
Compte-rendu

Classe abstraite

Sommaire

1) Qu'est-ce qu'une classe abstraite ?	p3
2) Définitions des méthodes sous forme algorithmique	p4
3) Diagramme de classe UML	p5
4) Conclusion	p6

1) Qu'est qu'une classe abstraite ?

Nous allons nous intéresser dans ce compte rendu à la notion de classes abstraites.

Une classe abstraite est une classe si et seulement si elle n'est pas instanciable. C'est une base à d'autres classes dérivées.

Cette classe est composée de méthodes abstraites, dites **abstract** en c#.

Une méthode abstraite ne contient pas de corps, mais doit être implémentée dans les sous classes non abstraites : **abstract** nomDeMéthode (<arguments>).

L'intérêt de déclarer une classe abstraite est qu'elle fournira un cadre plus strict lors de la définition des méthodes, obligeant donc ces méthodes à être définie.

Les classes qui héritent de cette classe (donc les classes filles) seront obligés de posséder des corps de méthodes, méthode héritée de la classe mère qui est abstraite.

2) Définition des méthodes sous forme algorithmique

Voici notre classe mère (abstraite) :

```
5 références
abstract class Equipement
{
    private string nom;
    private int prix;

    1 référence
    public Equipement(string nomSai, int prixSai)
    {
        this.nom = nomSai;
        this.prix = prixSai;
    }
}
```

Qui possède ses méthodes abstraites :

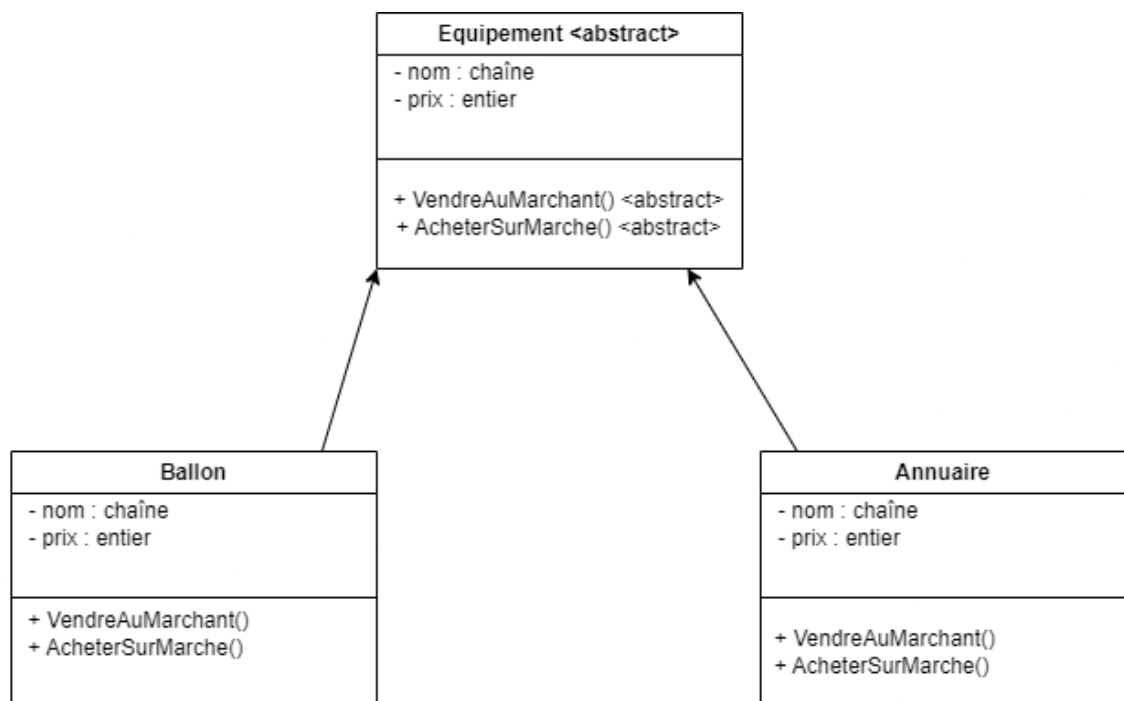
```
1 référence
public abstract void AcheterSurMarche();
```

Et voici la classe fille avec ses méthodes qui hérite de cette dernière :

```
2 références
class Annuaire : Equipement
{
    1 référence
    public override void AcheterSurMarche()
    {
        Console.WriteLine("J'achete sur l'annuaire");
    }

    1 référence
    public override void VendreAuMarchant()
    {
        throw new NotImplementedException();
    }
}
```

3) Diagramme de classe UML



4) Conclusion

En P.O.O, le terme de classes abstraites est très important, notamment en ce qui concerne les héritages.

En effet, les classes filles héritant des méthodes abstraites se verront obligé d'être définie et déclarée.

Les classes abstraites seront également utiles dans la notion d'interface, que nous étudierons un peu plus tard.