

# <sup>1</sup>Production finale d'intégration, *CRUDlard* TiGris à l'école des sorciers....



Hiver 2022

- Ce travail sera réalisé individuellement.
- À terminer pour le 31 mai avant minuit pour tous les groupes.
- Le travail compte **pour 20 %** de la note finale.

## Objectifs :

Ce travail vise principalement à vous faire expérimenter les aspects suivants:

- À partir d'un modèle de données, créer et exploiter une base de données relationnelle.
- Normaliser une base de données de petite taille.
- Réaliser une application C# ADO.NET

## Consignes :

1. Vous devez télécharger le fichier CRUDlard.sql et l'enregistrer sur votre disque.
2. Exécuter votre fichier CRUDlard.sql pour commencer à écrire les requêtes de la partie 1
3. Vos requêtes SQL doivent être bien présentées (voir exemple), sinon elles ne seront pas corrigées. **Sachez qu'une longue requête sur une seule ligne est difficile à corriger.**

Exemple :

```
SELECT nom_acteur, prenom_acteur
FROM ((acteurs
INNER JOIN Acteurs_film ON acteurs.id_acteur= Acteurs_film.id_acteur)
INNER JOIN films ON films.id_film= Acteurs_film.id_film)
WHERE titre = 'Pirates des Caraïbes,Jusqu"au bout du monde';
```

- Vous pouvez écrire vos requêtes tout en minuscules si vous le souhaitez.
- Lorsqu'il s'agit d'une jointure interne (INNER JOIN), il est conseillé de ne pas mettre les parenthèses pour laisser l'optimiseur de requêtes trouver la façon la plus rapide d'exécuter la requête.
- Lorsqu'il s'agit d'une jointure externe (LEFT/OUTER JOIN), les parenthèses sont nécessaires.

<sup>1</sup> Source de l'image

<https://pixabay.com/fr/illustrations/ch%C3%A2teau-de-sorcellerie-vieille-maison-5683182/>

## Mise en contexte :

Nous sommes à l'école des sorciers « CRUdlard » où les élèves apprennent à devenir des sorciers par la maîtrise de certaines techniques décrites dans différents livres, ici appelés *Grimoires*. Un grimoire est un livre de magie.

Un grimoire a un identifiant, un titre et un type de grimoire .

Les élèves sont identifiés par un numéro unique, ils ont un alias unique, un nom, un prénom , ils ont un solde en écus, un nombre de points.

Les élèves appartiennent à des groupes. Dans un groupe nous avons plusieurs élèves.

Pour lire un grimoire, les élèves doivent **payer** un certain montant. Après la lecture d'un grimoire, les élèves obtiennent des **points**.

Le nombre de points et le prix de lecture dépendent du type de grimoire. Nous avons 5 types de grimoires : Air, eau, terre, feu, sort.

Après la lecture d'un grimoire, les élèves peuvent écrire un commentaire et donner une évaluation pour le grimoire. Les commentaires et les évaluations sont stockés dans une table : evaluationgrimoire.

De temps en temps, l'école organise des tournois entre les élèves de groupes différents. Le tournoi se déroule entre deux élèves de groupes différents. Lors d'une partie d'un tournoi un élève peut obtenir un maximum 5 points. Exemple

Lorsque l'élève Kiwi affronte Fantomas, Kiwi obtient 3 et Fantomas 2.

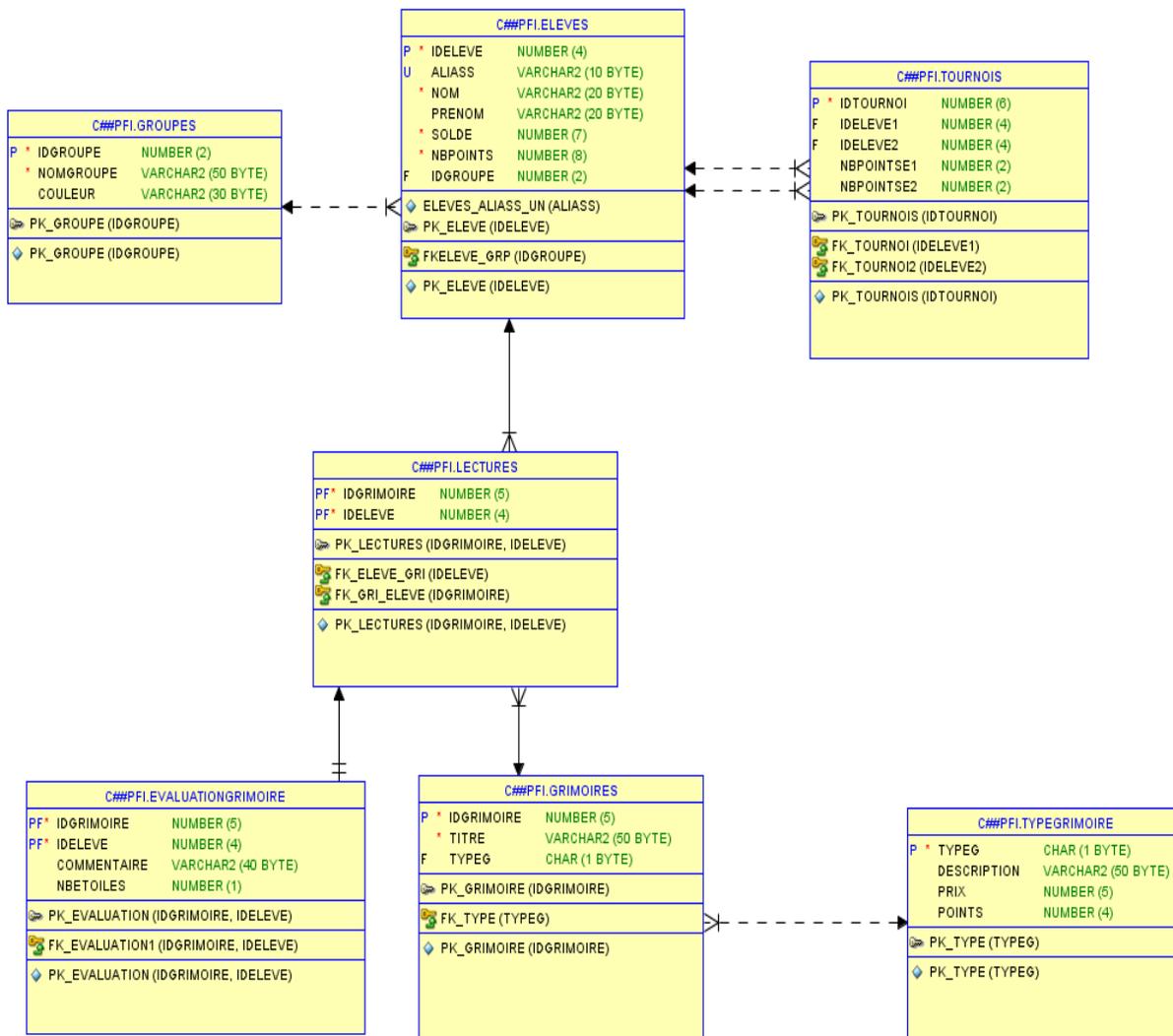
Lorsque l'élève Chubaka affronte Féroce, Chubaka obtient 4 points et Féroce 1 point.

ALIAS_ELEVE1	ALIAS_ELEVE2	NBPOINTSE1	NBPOINTSE2
Kiwi	Fantomas	3	2
Chubaka	Féroce	4	1

Finalement, le nombre total de points d'un élève est égal au nombre de total de points obtenus en lisant des grimoires plus (+) le nombre total de points obtenus dans les tournois.

Le modèle de la base de données De CRUdlard est donné dans le diagramme suivant.

## Modèle relationnel de la base de données de CRUDlard :



On vous demande de bien lire le modèle, de comprendre les liens entre les tables ainsi que leurs contenus. Vous ne pouvez pas réussir l'écriture des requêtes si vous n'avez pas compris le modèle

## Travail demandé : SQL (80 points)



Partie 1 : Écriture de requêtes SQL (exploiter une base de données).

Pour toutes les questions de cette étape, il faudra écrire les requêtes SQL.

Télécharger le script **CRUDIrad.sql** puis exécuter le script pour

- a. Créer les tables
- b. Faire les différentes insertions.

1. Afficher le titre et le prix de tous les grimoires. Cette sortie est ordonnée par prix.
2. Afficher les alias des élèves ainsi que le nom de leur groupe. Cette liste doit inclure aussi les élèves qui ne sont dans aucun groupe.
3. Afficher les élèves qui n'ont lu aucun grimoire. Afficher l'alias, le nom et le prénom.
4. Afficher les alias des élèves qui sont dans le même groupe que TiGris
5. Afficher le nombre de grimoires par type de grimoire. Afficher la description ainsi que le nombre.
6. Afficher la moyenne des évaluations pour chaque grimoire. Arrondir ce chiffre à deux chiffres après la virgule. Afficher le titre et la moyenne.
7. Afficher les meilleurs grimoires selon la moyenne des évaluations. Afficher le titre du grimoire, le id du grimoire ainsi que la moyenne d'évaluation.

Vous aurez la sortie suivante.

TITRE	IDGRIMOIRE	MOYENNE
Se déplacer sur l'eau	100	4
Contrôler les rivières	101	4

8. Afficher l'alias, le nom, le prénom et le nombre d'étoiles des élèves ayant donné la même note d'évaluation au grimoire : *Marcher dans le feu*.

Vous aurez la sortie suivante :

ALIAS	NOM	PRENOM	NBETOILES
Panthère	Lavoie	Linna	5
Malin	Gagnon	Nicolas	5
Gribouille	Lapointe	Vanessa	4
PiwiPiwi	Deblanc	Racha	4

9. Afficher l'alias, le nom, le prénom, le nom du groupe et le nombre de grimoires lus des élèves ayant lu 2 grimoires ou plus. Cette sortie est ordonnée par le nombre de grimoires :

ALIAS	NOM	PRENOM	NOMGROUPE	NOMBRE_GRIMOIRE
TiGris	Courtesy	Rosa	les poussins	5
Chubaka	Clément	Sylvain	les poussins	3
Malin	Gagnon	Nicolas	les poussins	2
Kiwi	Lépine	Fred	les poussins	2
Gribouille	Lapointe	Vanessa	les chevaux	2

10. Pour chaque groupe, afficher le nombre d'élèves ayant lu des grimoires. Afficher le nom du groupe et le nombre d'élèves. Vous aurez la sortie suivante :

NOMGROUPE	Nombre d'eleves
les poussins	7
les chevaux	2
les renards	1

11. Afficher le nom des groupes ayant le nombre d'élèves ayant lu des grimoires plus grand que 2. (Selon la figure précédente seul le groupe « Les poussins » sortira.)

NOMGROUPE	Nombre d'eleves
les poussins	7

12. Afficher le nom du groupe ayant lu un grimoire de même type que le grimoire « *Marcher dans le feu* ». Cette sortie ne doit pas inclure le grimoire « *Marcher dans le feu* ».

NOMGROUPE	TITRE
les poussins	Contrôler la braise

13. Afficher les alias des élèves ayant fait une lecture, mais n'ayant pas fait d'évaluation.

ALIAS
Patoche
Primogene

14. Afficher le coût des lectures de TiGris : à chaque fois que TiGris fait une lecture de grimoire, il paye un montant. On vous demande le coût total de ses lectures. (coût de lecture est la somme des coûts de lecture pour chaque grimoire lu)

ALIAS	COUT_DE_LECTURE
TiGris	390

15. Afficher les noms des groupes ayant dépensé pour les lectures moins que le groupe les « les poussins ». Afficher le cout des lectures aussi

NOMGROUPE	COÛT_DE_LECTURE
les renards	100
les chevaux	270

16. Afficher le nombre total des points que TiGris a obtenu grâce à ses lectures. À chaque lecture de grimoire, TiGris obtient des points. On vous demande le total de ses points (de TiGris) :

TOTAL_POINTS
350

17. Mettre à jour le nombre de points de TiGris par le total des points obtenu grâce à ses lectures. (Vous ne devez pas mettre à jour le nombre de points directement par 350. Le nombre de points évolue)
18. Mettre à jour le solde de TiGris après ses lectures. (Après que TiGris ait payé le coût de ses lectures, quel est le solde qui lui reste?)
19. Donner le classement des élèves selon les lectures qu'ils ont réalisés es. Afficher l'alias et le total des points.

ALIAS	TOTAL_POINTS
TiGris	350
Chubaka	150
Gribouille	150
Malin	150
Panthère	100
PiwiPiwi	100
Cacao	100
Kiwi	100
Patoche	50
Primogene	50

20. Écrire la requête qui permet d'afficher les alias des élèves ayant le nombre de points le plus faible. (selon la requête 19, ce sera Patoche et Primogene)

Pour les questions suivantes, et pour le classement des élèves, on vous demande d'utiliser la table **tournois**.

21. La table tournois contient les numéros des élèves avec le résultat des tournois. On vous demande d'afficher la sortie suivante : (alias à la place des numéro)

ALIAS_ELEVE1	ALIAS_ELEVE2	NBPOINTSE1	NBPOINTSE2
Kiwi	Fantomas	3	2
Chubaka	Féroce	4	1
TiGris	Gribouille	3	2
Patoche	Barakuda	3	2
Fantomas	Kiwi	4	1
Féroce	Chubaka	4	1
Gribouille	TiGris	1	4
Barakuda	Patoche	2	3

22. Créer la vue Vtournois, qui classe les élèves dans **les tournois**. Le select \* from Vtournois donnera le résultat suivant :

TOTALPOINTS	ELEVE
7	3
6	7
6	4
5	2
5	8
4	12
4	1
3	11

23. Utiliser la vue pour afficher le classement des élèves dans les tournois. Utiliser l'alias de l'élève. Vous aurez la sortie suivante :

TOTALPOINTS	ALIAS
7	TiGris
6	Patoche
6	Fantomas
5	Féroce
5	Chubaka
4	Barakuda
4	Kiwi
3	Gribouille

24. Afficher le classement des élèves du groupes : *les poussins*. Afficher l'alias et le total des points.

ALIAS	TOTALPOINTS
TiGris	7
Patoche	6
Chubaka	5
Kiwi	4

25. Afficher le nombre total des points pour chaque groupe. Afficher le nom du groupe et le total des points.

NOMGROUPE	TOTAL_POINTS
1 les poussins	22
2 les renards	11
3 les chevaux	7

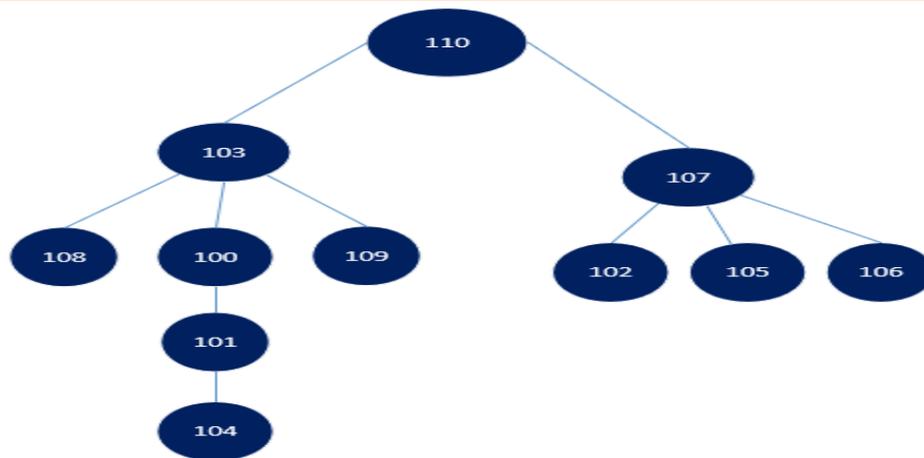
26. On vous demande d'afficher le nombre total de points que TiGris a obtenus grâce aux lectures et aux tournois. (Après exécution de votre requête vous obtiendrez 357.)

## Partie 2 : Compréhension

On veut implémenter un système de cours prérequis comme suit :

- Le grimoire final à lire est « le grimoire de Zéro » dont le numéro est 110
- Chaque grimoire a **0 ou plusieurs prérequis** (grimoires **qu'il faut avoir lus** pour pouvoir accéder à ce grimoire). *Exemple avant de lire le grimoire 100, il faut lire le grimoire 104, puis le grimoire 101.*
- Un grimoire a **1 ou zéro « objectif »** (grimoire **qu'on peut atteindre** après avoir lu ce grimoire). *Exemple après avoir lu le grimoire 102 on peut lire le grimoire 107*
- Le grimoire ayant **zéro objectif est « l'objectif final »** (une fois qu'on l'a lu, on n'atteint aucun autre grimoire). Dans notre exemple, c'est le grimoire 110.

110, Le grimoire de zéro est l'objectif final



- Comment allez-vous changer la table **grimoiresPlus** (voir le script plus bas) pour représenter ce système de prérequis?
- Mettre à jour **grimoiresPlus** pour refléter le système des prérequis (voir plus haut)
- Donner le modèle de données qui représente cette situation.
- Écrire la requête qui affiche de manière hiérarchique tous les grimoires prérequis à « **Parler aux arbres** »

Note : Pour ces questions (a, b, c), afin de ne pas détruire la structure de votre table grimoires, utilisez la table **grimoiresPlus** créé en exécutant le script suivant :

```

create table grimoiresPlus as select * from grimoires;
alter table grimoiresPlus add constraint pk_grimoireplus primary key(idgrimoire);
  
```

## Partie ADO.NET (10 points)

Vous devez concevoir un programme ADO.NET qui permet d'afficher l'information contenue dans la base de données CRUDIard. Au minimum le programme doit permettre de :

- saisir un nom d'utilisateur et un mot de passe
- se connecter à la base de données
- se déconnecter de la base de données
- consulter la liste des élèves et :
  - obtenir la liste de ses lectures. Le nombre de points total accumulés par les lectures, pour cet élève, doit aussi être affiché
  - obtenir la liste des tournois auxquels cet élève a participé. Le nombre de points accumulés par les tournois, pour cet élève, doit aussi s'afficher.

Inspirez-vous de la vue vtournois

The screenshot shows the CRUDIard application window. At the top right, there is a login section with two input fields: one containing 'c#marcb' and another with '?????????'. Below these are 'Connecter' and 'Déconnecter' buttons. The main area is divided into three sections: 'Élèves', 'Lectures', and 'Tournois'. The 'Élèves' section has a list box with 'Kiwi' selected. The 'Lectures' section contains a table with columns 'Titre', 'Desc', 'Prix', and 'Points'. The 'Tournois' section contains a table with columns 'Alias' and 'NbPoints'. Below each table is a 'Total points' label and a text box showing the sum of points for the selected student.

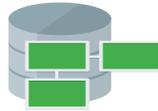
Titre	Desc	Prix	Points
Se déplacer sur l'eau	Eau	70	50
Contrôler les rivières	Eau	70	50

Total points: 100

Alias	NbPoints
Kiwi	1
Kiwi	3

Total points: 4

## Partie Normalisation (10 points)



CRUDlard souhaite développer une petite application pour la gestion de ses films de magie. Voici certaines données que nous souhaitons stocker dans la base de données **MaGilms**.

- Les films ont un numéro unique, un titre, une catégorie, l'année de sortie, la durée
- Les films sont classifiés (G, 8+, 13+). Les films sont joués par des acteurs et réalisés par un réalisateur.
- Les catégories ont un code, une description.
- Les acteurs, qui sont identifiés par un numéro unique, ont un nom, un prénom une nationalité, une date de naissance et une photo.
- Les réalisateurs, qui sont identifiés par un numéro unique, ont un nom, un prénom, une nationalité et une date de naissance.

Nous savons que :

- ✓ On peut avoir plusieurs films d'une même catégorie. Un film est dans une seule catégorie. Vous devez avoir une table catégories.
- ✓ Un acteur peut jouer dans plusieurs films. Dans un film nous avons plusieurs acteurs.
- ✓ Un film a un seul réalisateur. Un réalisateur peut réaliser plusieurs films.

Vous devez :

- 1- Faire le modèle relationnel en 3FN dans DataModeler, la remise est en PDF ou PNG.
- 2- Générer le code SQL pour la création de votre modèle. (On ne vous demande pas de créer les tables.) La remise est un simple fichier .sql.

## Modalités de remise :

1. La remise finale du travail est pour mardi le **31 mai avant minuit**
2. Le travail sera remis dans la boîte de remise.
3. Tous les fichiers doivent être clairement identifiés par Votrenom.
4. Pour les fichiers **SQL**, il faudra déposer :
  - a. Un fichier sql correspondant aux réponses aux questions de la partie SQL. À l'intérieur de ce fichier, votre nom doit y figurer. Toutes les requêtes doivent être identifiées par leur numéro.
  - b. Un fichier sql correspondant au code SQL généré par la question 2 de la partie normalisation.
5. Le modèle relationnel sera remis en format **PDF** ou en **PNG** et rien d'autre
6. La partie ADO.NET, vous devez remettre l'application zippée.
7. Tout travail non clairement identifié ne sera pas corrigé
8. Tout travail qui n'est pas dans la boîte de remise ne sera pas corrigé.
9. Jusqu'à 10 % des points seront enlevés si les consignes ne sont pas respectées.

## Évaluation, partie SQL (80% de la PFI)

Éléments évalués	Sur
Partie 2 : 1-6 (6 requêtes)	12 (6*2)
Partie 2 :7-26 (20 requêtes)	60(20*3)
Partie 2	8 (2*4)
<b>Total</b>	<b>80</b>

## Évaluation, partie ADO.NET (10% de la PFI)

Éléments évalués	Sur
Liste des élèves	2
Liste des lectures	3
Liste des tournois	3
Nombre de points cumulé par les lectures	1
Nombre de point cumulés dans le tournois	1
<b>Total</b>	<b>10</b>

## Évaluation, partie normalisation (10% de la PFI)

Éléments évalués	Sur
Le modèle normalisé	8
Le code SQL généré	2
<b>Total</b>	<b>10</b>