

Cours 4

(14 octobre)

Héritage

Cha e III

H age

,

|

|

Chaque III: Hage

- A) E e g a
 - Affec a e a age
- B) M h de
 - S cha ge e g a e
- C) M h de (, e)
 - Red f e a d a e
- D) C e ce
 - Le a ab e
- E) D e
 - S e acc , f a
- F) C' c e e h age

A) E e : g a

□ P c e de a g a a b e :

□ be ge' a e a de ch e'

■ ad c e e ca ac , e de che

■ e a de e a e

■ che e - e a f e e - e

■ O e d d ;
 □ H a ch e de c a e (Ob ec a ac e)
 □ e Be e e e de A a b e de
 B'e b e de A a ec de
 e a e

E e : g a

Q a d B e e e e de a c a e A:

■ T ' b e de B a e ' e d b e
de A (+ da e).

■ D c b e B e e c d ' c e
b e A.

■ D c e a a b e d f e ' b e de A
da e). (Ma e e e e b e de B (+
e occultées)

■ Ide ' e h de : Le h de de A
e e ' B e b e B e d f de
e e h de .

■ Ma B e redéfinir de h de de A.

E e de c a e

S B e e e e de A

□ e a ab e :

' ■ B e a e de a ab e (e e e
de e ce a *occultera* a a ab e de e
da A)



Re a e :

□ e a abe

□ c'e e de a a abe e e c e(a e e).

□ da, c e ed, cha e de a abe ec e d
e e ed ca a.

□ ('a ca e e e de a a abe e ad ca a e fa e
ac a)

□ e h: de

■ c'e a | g a e(+ e de a | a e) e e e
c e: |

□ e' a | de h de de e e de g a | e .
d ff e | e (cha ge)

□ da c | e ed de h de e e g a | e
c' e | d e e ed' f

□ ('a ca e e e de a h de e ad ca a e fa e
ac a a a 'a 'ca e) e e de a | h de e a
d f | e a fa e 'e c)

POO-L3 H. Fa c e

E e (c e)

- S B e e e e de A
(class B extends A)
- 'Le a abe e 'h de de A de
h de de B(a e e e e a e
acce be : private)
- B e a e de ' e e a abe (e *nom*
e de e a cc a)
- B e a e de e e h de ' a
signature e d ff e e
- B ed f de h de de A a g à e e
de e

Re a e :

□ Ja a e a gage

■ e a c e cha e a ab e a e: ce de
a d c a a

■ a c ' à , a f c a d age e ' e
e f a e è d ' a e d c a a (c e
e c ' e)

■ e c ' a e d f e a g a ' de ' a ' e
de h de e de acc a a ab e :

□ a. f () è ga ' e ' e de a a ' ab e a e e
e h de f () e ' a e ' b e de ce

□ a. e ga ' e e de ' a a ab e a e e e
a ab e e ' a e b e de ce e

E c e ce:

□ U e a a b e d c a e c e a d e c a e
 A e f e c e b e d e c a e B
 g ' a e e b e d e c a e
 d e d e A:

■ e | b e c e ce fa e
 ' b e d e c a e A

□ Pa c | e e a a b e d c a e d e c a e B
 e | e f e c e b e d e c a e A:
 a e e e c h e!

Affectation / ca

```
class A{
    public int i;
    //...
}
class B extends A{
    public int j;
    //...
}
public class Affecter{
    static void essai(){
        A a = new A();
        B b = new B();
        //b=a; impossible que signifierait b.j??
        a=b; // a référence un objet B
        // b=a;
        b=(B)a; // comme a est un objet B ok!!
    }
}
```

U ca g

- S B e e e e de A, a b e de B
 e e c d c e b e de A:
 ■ A a= e B();
- O a a' c e:
 ■ A a=(A) e B();
- ' ca g e e de c d e b e d' e
 c a e d e c e b e d' e c a e d e b a e
- U ca ' g de c f e e c f e (e
 e ha da a h a c h e d e c a e)
- ' ca ' g e ' e c e (' e a ' ' e!)
- a e
 ■ e a g a e e è d' e c' e : b' e ' e
 a d f

D ca g

□ S B e e e e de A, e b e ' b e de A e
fa b e de B. Da ce ca e e c d e b e
de B

■ A a= e B();

■ B b=(B)a;

□ I fa da ce ca | ca (a age) *explicite* (a "c e e d"
| e a b e | b e f e c e e a e d'
| e d de B)

□ A e c f e a e e ca e b e e e ' b e
c d e ' b e d' e d de B

□ d ca g: aff e e ' b e c d e d' (e e ba da
c f e e e e c e da a d ca a (e e ba da
ah a ch e de ca e)

□ e d ca g e e a e c e ('e a
b e!)

□ a e e ag d e e e d' e c e : ' b e 'e a d f

Ca g

□ O e e e a c a e a a de fa e
d "d ca g":

```
Base sref;  
Derive dref;  
if(sref instanceof Derive)  
    dref=(Derive) sref
```

B) M h de : S cha ge

□ M h de e g a e:

■ S g a e: e e e a g e a e c e e (a a
e e d e a n e e e)

■ Sè e a g a e c e:

□ f()

□ cha f()

□ Le de h de a e g a e: c'e e d

■ S cha ge b e:

□ De g a e d f f e e e

f()

f(d b e f)

□ Le compilateur d e e a e e d e a g e e e
f c e e (e a e g e)

S charge

□ U e def c

e f c d g e

a e g a e

(Ja a, C++, Ada e e e a ' cha ge)

E C / e

cha g

3/2 d

e

e -> 1

3.0/2, d, ,

e e -> 1,5

S change

```
public int f(int i){
    return i;
}
// public double f(int i){
//     return Math.sqrt( i);
// }
public int f(double i){
    return (int) Math.sqrt( i);
}
public int f(char c){
    return c;
}
```

Re a e

□ La de a cha ge a e a

c a

□ La g a , e d e e e c e ' e

□ (e , e c , ca d fa d ,

a age:

■ E e e: chà e ' c e è

■ E e e: ca g , ,

)

C) M h de : Red f

- U c a e h e de h de de
c a e a c e
- E e e a e de e e h de
- E e e cha ge de h de
- E e e a ed f de h de
de a c e .



E e e

```
class Mere{  
    void f(int i){  
        System.out.println("f("+i+") de Mere");  
    }  
    void f(String st){  
        System.out.println("f("+st+") de Mere");  
    }  
}
```

E e e (e)

```
class Fille extends Mere{
    void f(){ //surcharge
        System.out.println("f() de Fille");
    }
    // char f(int i){
    // même signature mais type de retour différent
    // }
    void g(){ //nouvelle méthode
        System.out.println("g() de Fille");
        f();
        f(3);
        f("bonjour");
    }
    void f(int i){ // redéfinition
        System.out.println("f("+i+") de Fille");
    }
}
```



E e e

```
public static void main(String[] args) {
```

```
    Mere m=new Mere();
    Fille f=new Fille();
    m.f(3);
    f.f(4);
    m=f;
    m.f(5);
    //m.g();
    ((Fille)m).g();
    f.g();
}
```

R a

f(3) de Me e

f(4) de F e

f(5) de F e

g() de F e

f() de F e

f(3) de F | e

f(b |) de Me e

g() de F | e

f() de F | e

f(3) de F | e

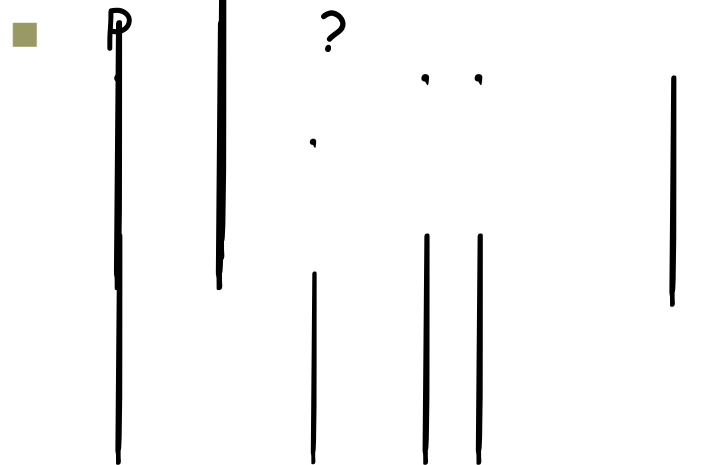
f(b |) de Me e

D) C e ce

□ E e a abe ?

■ U c e:

□ U e ' h de (e) d f e da e ca e A e
 e ' e a e e da ec e ede
 a abe d f e da ' aca e A.



POO-L3 H. Fa c e



E e e

```
class A{
    public int i=4;
    public void f(){
        System.out.println("f() de A, i="+i);
    }
    public void g(){
        System.out.println("g() de A, i="+i);
    }
}
class B extends A{
    public int i=3;
    public void f(){
        System.out.println("f() de B, i="+i);
        g();
    }
}
```

E e e e:

```
A a=new B();  
a.f();  
System.out.println("a.i="+a.i);  
System.out.println("(B) a.i="+((B)a).i);
```

D e a:

- f() de B, =3
- g() de A, =4
- a. =4
- ((B) a). =3

Re a e :

□ La a ab e de A e *occultée* a a a ab e
de B

□ La a ' ab e ' de A e e e da
b e de B

□ Le h de g de A a acc e e
a ab e d f e da A (e e e
ce e -)

□ La h de f de B *redéfinit* f. f() ed f e a
acc e e a ab e d f e da B