

Interfaces Java

- Elles jouent un rôle fondamental en Java
- Elles se définissent au même niveau que les classes
- Elles sont utilisées pour réaliser une espèce de multi-héritage

Interfaces Java

- Une interface est une collection d'opérations utilisée pour spécifier des services offerts par des classes
- Elles contiennent uniquement des définitions de constantes et déclarations de méthodes
- Une interface doit être implémentée par les classes offrant ces services (ces classes s'engagent à fournir ces services)

Interfaces Java

- En java on utilise le mot clé **Interface**:

```
interface NomInterface{  
    // constante ....  
    // entête de méthodes ....  
}
```

- Une constante se définit par le mot-clé **final**

Interfaces Java

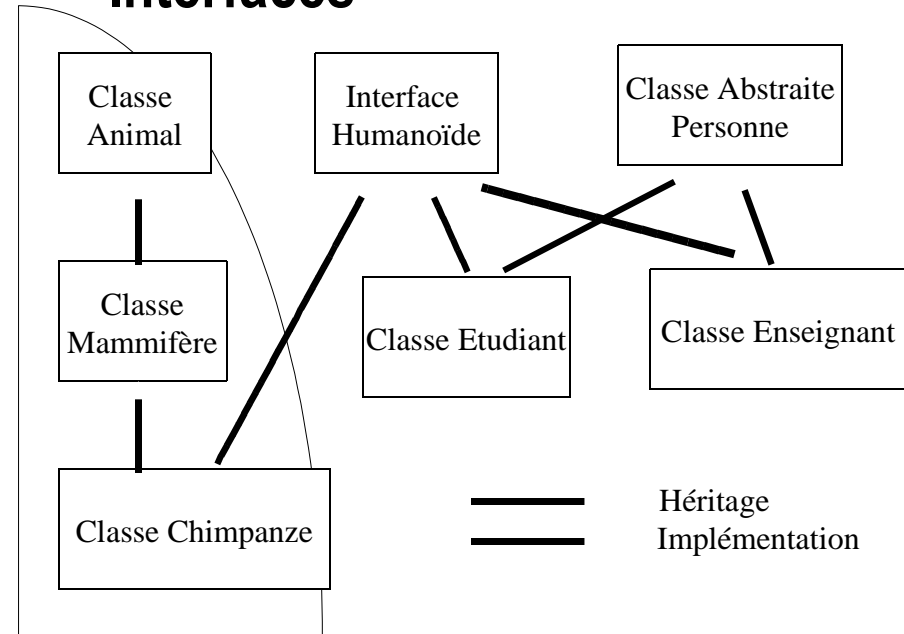
- Une classe peut implémenter une interface
- On utilise le mot clé **implements** :

```
Class NomClasse implements NomInterface{  
    ....  
    ....  
}
```

Interfaces Java

- Une classe qui implémente une interface :
 - dispose des constantes de l'interface
 - doit déclarer en public toutes les méthodes prototypées dans l'interface (sinon la classe est abstraite)

Interfaces



Interfaces

- Une classe peut implémenter plusieurs interfaces

```
Class NomClasse implements I1, I2, I3 {
    ....
    ....
}
```

Interface

- Une interface définit des objets
- Dans l'exemple précédent une instance de Chimpanzé :
 - est une instance de Mammifère
 - est une instance d'Animal
 - est un Humanoïde
- On peut donc définir des tableaux d'Humanoïde, des variables d'instance de type Humanoïde

Interface

- Une interface peut hériter d'une autre interface
- Une « sous-interface » hérite des constantes et des déclarations de sa « super interface »
- Une « sous-interface » peut définir ses propres constantes et déclarations de méthode

Interface

- A la différence d'une classe, une interface peut hériter de plusieurs interfaces !!

```
interface NomInterface extends I1,I2, ... In {  
    // constante ....  
    // entête de méthodes ....  
}
```