Création d’une bdd

* Interview pour savoir quelle information il faut dans la base de données
  + Etudiant : id, nom, prénom, @
  + Note : id, cc, exam, rattrapage
  + Relation entre Etudiant et matière : Etudiant -> avoir plusieurs -> Matière
* -> On fait une maquette entre les différents sous-groupes
* Pour la conception de la base de données : 2 modèles
  + EIA : entité association
  + UML :

Modèle entité association :

* Construction d’une base de données
  + Construire un schéma conceptuel modélisé sous forme d’entités et d’associations
  + Transformer le schéma E/A en schéma relationnel
  + Diapo
* Modèle cool car :
  + Langage graphique
  + Concept simple du « réel perçu » sous la forme de entités associations propriétés
* Modèle :
  + Entité : ID non nulle & unique
    - Propriété

Exercice 1 :

* Moi :
* Lecteur :
  + Numéro
  + Nom
  + Adresse
  + ID Société adhérente
  + ID Catégorie
* Livre :
  + ID
  + Auteur
  + Editeur
  + Nombre d’exemplaire
  + Libellé Catégorie
* Catégorie Livre :
  + Libellé
* Catégorie Lecteur :
  + ID
  + Nom/Type
* Emprunt :
  + ID Livre
  + ID Lecteur
  + Date de retour
* Société adhérente :
  + ID
  + Nom
  + Adresse
* Correction :
  + Lecteur :
    - **Num**
    - Nom
    - Adresse
  + Société :
    - **ID**
    - Nom
  + Livre :
    - **Num**
    - Nom\_auteur
    - Editeur
    - Nombre\_exemplaire
  + Type :
    - Num
    - Libellé
  + Catégorie
    - Num
    - Libéllé

Exercice 2 :

* Moi

Nouveau cours :

* Entité = une table (étudiant)
* Association : le lien implicite entre deux entités, on relit les numéros
* Cardinalité :
  + Diapo
  + Cardinalité minimale
  + Cardinalité maximale
* Double contrainte sur une patte
  + Diapo
  + X n’a pas besoin de Y mais Y à besoin de X pour être créé
* Association
  + Binaire fonctionnelle : Patte (x,y) si y = 1 -> Patte **fonctionnelle**
  + **Combinaisons de cardinalité possible** 
    - **(1,1)/(1,1) Interdit !! -> Il faudra X pour créer Y et Y pour créer X**
    - **(1,n)/(1,n) interdit !! -> Cardinalité min = 1 : il faut au moins 1 X pour créer un Y et au moins 1 Y pour créer un X**
    - **(0,1)/(1,1) rare et délicat :**
      * **Rare car on pourrait mettre Y dans la table X**
      * **Si on veut les mettre dans deux tables : il faut expliquer pourquoi dans l’explication textuel de notre modèle entité association**
    - **Une association sans contrainte = (0,n)/(0,n)**
* Propriété dans les associations :
  + == un raccourci je crois
  + Porté par une association
    - Pas trop compris/écouté
  + Règle absolue
    - On met des propriétés dans les associations uniquement si cardinalité n/n
  + Correction du modèle et des exemples
    - Client/commande : les cardinalités sont bonnes mais il faut mettre taux dans Client ; car un taux est le même par client
    - Donateur/Année : Cardinalité fausse : Un donateur peut donner (0,n) dont par ans, pas uniquement 1 min/max -> une fois la cardinalité changé la propriété est correct
* CCL règle ;
  + **Existence d’un identifiant pour chaque entité**
  + Pour chaque occurrence d’une entité, chaque propriété ne peut prendre qu’une valeur
  + Toutes les propriétés doivent être élémentaires (non décomposable) comme une adresse qui est décomposable
  + Toutes les propriété autre que l’identifiant doivent dépendre pleinement et directement de l’identifiant (pas de matière dans une table Etudiant)
* CCL règle association :
  + A chaque occurrence d’une association correspond une et une seule occurrence de chaque entité qui participe à la relation
  + Pour chaque occurrence de l’association, il ne peut exister qu’une et une seule valeur pour chaque propriété de l’association
  + Toutes les propriétés d’une association doivent dépendre de pleinement de l’identifiant de la relation

New Cours :

* Association ternaire : éviter
  + Garder une association ternaire ⬄ toutes les cards max = n
* Association entre occurrences d’une même entité
  + Lire les exemple
  + Pas de car min = 1
* Historisation
  + Solution au truc complexe -> on ajoute un (H)
  + On fait une sorte de tuple avec la date
  + Allège le modèle
* Héritage
  + Hériter des propriété de l’autre
  + Légataire = celui qui donne ces props
  + Héritier celui qui hérite

**Résumé des règles :**

* **Toujours avoir un ID**
* **Association :**
  + **(1,1)/(1,1) Interdit !! -> Il faudra X pour créer Y et Y pour créer X**
  + **(1,n)/(1,n) interdit !! -> Cardinalité min = 1 : il faut au moins 1 X pour créer un Y et au moins 1 Y pour créer un X**
  + **Diagram

    Description automatically generated(0,1)/(1,1) rare et délicat :**
* **Association entre occurrence d’une même entité** 
  + **Toutes les cardinalités minimum doivent être égales à zéro**
* **Historisation = tuple avec une date**
* **Héritage**

Chapitre 2 : RCD

* RCD -> RLD -> création table

Passage d’une modèle à l’autre :

* C’est des points gratuits au DS comme c’est automatique dans le logiciel
* Recopier les tables tel qu’elles sont en soulignant les clé primaire
* Cardinalité maximal : (.,1) et (.,n)
  + On retire l’association et on met une fleche
  + Puis on ajoute une clé étrangère du coté de l’association (.,1)
* Cardinalité max : (.,n) et (.,n)
  + L’association vas devenir une table avec plusieurs clé primaire qui peuvent être clé étrangère en même temps.
  + On ajoute les propriété de l’association en tant que propriétés de base de la table

Chapitre 3 : SQL et contrainte d’intégrité :