

Nouvelle Calédonie – 2025 – sujet1 - Correction

Exercice 2 (6 points)

1) Requête

```
SELECT nom  
FROM lac  
WHERE altitude <= 2000;
```

Dans l'extrait fourni :

nom	altitude
Lac Blanc	2354
Lacs Noirs	2564
Lac Vert	1266
Lac Rouge	2585

Seul lac dont altitude ≤ 2000 :

👉 **Lac Vert**

2) Lacs atteignables depuis Chamonix

Dans la table parking : Chamonix \rightarrow idP = 1

Dans la table rando :

depart	arrivee
1	1
1	2

Donc lacs accessibles :

- idL = 1 → Lac Blanc
- idL = 2 → Lacs Noirs

👉 Réponse : **Lac Blanc et Lacs Noirs**

3) Coordonnées GPS des parkings de Passy entre 800 et 1000 m

```
SELECT coord_GPS
FROM parking
WHERE commune = 'Passy'
AND altitude > 800
AND altitude < 1000;
```

👉 *Commentaire : Strictement entre → > et <*

4) Lacs atteignables depuis parking à 1300 m à Cordon

```
SELECT lac.nom
FROM parking
JOIN rando ON parking.idP = rando.depart
JOIN lac ON rando.arrivee = lac.idL
WHERE parking.commune = 'Cordon'
AND parking.altitude = 1300;
```

👉 *Commentaire : Double JOIN nécessaire.*

5) Ajouter le lac et la randonnée

⚠️ **Ordre important** : Il faut insérer le lac AVANT la randonnée (clé étrangère).

Ajouter le lac

```
INSERT INTO lac VALUES (42, 'Lac d Anterne', 2059);
```

Ajouter la randonnée

```
INSERT INTO rando VALUES (100, 3, 42);
```

6) Corriger l'erreur

```
UPDATE lac
SET nom = 'Lac d Anterne'
WHERE idL = 42;
```

7) Supprimer le parking 28

⚠ Il faut d'abord supprimer les randonnées associées (clé étrangère).

```
DELETE FROM rando
WHERE depart = 28;
```

```
DELETE FROM parking
WHERE idP = 28;
```

8) Fonction get_parking

```
def get_parking(randos):
    parkings = []
    for rando in randos:      # ligne 3
        if rando.depart not in parkings:
            parkings.append(rando.depart)  # ligne 5
    return parkings
```

👉 *Commentaire : On évite les doublons avec not in.*

9) Fonction get_nb_rando

```
def get_nb_rando(parking, randos):  
  
    nb = 0  
  
    for rando in randos:  
        if rando.depart == parking: # ligne 4  
            nb = nb + 1  
  
    return nb
```

10) Fonction nb_rando_par_parking

```
def nb_rando_par_parking(randos):  
  
    resultat = {}  
  
    for rando in randos:  
  
        if rando.depart not in resultat:  
            resultat[rando.depart] = 1  
        else:  
            resultat[rando.depart] += 1  
  
    return resultat
```
