

TP 4 : Linq

Exercice 1

Considérons la classe **Stagiaire** ayant les propriétés suivantes :

- Nom de type chaîne de caractère
- Notes de type tableau de réels (float [] Notes)

Et la classe **Formation** avec les propriétés suivantes :

- Intitulé de type chaîne de caractère
- NbrJours de type entier
- Stagiaires de type tableau de **Stagiaire**

Pour chaque classe, générez **ToString** et deux constructeurs sans et avec tous les paramètres.

Dans Main, utilisons la liste des formations suivante :

```
Stagiaire stagiaire1 = new Stagiaire("Wick", [12, 14, 16]);
Stagiaire stagiaire2 = new Stagiaire("Dalton", [4, 6, 8]);
Stagiaire stagiaire3 = new Stagiaire("Widmore", [10, 13, 16]);
Stagiaire stagiaire4 = new Stagiaire("Clinton", [5, 15, 10]);
Stagiaire stagiaire5 = new Stagiaire("Scofield", [20, 20, 11]);

Formation[] formations = [
    new Formation("C#", 57, [stagiaire1, stagiaire2, stagiaire4]),
    new Formation("Java", 15, [stagiaire1, stagiaire3, stagiaire4, stagiaire5]),
    new Formation("PHP", 57, [stagiaire2, stagiaire4])
];
```

Utilisez Linq pour répondre aux questions suivantes :

1. Afficher l'intitulé et le nombre de jours de chaque formation.
2. Afficher l'intitulé de chaque formation ainsi que le nombre des stagiaires qui y sont inscrits.
3. Afficher l'intitulé de chaque formation ainsi que les noms des stagiaires qui y sont inscrits.
4. Afficher les noms des stagiaires qui participent à au moins une formation (Pas de doublons).
5. Afficher les noms des stagiaires et leurs moyennes.
6. Afficher les trois meilleures moyennes.
7. Afficher les notes des stagiaires qui participent à la formation C#.
8. Afficher la meilleure note de tous les stagiaires qui participent à la formation Java.
9. Afficher les noms des stagiaires qui participent à la fois à la formation C# et Java.
10. Afficher les noms des stagiaires qui participent à la fois à la formation C# et Java mais pas PHP.
11. Afficher la moyenne de la formation C# (la moyenne des moyennes des stagiaires qui y participent).
12. Afficher l'intitulé de la formation qui a la meilleure moyenne de formation.

Exercice 2

Dans un nouveau projet, créez deux classes **Personne** et **Adresse**.

Les propriétés de la classe **Adresse** sont :

- Rue de type chaîne de caractère,
- Ville de type chaîne de caractère,
- CodePostal de type chaîne de caractère.

Les propriétés de la classe **Personne** sont :

- Nom de type chaîne de caractère,
- Prenom de type chaîne de caractère,
- DateNaissance de type DateTime,
- Adresses de type List<Adresse>.

Pour chaque classe, générez ToString et deux constructeurs sans et avec tous les paramètres.

Dans Main, utilisons la liste des personnes suivante :

```
Adresse adresse1 = new("paradis", "Marseille", "13006");
Adresse adresse2 = new("prado", "Marseille", "13008");
Adresse adresse3 = new("plantes", "Paris", "75014");
Adresse adresse4 = new("victor_hugo", "Lyon", "69002");
Adresse adresse5 = new("tunis", "Toulouse", "31200");

List<Personne> personnes = [
    new Personne("wick", "john", new DateTime(1988, 2, 11), [adresse1, adresse3]),
    new Personne("dalton", "jack", new DateTime(1985, 7, 30), [adresse2, adresse4]),
    new Personne("maggio", "carol", new DateTime(2002, 10, 25), [adresse1]),
    new Personne("benamar", "sophie", new DateTime(1990, 5, 5), [adresse2, adresse3, adresse5])
];
```

Utilisez Linq pour répondre aux questions suivantes :

1. afficher Nom + " " + Prénom de chaque personne de la liste.
2. afficher l'âge de chaque personne de la liste.
3. afficher les noms des personnes ayant au moins deux adresses.
4. afficher les noms des personnes ayant une adresse à Paris.
5. afficher toutes les adresses (n'affichez pas deux fois la même adresse).
6. afficher le nom et le nombre d'adresses à Paris de chaque personne de la liste.
7. afficher les noms des personnes nées en une année bissextile.
8. afficher l'âge max de la liste.
9. afficher le nom de la personne la plus âgée.
10. vérifier s'il existe une personne majeure ayant une adresse à Paris.
11. supprimer les personnes qui n'ont aucune adresse à Paris.
12. afficher le nombre de jours qui nous sépare de l'anniversaire le plus proche.

13. vérifier que toute personne ayant une adresse à **Paris** a également une adresse ailleurs.
14. afficher les villes et leurs nombres d'occurrence dans le tableau **personnes**.
15. vérifier que toute personne ayant une adresse à **Lyon** n'a pas une adresse à **Toulouse** et inversement.
16. afficher le nom de chaque personne et son nombre d'adresses.
17. afficher les noms des personnes qui n'ont ni le plus grand nombre d'adresses, ni le plus petit.
18. afficher les noms des personnes ayant le plus grand nombre d'adresses.