

## CORRIGÉ - EXERCICES BDD

## Exercice 1

1. Nom des clients occupant un emplacement à l'ombre.

Optimisation :  $\pi_{nom}(\sigma_{ombre=1}(\text{emplacement}) \bowtie_{nE} \text{occupation} \bowtie_{nC} \text{client})$

2. Numéro des clients ayant loué un emplacement à côté duquel il y a une activité coûtant plus cher que tous les emplacements de la zone.

3. Zones dans lesquelles le prix moyen des activités est inférieur au prix moyen des emplacements.

4.  $\pi_{nE} \cdot \sigma_{type='vide'}(\text{emplacement})$ .

```
SELECT nE FROM emplacement WHERE type = 'vide';
```

5.  $\pi_{nE} \cdot \sigma_{ombre=1 \wedge raccordement=1}(\text{emplacement})$ .

```
SELECT nE FROM emplacement WHERE ombre = 1 AND raccordement = 1;
```

6.  $\pi_{nom, prenom} \cdot \sigma_{email=NULL}(\text{client})$ .

```
SELECT nom, prenom FROM client WHERE email IS NULL;
```

7.  $\pi_{nom, prenom}(\text{client} \bowtie_{nC} \text{occupation} \bowtie_{nE} \sigma_{ombre=1}(\text{emplacement}))$ .

```
SELECT nom, prenom FROM client C
JOIN occupation O ON C.nC = O.nC
JOIN emplacement E ON E.nE = O.nE
WHERE ombre = 1;
```

8.  $\pi_{nE}(\text{emplacement} \bowtie_{zoneE=zoneA} \sigma_{nomA='Piscine'}(\text{activite}))$ .

```
SELECT nE FROM activite
JOIN emplacement ON zoneA = zoneE
WHERE nomA = 'Piscine';
```

9.  $zone - \pi_{nZ, description}(zone \bowtie_{nZ=zoneE} \sigma_{ombre=0}(\text{emplacement}))$ .

```
SELECT * FROM zone
WHERE NOT EXISTS (SELECT * FROM emplacement WHERE ombre = 0 and zoneE = nZ);
```

```
SELECT * FROM zone
EXCEPT
SELECT nZ, description FROM zone
JOIN emplacement on nZ = zoneE
WHERE ombre = 0;
```

```
SELECT * FROM zone
WHERE 0 = (SELECT COUNT(*) FROM emplacement WHERE ombre = 0 and zoneE = nZ);
```

```
SELECT * FROM zone
WHERE nZ NOT IN (SELECT zoneE FROM emplacement WHERE ombre = 0);
```

10.  $\pi_{nC, zoneE}(\text{emplacement} \bowtie_{nE} \text{occupation} \bowtie_{nC} \text{client}) \div \rho_{nZ \rightarrow zoneE} \cdot \pi_{nZ}(\text{zone})$ .

```
SELECT nC FROM client C
WHERE NOT EXISTS (SELECT * FROM zone
WHERE nZ NOT IN (SELECT zoneE FROM occupation O
JOIN emplacement E ON O.nE = E.nE
WHERE C.nC = O.nC)
);
```

```
SELECT nC FROM client C
WHERE NOT EXISTS (SELECT * FROM zone
WHERE NOT EXISTS (SELECT * FROM occupation O
JOIN emplacement E ON O.nE = E.nE
WHERE C.nC = O.nC AND nZ = zoneE)
);
```

```
SELECT nC FROM client C
WHERE (SELECT COUNT(nZ) FROM zone)
= (SELECT COUNT(DISTINCT zoneE)
FROM occupation O
JOIN emplacement E ON O.nE = E.nE
WHERE nC = C.nC);
```

```
SELECT nC FROM occupation
JOIN emplacement E ON E.nE = O.nE
GROUP BY nC
HAVING COUNT(DISTINCT zoneE) = (SELECT COUNT(nZ) FROM zone);
```

- 11.

```
SELECT nE FROM emplacement E
WHERE NOT EXISTS (SELECT * FROM activite A WHERE zoneA = zoneE);
```

```
SELECT nE FROM emplacement WHERE zoneE NOT IN (SELECT zoneA FROM activite);
```

- 12.

```
SELECT nZ, description FROM zone
JOIN emplacement ON nZ = zoneE
WHERE prix = (SELECT MIN(prix) FROM emplacement);
```

- 13.

```
SELECT nC, SUM(prix) AS montant, COUNT(*) AS nombre_de_sejours
FROM client C
JOIN occupation O ON C.nC = O.nC
JOIN emplacement E ON E.nE = O.nE
GROUP BY nC
ORDER BY montant;
```

## Exercice 2

1.  $\pi_{e, prenom} \left( \text{personne } E \bowtie_{\substack{P.numP \\ = E.numPere}} \sigma_{\substack{nom='Bernoulli' \\ \wedge prenom='Jean' \\ \wedge anneeNaissance=1710}} (\text{personne } P) \right)$

```
SELECT E.prenom
FROM personne E
JOIN personne P ON P.numP = E.numPere
WHERE P.nom = 'Bernoulli' AND P.prenom = 'Jean' AND P.anneeNaissance = 1710;
```

2.  $\pi_{F, prenom} \left( \text{personne } F \bowtie_{\substack{F.numP \\ = numEpouse}} \text{union} \bowtie_{\substack{M.numP \\ = numEpoux}} \sigma_{\substack{nom='Einstein' \\ \wedge prenom='Albert'}} (\text{personne } M) \right)$

```
SELECT F.prenom
FROM personne F
JOIN union ON numEpouse = F.numP
JOIN personne M ON numEpoux = M.numP
WHERE M.nom = 'Einstein' AND M.prenom = 'Albert';
```

- 3.

```

SELECT F.anneeNaissance
FROM personne F
JOIN union ON numEpouse = F.numP
JOIN personne M ON numEpoux = M.numP
WHERE M.nom = 'Einstein' AND M.prenom = 'Albert'
ORDER BY annee
LIMIT 1;

```

```

SELECT F.anneeNaissance
FROM personne F
JOIN union ON numEpouse = F.numP
JOIN personne M ON numEpoux = M.numP
WHERE M.nom = 'Einstein' AND M.prenom = 'Albert'
AND annee = (SELECT MIN(annee) FROM union
             JOIN personne ON numEpoux = numP
             WHERE nom = 'Einstein' AND prenom = 'Albert');

```

4.

```

SELECT COUNT(*)
FROM personne E
JOIN personne P ON E.numPere = P.numP
WHERE P.nom = 'Gauss' AND P.prenom = 'Carl';

```

5.  $\pi_{G.nom, G.prenom}$  ()

```

SELECT G.nom, G.prenom
FROM personne G JOIN personne P ON G.numP IN (P.numPere, P.numMere)
             JOIN personne E ON P.numP = E.numPere
WHERE E.nom = 'Turing' AND E.prenom = 'Alan';

```

```

SELECT G.nom, G.prenom
FROM personne G JOIN personne P ON G.numP = P.numPere
             JOIN personne E ON P.numP = E.numPere
WHERE E.nom = 'Turing' AND E.prenom = 'Alan'

```

UNION

```

SELECT G.nom, G.prenom
FROM personne G JOIN personne P ON G.numP = P.numMere
             JOIN personne E ON P.numP = E.numPere
WHERE E.nom = 'Turing' AND E.prenom = 'Alan';

```

6.

```

SELECT O.anneeNaissance
FROM personne O JOIN personne M ON O.numPere = M.numPere
             JOIN personne JM ON JM.numMere = M.numP
WHERE JM.prenom = 'John' AND JM.nom = 'Maxwell';

```