

## Corrigé du R1.05-Intro BD et SQL (Séance n° 7) Interrogations en sqlplus

Première étape : Expression des projections et sélections
---

Q1 : Quelles sont les villes d'arrivée des voyages à destination du Maroc ?

```
SELECT DISTINCT VILLEARR FROM VOYAGE WHERE PAYSARR = 'MAROC' ;
```

Q2 : Donnez toutes les informations sur les options dont le libellé comporte le mot Visite.

```
SELECT * FROM OPTIONV WHERE LIBELLE LIKE '%VISITE%' ;
```

Q3 : Donnez la liste, triée en ordre croissant, des dates de départ possibles pour le voyage d'identifiant 927 entre le 1<sup>er</sup> juin et le 30 juillet 2004.

```
SELECT DATEDEP FROM PLANNING  
WHERE IDV = 927 AND DATEDEP BETWEEN '01/06/04' AND '30/07/04'  
ORDER BY DATEDEP ;
```

```
SELECT DATEDEP FROM PLANNING  
WHERE IDV = 927 AND DATEDEP >= '01/06/04' AND DATEDEP <= '30/07/04'  
ORDER BY DATEDEP ;
```

Q4 : Donnez la liste des clients (leur identifiant, nom et prénom) n'habitant ni Paris ni Marseille. Présentez cette liste triée selon l'identifiant.

```
SELECT NUMCL, NOM, PRENOM FROM CLIENT  
WHERE VILLE NOT IN ('PARIS', 'MARSEILLE') ;  
ORDER BY NUMCL
```

```
SELECT NUMCL, NOM, PRENOM FROM CLIENT  
WHERE VILLE <> 'PARIS' AND VILLE <> 'MARSEILLE'  
ORDER BY NUMCL ;
```

Q5 : Quels sont les clients (identifiant et nom) dont l'adresse n'est pas valorisée dans la base ?

```
SELECT NUMCL, NOM FROM CLIENT WHERE ADRESSE IS NULL ;
```

## Deuxième étape : Expression des jointures

Q6 : Donnez, pour le client Hubert Marin, la liste des voyages (identifiant) pour lequel il a fait une réservation.

```
SELECT IDV FROM RESERVATION, CLIENT
WHERE NOM = 'MARIN' AND PRENOM = 'HUBERT' AND
RESERVATION.NUMCL = CLIENT.NUMCL ;
```

```
SELECT IDV FROM RESERVATION
WHERE NUMCL IN
  (SELECT NUMCL FROM CLIENT
   WHERE NOM = 'MARIN' AND PRENOM = 'HUBERT') ;
```

```
SELECT IDV
FROM RESERVATION
JOIN CLIENT
ON   RESERVATION.NUMCL = CLIENT.NUMCL
WHERE NOM = 'MARIN' AND PRENOM = 'HUBERT' ;
```

Q7 : Donnez la liste des réservations (numéro client, date de départ et ville d'arrivée du voyage) des clients habitant Marseille.

```
SELECT NUMCL, DATEDEP, VILLEARR
FROM VOYAGE, RESERVATION
WHERE VOYAGE.IDV = RESERVATION.IDV
AND NUMCL IN
  (SELECT NUMCL FROM CLIENT
   WHERE VILLE = 'MARSEILLE') ;
```

-- Comme DATEDEP et VILLEARR sont  
-- dans des relations différentes,  
-- il faut que ces relations  
-- figurent dans la clause FROM.

```
SELECT CLIENT.NUMCL, DATEDEP, VILLEARR
FROM VOYAGE, RESERVATION, CLIENT
WHERE VOYAGE.IDV = RESERVATION.IDV
AND RESERVATION.NUMCL = CLIENT.NUMCL
AND VILLE = 'MARSEILLE' ;
```

```
SELECT CLIENT.NUMCL, DATEDEP, VILLEARR
FROM VOYAGE
JOIN RESERVATION
ON VOYAGE.IDV = RESERVATION.IDV
JOIN CLIENT
ON RESERVATION.NUMCL = CLIENT.NUMCL
WHERE VILLE = 'MARSEILLE' ;
```

Q8 : Quelles sont les options (leur libellé) proposées pour le voyage d'identifiant 862 ?

```
SELECT LIBELLE FROM OPTIONV, CARAC
WHERE OPTIONV.CODE = CARAC.CODE AND IDV= 862 ;

SELECT LIBELLE FROM OPTIONV WHERE CODE IN
  (SELECT CODE FROM CARAC WHERE IDV = 862) ;
```

```

SELECT LIBELLE
FROM OPTIONV
JOIN CARAC
ON OPTIONV.CODE = CARAC.CODE
WHERE IDV = 862) ;

```

Q9 : Quelles sont les options (leur libellé) proposées pour les voyages réservés par le client Thomas Jarolim ?

```

SELECT DISTINCT LIBELLE
FROM OPTIONV, CARAC, RESERVATION, CLIENT
WHERE OPTIONV.CODE = CARAC.CODE AND CARAC.IDV = RESERVATION.IDV AND
RESERVATION.NUMCL = CLIENT.NUMCL AND NOM = 'JAROLIM'
AND PRENOM = 'THOMAS' ;

```

```

SELECT DISTINCT LIBELLE
FROM OPTIONV, CARAC
WHERE OPTIONV.CODE = CARAC.CODE AND IDV IN
(SELECT IDV FROM RESERVATION, CLIENT
WHERE RESERVATION.NUMCL = CLIENT.NUMCL AND NOM = 'JAROLIM' AND
PRENOM = 'THOMAS') ;

```

```

SELECT DISTINCT LIBELLE
FROM OPTIONV
JOIN CARAC
ON OPTIONV.CODE = CARAC.CODE
JOIN RESERVATION
ON CARAC.IDV = RESERVATION.IDV
JOIN CLIENT
ON RESERVATION.NUMCL = CLIENT.NUMCL
WHERE NOM = 'JAROLIM' AND PRENOM = 'THOMAS' ;

```

/\* une option pouvant être proposée par plusieurs voyages, le mot-clef **DISTINCT** est nécessaire, dans les versions précédentes, pour éliminer les duplicats. Ce n'est pas le cas, pour la requête suivante car CODE est clef primaire de la relation spécifiée dans le 1<sup>er</sup> bloc \*/

```

SELECT LIBELLE FROM OPTIONV WHERE CODE IN
(SELECT CODE FROM CARAC WHERE IDV IN
(SELECT IDV FROM RESERVATION WHERE NUMCL IN
(SELECT NUMCL FROM CLIENT WHERE NOM = 'JAROLIM'
AND PRENOM = 'THOMAS')))) ;

```

Q10: Quels sont les clients (nom, prénom, ville) ayant réservé un voyage partant de leur ville de résidence ?

```

SELECT DISTINCT NOM, PRENOM, VILLE
FROM CLIENT, RESERVATION, VOYAGE
WHERE CLIENT.NUMCL = RESERVATION.NUMCL AND
RESERVATION.IDV = VOYAGE.IDV AND VILLE = VILLEDEP ;

```

/\* Dans la requête suivante, la jointure imbriquée doit se faire **simultanément** sur les attributs VILLE de la relation CLIENT et IDV de la relation RESERVATION. Il faut donc que ces deux relations soient spécifiées dans le 1<sup>er</sup> bloc. \*/

```

SELECT DISTINCT NOM, PRENOM, VILLE
FROM CLIENT, RESERVATION
WHERE CLIENT.NUMCL = RESERVATION.NUMCL

```

```

AND (VILLE, IDV) IN (SELECT VILLEDEP, IDV FROM VOYAGE) ;
SELECT DISTINCT NOM, PRENOM, VILLE
FROM CLIENT
JOIN RESERVATION
ON CLIENT.NUMCL = RESERVATION.NUMCL
JOIN VOYAGE
ON RESERVATION.IDV = VOYAGE.IDV AND VILLE = VILLEDEP ;

```

<b>Troisième étape : Formulation de calculs verticaux et horizontaux</b>
--

Q11: Combien y a-t-il de réservations actuellement saisies dans la base ?

```

SELECT COUNT(*) FROM RESERVATION ;

```

Q12: Quelle est la durée moyenne des séjours au Kenya ?

```

SELECT AVG(DUREE) FROM VOYAGE WHERE PAYSARR = 'KENYA' ;

```

Q13: Combien y a-t'il de places réservées pour le voyage 122 ?

```

SELECT SUM(NBPERS) FROM RESERVATION WHERE IDV = 122 ;

```

Q14: Donnez, pour les voyages 122 les dates de départ et de retour possibles.

```

SELECT DATEDEP, DATEDEP + NVL(DUREE, 0) DATERETOUR
FROM PLANNING, VOYAGE
WHERE PLANNING.IDV = VOYAGE.IDV AND
VOYAGE.IDV = 122 ;

```

```

/* L'attribut DATEDEP faisant partie de la clef primaire de la relation
PLANNING, les valeurs nulles ne sont pas acceptées. En revanche, il
convient de gérer les valeurs manquantes pour l'attribut DUREE. L'alias
DATERETOUR est simplement introduit pour l'affichage des résultats. */

```

Q15: Quel est le tarif le plus faible ?

```

SELECT MIN(TARIF) FROM PLANNING ;

```

Q16: Quels sont les voyages (identifiant et destination) dont le tarif est le plus faible ?

```

SELECT DISTINCT VOYAGE.IDV, VILLEARR, PAYSARR
FROM VOYAGE, PLANNING
WHERE VOYAGE.IDV = PLANNING.IDV AND TARIF =
      (SELECT MIN(TARIF) FROM PLANNING) ;

SELECT DISTINCT VOYAGE.IDV, VILLEARR, PAYSARR
FROM VOYAGE, PLANNING
WHERE VOYAGE.IDV = PLANNING.IDV AND TARIF <= ALL
      (SELECT TARIF FROM PLANNING) ;

```

Q17: Donnez le prix total des réservations du client Arnaud Peyroche (en tenant compte du nombre de personnes).

```
SELECT SUM(TARIF * NBPERS)
FROM CLIENT, RESERVATION, PLANNING
WHERE CLIENT.NUMCL = RESERVATION.NUMCL AND
RESERVATION.IDV = PLANNING.IDV AND
RESERVATION.DATEDEP = PLANNING.DATEDEP AND NOM = 'PEYROCHE'
AND PRENOM = 'ARNAUD' ;
```

/\* si l'un des deux attributs a une valeur nulle, le calcul horizontal rend NULL et la fonction agrégative SUM n'en tient pas compte. \*/

```
SELECT SUM(TARIF * NBPERS)
FROM RESERVATION, PLANNING
WHERE RESERVATION.IDV = PLANNING.IDV AND
RESERVATION.DATEDEP = PLANNING.DATEDEP AND NUMCL IN
  (SELECT NUMCL
   FROM CLIENT
   WHERE NOM = 'PEYROCHE' AND PRENOM = 'ARNAUD') ;
```

Q18: Pour quelle destination part le voyage le plus cher ?

```
SELECT DISTINCT VILLEARR, PAYSARR FROM VOYAGE, PLANNING
WHERE VOYAGE.IDV = PLANNING.IDV AND TARIF =
(SELECT MAX(TARIF) FROM PLANNING) ;
```

/\* Il faut procéder à l'élimination des duplicats (par **DISTINCT**) car plusieurs planifications du même voyage ou de différents voyages peuvent être au tarif maximal. \*/

```
SELECT DISTINCT VILLEARR, PAYSARR FROM VOYAGE, PLANNING
WHERE VOYAGE.IDV = PLANNING.IDV AND TARIF >= ALL
(SELECT TARIF FROM PLANNING) ;
```

Q19: Pour le voyage réservé par le client Nicolas Potier, donnez le prix total des options proposées ?

```
SELECT SUM(PRIX)
FROM CLIENT, RESERVATION, CARAC
WHERE CLIENT.NUMCL = RESERVATION.NUMCL AND RESERVATION.IDV = CARAC.IDV
AND NOM = 'POTIER' AND PRENOM = 'NICOLAS' ;
```

#### Quatrième étape : Test d'absence de données

*Proposer deux formulations différentes des requêtes suivantes, en ne faisant pas appel aux opérateurs ensemblistes.*

Q20: Existe-t-il des clients, si oui, donnez leur nom et prénom, qui n'ont aucune réservation ?

```
SELECT NOM, PRENOM FROM CLIENT WHERE NUMCL NOT IN  
(SELECT NUMCL FROM RESERVATION) ;
```

```
SELECT NOM, PRENOM FROM CLIENT WHERE NUMCL <> ALL  
(SELECT NUMCL FROM RESERVATION) ;
```

Q21: Donner les voyages (identifiant et destination) pour lesquels il n'y a aucune réservation.

```
SELECT IDV, VILLEARR, PAYSARR FROM VOYAGE WHERE IDV NOT IN  
(SELECT IDV FROM RESERVATION) ;
```

```
SELECT IDV, VILLEARR, PAYSARR FROM VOYAGE WHERE IDV <> ALL  
(SELECT IDV FROM RESERVATION) ;
```

Q22: Quelles sont les destinations (ville, pays) qui ne sont pas proposées au départ de Marseille ?

```
SELECT DISTINCT VILLEARR, PAYSARR FROM VOYAGE  
WHERE (VILLEARR, PAYSARR) NOT IN  
(SELECT VILLEARR, PAYSARR FROM VOYAGE WHERE VILLEDEP = 'MARSEILLE') ;
```

```
SELECT DISTINCT VILLEARR, PAYSARR FROM VOYAGE  
WHERE (VILLEARR, PAYSARR) <> ALL  
(SELECT VILLEARR, PAYSARR FROM VOYAGE WHERE VILLEDEP = 'MARSEILLE') ;
```

Q23: Quelles sont les options (code et libellé) qui ne sont pas proposées dans les voyages pour Chypre ?

```
SELECT CODE, LIBELLE FROM OPTIONV  
WHERE CODE NOT IN  
(SELECT CODE FROM CARAC, VOYAGE  
WHERE CARAC.IDV = VOYAGE.IDV AND PAYSARR = 'CHYPRE') ;
```

```
SELECT CODE, LIBELLE FROM OPTIONV  
WHERE CODE <> ALL  
(SELECT DISTINCT CODE FROM CARAC, VOYAGE  
WHERE CARAC.IDV = VOYAGE.IDV  
AND PAYSARR = 'CHYPRE') ;
```