Ingénierie des systèmes d'information.
Requêtes SQL

Manon Ansart

ESIREM, 2023

Pour la totalité de l'activité, on considère les tables suivantes :
- recette(id_recette, titre, type, sous_type)
- ingredient(id_ingredient, nom_ingredient)
- correspondance(id_recette, id_ingredient quantite)
- invite(id, nom, prenom)
- allergie(id_invite, id_ingredient, est_allergique) : est_allergique vaut 1 quand la personne est allergique à l'ingrédient, 0 sinon

1 Requêtes de base

1.1 Insert

La syntaxe pour les insertions est la suivante :

```
INSERT INTO table (colonne1, colonne2, colonne3)
VALUES value1, value2, value3
```

Si une valeur est insérée pour chaque colonne, dans le même ordre, il est possible de ne pas renseigner le nom des colonnes. C'est cependant mieux de les indiquer, pour plus de clareté.

1. Insérez l'ingrédient banane ayant pour id 259

1.2 Select

La syntaxe pour selectionner et afficher les attributs colonne1 et colonne2 de la table table pour les lignes respectant une condition est la suivante :

```
SELECT colonne1, colonne2
FROM table
WHERE condition
```

1. Affichez tous les titres des recettes.

2. Affichez les titres des recettes de type 'dessert'.

3. Affichez les types possibles dans la table recette. Chaque type ne doit apparaitre qu'une fois

4. Affichez le nombre de types possibles dans la table recette.

1.3 Update

La syntaxe pour modifier les attributs colonne1 et colonne2 de la table t1 pour les lignes respectant une condition est la suivante :

```
UPDATE table
SET colonne1='a', colonne2='b'
WHERE condition
```

1. Modifiez l'invite d'id 42 en lui donnant le prénom Douglas et le nom Adams

2. Modifiez l'invité Manon Ansard en Manon Ansart (s'il vous plait)

1.4 Delete

La syntaxe est la suivante :

```
DELETE FROM table
WHERE condition
```

1. Vous réalisez qu'Adam Douglas est décédé il y a 22 ans. Supprimez-le de la liste d'invités.

2. Voyez vous un problème avec cette opération ? Quelle qualité d'un SI cela concerne ?

2 Jointures

Une jointure permet de combiner plusieurs tables. Elle s'utilise quand on souhaite avoir accès à des colonnes situées dans différentes tables tout en faisant correspondre les lignes. Il faut donc indiquer comment effectuer cette correspondance entre les lignes. Cela peut se faire de 2 façons :

```
SELECT *
FROM table1 t1, table2 t2
WHERE t1.att = t2.autre

SELECT *
FROM table1 t1
JOIN table2 t2 ON t1.att = t2.autre
```

Attention : on utilise toujours t1.att et non juste att pour faire référence à un attribut de la table 1, même si l'attribut en question ne se situe que dans la table t1. La requête peut fonctionner si le préfixe n'est pas mis, mais c'est une très mauvaise pratique, qui sera sanctionnée en examen.

1. Question corrigée : Afficher toutes les colonnes de la table correspondance ainsi que les noms des ingredients

correspondants (2 solutions)

.....

.....

.....

.....

.....

.....

.....

.....

Solution:

```
SELECT *
FROM correspondance c, ingredient i
WHERE c.id_ingredient = i.id_ingredient

SELECT *
FROM correspondance c
JOIN ingredient i ON c.id_ingredient = i.id_ingredient
```

2. Afficher toutes les colonnes de la table correspondance ainsi que les informations sur les recettes correspondantes (2 solutions)

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

3. Afficher les id des ingrédients utilisés dans des recettes de type dessert (2 solutions)

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

4. (Bonus) Afficher les noms des ingrédients utilisés dans des recettes de type dessert (2 solutions)

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

5. (Bonus) Afficher les nom et prenom des invités ayant une ou plusieurs allergies (est_allergique à 1). (2 solutions)
   Note : le mot clé DISTINCT permet d'afficher chaque groupe de valeur une seule fois

. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .

3 Agrégats

Les agrégats permettent de regrouper ensemble les lignes ayant la même valeur pour un attribut (par exemple regrouper les recettes par type, ou les allergies par invité), et de calculer une valeur par groupe. On utilise le mot clé GROUP BY **après les clauses FROM et WHERE** pour identifier l'attribut à utiliser pour créer les groupes. L'attribut utilisé pour créer les groupes est toujours retourné en premier dans le SELECT. Les agrégats (valeur par groupe) peuvent par exemple être :

- une moyenne : AVG(attribut)
- une somme : SUM(attribut)
- un décompte du nombre de lignes ou de valeurs différentes : COUNT(attribut)

Les agrégats peuvent être calculés dans la clause SELECT :

```
SELECT groupe, AVG(note)
FROM table
GROUP BY groupe
```

Ils peuvent aussi être calculés dans l'équivalent d'une clause WHERE. Dans ce cas on utilise le mot clé HAVING :

```
SELECT groupe
FROM table
GROUP BY groupe
HAVING AVG(note) >= 10
```

Attention : les clauses GROUP BY et HAVING sont toujours après les clauses FROM et WHERE

1. Question corrigée : Afficher le nombre d'allergies de chaque invité

.....

.....

.....

.....

.....

.....

.....

.....

Solution:

```
SELECT id_invite, SUM(est_allergique)
FROM allergie
GROUP BY id_invite
```

2. Question corrigée : Afficher l'id des invités ayant au moins une allergie

.....

.....

.....

.....

.....

.....

.....

.....

Solution:

```
SELECT id_invite
FROM allergie
GROUP BY id_invite
HAVING SUM(est_allergique) > 0
```

3. Pour chaque ingrédient, affichier l'id de l'ingrédient et si il y a au moins une personne allergique ou non

4. Afficher uniquement les id des ingrédients auxquels personne n'est allergique

5. (Bonus) Afficher le nombre d'ingrédients utilisés dans chaque recette

6. (Bonus) Afficher la quantité totale par ingrédient

.....
.....
.....
.....
.....
.....
.....
.....

4 Combinaison d'agrégat et jointure

1. Afficher le nombre d'ingrédients utilisés dans chaque recette, avec le titre des recettes

.....
.....
.....
.....
.....
.....
.....
.....

2. Afficher la quantité totale par ingrédient, avec le nom des ingrédients

.....
.....
.....
.....
.....
.....
.....
.....

3. En ne prenant en compte que les recettes de type dessert, afficher la quantité totale par ingrédient, avec le nom des ingrédients

. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .

4. Afficher l'id des recettes ne contenant aucun ingrédient avec des allergies

. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .
. . . . . .