
Compte-rendu

Dictionnaire

Sommaire

| | |
|---|----|
| 1) Qu'est-ce qu'un dictionnaire ? | p3 |
| 2) Et les assertions ? | p4 |
| 3) Définitions des méthodes sous forme algorithmique | p5 |
| 4) Diagramme de classe UML | p6 |
| 5) Conclusion | p7 |

1) Qu'est-ce qu'un dictionnaire ?

Précédemment, nous avons vu que les collections nous permettaient de nous passer des tableaux en P.O.O.

Maintenant, intéressons-nous maintenant aux dictionnaires, que l'on peut très simplement appeler des tableaux associatifs.

Les dictionnaires sont des classes « génériques », les classes génériques sont des classes qui encapsulent des opérations qui ne sont pas spécifiques à un type de données particulier.

Ils fonctionnent avec un système de collection de clefs, auquel l'on associe une valeur, donc une collection de pair clef/valeur.

Voici un schéma descriptif :

| | | | |
|----------|--------|---------|---------------|
| Clef 1 | Clef 2 | Clef 3 | Clef 4 |
| Valentin | Simon | Charles | Robot Farceur |

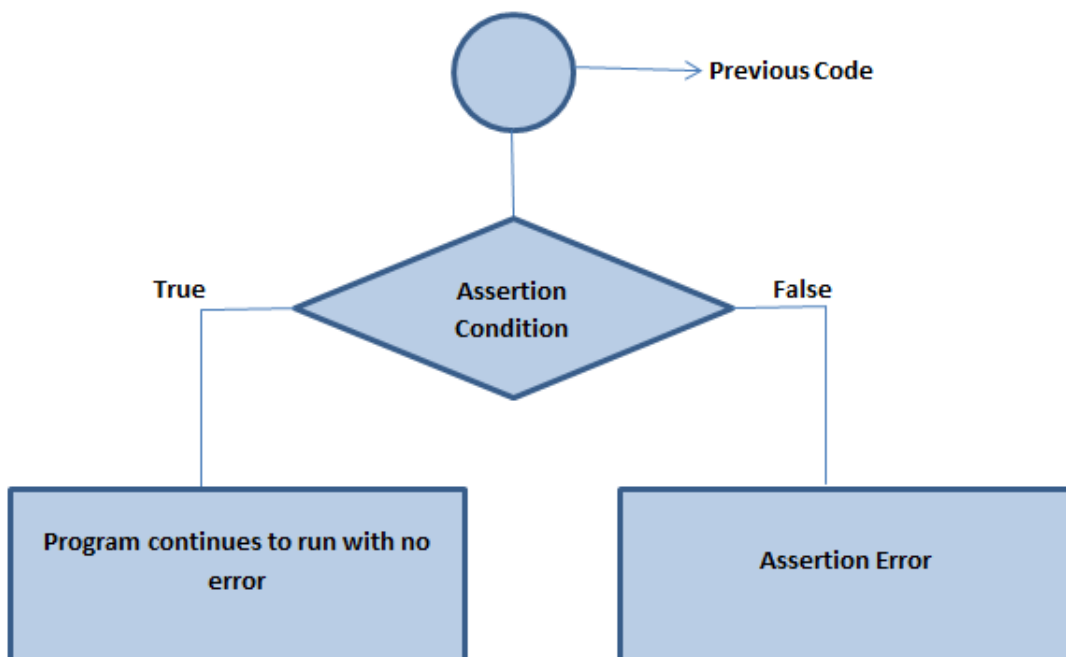
Un dictionnaire s'écrit comme ceci : dictionnaire<Tkey, Tvalue>.

2) Et les assertions ?

Une assertion, en programmation, est un énoncé présenté comme vrai mais qui n'est pas encore vérifié voire non vérifiable, et potentiellement faux.

C'est une proposition que l'on avance et qu'on prétend vraie.

L'instruction Assert permet de tester des suppositions sur le comportement du code. L'intérêt de cette notion est de détecter rapidement et avec efficacité les erreurs de programmation.



3) Définition des méthodes sous forme algorithmique

Voici comment on déclare un dictionnaire dans le langage C# :

```
private Dictionary<string, int> releveHoraire = new Dictionary<string, int>();
```

Dans ce cas-ci, notre dictionnaire s'appelle releveHoraire.

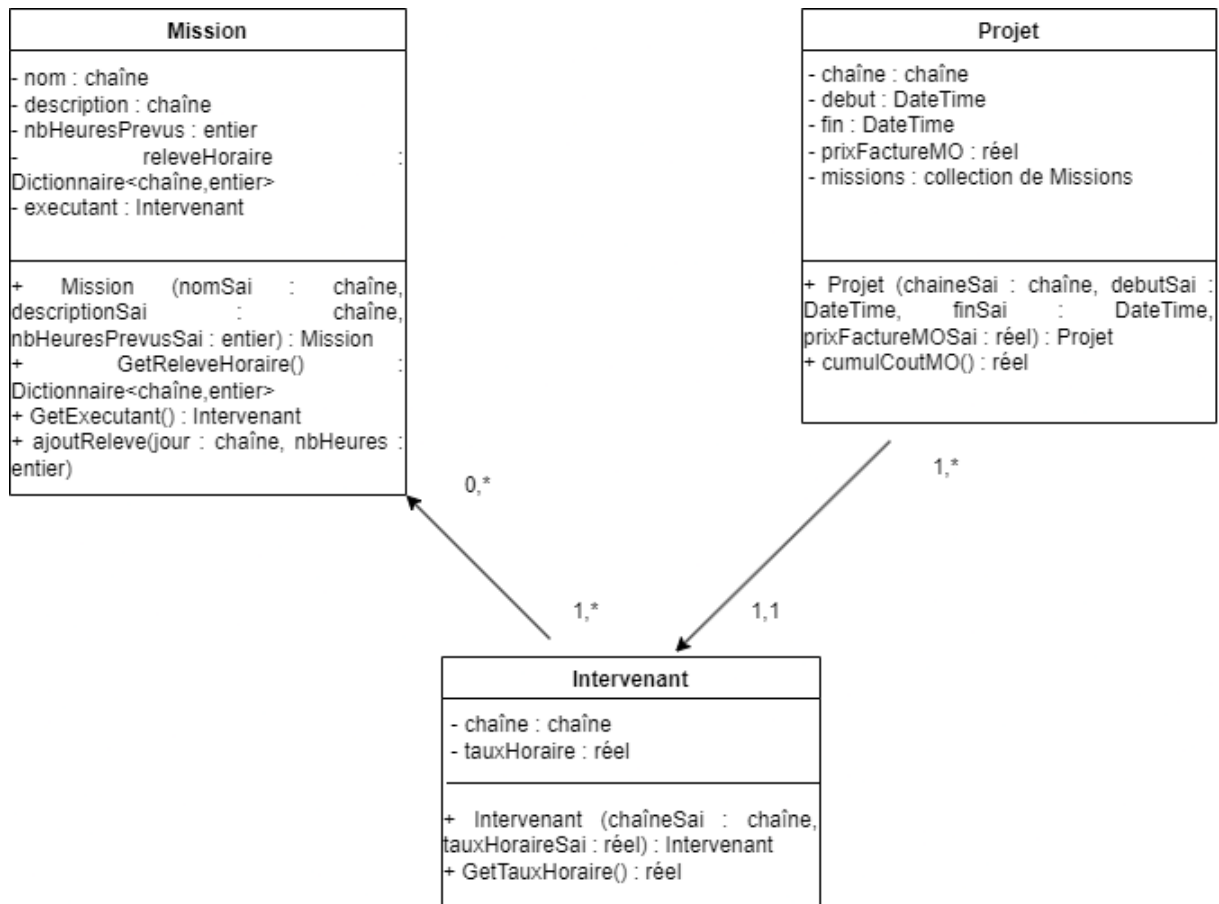
N'oublions pas le getReleveHoraire() :

```
1 référence  
public Dictionary<string, int> getReleveHoraire()  
{  
    return releveHoraire;  
}
```

Maintenant, ajoutons la méthode ajoutReleve, faisant intervenir des méthodes de dictionnaire :

```
2 références  
public void ajoutReleve(string jour, int nbHeures)  
{  
    if (releveHoraire.ContainsKey(jour))  
    {  
        nbHeures = nbHeures + releveHoraire.GetValueOrDefault(jour);  
        releveHoraire.Remove(jour);  
    }  
    if (nbHeures <= 8)  
    {  
        releveHoraire.Add(jour, nbHeures);  
    }  
    else  
    {  
        releveHoraire.Add(jour, 8);  
    }  
}
```

4) Diagramme de classe UML



5) Conclusion

Les dictionnaires sont donc une approche des classes « génériques » permettant d'associer des clefs à des valeurs, ce qui facilite le parcours des collections, et donc des dictionnaires en P.O.O.

Les assertions quant à elle sont un moyen rapide et efficace de détecter les erreurs de programmation. Elles sont visibles par des test unitaires (« frameworks de test »).

Ces deux notions sont donc utiles et indispensables à connaître en programmation.