

TD2 Bases de Données et interopérabilité

Ruiwen HE
Département d'Informatique

1. Quiz

- Q1. Comment définiriez-vous la notion de vue?
- A. C'est un moyen de pré-calculer un résultat
 - B. C'est un moyen de créer une deuxième base dérivée de la première par exécution de requêtes SQL
 - C. C'est un moyen de pré-définir des requêtes
- Q2. Quelles restrictions s'appliquent à l'interrogation d'une vue?
- A. On ne peut pas effectuer de jointure entre une vue et une table
 - B. Aucune
 - C. On ne peut pas créer de vue sur une autre vue
- Q3. Dans lequel des cas suivants une vue ne peut-elle pas être mise à jour ?
- A. contenir un opérateur ensembliste (UNION, MINUS, INTERSECT)
 - B. La clause select ne contient que les noms des attributs de la relation
 - C. Une clause GROUP BY et HAVING
 - D. La clause FROM contient plusieurs relations de base de données.
- Q4. Combien peut-on créer d'index en forme d'arbre B sur une table?
- A. un seul, car l'arbre B est non dense
 - B. deux, celui sur la clé primaire, et celui sur un autre attribut au choix
 - C. autant que l'on veut
- Q5. Indiquer quelles affirmations sont vraies
- A. une entrée dans une feuille peut référencer une autre feuille
 - B. une entrée dans une feuille référence toujours un enregistrement du fichier de données
 - C. une entrée dans un nœud interne référence toujours une feuille
 - D. une entrée dans un nœud interne référence toujours un autre nœud interne
 - E. une entrée dans un nœud interne référence un bloc de l'arbre, feuille ou nœud interne
- Q6. À propos de l'éclatement, que peut-on dire?
- A. tout éclatement entraîne l'ajout d'un niveau à l'arbre
 - B. un éclatement nécessite l'ajout d'au moins un bloc à l'arbre B
 - C. l'éclatement de la racine nécessite l'ajout d'exactly 2 blocs
 - D. un éclatement déplace les enregistrements d'un bloc plein vers les blocs voisins

2. Les vues

- 2.1. Créez une vue `vueEtu` qui donne pour chaque étudiant dont la base contient le numéro d'étudiant et le nom de chaque étudiant . La vue est-elle modifiable ? Pourquoi ? SI oui,insérez les données suivantes:

```
1 INSERT INTO list_attendance VALUES ( '15', 'Anna' );
```

Le tableau original est-il modifié ?

Si Anna est insérée dans la table `student`, la vue sera-t-elle modifiée ?

- 2.2. Créez une vue `vueEtuSc` qui donne pour chaque étudiant dont la base contient le numéro d'étudiant, le nom, les note du cours de chaque étudiant,la vue est-elle modifiable ? Pourquoi ?

- 2.3. Créez une vue `vueEtuSc_cv` qui donne pour chaque étudiant dont la base contient le numéro d'étudiant, le nom, le note du cours computer vision de chaque étudiant à partir de la vue `vueEtuSc`,la vue est-elle modifiable ? Pourquoi ?

- 2.4. L'école souhaite demander des bourses pour les étudiants en remplissant les conditions suivantes

- Condition 1 : Suivre au moins deux cours;
- Condition 2 : Né après 1990;
- Condition 3 : Note moyenne supérieur à 14;

Créez une vue qui remplit les trois conditions ci-dessus (le nom de la vue est `ViewPractice1_5`). La vue est-elle modifiable ? Pourquoi ?

- 2.5. Renommer la vue (`ViewPractice1_5`) en `list_bourse`.

- 2.6. Détruisez cette vue `list_bourse`

3. Les séquences

- 3.1. Ecrire et faire exécuter un script SQL pour créer une table Insérer dans votre table les lignes suivantes (id généré automatiquement) :

| ID | Prenom | Metier | Salaire | Date d'entrée | Date de sortir |
|----|----------|---------------------|---------|---------------|----------------|
| 1 | Bob | Data scientist | 43 | 2001-09-10 | 2003-03-09 |
| 2 | Janny | Data architecte | 40 | 2007-05-23 | 2013-07-22 |
| 3 | Tom | Lead data scientist | 60 | 2007-01-19 | |
| 4 | Steven | Data scientist | 45 | 2021-04-01 | |
| 5 | Alice | Data scientist | 42 | 2022-09-10 | |
| 6 | Felix | Lead data scientist | 55 | 2001-09-10 | 2006-11-01 |
| 7 | Laura | Data scientist | 43 | 2000-05-25 | 2003-11-09 |
| 8 | Louis | Data scientist | 40 | 2019-09-13 | 2022-09-10 |
| 9 | Aurelien | Data architecte | 41 | 2023-01-01 | |
| 10 | Sandro | Data engineer | 38 | 2023-09-01 | |
| 11 | Lucas | Data engineer | 40 | 2023-10-10 | |

- 3.2. Supprimez les salariés qui ont quitté l'entreprise, redéfinissez l'ID.

- 3.3. Recréez une table `HR`, ajoutez les trois employés suivants, ID commence par 7

- Sophie, Gestion_paie, 35, 2024-01-01;

- David, responsable_RH, 38, 2024-01-01;
- Annick, charge_recrutement, 30, 2024-01-01;

4. Les indexes

4.1. Laquelle des instructions SQL suivantes est la plus efficace à exécuter ? Pourquoi ?

```
1 SELECT * FROM user WHERE sid = 06;
2 SELECT * FROM user WHERE sname = 'sophie' ;
```

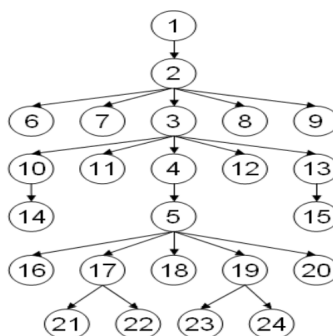
4.2. Soit les deux tables suivantes:

```
1 CREATE TABLE R (
2   idR VARCHAR(20) NOT NULL,
3   PRIMARY KEY (idR));
4 CREATE TABLE S (
5   idS INT NOT NULL,
6   idR VARCHAR(20) NOT NULL,
7   PRIMARY KEY (idS),
8   FOREIGN KEY idR
9   REFERENCES R);
```

Indiquez, pour les ordres SQL suivants, quels index peuvent améliorer les performances ou optimiser la vérification des contraintes primary key et foreign key.

```
1 SELECT * FROM R WHERE idR = 'Bou'
2 SELECT * FROM R WHERE idR LIKE 'B%'
3 SELECT * FROM R WHERE length(idR) = 3
4 SELECT * FROM R WHERE idR LIKE '_ou'
5 INSERT INTO S VALUES (1, 'Bou')
6 SELECT * FROM S WHERE idS BETWEEN 10 AND 20
7 DELETE FROM R WHERE idR LIKE 'Z%'
```

4.3. Etant donné l'arbre T suivant :



- Déterminer pour l'arbre T, sa racine, sa taille, sa hauteur, sa profondeur, ses nœuds intérieurs et ses feuilles.
- Pour le nœud 4, déterminer son parent, ses frères, sa hauteur, sa profondeur, ses ancêtres et ses descendants propres.
- Déterminer les nœuds qui sont à gauche du nœud 4, et ceux qui sont à sa droite.

4.4. On considère des arbres B d'ordre 2 (les noeuds et les feuilles ont entre 2 et 4 valeurs). La racine a entre 1 et 4 valeurs.

- Soit un arbre A1, contenant les valeurs suivantes : 2, 9, 13, 16, 18, 23, 24, 27, 30, 38, 41, 42, 49, 55, 58. Dessinez l'arbre A1, sachant que la racine contient les valeurs 18, 26, 34, 55.
- On insère successivement dans A1 les valeurs 28, 31, puis 36. Dessinez l'arbre A2 après insertion de ces valeurs.
- Dans l'arbre A1 d'origine, on supprime successivement les valeurs 55, 38, puis 42. Dessinez l'arbre A3 après suppression de ces valeurs. En cas de fusion nécessaire, on considérera d'abord la fusion avec le voisin de gauche, puis si impossible, avec le voisin de droite.

4.5. Soit la liste des départements suivants, à lire de gauche à droite et de bas en haut.

```
1 3 Allier ; 36 Indre ; 18 Cher ; 75 Paris
2 39 Jura ; 9 Ariège ; 81 Tarn ; 11 Aude
3 12 Aveyron ; 25 Doubs ; 73 Savoie ; 55 Meuse ;
4 15 Cantal ; 51 Marne ; 42 Loire ; 40 Landes
5 14 Calvados ; 30 Gard ; 84 Vaucluse ; 7 Ardeche
```

- Construire, en prenant comme clé le numéro de département, un index dense à deux niveaux sur le fichier contenant les enregistrements dans l'ordre indiqué ci-dessus, en supposant 2 enregistrements par bloc pour les données, et 8 par bloc pour l'index.
- Construire un index non-dense sur le fichier trié par numéro de département, avec les mêmes hypothèses.
- Construire un arbre-B sur les numéros de département, en supposant qu'il y a au plus 4 entrées par bloc dans l'index, et en insérant les enregistrements dans l'ordre donné ci-dessus.
- **Devoir a la maison:** Construire un arbre-B sur les noms de département, en supposant qu'il y a au plus 4 entrées par bloc dans l'index, et en insérant les enregistrements dans l'ordre donné ci-dessus.