③ - ENCAPSULATION

Internal based

↓

wrapping

④ - ABSTRACTION

↓

External based.

Hide unnecessary & reveal necessary

Exp. No. : .....

Date : .....

cannot overide a final method;

Over riding depends on ... objects
Statict doesnot depend on object
so static does not overriden.

It is all done by

DYNAMIC METHOD DISPATCH

Check during runtime for a overriden method.

Lets say

```
A a = new A();
B b = new B();
a.info()
b.info()
```

O/P : From A

O/P : From B

Now, both have same methods....

```
A aa = new b()   // compile time Success
aa.info();       // compile time verifies LHS variables has such method or not.
```

Runtime it checks RHS, (object) has same method it overrides and prints.

O/P : From B

Case-1

① if nothing in B, It prints From A

Case-2

② if nothing in A, it will be compile time error

Exp. No. : .....

Date : .....

② POLYMORPHISM

Poly - many

Morphism - ways to represent

[:: Determined by parameter - representation]

Types

Method Overloading [Compile time / static]

↳ Parameters will change, like types, values, variables

Method Overriding [runtime / Dynamic]

```
Parent JK = new extendable();

JK.info()  // compile time this checks the parent class if method present or not & throw error
```

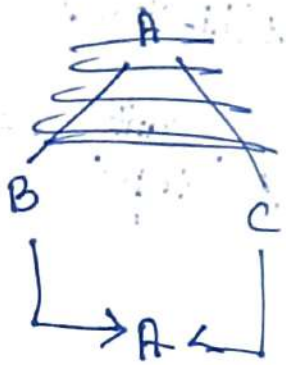
but, run time the real object's method is overiden.

TYPES

Single level

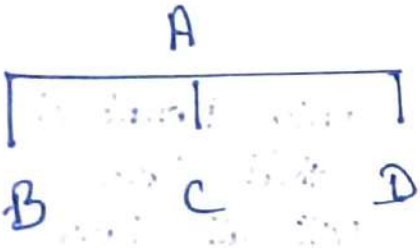
Multi level

Multiple

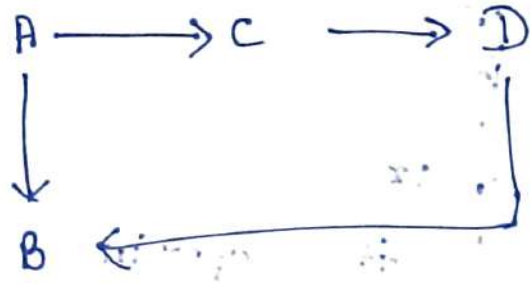


Not allowed in Java

Hierarchy



Hybrid



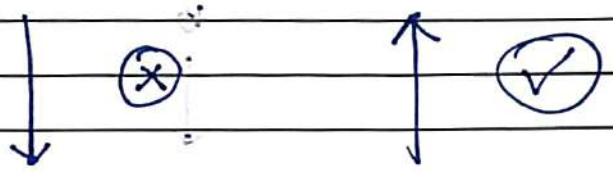
Not allocoed in Java

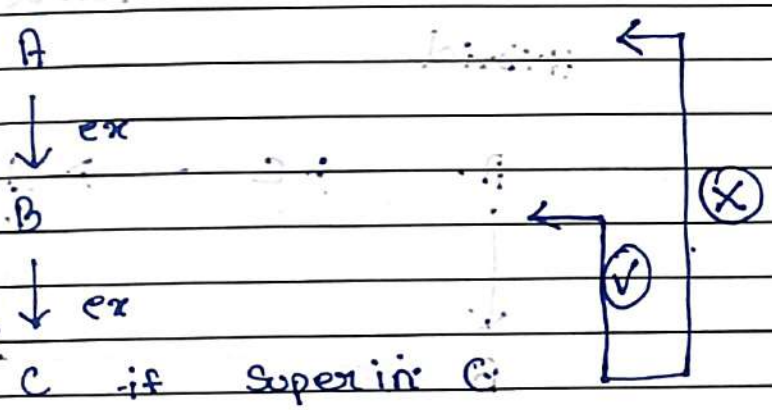
Exp. No. : .....

Date : .....

sout(hi.sec) // error ② -phase code

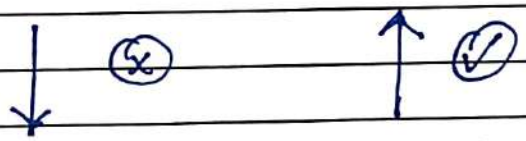
can't access child at 'bottom' by parent at 'top'





only Parent of that class not all class Parent (ie root).

Super also covers same pattern.



Only access above not bellow (in other word)
Only parent not child.

Page No.:

Why ?

class A

class B ex class A

here, the top is parent class,

In OOPS only the top way can be accessed not the bottom

In other words

See how the extend works, when creating object for 'child' which extends 'parent' then it can access parent

But while the parent object only access 'parent' itself it cant access child.

— x —

Not only here,

lets see in object way

A hi = new B (12);

here, I can initialize Parent via child

But access belongs to the parent and parent cannot access childs variable

[∵ It can access by only over riding]

Polymorphism

Exp. No. : .....

Date : .....

② Phasie

Using Super, You can modify the Parent class.

```
Class B extends A {
    int Sec = 21;
    Public B (int JK) {
        Super (JK);
    }
}
```

Super(JK) will find a constructor which can hold this value and insert it there, like modifies ↓ calls parent class constructor.

In main.

```
B h = new B (11);
Sout (h.a);
h.Info();
```

O/P

11

..... a = 11

[Note : You can access parent variable using child.

You cannot access child variable via parent.

] why?

↳ next page

Page No.:

in main with ① phase

```
B b = new B();
Sout(b.a);
b.Info();
```

OP:

-1

..... a = -1

Passing parameter - Only possible via Parent class:

[To pass by child need super] → ② in 2nd phase

```
A a = new A(10);
Sout(a.a);
a.Info();
```

OP: 10

..... a = 10

Pass by object 'a'

```
A x = new A(a);
Sout(x.a);
x.Info()
```

OP: 10

..... a = 10

Exp. No. : .....

Date : .....

One class inherits the property of another :-

```
Class A {

    int a

    A () {
        this.a = -1;  // default -1
        }
    A (int n) {
        this.a = n;  // change by parameter
        }
    A (A object) {
        this.a = object.a;  // change by object.
        }

    Void info () {
            sout (" Info from parent a = " + a );
        }
    }
```

(1) Phase

```
Class B extends A {

        }
```

OOPS-3

PRINCIPLES :- API-E

Inheritance
Polymorphism
Encapsulation
Abstraction.

(1) INHERITANCE :-

↳ Inheriting the property of a class.

class Parent — # 7 methods brought 7 apples

class extends [Inherits] Parent
child. use or eat those apple
uses those method. of parent

Ex:

```
class parent {

    void sum() {
        sout("Hi"); } }
class child extends parent {
        sum();
    }
```