

TP 11 base de données SQLite : solutions (du début)

- (R0) Liste de noms de tous les hotels, avec leur numéros de station.

Vérif. : On doit obtenir 78 réponses. Le nombre de réponses est marqué en bas de la fenêtre SQLite Manager.

Solution 0 `SELECT nomh, ns FROM hotels`

- (R1) En faisant suivre la requête précédente de la commande `ORDER BY nomh`, on s'aperçoit que des hotels situés dans des stations différentes portent le même nom : essayez ! On se demande alors combien de *noms* distincts d'hotels, il y a dans la base. Pour cela, une commande est utile : `SELECT DISTINCT`.

Vérif. On ne trouve plus que 22 réponses, ce qui montre que l'imagination des hôteliers est ici limitée !

Solution 1 `SELECT DISTINCT nomh FROM hotels -- Facultatif, on peut rajouter ORDER BY nomh`

- (R2) Liste des numéros et noms des stations de montagne avec leur capacité en chambres.

Vérif. : On doit obtenir six réponses.

Solution 2 `SELECT ns, noms, capch FROM resorts WHERE types='montagne'.`

- (R3) Liste des noms des hôtels trois étoiles et quatre étoiles, avec leur numéros de station.

Vérif. : On doit obtenir 31 réponses.

Solution 3 `SELECT nomh,ns FROM hotels WHERE cath=3 OR cath=4`

- (R4) Liste des noms des hôtels avec leur catégorie et le nom de la station où ils se trouvent, se trouvant dans une station balnéaire.

Indication : Attention, il y a deux tables à croiser.... car le caractère balnéaire d'une station ne se voit pas dans la table hôtel.

Vérif. : On doit obtenir 37 réponses.

Solution 4

```
SELECT hotels.nomh, hotels.cath, resorts.noms  
FROM hotels, resorts WHERE hotels.ns=resorts.ns  
AND resorts.types="mer"
```

Variante avec la commande JOIN :

```
SELECT h.nomh,h.adrh,h.telh,h.cath  
FROM resorts r JOIN hotels h ON r.ns=h.ns  
WHERE r.types='mer'
```

- (R5) Liste des noms des stations au bord de la mer ayant au moins un hôtel 4 étoiles.

Vérif. : Vous obtiendrez trois réponses. Si vous ne faites pas de `SELECT DISTINCT`, vous obtiendrez quatre réponse : qu'est ce que cela signifie ?

Solution 5 `SELECT DISTINCT r.noms`

```
FROM resorts r, hotels h  
WHERE r.ns=h.ns AND r.types='mer' AND cath=4
```

Le fait d'obtenir deux fois le nom de la station **Dunkerque** (magnifique station ensoleillée) signifie qu'il y a deux hotels 4 étoiles à Dunkerque. En projetant sur l'attribut `r.noms` on a rendu identique deux lignes différentes du tableau.

- (R6) Liste des noms et adresses email des clients dont l'adresse email est l'adresse d'un hôtel de la base (des hoteliers quoi !)

Vérif. : 2 réponses avec SELECT DISTINCT, et quatre réponses avec SELECT simple, expliquer pourquoi !

```
Solution 6 SELECT DISTINCT g.NomCL, g.AdrCL
FROM Guests g, Hotels h
WHERE g.AdrCL=h.AdrH
```

Pourquoi obtient-on quatre réponses avec SELECT contre deux avec SELECT DISTINCTS ? Pour le voir faisons :

```
SELECT *
FROM Guests g, Hotels h
WHERE g.AdrCL=h.AdrH
```

On s'aperçoit que pour chacune des deux personnes, avec la même adresse email, on a deux hotels, (situés dans des stations différentes) : la même personne gère donc deux hotels.

- (R7) Liste des noms et adresses des clients ayant réservé à la montagne.

Remarque : *Ici deux jointures. Quand elle est possible dans les deux sens, la jointure est associative.*

Vérif. : 889 réponses.

```
Solution 7 SELECT DISTINCT g.nomcl,g.adrcl
FROM resorts r, bookings b, guests g
WHERE r.ns=b.ns AND g.ncl=b.ncl AND r.types='montagne'

SELECT DISTINCT g.nomcl,g.adrcl
FROM bookings b JOIN guests g ON g.ncl=b.ncl
JOIN resorts r ON r.ns=b.ns
WHERE r.types='montagne'
```

- (R8) Nom de l'hôtel ayant le plus grand nombre de chambres, avec son numéro de station : attention, on veut que ce nom soit une valeur de retour de SQL, il ne s'agit pas de faire simplement ORDER BY nbch pour lire le résultat dans la liste.

```
Solution 8 SELECT nomh, ns FROM hotels WHERE nbch=(SELECT MAX(nbch) FROM hotels)
```

Vérif. : Sea Hotel, 1.

- (R9) Liste des numéros des hôtels avec leur numéro de station ayant *au moins* une chambre dont le prix est supérieur strictement à 40 euros.

Liste des *vrais noms* des hôtels avec leur numéro de station ayant *au moins* une chambre dont le prix est supérieur strictement à 40 euros

En déduire la liste des noms des hôtels avec leur numéro de station ayant *toutes leurs chambres* de prix inférieur ou égal à 40 euros.

Vérif. 31 réponses pour la dernière question.

Solution 9 Pour la première requête, on peut n'utiliser que la table rooms.

```
SELECT DISTINCT ns,nh FROM rooms WHERE prix >40
```

Si en plus, pour la même requête, on voulait le vrai nom des hotels, on a besoin de la jointure : SELECT DISTINCT r.ns, h.nomh FROM rooms r, hotels h WHERE r.prix>40 AND r.ns=h.ns AND r.nh=h.nh

Pour la deuxième requête demandée : il suffit de prendre le complémentaire dans la liste précédente, avec EXCEPT.

```
SELECT DISTINCT r.ns, h.nomh FROM rooms r, hotels h WHERE r.ns=h.ns AND r.nh=h.nh
EXCEPT SELECT DISTINCT r.ns, h.nomh FROM rooms r, hotels h
WHERE r.prix>40 AND r.ns=h.ns AND r.nh=h.nh
```

- (R10)** Prix de la chambre la moins chère située dans un hôtel trois étoiles d'une station balnéaire ?
Vérif. 65 euros.

```
Solution 10 SELECT MIN(ro.prix) AS prix
FROM Resorts re, Hotels h, Rooms ro
WHERE re.ns=h.ns AND h.ns=ro.ns AND h.nh=ro.nh AND h.cath=3
AND re.types='mer'
```

- (R11)** Liste des numéros de chambres d'hôtels avec leur noms d'hôtels et leur numéro de station, ayant un prix inférieur au prix moyen de toutes les chambres d'hôtels. **Vérif.** : 1726 réponses.

Puis la liste des hotels (nomh, ns), avec leur *nombre de chambres* vérifiant la condition de prix précédente.
(On utilisera la fonction COUNT)
Vérif. : 47 réponses.

```
Solution 11 SELECT h.nomh, h.ns, r.nch FROM rooms r, hotels h WHERE r.ns=h.ns AND r.nh=h.ns
AND r.prix < (SELECT avg(prix) FROM rooms)
Puis :
```

```
SELECT h.nomh, h.ns, COUNT(r.nch) FROM rooms r, hotels h WHERE r.ns=h.ns AND r.nh=h.ns
AND r.prix < (SELECT avg(prix) FROM rooms) GROUP BY h.nomh, h.ns
```

- (R12)** Obtenir la liste des numéros d'hôtels avec leur numéro de station des hôtels 4 étoiles n'ayant que des chambres avec salles de bain.

Vérif. 11 réponses.

```
Solution 12 SELECT DISTINCT ns, nh FROM hotels WHERE cath=4 EXCEPT
SELECT DISTINCT ns,nh FROM rooms WHERE typch<>"SDB"
```

Remarque : comment ferait-on, pour la requête précédente, si on voulait faire apparaître le nom de l'hôtel, sachant qu'il n'apparaît pas dans la table rooms et que le EXCEPT doit faire la différence de tables ayant les mêmes attributs ? Il faudrait faire une jointure supplémentaire de ce résultatat avec la table hôtel.

On peut aussi utiliser un NOT IN, qui permet d'éviter cet inconvénient :

```
SELECT ns, nh, nomh FROM hotels WHERE cath=4 AND (ns,nh) NOT IN (
SELECT ns,nh FROM rooms WHERE typch<>"SDB")
```

N.B. La requête précédente marche "théoriquement" mais il semble qu'en SQLite le IN (et le NOT IN) n'est pas capable de gérer une liste d'attributs.

- (R13)** (i) Listes des hôtels avec leur nom, adresse et leur nombre de réservations dans l'année affiché dans une colonne appelée NbResa, ordonné par nombre de réservation décroissante.

Vérif. on devrait trouver 78 réponses, car on sait bien qu'il y a 78 hotels mais :

Attention : un hôtel n'a pas de réservation, donc n'apparaît pas dans la table bookings. Donc la jointure avec bookings ne fera apparaître que 77 hotels. Donc :

(ii) Ecrire une requête qui permet de détecter l'hôtel (disons avec ns, nh) qui n'est pas dans bookings.

Solution 13 (i) Si on veut compter le nombre de réservations :

```
SELECT h.nomh, h.adrh, COUNT(*) AS NbResa FROM hotels h, bookings b
WHERE h.ns=b.ns AND h.nh=b.nh GROUP BY h.ns,h.nh
```

Mais il n'y a que 77 hotels qui apparaissent car l'hôtel qui n'a pas de réservations n'apparaît pas dans la jointure avec la table bookings.

(ii)

```
SELECT ns, nh FROM hotels EXCEPT SELECT ns,nh FROM bookings
```

- (R14)** -Nom et adresse de l'hôtel de la station « Chamonix » ayant eu le plus de réservations dans l'année.

Indication On peut bien sûr utiliser des requêtes imbriquées, mais cela en fait pas mal, les tester au fur et à mesure. Une autre méthode peut-être plus agréable (requête moins lourde) est d'utiliser un HAVING.

Vérification On trouvera l'Hôtel de la Gentiane.

Solution 14 On reprend le code précédent qui comptait le nombre de réservations dans chaque hôtel, en se restreignant à la station chamonix.

```
SELECT h.nomh, h.adrh, COUNT(*) AS NbResa FROM hotels h, bookings b, resorts r
WHERE h.ns=b.ns AND h.nh=b.nh AND h.ns=r.ns AND r.noms="Chamonix" GROUP BY h.ns,h.nh
```

Puis de ce code on va extraire le uplet réalisant le MAX de la réservation, en faisant précéder la requête précédente de `SELECT MAX(NbResa) FROM ()` : on trouve 790. Et enfin on trouve l'hôtel dont le nombre de réservations est égale à ce MAX (ouf) : ce qui donne la requête complète suivante :

```
SELECT nomh,NbResa FROM (SELECT nomh, adrh, COUNT(*) AS NbResa
FROM hotels h, bookings b, resorts r
WHERE h.ns=b.ns AND h.nh=b.nh AND r.noms="Chamonix" GROUP BY h.ns,h.nh)
WHERE NbResa = (SELECT MAX(NbResa) FROM (SELECT nomh, adrh, COUNT(*) AS NbResa
FROM hotels h, bookings b, resorts r
WHERE h.ns=b.ns AND h.nh=b.nh AND r.ns=h.ns AND r.noms="Chamonix" GROUP BY h.ns,h.nh) )
```

- (R15)** Nom de clients ayant réservé au moins deux jours consécutifs une même chambre dans le même hôtel de la même station.

Vérif. 976 réponses.

Indication – L'idée est de faire une *auto-jointure* c'est-à-dire une jointure de la table bookings avec elle-même. En pratique, la requête commencera par :

```
SELECT DISTINCT g.nomcl FROM guests g, bookings b1, bookings b2
```

Solution 15 `SELECT DISTINCT g.nomcl FROM guests g, bookings b1, bookings b2
WHERE g.ncl=b1.ncl AND g.ncl=b2.ncl
AND b1.ns=b2.ns
AND b1.nh=b2.nh
AND b1.nch=b2.nch
AND b1.Jour=b2.Jour-1`

- (R16)** Jour de l'année où l'hôtel « Bon séjour » de la station « Chamonix » a eu le plus de réservations.

Vérification On trouvera deux jours : le jour 99 et le jour 100.

Solution 16 `SELECT b.jour
FROM Bookings b, Resorts r, Hotels h
WHERE r.ns = h.ns AND h.ns = b.ns AND h.nh = b.nh
AND r.NomS = "Chamonix" AND h.NomH = "Bon Séjour" GROUP BY b.jour
HAVING COUNT(*) = (
SELECT MAX(Nb) FROM (SELECT COUNT(*) AS Nb
FROM Bookings b, Resorts r, Hotels h
WHERE r.ns = h.ns AND h.ns = b.ns AND h.nh = b.nh
AND r.NomS = "Chamonix" AND h.NomH = "Bon Séjour" GROUP BY b.jour))`

- (R17)** Noms des clients ayant réservé dans tous les hôtels 2 étoiles de la station « Chamonix ».

Vérifications 5 réponses.

```
Solution 17 SELECT g.NC1, g.NomCl
FROM Guests g, Bookings b, Hotels h, Resorts r
WHERE g.NC1 = b.NC1 AND h.NS = r.NS AND h.NS = b.NS AND h.NH = b.NH
AND r.nomS ="Chamonix" AND h.CatH = 2 GROUP BY g.NC1, g.NomCl
HAVING COUNT(DISTINCT h.NH) = (
SELECT COUNT (*)
FROM Hotels h, Resorts r
WHERE h.NS = r.NS AND r.NomS = "Chamonix" AND h.CatH = 2
);
```