

3 MIDI'HOME CAMPING™ - NOT IN - NOT EXISTS - MINUS

R3A : le nom et le prénom des employés qui n'ont pas de chef.

```
SELECT nomEmploye, prenomEmploye
FROM Employes
WHERE idEmployeChef IS NULL;
```

R3B : le nom des bungalows qui n'ont jamais été loués.

```
SELECT nomBungalow
FROM Bungalows
WHERE idBungalow NOT IN (SELECT idBungalow
                          FROM Locations);
```

Ou

```
SELECT nomBungalow
FROM Bungalows
WHERE idBungalow IN (SELECT idBungalow
                      FROM Bungalows
                      MINUS
                      SELECT idBungalow
                      FROM Locations);
```

Ou

```
SELECT nomBungalow
FROM Bungalows b
WHERE NOT EXISTS (SELECT *
                  FROM Locations c
                  WHERE c.idBungalow = b.idBungalow);
```

R30 : le nom des campings qui n'ont pas d'employé.

```
SELECT nomCamping
FROM Campings
WHERE idCamping IN (SELECT idCamping
                    FROM Campings
                    MINUS
                    SELECT idCamping
                    FROM Employes);
```

Ou

```
SELECT nomCamping
FROM Campings c
WHERE NOT EXISTS (SELECT *
                  FROM Employes e
                  WHERE e.idCamping = c.idCamping);
```

Ou

```
SELECT nomCamping
FROM Campings
WHERE idCamping NOT IN (SELECT idCamping
                       FROM Employes
                       WHERE idCamping IS NOT NULL);
```

R31 : le nombre de bungalows qui ne proposent pas de service.

```
SELECT COUNT(*) AS "NB BUNGALOWS"
FROM Bungalows b
WHERE NOT EXISTS (SELECT *
                  FROM Proposer p
                  WHERE p.idBungalow = b.idBungalow)
```

R32 : le nom des clients qui ont réalisé des locations, mais jamais dans des bungalows de moins de 58 m².

```
SELECT nomClient
FROM Clients c1
WHERE NOT EXISTS (SELECT *
                  FROM Locations l2
                  JOIN Bungalows b2 ON l2.idBungalow = b2.idBungalow
                  WHERE superficieBungalow < 58
                  AND l2.idClient = c1.idClient)
AND EXISTS (SELECT *
            FROM Locations l3
            WHERE l3.idClient = c1.idClient);
```

Ou encore

```
SELECT nomClient
FROM Clients
WHERE idClient IN (SELECT idClient
                  FROM Locations
                  MINUS
                  SELECT idClient
                  FROM Locations l
                  JOIN Bungalows b ON l.idBungalow = b.idBungalow
                  WHERE superficieBungalow < 58);
```

R33 : le nom des campings où tous les employés ont un salaire supérieur ou égal à 1 000 €.

```
SELECT nomCamping
FROM Campings c
WHERE EXISTS (SELECT *
              FROM Employes e1
              WHERE c.idCamping = e1.idCamping)
AND NOT EXISTS (SELECT *
                FROM Employes e2
                WHERE c.idCamping = e2.idCamping
                AND salaireEmploye < 1000);
```

Ou

```
SELECT nomCamping
FROM Campings c
WHERE idCamping IN (SELECT idCamping
                    FROM Employes e1)
AND idCamping NOT IN (SELECT idCamping
                     FROM Employes e2
                     WHERE salaireEmploye < 1000
                     AND idCamping IS NOT NULL);
```

R34 : le nom des clients montpellierains qui n'ont jamais réalisé de location dans un bungalow qui ne propose pas de service.

```
SELECT nomClient
FROM Clients c1
WHERE villeClient = 'Montpellier'
AND NOT EXISTS (SELECT *
                FROM Locations l2
                WHERE l2.idClient = c1.idClient
                AND NOT EXISTS (SELECT *
                                FROM Proposer p3
                                WHERE p3.idBungalow = l2.idBungalow));
```

Ou

```
SELECT nomClient
FROM Clients
WHERE villeClient = 'Montpellier'
AND idClient NOT IN (SELECT idClient
                    FROM Locations l
                    WHERE idBungalow NOT IN (SELECT idBungalow
                                              FROM Proposer p));
```

Ou

```
SELECT nomClient
FROM Clients
WHERE idClient IN (SELECT idClient
                  FROM Clients
                  WHERE villeClient = 'Montpellier'
                  MINUS
                  SELECT idClient
                  FROM Locations
                  WHERE idBungalow IN (SELECT idBungalow
                                       FROM Bungalows
                                       MINUS
                                       SELECT idBungalow
                                       FROM Proposer));
```

R40 : l'identifiant, le nom et le prénom des clients qui ont réalisé une location (au moins) dans un camping qui se trouve dans la ville où ils habitent.

```
SELECT DISTINCT cl.idClient, nomClient, prenomClient
FROM Clients cl
JOIN Locations l ON cl.idClient = l.idClient
JOIN Bungalows b ON l.idBungalow = b.idBungalow
JOIN Campings ca ON b.idCamping = ca.idCamping
WHERE villeClient = villeCamping;
```

R41 : le nombre de bungalows du camping 'La Décharge Monochrome' qui ont été loués par le client Agathe Zeblouse.

```
SELECT COUNT(DISTINCT b.idBungalow) "nb bungalows"
FROM Clients cl
JOIN Locations l ON cl.idClient = l.idClient
JOIN Bungalows b ON l.idBungalow = b.idBungalow
JOIN Campings ca ON b.idCamping = ca.idCamping
WHERE nomCamping = 'La Décharge Monochrome'
AND nomClient = 'Zeblouse'
AND prenomClient = 'Agathe';
```

R42 : le nom et le prénom des clients qui n'ont jamais réalisé de location dans le camping 'Les Flots Bleus'.

```
SELECT nomClient, prenomClient
FROM Clients cl
WHERE NOT EXISTS (SELECT *
FROM Locations l
JOIN Bungalows b ON l.idBungalow = b.idBungalow
JOIN Campings c ON b.idCamping = c.idCamping
WHERE nomCamping = 'Les Flots Bleus'
AND c.idClient = l.idClient);
```

R43 : pour chacun des employés de la table Employés, le nom, le prénom de l'employé ainsi que le nom de son chef. Les employés doivent être classés par ordre lexicographique de leur nom.

```
SELECT sub.nomEmploye, sub.prenomEmploye, NVL(chef.nomEmploye, 'Pas de chef')
FROM Employes sub
LEFT OUTER JOIN Employes chef ON sub.idEmployeChef = chef.idEmploye
ORDER BY sub.nomEmploye;
```

R44 : le nom et le prénom des employés dont un des subordonnés (au moins) possède lui-même un ou plusieurs subordonnés.

```
SELECT nomEmploye, prenomEmploye
FROM Employes
WHERE idEmploye IN (SELECT grandChef.idEmploye
FROM Employes grandChef
JOIN Employes sousChef
ON sousChef.idEmployeChef = grandChef.idEmploye
JOIN Employes subordonne
ON subordonne.idEmployeChef = sousChef.idEmploye);
```

Ou

```
SELECT nomEmploye, prenomEmploye
FROM Employes
WHERE idEmploye IN (SELECT idEmployeChef
FROM Employes
WHERE idEmploye IN (SELECT idEmployeChef
FROM Employes));
```

R45 : les villes dans lesquelles résident des clients mais où il n'y a pas de camping.

```
SELECT villeClient
FROM Clients
MINUS
SELECT villeCamping
FROM Campings;
```

R46 : le nom et le prénom du dernier client qui a loué le bungalow 'La Poubelle'.

```
SELECT nomClient, prenomClient
FROM Clients c
WHERE idClient IN
(SELECT idClient
FROM Locations l
JOIN Bungalows b ON l.idBungalow = b.idBungalow
WHERE nomBungalow = 'La Poubelle'
AND dateFin = (SELECT MAX(dateFin)
FROM Locations l
JOIN Bungalows b ON l.idBungalow = b.idBungalow
WHERE nomBungalow = 'La Poubelle'));
```

R47 : le nom et le prénom des clients pour lesquels toutes les locations ont duré au moins 10 jours.

```
SELECT nomClient, prenomClient
FROM Clients
WHERE idClient IN (SELECT idClient
FROM Locations
MINUS
SELECT idClient
FROM Locations
WHERE (dateFin-dateDebut) < 10 );
```

R48 : le nom et le prénom des clients qui ont effectué une location (au moins) dans le camping 'Les Flots Bleus' ou une location (au moins) dans le camping le camping 'La Décharge Monochrome' mais pas dans les deux.

```
SELECT nomClient, prenomClient
FROM Clients
WHERE idClient IN ((SELECT idClient
FROM Locations l
JOIN Bungalows b ON l.idBungalow = b.idBungalow
JOIN Campings c ON b.idCamping = c.idCamping
WHERE nomCamping = 'Les Flots Bleus'
OR nomCamping = 'La Décharge Monochrome')
MINUS
(SELECT idClient
FROM Locations l
JOIN Bungalows b ON l.idBungalow = b.idBungalow
JOIN Campings c ON b.idCamping = c.idCamping
WHERE nomCamping = 'Les Flots Bleus'
INTERSECT
SELECT idClient
FROM Locations l
JOIN Bungalows b ON l.idBungalow = b.idBungalow
JOIN Campings c ON b.idCamping = c.idCamping
WHERE nomCamping = 'La Décharge Monochrome')) ;
```

R49 : pour chacun des bungalows du camping qui a l'identifiant 'CAMP1', le nom des services de Luxe qui ne sont pas proposés.

```
SELECT nomBungalow, nomService
FROM Bungalows
CROSS JOIN Services
WHERE idCamping = 'CAMP1'
AND categorieService = 'Luxe'
MINUS
SELECT nomBungalow, nomService
FROM Bungalows b
JOIN Proposer p ON p.idBungalow = b.idBungalow
JOIN Services s ON s.idService = p.idService
WHERE idCamping = 'CAMP1';
```

Ou

```
SELECT nomBungalow, nomService
FROM Bungalows b
CROSS JOIN Services s
WHERE idCamping = 'CAMP1'
AND categorieService = 'Luxe'
AND NOT EXISTS (SELECT *
FROM Proposer p
WHERE p.idBungalow = b.idBungalow
AND p.idService = s.idService);
```