**Sous requêtes**

Une sous-requête permet d’exécuter une requête à l’intérieur d’une autre requête.

**Exemple 1 :**

**Table : Questions**

|  |  |  |
| --- | --- | --- |
| **N°\_Question** | **Date\_Question** | **Intitulé\_Question** |
| 1 | 24/03/2013 | Comment réparer un ordinateur ? |
| 2 | 26/03/2013 | Comment changer un pneu ? |
| 3 | 18/04/2013 | Que faire si un appareil est cassé ? |
| 4 | 22/04/2013 | Comment faire nettoyer un clavier d'ordinateur ? |

**Table : Réponses**

|  |  |  |  |
| --- | --- | --- | --- |
| **N°\_Réponse** | **N°\_Question\_fk** | **Date\_Réponse** | **Intitulé\_Question** |
| 1 | 1 | 27/03/2013 | Le plus simple consiste à faire appel à un professionnel. |
| 2 | 1 | 28/03/2013 | Des conseils son disponible sur internet sur ce sujet. |
| 3 | 2 | 22/04/2013 | Que faire si un appareil est cassé ? |
| 4 | 3 | 22/04/2013 | Comment faire nettoyer un clavier d'ordinateur ? |
| 5 | 2 | 29/03/2013 | Ça dépend de vos préférences vis-à-vis de l'écologie. |
| 6 | 3 | 01/04/2013 | L’amener chez un réparateur |

**Cette requête sélectionne que les questions qui ont eu des réponses**

SELECT \*

FROM Question

WHERE N°\_Question = (

    SELECT N°\_Question\_fk

    FROM Réponse

) ;

**Cette requête sélectionne que la question liée à la dernière réponse.**

SELECT \*

FROM Question

WHERE N°\_Question = (

    SELECT N°\_Question\_fk

    FROM Réponse

ORDER bY Date\_Réponse DESC

LIMIT 1

) ;

**Cette requête permet de récupérer les questions qui ont eu des réponses entre 2 dates.**

**C’est pratique pour éviter d’obtenir des questions qui n’ont pas eu de réponses du tout ou pas de nouvelles réponses depuis longtemps.**

SELECT \*

FROM Question

WHERE N°\_Question IN (

    SELECT N°\_Question\_fk

    FROM Réponse

    WHERE Date\_Réponse BETWEEN '2013-01-01' AND '2013-12-31'

  )

**Exemple 2 :**

**Table Animal**

| **Id\_animal** | **sexe** | **date\_naissance** | **nom** | **Id\_espèce\_fk** |
| --- | --- | --- | --- | --- |
| 4 | F | 2009-08-03 05:12:00 | NULL | 3 |
| 6 | F | 2009-06-13 08:17:00 | Bobosse | 3 |
| 45 | F | 2007-04-01 18:17:00 | Nakki | 3 |
| 46 | F | 2009-03-24 08:23:00 | Tortilla | 3 |
| 47 | F | 2009-03-26 01:24:00 | Scroupy | 3 |
| 48 | F | 2006-03-15 14:56:00 | Lulla | 3 |
| 49 | F | 2008-03-15 12:02:00 | Dana | 3 |
| 50 | F | 2009-05-25 19:57:00 | Cheli | 3 |
| 51 | F | 2007-04-01 03:54:00 | Chicaca | 3 |
| 52 | F | 2006-03-15 14:26:00 | Redbul | 3 |
| 60 | F | 2009-03-26 07:55:00 | Parlotte | 4 |

**Table Espèce**

| **Id\_espèce** | **nom** |
| --- | --- |
| 3 | Tortue |
| 4 | Perroquet |

**Table Race**

| **id** | **nom** | **espece\_id** |
| --- | --- | --- |
| 1 | Berger allemand | 1 |
| 2 | Berger blanc suisse | 1 |
| 3 | Boxer | 1 |
| 4 | Bleu russe | 2 |
| 5 | Maine coon | 2 |
| 6 | Singapura | 2 |
| 7 | Sphynx | 2 |

**Sélection de toutes les femelles parmi les perroquets et les tortues**

SELECT Animal.id, Animal.sexe, Animal.date\_naissance, Animal.nom, Animal.espece\_id

FROM Animal

INNER JOIN Espece

ON Espece.id\_espèce = Animal.id\_espece\_fk

WHERE sexe = 'F'

AND Espece.nom\_courant IN ('Tortue d''Hermann', 'Perroquet amazone');

**Parmi ces femelles perroquets et tortues, on veut connaître la date de naissance de la plus âgée. On va donc faire une sélection dans la table des résultats de la requête.**

SELECT MIN(date\_naissance)

FROM (

SELECT Animal.id, Animal.sexe, Animal.date\_naissance, Animal.nom, Animal.espece\_id

FROM Animal

INNER JOIN Espece

ON Espece.id = Animal.espece\_id

WHERE sexe = 'F'

AND Espece.nom\_courant IN ('Tortue d''Hermann', 'Perroquet amazone')

) AS tortues\_perroquets\_F;

**Sélection des animaux de la race Berger Allemand**

SELECT id, sexe, nom, commentaires, espece\_id, race\_id

FROM Animal

WHERE race\_id =

(SELECT id FROM Race WHERE nom = 'Berger Allemand') ;

**Cette requête peut s’écrire avec des jointures internes**

SELECT Animal.id, sexe, Animal.nom, commentaires, Animal.espece\_id, race\_id

FROM Animal

INNER JOIN Race ON Race.id = Animal.race\_id

WHERE Race.nom = 'Berger Allemand';

**Les races qui ont le id\_espèce le plus petit**

SELECT id, nom, espece\_id

FROM Race

WHERE espece\_id = (

SELECT MIN(id) FROM Espece

);

| **id** | **nom** | **espece\_id** |
| --- | --- | --- |
| 1 | Berger allemand | 1 |
| 2 | Berger blanc suisse | 1 |
| 3 | Boxer | 1 |

**Les espèces Tortue d’hermann**

SELECT id, nom, espece\_id

FROM Race

WHERE espece\_id < (

SELECT id

FROM Espece

WHERE nom\_courant = 'Tortue d''Hermann');

Ici, la sous-requête renvoie 3, donc nous avons les races dont l'espèce a un id inférieur à 3.

| **id** | **nom** | **espece\_id** |
| --- | --- | --- |
| 1 | Berger allemand | 1 |
| 2 | Berger blanc suisse | 1 |
| 3 | Boxer | 1 |
| 4 | Bleu russe | 2 |
| 5 | Maine coon | 2 |
| 6 | Singapura | 2 |
| 7 | Sphynx | 2 |

**Sélection des animaux qui ont un id race=7 et id espèce=2**

SELECT id, sexe, nom, espece\_id, race\_id

FROM Animal

WHERE (id, race\_id) = (

SELECT id, espece\_id

FROM Race

WHERE id = 7);

| **id** | **sexe** | **nom** | **espece\_id** | **race\_id** |
| --- | --- | --- | --- | --- |
| 7 | F | Caroline | 1 | 2 |

En décomposant :

SELECT id, espece\_id

FROM Race

WHERE id = 7;

Donne comme résultat

| **id** | **espece\_id** |
| --- | --- |
| 7 | 2 |

Qui va être utilisé dans le WHERE