BD Examen

# *Réponses aux questions :*

1. ***Spécialisation*** :



***Intérêts*** : permet de cibler plus spécifiquement

***Représentation des valeurs distinctes de TypeAssujetti*** :



***Intérêts*** : utile dans le cas si on veut disposer des valeurs distinctes des NumTVA

1. L'utilisation d'un index simplifie et accélère les opérations de recherche, de tri, de jointure ou d'agrégation effectuées par le SGBD

Index nécessaire quand lien 0-1/1-1. Sans index, la cardinalité 0-1 serait perdue. Il faut mettre un index sur la clé étrangère de la table FacturetImp spécifiant une classe unique.



Chemin 1



Chemin 2

On sait retrouver le représentant en passant par Client et Région. Mais pas redondant car dans commande(CDE) il y a la date donc par le premier chemin on trouve le représentant au moment de la commande. Pour le chemin deux c’est le représentant actuel de la région au moment de la commande. C’est donc deux informations différentes donc pas de redondance.

C’est souvent au niveau des dates où il faut réfléchir.

1. C’est utile lorsque des liens se font entre plusieurs tables. Les index permettent de faciliter les jointures et donc de trouver ce que l’on cherche en passant par moins de valeurs.

Exemple :

Il y a 100 000 commandes et 10 000 client, sans index, il faudra passer par toutes les commandes et puis tous les clients. S’il y a des indexes, comme la commandes sera lié au client, alors si on trouve la commande on trouve le client ce qui réduit énormément le nombre de recherche.

1. Le problème avec les techniques procédurales et que cela se situe dans le programme, du coup si on doit modifier le programme et qu’on oublie la vérification, il y aura un problème alors qu’avec les triggers et les checks, comme le code de la BD lui n’aura pas changé, il y aura toujours une vérification et donc il y aura une erreur SQL.
2. Fichier système contenant d’une part les valeurs(images) après modifications des pages modifiées, dans l’ordre des modifications avec les identifiants des transactions modifiantes, et d’autre part des enregistrements indiquant les débuts, validation et annulation de transactions.

Cela permet donc en cas de coupure de pouvoir récupérer ce qu’on a fait.

1. La forme normale de BOYCE-CODD est plus normalisée que la 3FN, elle élimine les dépendances fonctionnelles à l’intérieur de l’identifiant.

Exemple :

COURS (Matière, Etudiant, professeur), id= (Matière, Etudiant). Dans cet exemple c’est le professeur qui détermine la matière alors que normalement c’est l’id qui doit tout déterminer.

1. P106
2. D
3. Date entrée doit être inférieur à Date sortie.

Date Imm doit être inférieur à Date entrée(VISITE) et Date(DEVIS)

1. ***Contrainte additionnelle au niveau conceptuel*** :

C’est une contrainte d’intégrité qui, en plus, ne peut pas être représentée graphiquement dans le modèle utilisé

Exemple :

Le fait qu’un produit doit soit être composé d’une matière première et d’un produit non fini ou deux produits non fini ou encore deux matières premières.

***Limite au niveau logique ou physique*** :

Une limite d’implémentation est une contrainte liée au besoin fonctionnel qui ne p…

Ou pas entièrement, dans la solution proposée. Elle doit, dans tous les cas, être documenté…

Délibérée. La décision de poser une limite peut se produire à chaque stade de l’élaboration…

Exemple :

1. D
2. Plus l’identifiant est petit, plus il est restrictif tandis que s’il est grand il est moins restrictif car par exemple si une visite est juste caractérisée par une date alors on aura les visites que pour cette date tandis que si la visite est caractérisé par la date et le véhicule au aura toute le visite pour la date et le véhicule.



1. 

Id(AffiliationHonneur) = AFFILIATION

1. CLIENT(N°\_Cli,Adresse,NomCli)

PRODUIT(N°\_Prod,PrixUn)

COMMANDE(CLIENT,PRODUIT,Date)





1. Valeurs distinctes car :

* On peut avoir la liste
* On peut réutiliser
* Si on a une envie de contrôle



1. Représentation des instances car ainsi il y aura N numéro de téléphone pour un ouvrier



Descriptif

DateDernierContract

IdVehicule

NumClient

NomMarque

NumPlaque

NomModèle

NumChassis

DateIm

Concessionnaire









1. L’intérêt est qu’il y a bien une séparation entre employé et ouvrier.

***Contraintes additionnelles :***

* Il y a une contraintes d’exclusion.



1. ***Type spécifique*** : Employé ou Ouvrier

Il permet de faire la distinction entre employé et ouvrier

***Contraintes additionnelles :***

* Si le matricule existe dans ouvrier, il ne doit pas exister dans employé et inversement.
* Tu es soit employé soit ouvrier.



1. ***Type générique*** : Personne

Grâce à ceux-ci, il y a moins de tables

***Contraintes additionnelles*** :

* Il ne faut pas oublier de rajouter le type



1. ***Attribut mutlivalué*** :

***Représentation des instances*** :

id(Téléphone) : Client, NumTel

***Représentation des valeurs distinctes*** :

***Représentation par instanciation*** :

1. ***Attribut décomposable*** :

***Représentation des instances*** :



***Représentation des valeurs*** :

***Représentation par désagrégation*** :

***Représentation par concaténation*** :

1. 
2. La commande est liée au représentant or la commande est aussi liée client qui est elle-même liée à la région qui est elle-même lié au représentant. On pourrait alors penser à une redondance car on peut trouver le représentant par deux chemins mais en réalité ce n’est pas le cas. En effet Le lien entre commande et représentant est le représentant qu’il y avait au moment de la commande car commande à un attribut date alors que l’autre lien est le représentant actuel qui n’est pas forcément le même que celui au moment de la commande. C’est donc une fausse redondance.
3. Il pourrait être un faux ternaire si par exemple il y avait plusieurs lignes pour la commande. Dans le cas ici si la ternaire est vrai alors la commande ne comporte qu’une seule ligne sinon la ternaire est fausse car il y aura pu avoir plusieurs produits pour la commande.



Avec la transformation on voit bien que la ternaire est vrai s’il y a juste une ligne pour la commande.

Identifiant : id(Commande) : Client, Magasin, Produit

1. On sait quelle opération l'ouvrier peut faire et celle qu'il ne peut pas pour pouvoir choisir l’ouvrier adéquat pour l’opération.
2. Comme ça on peut savoir si on fait des erreurs par rapport aux estimations qu'on a eu pour la création du devis et adapter en fonction de ça pour les prochains.
3. Car comme ça on peut savoir si en fonction des années, il y a eu des changements ou non.



Il faut rajouter le libellé dans LangueOp pour savoir quelle traduction va avec quelle langue pour l’opération.