

# Conception Orientée Objet & Programmation JAVA

## Chapitre 6 : Les interfaces

ESPRIT - UP JAVA

Année universitaire 2020/2021



# PLAN



- Introduction
- Classe et Objet
- Encapsulation
- Héritage
- Polymorphisme
- Exceptions
- **Interfaces**
- Collection
- Interface Fonctionnelle
- Expression Lambda
- Stream



# Objectifs



- Comprendre la notion de classe abstraite
- Se familiariser avec le concept d'interfaces
- Manipuler une interface.

# ***Classe et Méthode Abstraite***



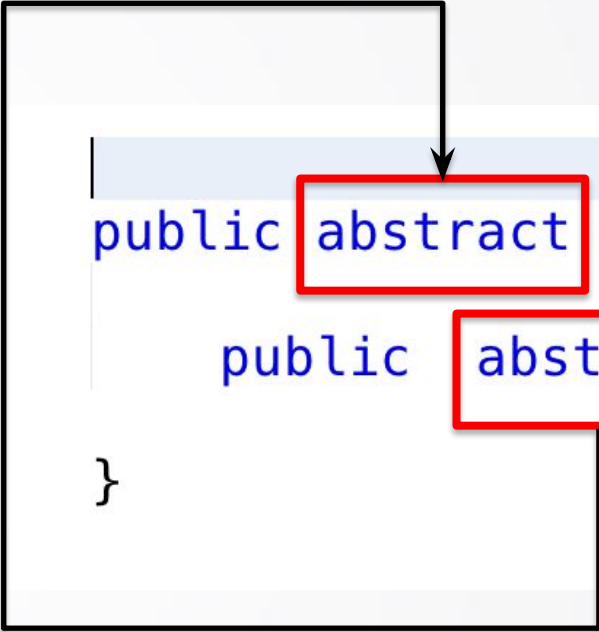


# Méthode et Classe Abstraite



- On définit une **méthode abstraite** comme étant une méthode n'ayant pas de corps
- Une fois on déclare une **méthode abstraite** dans une classe A, cette classe devient **abstraite** aussi
- Une **méthode abstraite** ne peut pas être déclarée **static** ou **private** ou **final**.

```
public abstract class Atest {  
    public abstract void m();  
}
```



# Classe et Méthode Abstraite



**<abstract>**  
**Personnel**

```
public class Test {  
  
    public static void main(String[] args) {  
        Personnel p = new Personnel();  
    }  
  
}
```

Une classe **abstraite** ne peut pas être instanciée



# Les interfaces



```
public abstract class ClassA {  
    public abstract void m1();  
    public abstract void m2();  
    public void m3() {  
        System.out.println("hello word");  
    }  
}
```

**ClassA** : classe abstraite dans laquelle on peut déclarer des méthodes concrètes (c'est le cas de la méthode ***m3()***)



# Les interfaces

```
public abstract class ClassA {  
    public abstract void m1();  
    public abstract void m2();  
    public abstract void m3();  
}
```



<<ClassA>>

Si la classe **ClassA** contient que des méthodes abstraites -> **Interface**





# Les interfaces

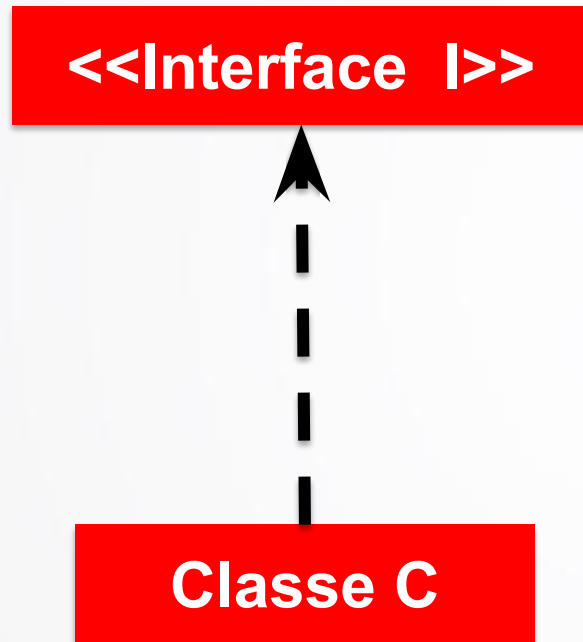


```
public interface I
{
    void f (int n) ;    //les mots clés public et abstract sont
                        facultatifs
    public abstract void g () ;
}
```

- Une interface définit un comportement (d'une classe) qui doit être implémenté par une classe, **sans implémenter ce comportement.**
- C'est un ensemble de méthodes **abstraites**
- On peut déclarer que des constantes dans une interface



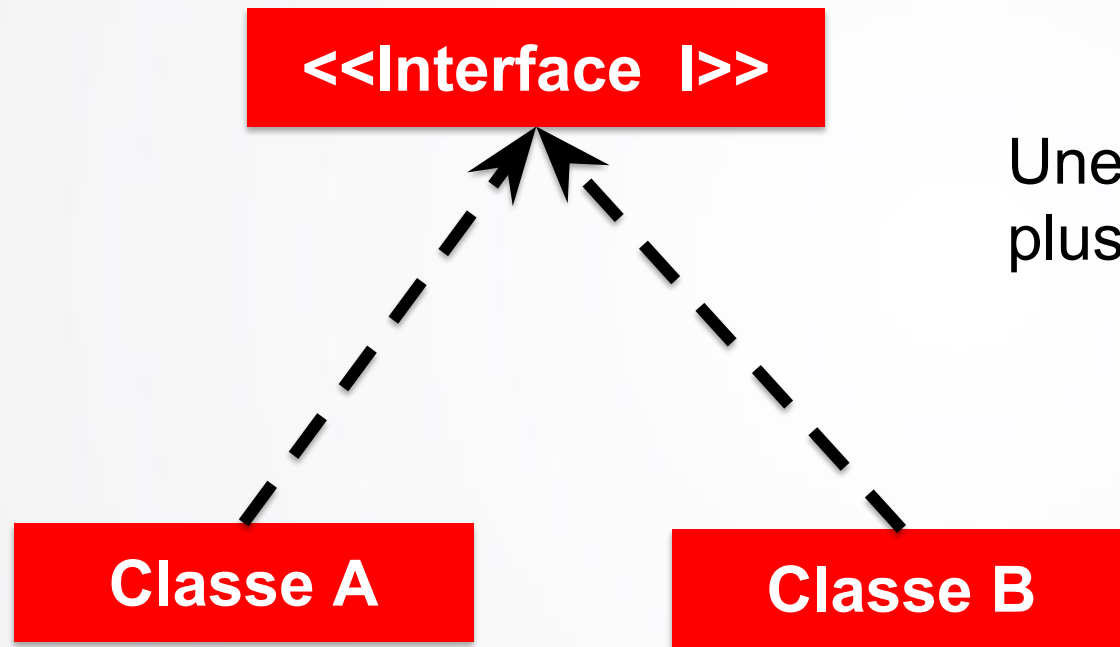
# Les interfaces



- Une classe C implémente une interface I
- La classe C doit implémenter toutes les méthodes de l'interface



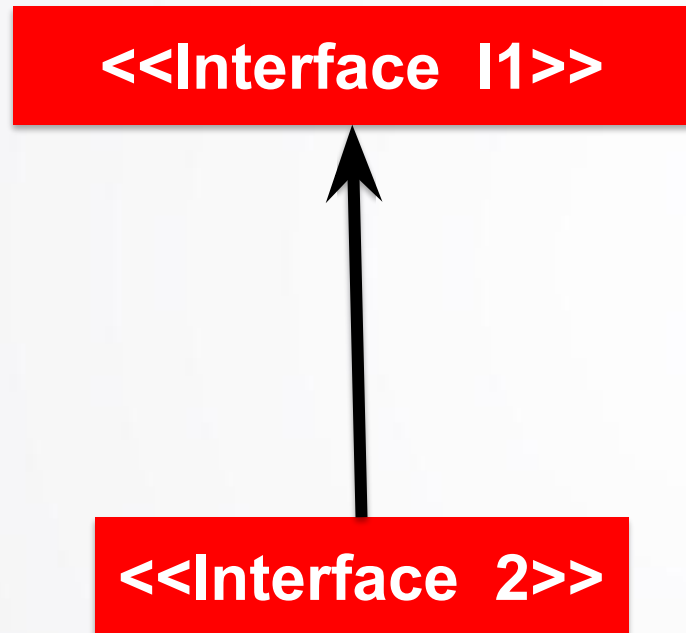
# Les interfaces



Une classe dérivée peut implémenter 1 ou plusieurs interfaces



# Les interfaces



- Les interfaces peuvent se dériver cad une interface peut hériter d'une autre interface



# Merci pour votre attention

