# Exercices LINQ (2)

## Objectifs

* Etre capable d’écrire une requête(query) LINQ
* Pouvoir utiliser les opérateurs LINQ suivants :
  + Select
  + Where, OrderBy
  + Count, Min, Max, Sum, Average
  + GroupBy, Join, GroupJoin

## Exercice 4– Opérateurs Count, Min, Max, Sum, Average

### Exercice 4.1

Donner le résultat annuel moyen pour l’ensemble des étudiants.

### Exercice 4.2

Donner le plus haut résultat annuel obtenu par un étudiant.

### Exercice 4.3

Donner la somme des résultats annuels.

### Exercice 4.4

Donner le résultat annuel le plus faible.

### Exercice 4.5

Donner le nombre de lignes qui composent la « table » STUDENT.

## Exercice 5– Opérateurs GroupBy, Join, GroupJoin

### Exercice 5.1

Donner pour chaque section, le résultat maximum (Max\_Result) obtenu par les étudiants.

### Exercice 5.2

Donner pour toutes les sections commencant par 10, le résultat annuel moyen (AVG\_Result) obtenu par les étudiants.

### Exercice 5.3

Donner le résultat moyen (AVG\_Result) et le mois en chiffre (BirtMonth) pour les étudiants né le même mois entre 1970 et 1985.

### Exercice 5.4

Donner pour toutes les sections comptant plus de 3 étudiants, la moyenne des résultats annuels ( AVG\_Results).

### Exercice 5.5

Donner pour chaque cours le nom du professeur responsable ainsi que la section dont il fait partie.

|  |  |  |
| --- | --- | --- |
| **Course\_name** | **Section\_name** | **Professor\_name** |
| Derivatives | BA Sociology | giot |
| Marketing management | MSc Economics | decrop |
| Marketing engineering | MSc Management | zidda |
| Financial Management | BA Sociology | giot |
| Supply chain management … | MSc Management | scheppens |

### Exercice 5.6

Donner pour chaque section, l’id, le nom et le nom de son délégué. Classer les sections dans l’ordre inverse des ids de section.

|  |  |  |
| --- | --- | --- |
| **Section\_id** | **Section\_name** | **Last\_name** |
| 1320 | MA Sociology | Basinger |
| 1310 | BA Sociology | Reeves |
| 1120 | MSc Economics | Basinger |
| 1110 | BSc Economics | Marceau |
| 1020 | MSc Management | Portman |
| 1010 | BSc Management | Willis |

### Exercice 5.7

Donner pour toutes les sections les professeurs qui en sont membres.

section\_id -> section\_name :

professor\_name1

professor\_name2

…

### Exercice 5.8

Même objectif que le 5.7 mais seules les sections comportant au moins 1 professeur doivent être prises en compte.

### Exercice 5.9 (BONUS)

Donner à chaque étudiant ayant obtenu un résultat annuel supérieur ou égal à 12 son grade en fonction de son résultat annuel et sur base de la table grade. La liste doit être classée dans l’ordre alphabétique des grades attribués.

|  |  |  |
| --- | --- | --- |
| **Last\_name** | **Year\_result** | **Grade** |
| Berry | 18 | E |
| Garner | 18 | E |
| Garcia | 19 | E |
| Basinger | 19 | E |
| Connery | 12 | S |
| Roberts | 17 | TB |
| Bacon | 16 | TB |

### Exercice 5.10 (BONUS)

Donner la liste des professeurs et la section à laquelle ils se rapportent ainsi que le(s) cours ( nom du cours et crédits) dont le professeur est responsable. La liste est triée par ordre décroissant de crédits attribués à un cours.

|  |  |  |  |
| --- | --- | --- | --- |
| **Professor\_name** | **Section\_name** | **Course\_name** | **Course\_ects** |
| Giot | BA Sociology | Financial Management | 4.0 |
| Zidda | MSc Management | Marketing engineering | 4.0 |
| Decrop | MSc Economics | Marketing Management | 3.5 |
| Giot | BA Sociology | Derivatives | 3.0 |
| Scheppens | MSc Management | Supply chain …. | 2.5 |
| Lecourt | BA Sociology | NULL | NULL |
| Louveoux | BSc Economics | NULL | NULL |

### Exercice 5.11 (BONUS

Donner pour chaque professeur son id et le total des crédits ECTS (ECTSTOT) qui lui sont attribués. La liste est triée par ordre décroissant de la somme des crédits alloués.

|  |  |
| --- | --- |
| **Professor\_id** | **ECTSTOT** |
| 3 | 7 |
| 1 | 4 |
| 2 | 3.5 |
| 5 | 2.5 |
| 6 |  |
| 4 |  |