



Recherche de données 2

1



Structure d'une requête SQL

- Une requête SQL est composée à partir d'un opérateur de base qui produit une table

```
SELECT  
FROM  
[WHERE]  
[GROUP BY]  
[HAVING]  
[ORDER BY]
```

- Des opérateurs d'union, d'intersection ou de différence
- Des opérateurs de jointures
- Des expressions d'attributs et des fonctions de calcul

2



Sous-requêtes

Q20. Quels sont les noms des clients ayant choisi de faire du sport ?

```
SELECT  NomCl
FROM    Sejour
WHERE   Num IN ( SELECT Num
                  FROM    Location)
```

```
SELECT  NomCl
FROM    Sejour, Location
WHERE   Sejour.Num=Location.Num
```

3



Sous-requêtes

*Q20. Quels sont les noms des clients ayant choisi **de faire du sport** ?*

```
SELECT  NomCl
FROM    Sejour
WHERE   Num IN ( SELECT Num FROM LOCATION)
```

*Q21. Quels sont les noms des clients ayant choisi **de ne pas faire du sport** ?*

```
SELECT  NomCl
FROM    Sejour
WHERE   Num NOT IN ( SELECT Num FROM LOCATION)
```

4

Le prédicat est vrai ssi le résultat de la sous-requête est non vide

Sous-requêtes : exists

*Q21. Quels sont les noms des clients ayant choisi **de faire du sport** ?*

```
SELECT NomCl
FROM Sejour
WHERE Num IN ( SELECT Num FROM LOCATION)
```

```
SELECT NomCl
FROM Sejour S
WHERE EXISTS ( SELECT *
                FROM Location L
                WHERE S.Num = L.Num)
```

5

Sous-requêtes : exists

*Q21. Quels sont les noms des clients ayant choisi **de ne pas faire du sport** ?*

```
SELECT NomCl
FROM Sejour
WHERE Num NOT IN ( SELECT Num FROM LOCATION)
```

```
SELECT NomCl
FROM Sejour S
WHERE NOT EXISTS ( SELECT *
                   FROM Location L
                   WHERE S.Num = L.Num)
```

6

Groupement et agrégation

EMPLACEMENT			
<i>numE</i>	<i>Surface</i>	<i>nbrePersMax</i>	<i>TypeEmpl</i>
1	12	4	tente
2	12	4	tente
3	15	6	tente
4	20	8	tente
15	14	4	caravane
16	16	6	caravane
17	19	6	caravane
33	16	4	caravane
34	16	4	camping-car
58	30	4	camping-car
59	36	4	Bungalow
60	50	5	Bungalow
61	60	6	Bungalow

Q28. Combien y a-t-il d'emplacements par type ?

Q29. Quels sont les types d'emplacement où il y a plus de 3 places ?

7

Groupement et agrégation

EMPLACEMENT			
<i>numE</i>	<i>Surface</i>	<i>nbrePersMax</i>	<i>TypeEmpl</i>
1	12	4	tente
2	12	4	tente
3	15	6	tente
4	20	8	tente
15	14	4	caravane
16	16	6	caravane
17	19	6	caravane
33	16	4	caravane
34	16	4	camping-car
58	30	4	camping-car
59	36	4	Bungalow
60	50	5	Bungalow
61	60	6	Bungalow

<i>TypeEmpl</i>	<i>NbreEmplacement</i>
tente	4
caravane	4
camping-car	2
bungalow	3

Q28. Combien y a-t-il d'emplacements par type ?

8

Groupement et agrégation

EMPLACEMENT			
<i>numE</i>	<i>Surface</i>	<i>nbrePersMax</i>	<i>TypeEmpl</i>
1	12	4	tente
2	12	4	tente
3	15	6	tente
4	20	8	tente
15	14	4	caravane
16	16	6	caravane
17	19	6	caravane
33	16	4	caravane
34	16	4	camping-car
58	30	4	camping-car
59	36	4	Bungalow
60	50	5	Bungalow
61	60	6	Bungalow

<i>TypeEmpl</i>	<i>NbreEmplacement</i>
tente	4
caravane	4
camping-car	2
bungalow	3

Q29. Quels sont les types d'emplacement où il y a plus de 3 places ?

<i>TypeEmpl</i>
tente
caravane

9

Groupement et agrégation

```

SELECT      <liste_attributs>
FROM        <relations>
[WHERE      <conditions>]
[GROUP BY  <attribut_partitionnement>)]>
[HAVING    <condition_partitionnement>]
[ORDER BY  <attribut> <ordre> [, <attribut> <ordre>]]

```

10

Q28. Combien y a-t-il d'emplacements par type ?

Groupement et agrégation

EMPLACEMENT			
<i>numE</i>	<i>Surface</i>	<i>nbrePersMax</i>	<i>TypeEmpl</i>
1	12	4	tente
2	12	4	tente
3	15	6	tente
4	20	8	tente
15	14	4	caravane
16	16	6	caravane
17	19	6	caravane
33	16	4	caravane
34	16	4	camping-car
58	30	4	camping-car
59	36	4	Bungalow
60	50	5	Bungalow
61	60	6	Bungalow

```
SELECT      typeEmpl, count(*)
FROM        Emplacement
GROUP BY    typeEmpl
```

<i>TypeEmpl</i>	<i>NbreEmplacement</i>
tente	4
caravane	4
camping-car	2
bungalow	3

11

count(*) équivalent à count(numE)

Groupement et agrégation

Q28. Combien y a-t-il d'emplacements par type ?

```
SELECT      typeEmpl, count(*) AS NbreEmplacements
FROM        Emplacement
GROUP BY    typeEmpl
```

■

12

Q29. Quels sont les types d'emplacement où il y a plus de 3 places?

Groupement et agrégation

TypeEmpl	NbreEmplacement
tente	4
caravane	4
camping-car	2
bungalow	3

TypeEmpl
tente
caravane

```
SELECT      typeEmpl
FROM        Emplacement
GROUP BY    typeEmpl
HAVING      count(numE) > 3
```

13

Groupement et agrégation

Q29b. Quels sont les types d'emplacement où il y a plus de 3 places, précisez ce nombre de places?

```
SELECT      typeEmpl, count(*) AS NbreEmplacements
FROM        Emplacement
GROUP BY    typeEmpl
HAVING      count(*) > 3
```

14



Groupement et agrégation

Q33. Combien y a-t-il d'emplacements de type tente et caravane?

```
SELECT      typeEmpl, count(*) AS NbreEmplacements
FROM        Emplacement
WHERE       typeEmpl in ('tente','caravane')
GROUP BY    typeEmpl
```

15



Groupement et agrégation

Q31. Quel est le type d'emplacement où il y a le plus de places ?

```
SELECT      typeEmpl
FROM        Emplacement
GROUP BY    typeEmpl
HAVING      count(*) >= ALL
              ( SELECT count(*)
                FROM Emplacement
                GROUP BY typeEmpl
              )
```

16



Définition de vues

17



Vue

```
CREATE VIEW <relation>[( <attribut>+ )]  
AS  
    <requêteSQL>
```

18

Vue

- Il s'agit d'une relation virtuelle, qui s'exprime à l'aide d'une requête SQL
- L'idée est
 - De simplifier l'expression de certaines requêtes
 - De définir des sous-ensembles de la base de données auxquels on pourra associer des droits d'accès
- On peut ensuite l'interroger comme une relation normale

19

Vue

Q32. Créer une vue contenant l'information concernant les noms de clients des séjours de plus de 10 personnes, leur nombre et le type d'emplacement utilisé pour les placer

```
CREATE VIEW GroupeImportant (RepresentantGroupe,
                             nbGroupe,
                             typeEmplacement)
AS
SELECT    nomcl, nbrepers, typeE
FROM      Emplacement, Sejour
WHERE     Emplacement.numE=Sejour.numE
AND       nbrepers > 10
```

20



Mise à jour via une vue

- Pas de clause `DISTINCT`
- Pas de calcul dans la clause `SELECT`
- Pas de Jointure
- Pas de `GROUP BY` ni de `HAVING`

21



Confidentialité (1)

- Une BD doit être protégée d'accès malveillants
- Un utilisateur ne peut effectuer une opération sur un objet que s'il en a le droit (*privèlège*)
- L'identification des utilisateurs se fait en général par leur nom et par un mot de passe

22

Confidentialité (2)

- Dans un SGBD relationnel :
 - les objets à protéger sont les tables et les vues
 - les opérations sont SELECT, INSERT, UPDATE ou DELETE
- L'attribution des droits peut être :
 - centralisé : l'administrateur a tous les droits sur tous les objets de la BD et peut transmettre certains de ces droits à d'autres utilisateurs
 - décentralisée : l'utilisateur qui crée un objet a tous les droits sur cet objet et peut les transmettre en totalité ou en partie à d'autres utilisateurs

23

Grant et Revoke

```
GRANT SELECT, UPDATE (animaux)
ON    Sejour
TO    Tartanpion
```

```
GRANT SELECT
ON    Sport
TO    PUBLIC
```

```
REVOKE UPDATE (animaux)
ON    Sejour
TO    Tartanpion
```

24