

## ② MIDI'HOME CAMPING™ – Suite des Révisions du premier semestre

**R11 :** nom et prénom des employés qui ont un salaire supérieur à 2 000 €.

```
SELECT nomEmploye, prenomEmploye
FROM Employes
WHERE salaireEmploye > 2000;
```

**R12 :** le nom des campings de la ville de Palavas qui possèdent 5 étoiles.

```
SELECT nomCamping
FROM Campings
WHERE villeCamping = 'Palavas'
AND nbEtoilesCamping = 5;
```

**R13 :** le nom et le prénom des employés qui travaillent dans le camping 'Les Flots Bleus'.  
Les employés doivent être classés par rapport à leur salaire (du moins bien payé au mieux payé).

```
SELECT nomEmploye, prenomEmploye
FROM Employes e
[INNER] JOIN Campings c ON e.idCamping = c.idCamping
WHERE nomCamping = 'Les Flots Bleus'
ORDER BY salaireEmploye ASC;
```

**R14 :** le nom et le prénom des subordonnés de Gaspard Alizan (c'est à dire des employés qui ont pour chef Gaspard Alizan).

```
SELECT nomEmploye, prenomEmploye
FROM Employes
WHERE idEmployeChef IN (SELECT idEmploye
                        FROM Employes
                        WHERE nomEmploye = 'Alizan'
                        AND prenomEmploye = 'Gaspard');
```

Ou

```
SELECT sub.nomEmploye, sub.prenomEmploye
FROM Employes chef
JOIN Employes sub ON chef.idEmploye = sub.idEmployeChef
WHERE chef.nomEmploye = 'Alizan'
AND chef.prenomEmploye = 'Gaspard';
```

**R15 :** le nom et le prénom des clients qui étaient dans un camping 'Les Flots Bleus' le 14 juillet 2017 (c'est-à-dire qui ont une location qui couvre le 14 juillet 2017).

```
SELECT 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 nomCamping = 'Les Flots Bleus'
AND dateDebut <= '14/07/2017'
AND dateFin >= '14/07/2017';
```

**R16 :** le nom et le prénom des clients qui ont été dans le camping 'Les Flots Bleus' en juillet 2017.

```
SELECT nomClient, prenomClient
FROM Clients
WHERE idClient IN (SELECT idClient
                  FROM Locations l
                  JOIN Bungalows b ON l.idBungalow = b.idBungalow
                  JOIN Campings ca ON b.idCamping = ca.idCamping
                  WHERE nomCamping = 'Les Flots Bleus'
                  AND dateDebut <= '31/07/2017'
                  AND dateFin >= '01/07/2017');
```

**R17 :** le nom et le prénom des clients qui habitent dans une ville où il y a un camping.

```
SELECT nomClient, prenomClient
FROM Clients
WHERE villeClient IN (SELECT villeCamping
                     FROM Campings);
```

**R18 :** le nombre de services proposés par le bungalow 'Le Titanic'.

```
SELECT COUNT(*) AS "nb de services"
FROM Bungalows b
JOIN Proposer p ON b.idBungalow = p.idBungalow
WHERE nomBungalow = 'Le Titanic';
```

**R19 :** le salaire de l'employé le mieux payé du camping 'Les Flots Bleus'.

```
SELECT MAX(salaireEmploye) AS "salaire max"
FROM Employes e
JOIN Campings c ON e.idCamping = c.idCamping
WHERE nomCamping = 'Les Flots Bleus';
```

**R20 :** le nombre de campings dans lesquels la cliente Agathe Zeblouse a séjourné.

```
SELECT COUNT(DISTINCT idCamping) AS "nb campings"
FROM Clients c
JOIN Locations l ON c.idClient = l.idClient
JOIN Bungalows b ON l.idBungalow = b.idBungalow
WHERE nomClient = 'Zeblouse'
AND prenomClient = 'Agathe';
```

**R21 :** le nom du plus grand bungalow.

```
SELECT nomBungalow
FROM Bungalows
WHERE superficieBungalow = (SELECT MAX(superficieBungalow)
                           FROM Bungalows);
```

**R22 :** le nom et prénom de l'employé le plus mal payé du camping 'Les Flots Bleus'.

```
SELECT nomEmploye, prenomEmploye
FROM Employes e
JOIN Campings c ON e.idCamping = c.idCamping
WHERE nomCamping = 'Les Flots Bleus'
AND salaireEmploye = (SELECT MIN(salaireEmploye)
                     FROM Employes e
                     JOIN Campings c ON e.idCamping = c.idCamping
                     WHERE nomCamping = 'Les Flots Bleus');
```

**R23 :** le nom des bungalows qui ne proposent pas de service.

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

Ou

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

**R24 :** le nom et prénom des employés qui n'ont pas de subordonné.

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

Ou

```
SELECT nomEmploye, prenomEmploye
FROM Employes
WHERE idEmploye NOT IN (SELECT idEmployeChef
                      FROM Employes
                      WHERE idEmployeChef IS NOT NULL);
```

**R25 :** le nom des bungalows qui ne proposent ni le service ‘Climatisation’ ni le service ‘TV’.

```
SELECT nomBungalow
FROM Bungalows
WHERE idBungalow IN (SELECT idBungalow
FROM Bungalows
MINUS
SELECT idBungalow
FROM Proposer p
JOIN Services s ON p.idService = s.idService
WHERE nomService = 'Climatisation'
MINUS
SELECT idBungalow
FROM Proposer p
JOIN Services s ON p.idService = s.idService
WHERE nomService = 'TV');

Ou

SELECT nomBungalow
FROM Bungalows
WHERE idBungalow NOT IN (SELECT idBungalow
FROM Proposer p
JOIN Services s ON p.idService = s.idService
WHERE nomService = 'Climatisation'
UNION
SELECT idBungalow
FROM Proposer p
JOIN Services s ON p.idService = s.idService
WHERE nomService = 'TV');

Ou

SELECT nomBungalow
FROM Bungalows
WHERE idBungalow NOT IN (SELECT idBungalow
FROM Proposer p
JOIN Services s ON p.idService = s.idService
WHERE nomService = 'Climatisation'
OR nomService = 'TV');

Ou

SELECT nomBungalow
FROM Bungalows
WHERE idBungalow NOT IN (SELECT idBungalow
FROM Proposer p
JOIN Services s ON p.idService = s.idService
WHERE nomService = 'Climatisation')
AND idBungalow NOT IN (SELECT idBungalow
FROM Proposer p
JOIN Services s ON p.idService = s.idService
WHERE nomService = 'TV');
```

**R26 :** le nom des services qui sont proposés dans le camping ‘La Décharge Monochrome’ et qui sont également proposés dans le camping ‘The White Majestic’.

```
SELECT nomService
FROM Services
WHERE idService IN (SELECT idService
FROM Campings c
JOIN Bungalows b ON c.idCamping = b.idCamping
JOIN Proposer p ON b.idBungalow = p.idBungalow
WHERE nomCamping = 'La Décharge Monochrome'
INTERSECT
SELECT idService
FROM Campings c
JOIN Bungalows b ON c.idCamping = b.idCamping
JOIN Proposer p ON b.idBungalow = p.idBungalow
WHERE nomCamping = 'The White Majestic');
```

**R27 :** le nom et le prénom des clients qui ont effectué une location (au moins) dans le camping ‘The White Majestic’ ou une location (au moins) dans le camping ‘La Décharge Monochrome’. Les clients doivent être classés dans l’ordre lexicographique de leur nom (et de leur prénom s’ils ont le même nom)

```
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 = 'The White Majestic'
OR nomCamping = 'La Décharge Monochrome')
ORDER BY nomClient, prenomClient ;

Ou

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 = 'The White Majestic'
UNION
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')
ORDER BY nomClient, prenomClient ;
```

**R28 :** pour chacun des employés de la table Employés, le nom, le prénom de l’employé ainsi que le nom du camping dans lequel il travaille. Les employés doivent être classés par ordre lexicographique de leur nom.

```
SELECT nomEmploye, prenomEmploye, nomCamping
FROM Employes e
JOIN Campings c ON e.idCamping = c.idCamping
UNION
SELECT nomEmploye, prenomEmploye, 'Pas affecté à un camping'
FROM Employes e
WHERE idCamping IS NULL
ORDER BY nomEmploye;

Ou

SELECT nomEmploye, prenomEmploye, NVL(nomCamping, 'Pas affecté à un camping')
FROM Employes e
LEFT OUTER JOIN Campings c ON e.idCamping = c.idCamping
ORDER BY nomEmploye;
```

**R29 :** le nom et le prénom des clients qui ont pu croiser Judas Bricot dans un camping (c'est-à-dire que pendant au moins une journée, ils ont eu une location dans le même camping que Judas Bricot).

```
SELECT c1.nomClient, c1.prenomClient
FROM Clients c1
JOIN Locations l1 ON c1.idClient = l1.idClient
JOIN Bungalows b1 ON l1.idBungalow = b1.idBungalow
JOIN Bungalows b2_JB ON b1.idCamping = b2_JB.idCamping
JOIN Locations l2_JB ON b2_JB.idBungalow = l2_JB.idBungalow
JOIN Clients c2_JB ON l2_JB.idClient = c2_JB.idClient
WHERE c2_JB.nomClient = 'Bricot'
AND c2_JB.prenomClient = 'Judas'
AND l1.dateFin >= l2_JB.dateDebut
AND l1.dateDebut <= l2_JB.dateFin
AND c1.idClient != c2_JB.idClient;

Ou

SELECT nomClient, prenomClient
FROM Clients c1
JOIN Locations l1 ON c1.idClient = l1.idClient
JOIN Bungalows b1 ON l1.idBungalow = b1.idBungalow
WHERE EXISTS (SELECT *
FROM Clients c2
JOIN Locations l2 ON c2.idClient = l2.idClient
JOIN Bungalows b2 ON l2.idBungalow = b2.idBungalow
WHERE nomClient = 'Bricot'
AND prenomClient = 'Judas'
AND b1.idCamping = b2.idCamping
AND l1.dateFin >= l2.dateDebut
AND l1.dateDebut <= l2.dateFin
AND c1.idClient != c2.idClient);
```