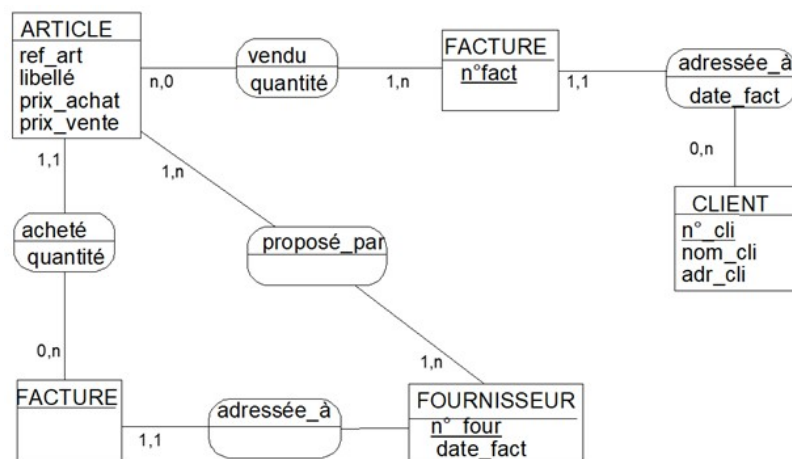


Bien sûr, voici une phrase pour chaque relation :

1. **Produits à Facturer** : Les produits sélectionnés sont prêts à être facturés.
2. **Facturer à Factures** : Le processus de facturation génère des factures.
3. **Factures à Clients** : Chaque facture est associée à un client spécifique.

1 Dans le schéma ci-dessous, repérer les erreurs, les oublis et les incertitudes :



Un client peut il être un client sans avoir de facture, doit plutôt être 1, n

La cardinalité entre article et facture, ne peut être n au minimum, et 0 au maximum. La cardinalité devrait être (0, n).

Facture fournisseur n'a pas de clé primaire

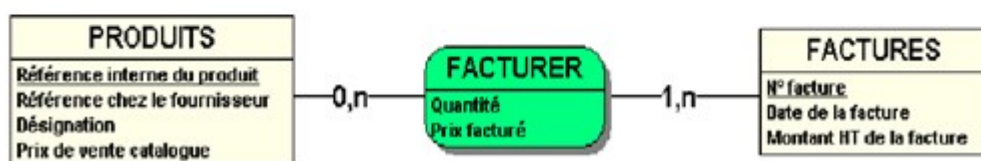
Facture ne peut pas exister sans avoir d'article, doit plutôt être (1, n). la cardinalité article facture doit être (0, n)

Cardinalité fournisseur facture est (1, n) car un fournisseur n'est fournisseur qu'à cause des factures.

Date facture doit être dans

2 Soit le modèle conceptuel des données ci-dessous concernant une partie du domaine de la gestion de la facturation des clients d'une entreprise donnée.

Approvisionnement
Domaine facturation client chez XXX
Modèle Conceptuel de Données
Projet : Exercice 1 de la fiche n° 25
Auteur : Nathan



- 1. facturer est une association binaire non fonctionnelle. Pourquoi ? Quel est l'identifiant de l'association facturer ?**

Une association binaire non fonctionnelle est une association entre 2 entités où l'élément lien entre les 2 entités n'est pas associé à un seul élément de l'autre table, cela veut dire qu'un produit n'apparaît que sur un seul article, ou une facture n'apparaît que pour un seul produit. **Dans le cas où une table dépend d'une autre**

Dans ce cas l'association est bien non fonctionnelle, on le voit par les cardinalités ayant n pour maximum. Càd qu'un produit peut apparaître sur plusieurs factures et vice versa

- 2. Justifiez par une phrase chaque couple de cardinalités de l'association Facturer. Expliquez brièvement la valeur de chaque cardinalité minimum.**

Un produit peut être facturé 0 ou plusieurs fois

La facture concerne un ou plusieurs produits

Un produit peut exister sur une ou plusieurs factures, mais il n'est pas obligé d'apparaître sur une facture. La cardinalité est de 0 à n, ce qui signifie qu'un produit peut ne pas être facturé du tout ou être facturé sur plusieurs factures.

Une facture ne peut exister que si elle contient au moins un produit. La cardinalité minimale est de 1. Une facture peut être associée à plusieurs produits, ce qui signifie qu'une facture peut regrouper plusieurs articles.

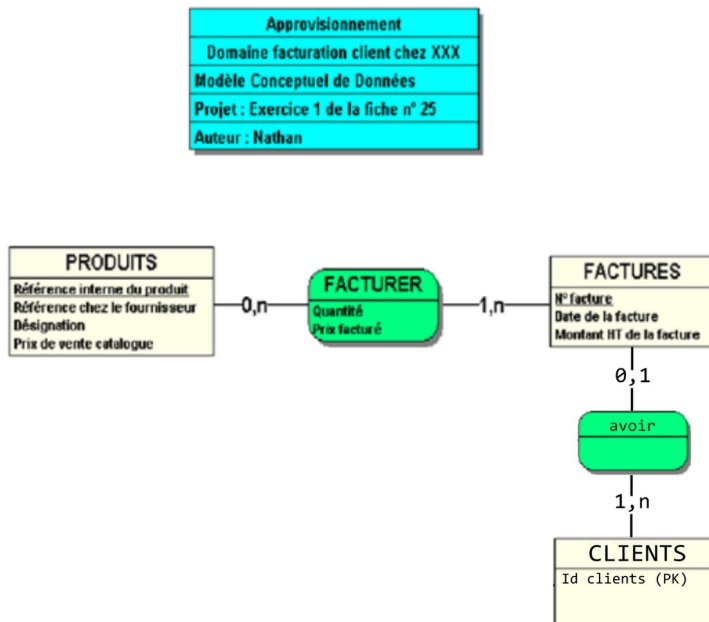
- 3. Expliquez pourquoi la propriété Prix facturé n'est pas portée par l'entité produits.**

La propriété "prix facturé" n'est pas portée par l'entité produits car elle est calculée en fonction des données présentes dans les factures. Cette information sera plutôt enregistrée dans une autre table.

Les données calculées ne sont pas souvent mises dans les tableaux, dans certain cas oui, car cela permet d'aller plus vite

Id clients

4. Ajoutez l'entité **CLIENTS** et reliez-la à **FACTURES**.

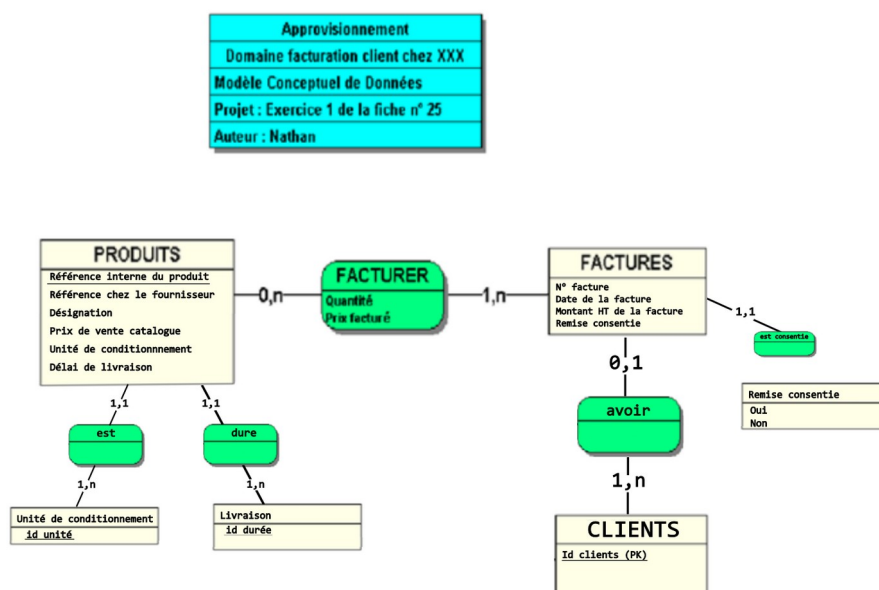


5. Ajouter dans le schéma ci-dessus les propriétés suivantes :

- Unité de conditionnement (kg, l, ml, un...) ;
- Remise consentie (variable selon la facture) ;
- Délai de livraison du produit ;
- Mode de paiement ;
- Date de livraison du produit facturé (une facture regroupe des livraisons différentes).

La valeur de la propriété Montant HT de la facture sera-t-elle saisie ? Quelle est alors sa pertinence ?

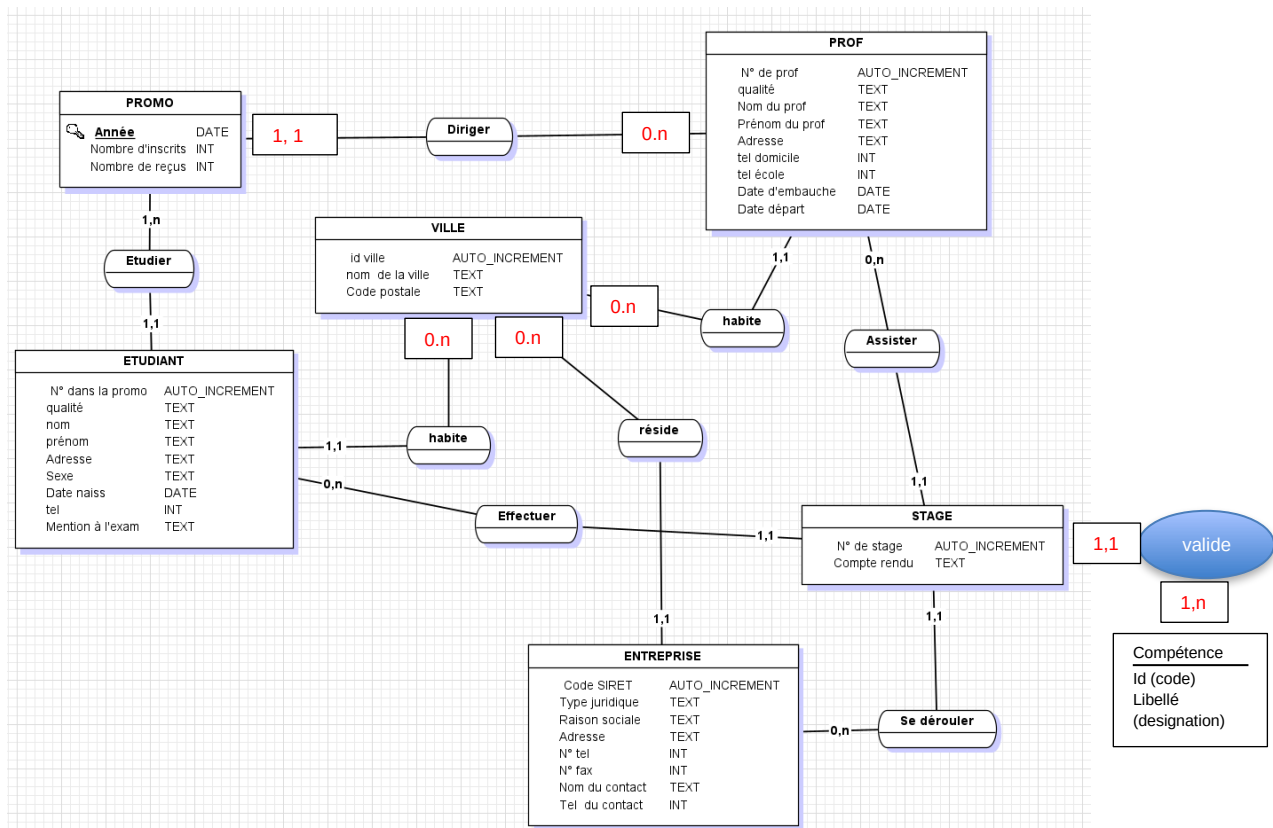
Le montant HT est une donnée pertinente, c'est la prix total des produits associé à chaque facture, il serait utile de prévoir une propriété



Descriptions :

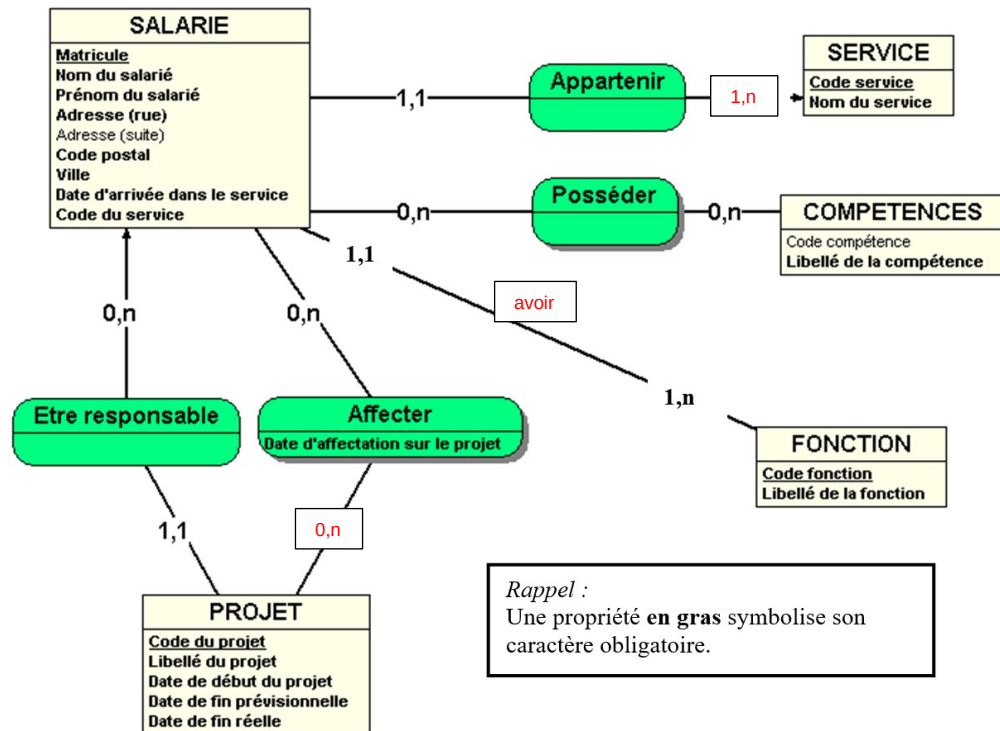
Les tables: unité de conditionnement, livraison, remise consentie. Sont des tables qui n'ont aucun but que d'optimiser la performance de la base de données. Et la rendre plus rapide. Cardinalité manquante, remise consentie, (0S.n)

3 Soit le MCD ci-dessous :



1. Relation PROMO-ETUDIANT : Une promotion peut comprendre plusieurs étudiants, tandis qu'un étudiant appartient spécifiquement à une seule promotion.
2. Relation ETUDIANT-VILLE : Un étudiant habite dans une ville spécifique, et inversement, une ville peut être le lieu de résidence de plusieurs étudiants.
3. Relation ETUDIANT-ENTREPRISE : Un étudiant effectue un stage dans une entreprise spécifique, et une entreprise peut accueillir des stages pour plusieurs étudiants.
4. Relation PROF-VILLE : Un professeur habite dans une ville spécifique, et une ville peut être le lieu de résidence de plusieurs professeurs.
5. Relation PROF-STAGE : Un professeur peut assister à plusieurs stages, tandis qu'un stage est supervisé par un seul professeur.

4 Le modèle conceptuel des données qui suit comporte des erreurs de syntaxe et de construction. Ces erreurs portent sur la syntaxe du modèle et non sur le contexte des éléments de gestion modélisés.



Retrouvez ces erreurs, indiquez en quoi la syntaxe est fausse et corrigez-les.

1. Cardinalités :

La cardinalité (n,1) n'existe pas, elle doit être remplacée par (1, n) ou (1,1) en fonction de si le salarié peut travailler dans plusieurs services ou non.

La relation PROJET SALARIE, doit être (1,n), ce n'est pas logique d'avoir un projet sans salariés, et un projet peut avoir plusieurs salariés, de

2. Clés primaires :

Clé primaire de code compétence doit être en gras, car elle est obligatoire (obligation DBMS), doit également être souligné