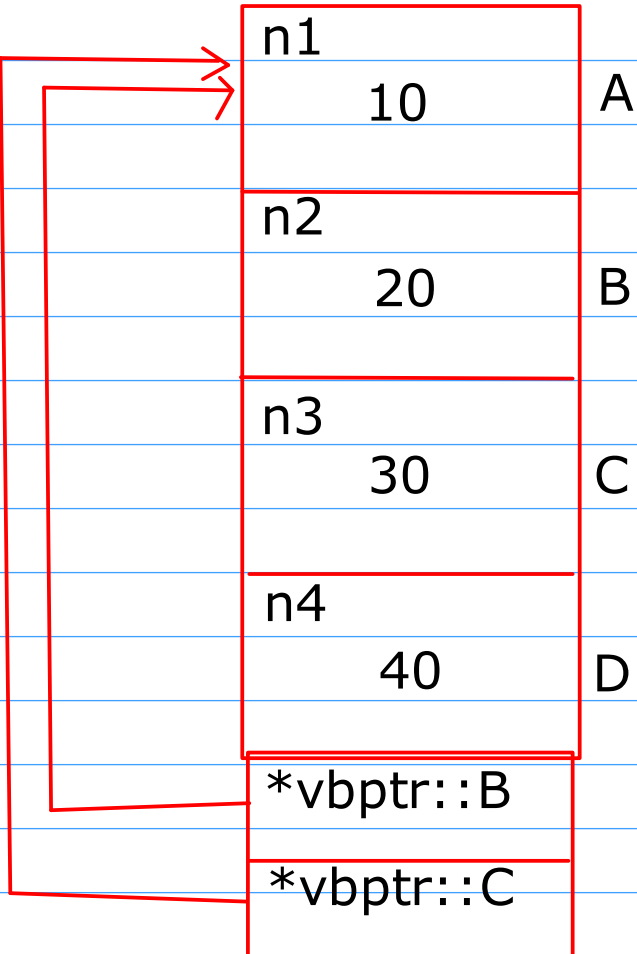Single Inheritance   Multiple Inheritance   Hirerachical Inheritance

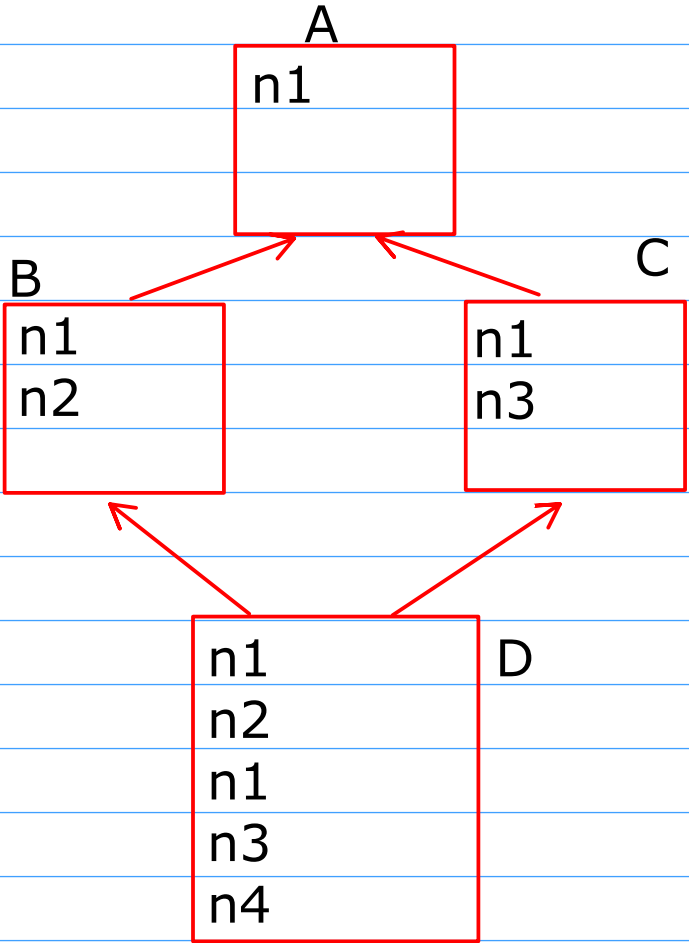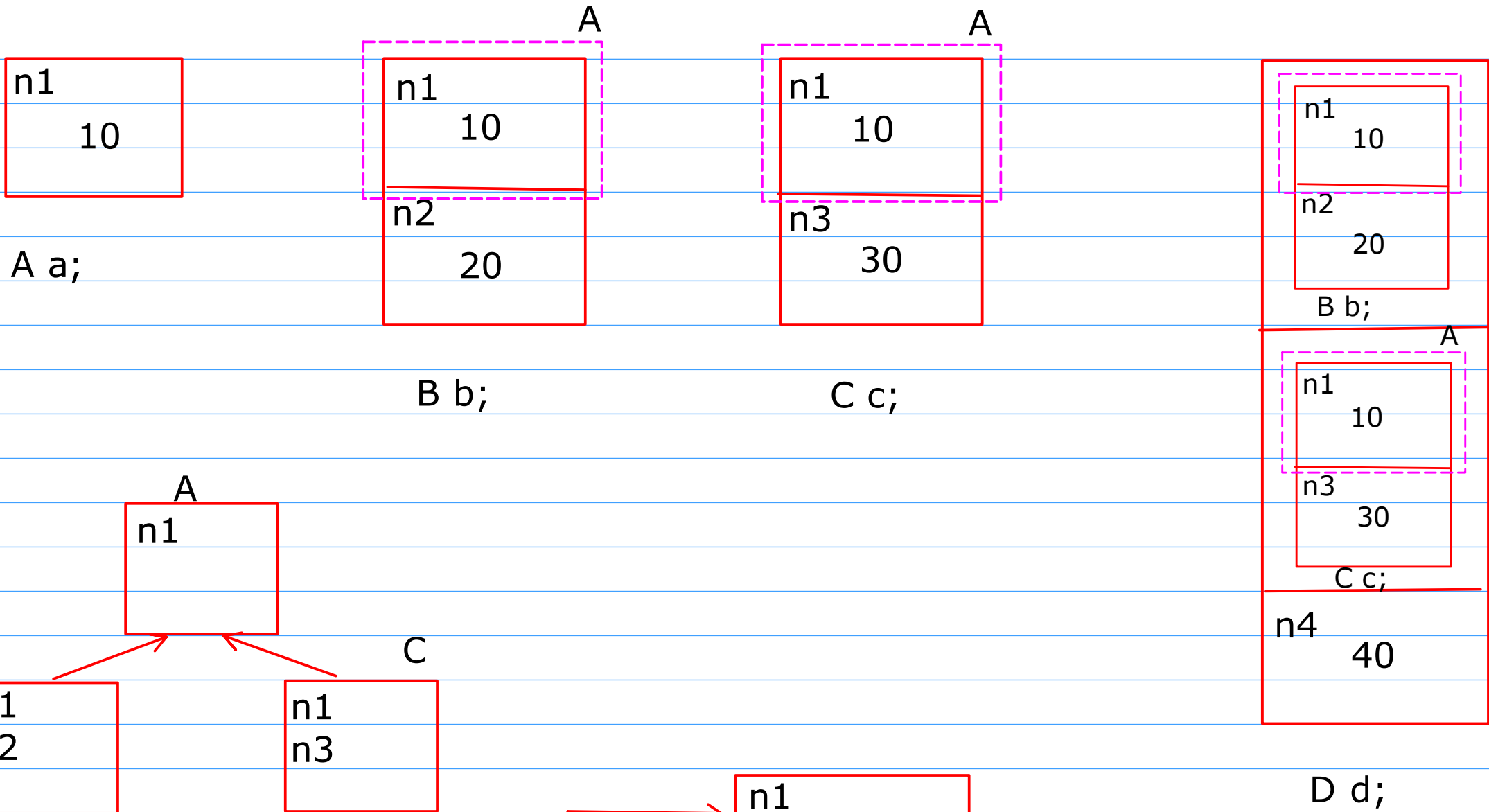
Multilevel Inheritance   Hybrid Inheritance
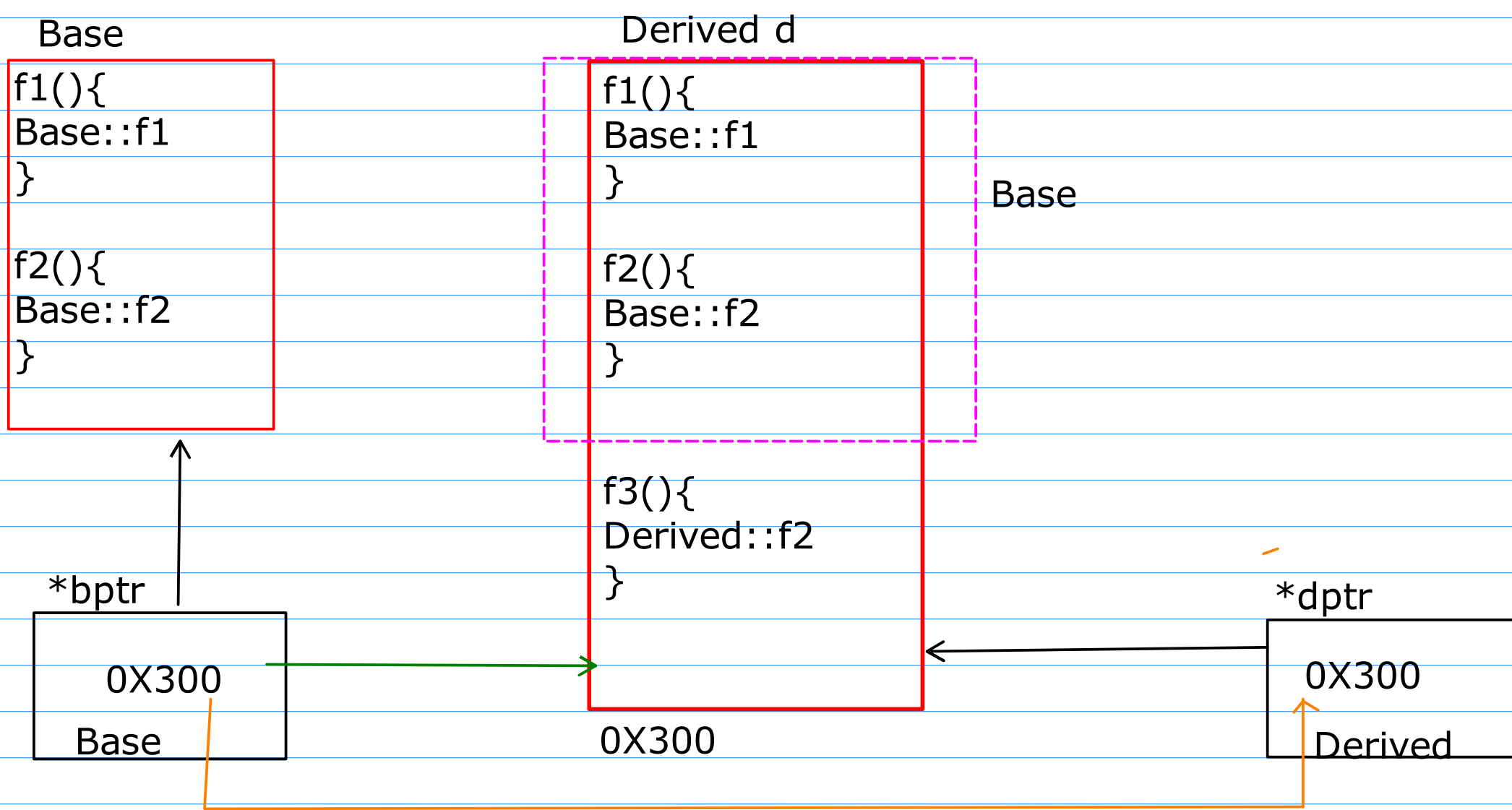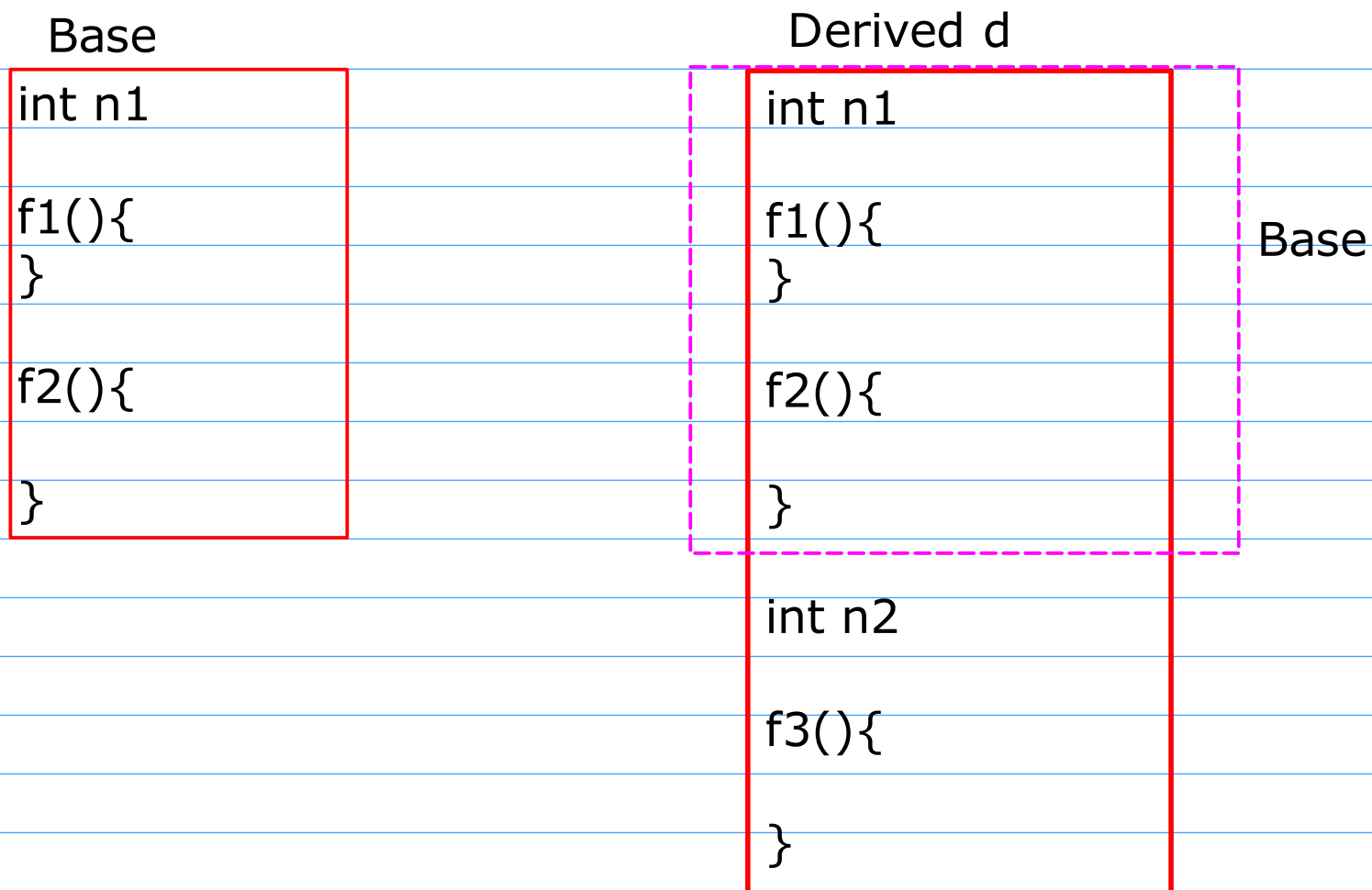
#Mode of Inheritance
- If private,public and protected keywords are used at the time of inheritance then it is called as mode of inheritance
- By default the mode of inheritance is private
- In private mode of inheritance the protected and public members of the base class are made as private inside the derived class
- In protected mode of inheritance the public members of the base class are made as protected inside the derived class
- In public mode of inheritance no any changes is done for the members of the base class inherited inside the derived class

A a;

B b;

C c;

B b;

C c;

D d;

A
n1

B
n1
n2

C
n1
n3

D
n1
n2
n1
n3
n4

n1
10   A
n2
20   B
n3
30   C
n4
40   D
*vbptr::B
*vbptr::C

0   n2
4   *vbptr   B
        16
12 n3
16 *vbptr   C
        4
24 n4   D
28 n1
32   A

Base

int n1

f1(){
}

f2(){

}

Derived d

int n1

f1(){
}

f2(){

}

Base

int n2

f3(){

}

Base

f1(){
Base::f1
}

f2(){
Base::f2
}

Derived d

f1(){
Base::f1
}

f2(){
Base::f2
}

Base

f3(){
Derived::f2
}

*bptr

0X300

Base

0X300

*dptr

0X300
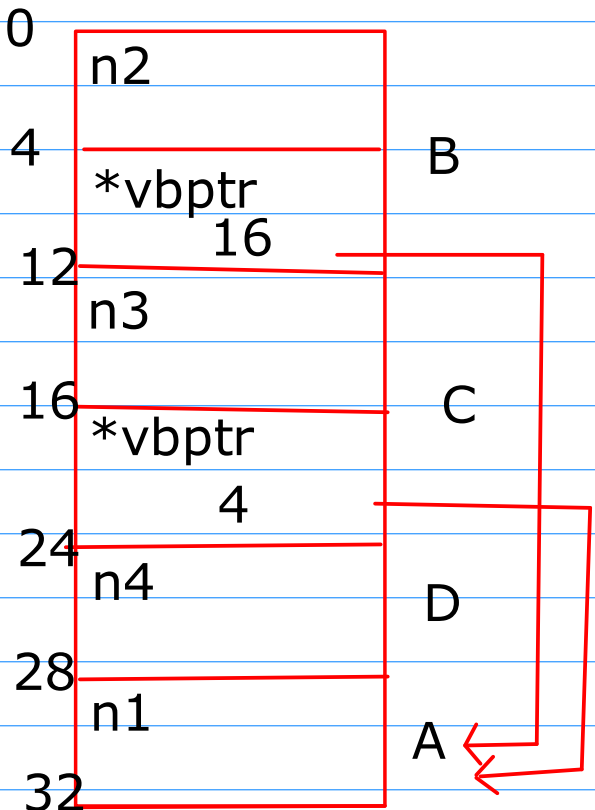
Derived

Upcasting -
Storing the object of derived class into base class pointer or reference is called as upcasting