



# B I N G G O

.net™

## Les interfaces

- une interface représente **un contrat** que la classe qui l'implémente doit **remplir**
- une interface contient uniquement des **méthodes** (ainsi qu'éventuellement des propriétés et des indexeurs) **non implémentées**
- une interface ne peut pas être implémentée
- de ce fait on ne dit pas qu'une classe hérite d'une interface, mais qu'elle **implémente une interface**.
- traditionnellement le nom d'une interface commence par la lettre I (majuscule). Ce n'est pas une obligation, mais c'est fortement conseillé. Par exemple: IEnumerable, ILisible...



# B I N G G O

.net™

---

## Exemple d'interface

- un four à micro-ondes et une télévision ne vont pas hériter l'un de l'autre
- on peut déclarer une interface contenant les méthodes Allumer() et Eteindre()
- ainsi le four et la télévision peuvent implémenter cette interface
- le four et la télévision doivent impérativement définir, chacun, une méthode Allumer() et une méthode Eteindre()



## Définition d'une interface

```
interface IControlable
{
    void Allumer();
    void Eteindre();
}
```

- une interface est souvent déclarée "public" ou "internal". Par défaut elle est "private"
- toutes les méthodes d'une interface sont "public" et "abstract". Il est donc inutile, et même interdit, d'indiquer ces deux mots-clés
- bien entendu les les méthodes peuvent retourner une valeur et avoir des paramètres.



## "Utilisation" d'une interface

```
class Television: AppareilMenager, IControlable
{
    public void Allumer()
    {
    }
    public void Eteindre()
    {
    }
}
```

- la classe (ici Television) qui implémente l'interface doit implémenter toutes les méthodes de l'interface
- dans notre exemple, la classe Television **hérite de la classe AppareilMenager** et **implémente l'interface IControlable**



## "Utilisation" d'une interface (suite)

Une utilisation possible des interfaces consiste à parcourir les éléments d'une collection. Ces éléments peuvent être de type différent:

```
ArrayList mesAppareils = new ArrayList();
mesAppareils.Add (new Television());
mesAppareils.Add (new Four());

foreach (IControlable app in mesAppareils)
{
    app.Eteindre();
}
```



B I N G G O

.net