

POLYTECHNIQUE
MONTRÉAL

Questionnaire examen final

INF3710

Sigle du cours

Identification de l'étudiant(e)			
Nom :	Prénom :		
Signature :	Matricule :	Groupe :	

Sigle et titre du cours			
INF3710 - Fichiers et bases de données			
Professeur	Téléphone	Groupe	Trimestre
Prof. Alejandro Quintero – Coordonnateur Dre. Franjeh El Khoury – Chargée de cours	5077	Tous	2023-3
Jour	Date	Durée	Heures
Vendredi	8 décembre 2023	2 h 30	9 h 30
Documentation	Calculatrice	Outils électroniques	
<input type="checkbox"/> Aucune <input type="checkbox"/> Toute <input checked="" type="checkbox"/> Voir directives particulières	<input checked="" type="checkbox"/> Aucune <input type="checkbox"/> Toutes <input type="checkbox"/> Non programmable (AEP)	Les appareils électroniques personnels sont interdits.	

Directives particulières	
<ul style="list-style-type: none">- Deux feuilles 8,5 po x 11 po écrites à la main en recto – verso comme documentation autorisée.- Écrivez lisiblement.- Ne détachez pas vos feuilles.- Écrivez vos brouillons à la page 18 du questionnaire.	

Important	Cet examen contient 4 questions sur un total de 18 pages (excluant cette page).
	La pondération de cet examen est de 40 %
	Vous devez répondre sur : <input checked="" type="checkbox"/> le questionnaire <input type="checkbox"/> le cahier <input type="checkbox"/> les deux
	Vous devez remettre le questionnaire : <input checked="" type="checkbox"/> oui <input type="checkbox"/> non

L'étudiant doit honorer l'engagement pris lors de la signature du code de conduite.

Exercice 1 – Conception d'une Base de Données relationnelle : Gestion de formations continues. (17 points)

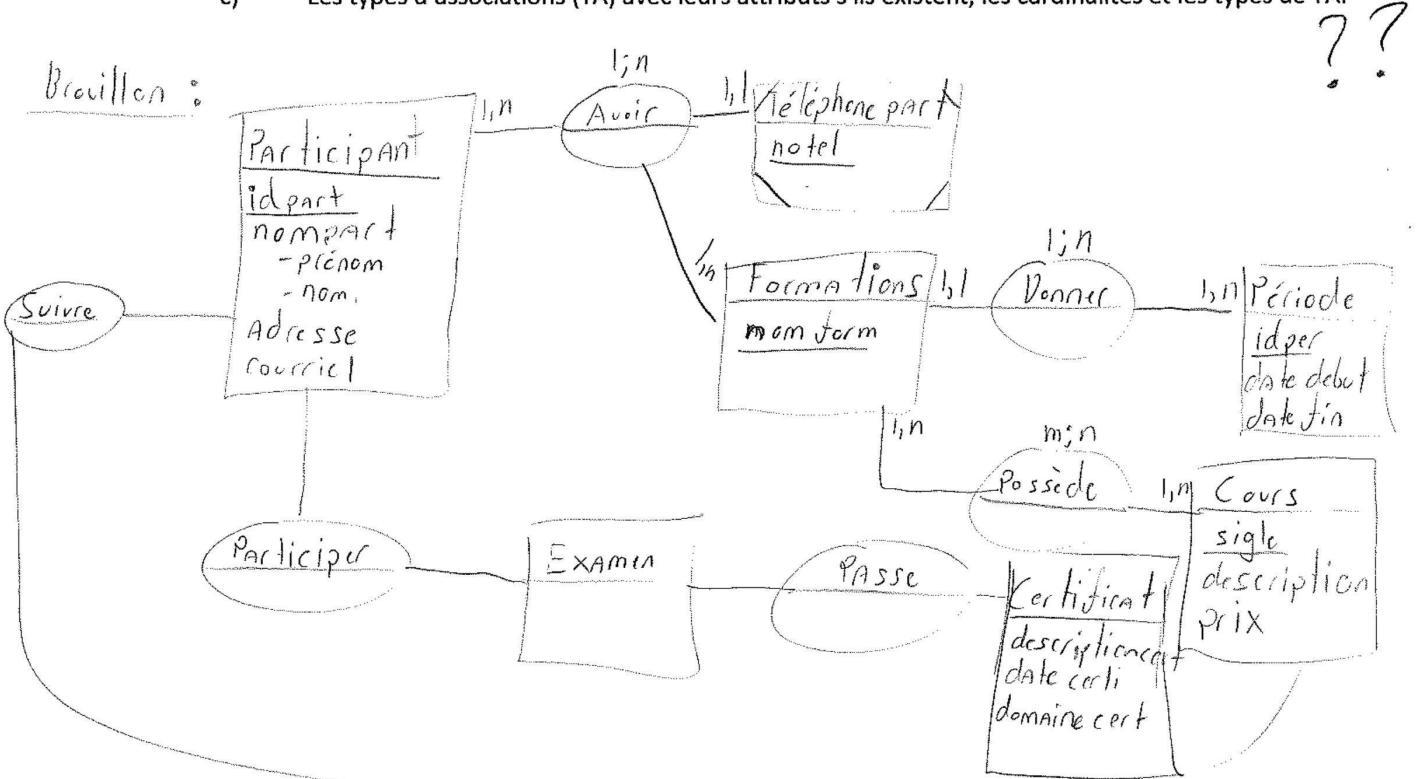
Considérons un centre de formations continues dans lequel des participants suivent des cours pour obtenir des certificats dans divers domaines. Le centre de formation souhaite informatiser le système d'information en développant une base de données pour la gestion de formations continues.

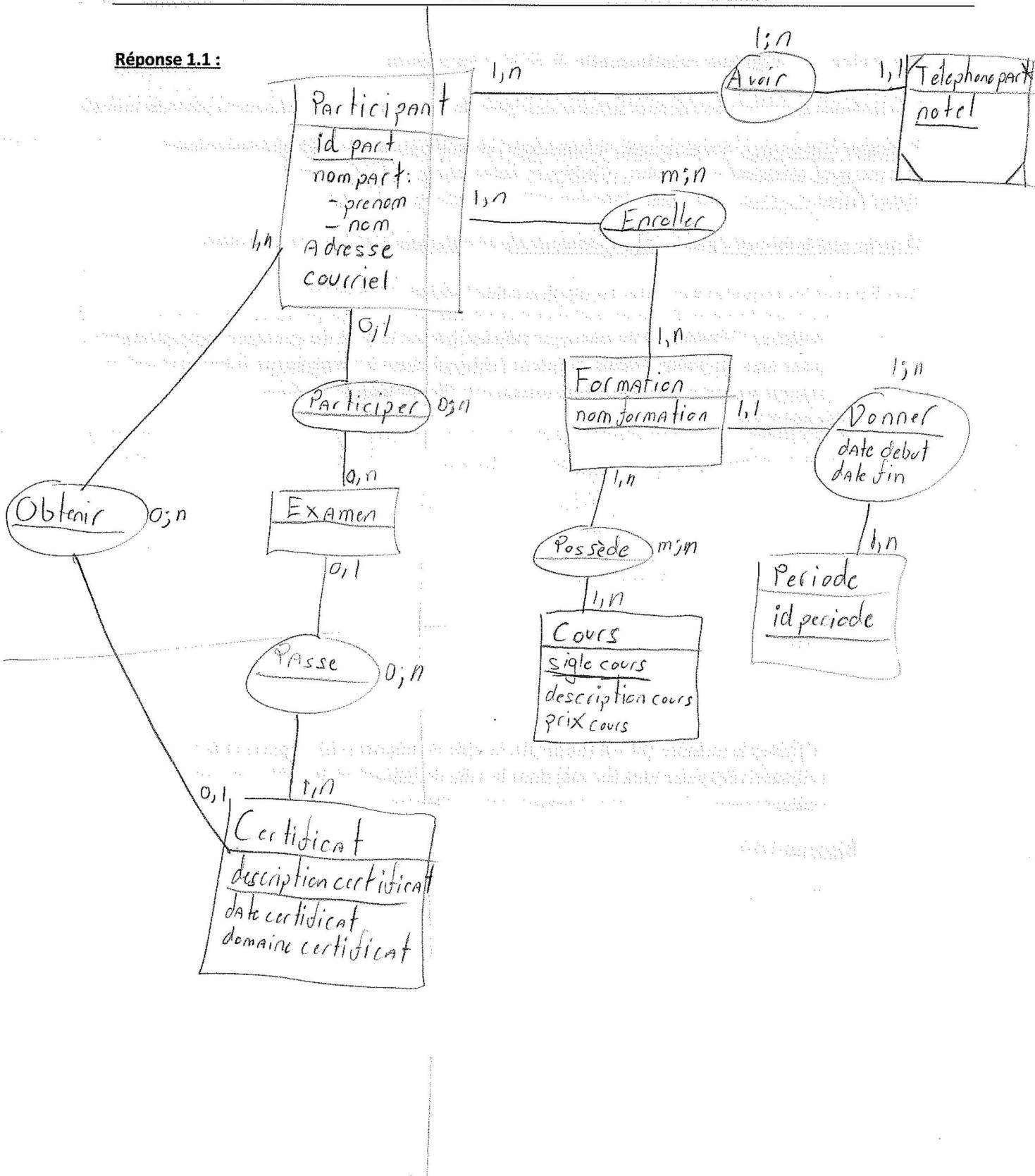
- Chaque participant est caractérisé par un identifiant, un nom complet composé d'un prénom et d'un nom, une adresse, un courriel, plusieurs numéros de téléphones propres à lui, et ses formations.
- Chaque formation est caractérisée par un nom unique, et ses propres cours. Chaque formation sera donnée sur différentes périodes. Chaque période est caractérisée par un identifiant, une date de début et une date de fin de la formation. Notons que plusieurs formations peuvent être données en parallèle pour une période déterminée.
- Chaque cours est caractérisé par un sigle, une description, un prix et sa formation.
- Chaque participant doit participer à un examen à la fin de la formation. Dans le cas où le participant passe l'examen, il obtient un certificat. Chaque certificat est caractérisé par une description, une date, un domaine et son participant.

L'étudiant suit un cours ou une formation ??

1.1- Présentez le modèle Entité-Association avec la méthode Merise en intégrant : (17 pts)

- Les types d'entités avec leur(s) identifiant(s) et leur(s) attribut(s); étudiant → Cours → formation
- Les types (fort ou faible) des types d'entités; étudiant → formation → cours
- Les types d'associations (TA) avec leurs attributs s'ils existent, les cardinalités et les types de TA.



Réponse 1.1 :

Exercice 2 – Algèbre relationnelle & SQL (34 points)

Considérons le schéma relationnel suivant pour gérer la réservation d'un vol pour la période estivale :

Passager (idpassager, nompassager, villepassager, courrielpassager, notéléphonepassager)

Vol (numvol, classevol, datedépart, villedépart, datearrivée, villearrivée, tarif)

Réservation (numvol, idpassager, dateréservation, montantréservation)

À noter que le format de datedépart, datearrivée et dateréservation est 'AAAA-MM-JJ'.

2.1- Écrivez les requêtes suivantes en algèbre relationnelle. (12.5 points)

- a) Affichez l'identifiant du passager (**idpassager**) et le nom du passager (**nompassager**) pour tous les passagers qui habitent (**villepassager**) à **Montréal** ou à **Laval** et qui ont réservé un vol ayant une classe (**classevol**) '**Économique**'. (3.5 pts)

Réponse 2.1-a :

$$\pi_{[idpassager, nompassager]} \sigma_{[villepassager = 'Montréal' \vee villepassager = 'Laval']} \wedge classevol = 'Économique'$$

$$\begin{aligned} & \left(\text{Passager} \bowtie [\text{Passager}.idpassager = \text{Réservation}.idpassager] \text{Réservation} \right. \\ & \quad \left. \bowtie [\text{Réservation}.numvol = \text{Vol}.numvol] \text{Vol} \right) \end{aligned}$$

- b) Affichez le numéro du vol (**numvol**), la ville de départ (**villedépart**) et la ville d'arrivée (**villearrivée**) pour tous les vols dont la ville de départ est la même que celle (**villepassager**) du passager (**nompassager**) '**Martin**'. (4 pts)

Réponse 2.1-b :

$$\pi_{[numvol, \delta_{[villedépart \rightarrow ville]}, villearivée]}$$

$$\left(\text{Vol} \right) \div \pi_{[\delta_{[villepassager \rightarrow ville]} \sigma_{[nompassager = 'Martin']}]} \left(\text{Passager} \bowtie \text{Vol} \right)$$

- c) Affichez le numéro du vol (**numvol**), la classe du vol (**classevol**) et le tarif du vol (**tarif**) pour tous les vols qui ont un tarif (**tarif**) compris entre 500\$ et 900\$ inclusivement, une date de départ (**datedépart**) en mars 2023, une date d'arrivée (**datearrivée**) qui n'est pas en juin 2023, et qui n'ont pas été réservés par des passagers. (5 pts)

Réponse 2.1-c :

$$\begin{aligned}
 & \pi[\text{numvol}, \text{classevol}, \text{tarif}] \sigma [\text{tarif} \geq 500 \wedge \text{tarif} \leq 900 \\
 & \quad \wedge \text{datedépart} \geq '2023-03-01' \wedge \text{datedépart} \leq '2023-03-31' \\
 & \quad \vee \text{datearrivée} \leq '2023-06-01' \vee \text{datearrivée} \geq '2023-06-31'] \\
 & \quad (\text{Vol}) \div (\text{Passager} \bowtie [\text{Passager}.id_{\text{passager}} = \text{Réservation}.id_{\text{passager}}] \\
 & \quad \text{Réservation} \bowtie [\text{Réservation}.numvol = \text{Vol}.numvol] \text{ Vol})
 \end{aligned}$$

~~erreur Adresselog ↗~~

2.2. Écrivez la requête suivante en algèbre relationnelle et en SQL. (6.5 points)

Affichez le nom du passager (**nompassager**), le numéro du vol (**numvol**), la classe du vol (**classevol**), la ville de départ (**adresselog**) et la date de réservation (**dateréservation**) pour tous les vols dont la date de réservation (**datedateréservation**) n'est pas dans l'année 2023 et le montant de réservation (**montanréservation**) n'est pas compris entre 700\$ et 1000\$ inclusivement.

Réponse 2.2 : Requête en algèbre relationnelle (3.5 pts)

$$\pi_{[nom\ passager,\ numvol,\ classe\ vol,\ ville\ de\ départ,\ date\ réservation]}$$

$$\sigma_{[date\ réservation < '2023-01-01' \vee date\ réservation > '2023-12-31']}$$

$$\wedge [montant\ réservation < 700 \vee montant\ réservation > 1000]$$

$$(Passager \bowtie [Passager.idpassager = Réservation.idpassager])$$

$$Réservation \bowtie [Réservation.numvol = Vol.numvol] \quad Vol)$$

RR
R

Réponse 2.2 : Requête en SQL (3 pts)

Select P.nompassager, V.numvol, V.classevol, V.ville départ, R.datereservation
From Passager P, Réservation R, Vol V
Where P.idpassager = R.idpassager
AND R.numvol = V.numvol
AND R.datereservation Not Between '2023-01-01' And '2023-12-31'
AND R.montantreservation Not Between 700 And 1000

2.3. Écrivez les requêtes suivantes en SQL. (11.5 points)

- a) Créez une vue v_vol qui contient la classe du vol (classevol) et le total des tarifs par classe du vol (vtottarif); dont la ville de départ (villedépart) est différente de Montréal et de Laval; le tarif du vol (tarif) est compris entre 1000\$ et 1500\$ inclusivement; le nom du passager (nompassager) ne contient pas à la quatrième position de la fin la lettre 's' mais qui contient la lettre 'a'; et la date de réservation (dateréservation) n'est pas dans l'année 2022; triés par ordre alphabétique selon la classe du vol (classevol). (5.5 pts)

Réponse 2.3-a :

```

Create View v_vol (classevol, vtottarif) As
Select V.classevol, V.tarif
From Réservation R, Passager P, Vol V
Where P.idpassager = R.idpassager
AND R.numvol = V.numvol
AND V.villedépart <> ('Montréal')
AND V.villedépart <> ('Laval')
AND V.tarif Between 1000 And 1500
AND P.nompassager Not Like '%.s--'
AND P.nompassager Like '%.a%'
AND R.dateréservation Not Between '2022-01-01' And '2022-12-31'
Order By V.classevol;

```

SQL

1h30 } reste 1h
2h30 }

- b) Affichez les classes des vols (**classevol**) et le nombre total des vols représenté comme (**Nombre Total des Vols**) par classe du vol pour tous les vols qui n'ont pas été réservés par le passager (**nompassager**) « Mathieu »; dont la ville de départ (**villedépart**) n'est pas à **Toronto**; la date de réservation (**dateréservation**) n'est pas dans les années 2021 et 2023; et la classe du vol ne contient pas la lettre '**n**' à la troisième position. (3.5 pts)

Réponse 2.3-b :

Select V.classevol, Count(V.numvol) et pas en 2023
 From Vol V, Passager P, Réservation R mais 2022 good ?
 Where P.idpassager = R.idpassager
 AND R.numvol = V.numvol
 AND P.nompassager <> 'Mathieu'
 AND V.villedépart <> 'Toronto'
 AND R.dateréservation not Between '2021-01-01' And '2023-12-31'
 AND V.classevol not Like '_n%'

- c) Incrémentez le tarif des vols (**tarif**) de 7% pour tous les vols qui ont une ville de départ (**villedépart**) « Laval »; qui ont été réservés par les passagers (**nompassager**) **Sylvain** ou **Martin** dont la date de réservation (**dateréservation**) est en mars 2023; et ayant un tarif (**tarif**) compris entre **700\$** et **1500\$**. (2.5 pts)

Réponse 2.3-c :

Select R.tarif
 From Réservation R, Vol V, Passager P
 WHERE P.idpassager = R.idpassager
 AND R.numvol = V.numvol
 AND V.villedépart = 'Laval'
 AND P.nompassager = 'Sylvain' OR P.nompassager = 'Martin'
 AND R.dateréservation Between '2023-03-01' And '2023-03-31'
 AND R.tarif Between 700 And 1500

2.4 – Déclencheurs (3.5 pts)

Supposons que nous avons les trois tables suivantes :

Employé (idemp, nomemp, salaire)

Entrepriseemployé (identreprise, idemp, moyennesalaire)

Entreprise (identreprise, nomentreprise, adresseentreprise)

Écrivez un déclencheur qui fait la mise à jour de la « moyennesalaire » dans la table « Entrepriseemployé » à chaque ajout d'un nouvel employé avec un salaire « salaire » dans la table « Employé ». À noter qu'il faut prendre en considération le cas de l'ajout d'un salaire de 700\$ d'un nouvel employé ayant par exemple un numéro (idemp) égal à 1148 et un nom (nomemp) Jean. (3.5 pts)

Réponse 2.4 :

```
CREATE OR REPLACE TRIGGER af_new_employe
AFTER INSERT ON Employé
FOR EACH ROW
Begin
UPDATE Entrepriseemployé
SET Entrepriseemployé.moyenne_salaire = Entrepriseemployé.moyenne_salaire
+ (NEW.salaire / Count(Entrepriseemployé.idemp))
```

C'est juste long à écrire, cela n'améliore pas notre compréhension

→ Trop de longs mots. Arrêter.

Exercice 3 – Normalisation des relations (36 points) Pas besoin de 6 noms
de variables longs. 1 ou 2 cela suffit.

Considérez le schéma relationnel suivant non normalisé et comportant des clés primaires potentiellement incorrectes pour la gestion de vente d'articles dans un magasin :

Magasin (idmagasin, nommagasin, addressmagasin, notéléphonemagasin, courrielmagasin, idclient)

Client (idclient, nomclient, notéléphoneclient, courrielclient, idmagasin, nommagasin)

Article (idarticle, descriptionarticle, unitémesurearticle, prixunitairearticle, idmagasin, nommagasin)

Facture (nofacture, idclient, datefacture, montantfacture, nomclient)

Lignefacture (nofacture, idarticle, quantitéfacturée, prixdevente)

Notez que :

- Chaque magasin possède plusieurs téléphones et plusieurs clients.
- Chaque client possède plusieurs téléphones et peut acheter ses articles de plusieurs magasins.
- Chaque article existe dans un seul magasin.
- Chaque facture est pour un seul client.
- Chaque lignefacture peut contenir plusieurs articles.

→ what do you mean ?

3.1- Présentez les dépendances fonctionnelles possibles et valides de toutes les relations mentionnées ci-dessus. (8 pts)

Réponse 3.1 :

Réponse 3.1 (suite) :

mag a plusieurs clients et plusieurs téléphones

3.2- Précisez la forme normale de chacune des relations dans le schéma relationnel ci-dessus.

Justifiez votre réponse. (12 pts)

Réponse 3.2 :

client + tel
+ mag
+ articles

Chaque article | magasin

Chaque facture | client

Chaque ligne facture + articles

Magasin n'est pas 1FN, car id client est multi valeurs.

client n'est pas 1FN, car id magasin est multi valeurs

Article est 1FN, est 2FN, n'est pas 3FN, car nom magasin
n'est pas direct et peut être déduit par transitivité
id article \rightarrow id magasin \rightarrow nom magasin
 \rightarrow nom magasin

Facture est 1FN, n'est pas 2FN, car nofacture, idclient \rightarrow nom client
id client \rightarrow nom client

n'est pas élémentaire

Ligne facture n'est pas 1FN, car id article est multi valeurs.

**3.3- Proposez une correction des relations précédentes pour arriver à une normalisation en FNBC.
(12 pts)**

Réponse 3.3 :



Réponse 3.3 (suite) :

- $R1 = MAGASIN(\underline{id_magasin}, nom_magasin, addr_magasin, courriel_magasin)$
- $R2 = MAGCLIENT(\underline{id_magasin}, \underline{id_client})$
- $R3 = MAGTELEPHONE(\underline{id_magasin}, no_telephone)$
- $R4 = CLIENT(\underline{id_client}, nom_client, courriel_client)$
- $R5 = CLIENTTELEPHONE(\underline{id_client}, no_telephone)$
- $R6 = CLIENTMAG(\underline{id_client}, \underline{id_magasin}, nom_magasin) \quad (\text{Redondant})$
- $R7 = ARTICLE(\underline{id_article}, description_article, unité_mesure_article, prix_unitaire_article)$
- $R8 = ARTMAGASIN(\underline{id_article}, \underline{id_magasin})$
- $R9 = FACTURE(\underline{no_facture}, \underline{id_client}, date_facture, montant_facture)$
- $R10 = LIGNEFACTURE(\underline{no_facture}, qte_facture, prix_unitc)$
- $R11 = LIGNEFACTUREARTICLE(\underline{no_facture}, \underline{id_article})$

- 3.4- Donnez le schéma relationnel final après normalisation en FNBC. (4 pts, -2 pts pour toute erreur au niveau du schéma relationnel final après normalisation)

Réponse 3.4 :

Pas le temps de tout retranscrire.

R1

R2

R3

R4

R5

R7

R8

R9

R10

R11

Exercice 4 – Organisation physique et méthodes d'accès (13 pts)**4.1- Énumérez les méthodes d'accès aux données. (3 pts)**

Accès unidimensionnelle
bi dimensionnelle

4.2- Encernez la lettre qui correspond à la bonne réponse dans les questions suivantes : (3 pts)

La méthode d'accès unidimensionnelle est basée sur : (1 pt)

- a. Une combinaison de la clé primaire et de la clé étrangère
b. Une super-clé
c. Une clé d'accès
d. Une clé étrangère
e. Une combinaison de la clé candidate et de la clé étrangère

La structure d'index est conservée : (1 pt)

- a. Avec les données
b. Séparément des données
 c. Au niveau de la mémoire

Dans un arbre-B, les nœuds du dernier niveau représentent : (1 pt)

- a. La racine
 b. Les clés d'index
c. Les enregistrements
d. Aucune de ces options

4.3- Considérons la table Produit (idproduit, descriptionproduit, unitémesure, prix) :

Qu'est-ce qui se passera pour la requête suivante? (3 pts)

Select *

From produit

Where idproduit like '%c%'

And unitémesure = 'kg'

And prix >= 5;

Cela ne fonctionnera pas

- 4.4-** Indiquez quelle est la différence entre un index primaire et secondaire. Soyez clair.e.s et précis.e.s. (4 pts)

L'index primaire l'avantage est qu'il est plus rapide, plus robuste.

C'est lui qu'on choisit par défaut.

Bon travail!