

Examen Final -Bases de données- Corrigé

Date : 07/05/2025

Durée : 1h30

Questions de cours : (5 Pts)

Q1 : Donnez deux avantages des **bases de données** par rapport aux **fichiers classiques**.

- Éliminer la redondance des données
- Centraliser et organiser correctement les données

Q2 : Quelle est la différence entre une **clé primaire** et une **clé candidate** ?

Une **clé candidate** c'est un attribut ou un ensemble d'attributs qui peut identifier de manière unique une ligne dans une table ; alors qu'une clé **primaire** c'est la clé **candidate choisie** pour représenter l'identifiant unique d'une ligne.

Q3 : Que signifie la **normalisation** des bases de données ?

La normalisation c'est l'éclatement d'une relation donnée en plusieurs relations normalisées, pour éliminer les redondances, organiser les données efficacement, et réduire le potentiel d'anomalies pendant les opérations sur les données.

Q4 : En SQL, que permet de faire la clause **GROUP BY** ?

La clause **GROUP BY** permet de regrouper les lignes d'une table selon les valeurs d'un ou plusieurs attributs.

Q5 : Reformuler cette requête SQL :

Select * **from** Client **where** Ville= 'ORAN' or Ville= 'Relizane' or Ville= 'Mostaganem'

SELECT * FROM Client **WHERE** Ville **IN** ('ORAN', 'Relizane', 'Mostaganem');

Exercice 01 : (8 Pts).

Une bibliothèque universitaire souhaite mettre en place un système de gestion informatisé pour suivre ses ouvrages, ses adhérents (étudiants, enseignants), et les emprunts.

Chaque ouvrage a un ISBN unique, un titre, un auteur, un éditeur, et une année de publication.

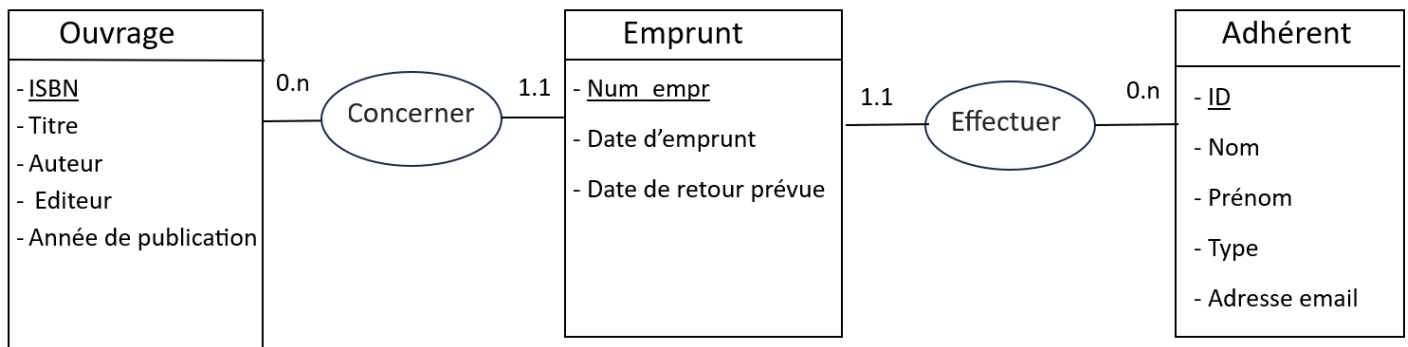
Chaque adhérent a un ID unique, un nom, un prénom, un type (étudiant ou enseignant), et une adresse email.

Chaque emprunt est identifié par un numéro unique. Il contient la date d'emprunt, et la date de retour prévue.

Un adhérent peut effectuer plusieurs emprunts. Mais un emprunt ne peut être effectué que par un seul adhérent.

Un emprunt concerne un seul ouvrage. Mais un même ouvrage peut être concerné par plusieurs emprunts.

Q1. Créez le Modèle Conceptuel de Données (MCD). (1,5)



Q2. Transformez le MCD en Modèle Logique de Données (MLD). (1,5)

OUVRAGE (ISBN, Titre, Auteur, Editeur, Annee_Publication)

ADHERENT (ID, Nom, Prenom, Type, Email)

EMPRUNT (Num_Empr, Date_Emprunt, Date_Retour_Prevue, #ID, #ISBN)

Q3. Écrivez les requêtes SQL suivantes :

3.1 Créer les tables : Ouvrage, Adherent, Emprunt . (1,5)

```
CREATE TABLE OUVRAGE (  
  ISBN VARCHAR(20) PRIMARY KEY,  
  Titre VARCHAR(100),  
  Auteur VARCHAR(100),  
  Editeur VARCHAR(100),  
  Annee_Publication INTEGER);
```

```
CREATE TABLE ADHERENT (  
  ID INTEGER PRIMARY KEY,  
  Nom VARCHAR(50),  
  Prénom VARCHAR(50),  
  Type VARCHAR(20),  
  Email VARCHAR(100) );
```

```
CREATE TABLE EMPRUNT (  
  Num_Emprunt INTEGER PRIMARY KEY,  
  Date_Emprunt DATE,  
  Date_Retour_Prevue DATE,  
  ID INTEGER,  
  ISBN VARCHAR(20),  
  FOREIGN KEY (ID) REFERENCES ADHERENT(ID),  
  FOREIGN KEY (ISBN) REFERENCES OUVRAGE(ISBN));
```

3.2 Insérer un étudiant et un enseignant dans la table Adherent. (0,5p)

```
INSERT INTO ADHERENT VALUES (202401, 'Benali', 'Amine', 'étudiant',  
'benali.amine@univ.dz');
```

```
INSERT INTO ADHERENT VALUES (202402, 'Sahraoui', 'Leila', 'enseignant',  
'sahraoui.leila@univ.dz');
```

3.3 Afficher le nombre total des adhérents. (0,5p)

```
SELECT COUNT(*) FROM ADHERENT;
```

3.4 Mettre à jour l'adresse email de l'adhérent avec l'ID 202401 à 'amine121@gmail.com' (0,5)

```
UPDATE ADHERENT  
SET Email = 'amine121@gmail.com'  
WHERE ID = 202401;
```

3.5 Afficher tous les ouvrages de l'auteur «Nadjib mahfoudh», classés par année de publication en ordre décroissant. **(0,5)**

```
SELECT * FROM OUVRAGE  
WHERE Auteur = 'Nadjib mahfoudh' ORDER BY Annee_Publication DESC;
```

3.6 Supprimer les ouvrages publiés avant l'année 1990. **(0,5)**

```
DELETE FROM OUVRAGE  
WHERE Annee_Publication < 1990;
```

3.7 Afficher l'ouvrage le plus ancien. **(0,5)**

```
SELECT * FROM OUVRAGE  
WHERE Annee_Publication = ( SELECT MIN(Annee_Publication) FROM OUVRAGE );
```

3.8 Afficher tous les adhérents ayant emprunté un ouvrage le 02 MAI 2025. **(0,5)**

```
SELECT * FROM ADHERENT  
WHERE ID IN ( SELECT ID FROM EMPRUNT  
              WHERE Date_Emprunt = '2025-05-02' );
```

Exercice 02 : (7 Pts).

Soit la relation "Vente", qui décrit les informations concernant des ventes de produits effectuées par des vendeurs dans différents magasins. La relation comprend les attributs suivants :

NoVente (Numéro de la vente), NoMagasin (Numéro du magasin), NoProduit (Numéro du produit), NomVendeur (Nom du vendeur), NomProduit (Nom du produit), DateVente (Date de la vente), et MontantVente (Montant de la vente).

Les dépendances fonctionnelles suivantes sont définies pour cette relation :

(NoVente, NoMagasin) → NoProduit

NoProduit → NomProduit

NoVente → NomVendeur

NoVente → DateVente

NoVente → MontantVente

Q1 : Quelle est la clé primaire de cette relation ? Justifiez votre réponse. (02 p)

Nous avons les dépendances fonctionnelles suivantes :

$(\text{NoVente}, \text{NoMagasin}) \rightarrow \text{NoProduit} \dots 1$

$(\text{NoVente}, \text{NoMagasin}) \rightarrow \text{NoVente} \dots 2$ (Par Réflexivité)

$(\text{NoVente}, \text{NoMagasin}) \rightarrow \text{NoMagasin} \dots 3$ (Par Réflexivité)

$\text{NoProduit} \rightarrow \text{NomProduit} \dots 4$

De 1 et 4 on trouve que $(\text{NoVente}, \text{NoMagasin}) \rightarrow \text{NomProduit} \dots 5$ (Par transitivité)

$\text{NoVente} \rightarrow \text{NomVendeur} \dots 6$

$\text{NoVente} \rightarrow \text{DateVente} \dots 7$

$\text{NoVente} \rightarrow \text{MontantVente} \dots 8$

De 2 et 6 on trouve que $(\text{NoVente}, \text{NoMagasin}) \rightarrow \text{NomVendeur} \dots 9$ (Par transitivité)

De 2 et 7 on trouve que $(\text{NoVente}, \text{NoMagasin}) \rightarrow \text{DateVente} \dots 10$ (Par transitivité)

De 2 et 8 on trouve que $(\text{NoVente}, \text{NoMagasin}) \rightarrow \text{MontantVente} \dots 11$ (Par transitivité)

De 1, 2, 3, 5, 9, 10, et 11 on déduit que l'ensemble $(\text{NoVente}, \text{NoMagasin})$ est une clé pour la relation Vente, car il détermine tous les attributs de cette relation

Q2 : Quelle est la forme normale de la relation "Vente" ? Justifiez votre réponse. (1,5p)

1. **1FN** :  Oui (la relation a une clé, et les attributs sont atomiques)

2. **2FN** :  Non

○ Car $\text{NoVente} \rightarrow \text{NomVendeur}, \text{DateVente}, \text{MontantVente}$: dépendance **partielle** de la clé primaire $(\text{NoVente}, \text{NoMagasin})$

3. **3FN** :  Non, car elle n'est pas en 2FN

Q3 : Proposez une décomposition pour cette relation en un ensemble de relations en 3FN si nécessaire. (1,5p)

VENTE (NoVente, NoMagasin, NoProduit)

PRODUIT (NoProduit, NomProduit)

INFO_VENTE (NoVente, NomVendeur, DateVente, MontantVente)

Q4 : Exprimez les requêtes suivantes en algèbre relationnelle :

4.1 : Les noms des produits vendus dans le magasin numéro 5. **(0,5)**

$\Pi \text{ NomProduit } (\sigma_{\text{NoMagasin} = 5} (\text{VENTE}))$

4.2 : Les informations de vente (Numéro de la vente, Vendeur, Montant) pour les ventes réalisées après le 1er janvier 2023. **(0,5)**

$\Pi \text{ NoVente, NomVendeur, MontantVente } (\sigma_{\text{DateVente} > '2023-01-01'} (\text{VENTE}))$

4.3 : Les numéros de produits vendus par le vendeur "Anis" au magasin numéro 10. **(0,5)**

$\Pi \text{ NoProduit } (\sigma_{\text{NomVendeur} = 'Anis' \wedge \text{NoMagasin} = 10} (\text{VENTE}))$

4.4 : Listez les noms des vendeurs qui ont réalisé des ventes supérieures à 5000 da. **(0,5)**

$\Pi \text{ NomVendeur } (\sigma_{\text{MontantVente} > 5000} (\text{VENTE}))$