

---

 Informatique, sujets de concours SQL, une correction
 

---

## Exercice 1 (Mines 2022)

## Recherche dans une base de données de matériaux magnétiques

Il existe des bases de données contenant les propriétés de nombreux matériaux, dont des propriétés magnétiques. Dans cette partie, on donne un modèle simplifié d'une telle base, et on souhaite effectuer quelques requêtes sur celle-ci. La base de données possède la structure suivante :

- La table `matériaux` contient un champ `id_materiau`, clé primaire de la table de valeur entière, un champ `nom` de type chaîne de caractères pour le nom du matériau et un champ `t_curie` de valeur entière pour la température de Curie du matériau en kelvin.

<code>id_materiau</code>	<code>nom</code>	<code>t_curie</code>
4534	cobalt	1 388
1254	dioxyde de chrome	386
8713	nickel	627
8284	YIG	560
...	...	...

- La table `fournisseurs`, contenant un champ `id_fournisseur`, clé primaire de type entier qui précise le code de chaque fournisseur, et un champ `nom_fournisseur` de type chaîne de caractères pour le nom du fournisseur.

<code>id_fournisseur</code>	<code>nom_fournisseur</code>
145	Worldwide Materials
13	Materials Company
...	...

- La table `prix` qui contient un champ `id_prix`, clé primaire de type entier, un champ `id_mat` dont les valeurs sont incluses dans l'ensemble des valeurs de la clé `id_materiau` de la table `matériaux`, un champ `id_four` dont les valeurs sont incluses dans l'ensemble des valeurs de la clé `id_fournisseur` de la table `fournisseurs`, et un champ `prix_kg` de type flottant qui précise le prix au kg que ce fournisseur propose pour ce matériau, en euros. Un fournisseur qui ne propose pas un matériau donné n'a pas d'entrée correspondante dans cette table.

<code>id_prix</code>	<code>id_mat</code>	<code>id_four</code>	<code>prix_kg</code>
1	4567	145	50.40
2	8671	13	1357.30
3	1763	145	52.75
...	...	...	...

1. Écrire une requête permettant d'obtenir le nom de tous les matériaux qui ont une température de Curie strictement inférieure à 500 kelvins.

Un client potentiel souhaite acheter 4,5 kilogrammes de nickel (d'identifiant 8713, que l'on pourra utiliser directement dans les requêtes) et sélectionner le fournisseur le moins cher.

2. Écrire une requête permettant d'obtenir les noms de tous les fournisseurs proposant du nickel et le prix proposé par chacun pour 4,5 kilogrammes de nickel.

3. Modifier ou compléter la requête précédente afin d'obtenir le nom du fournisseur de nickel le moins cher ainsi que le prix à payer chez ce fournisseur pour ces 4,5 kilogrammes de nickel. En cas d'égalité du prix optimal entre plusieurs fournisseurs, on obtiendra les noms de tous les fournisseurs possibles.
4. Écrire une requête permettant d'obtenir le nom de tous les matériaux et le prix moyen pour un kilogramme de chacun de ces matériaux (la moyenne étant calculée pour tous les fournisseurs proposant ce matériau), en se limitant aux prix moyens strictement inférieurs à 50 euros par kilogramme.

Une correction

1. Une requête permettant d'obtenir le nom de tous les matériaux qui ont une température de Curie strictement inférieure à 500 kelvins est :

```
SELECT
    nom
FROM
    materiaux
WHERE
    t_curie > 500;
```

Un client potentiel souhaite acheter 4,5 kilogrammes de nickel (d'identifiant 8713, que l'on pourra utiliser directement dans les requêtes) et sélectionner le fournisseur le moins cher.

2. Une requête permettant d'obtenir les noms de tous les fournisseurs proposant du nickel et le prix proposé par chacun pour 4,5 kilogrammes de nickel est :

```
SELECT
    nom_fournisseur,
    prix_kg * 4.5
FROM
    fournisseurs
INNER JOIN
    prix
        ON id_four = id_fournisseur
WHERE
    id_mat = 8713;
```

3. On peut modifier ou compléter la requête précédente afin d'obtenir le nom du fournisseur de nickel le moins cher ainsi que le prix à payer chez ce fournisseur pour ces 4,5 kilogrammes de nickel de la manière suivante, avec ,en cas d'égalité du prix optimal entre plusieurs fournisseurs, les noms de tous les fournisseurs possibles.) de la manière suivante.

```
SELECT
    nom_fournisseur,
    prix_kg * 4.5
FROM
    fournisseurs
INNER JOIN
    prix
        ON id_four = id_fournisseur
WHERE
    id_mat = 8713
    AND prix_kg =( SELECT
                    MIN(prix_kg)
                  FROM
                    prix
                  WHERE
                    id_mat = 8713);
```

4. Une requête permettant d'obtenir le nom de tous les matériaux et le prix moyen pour un kilogramme de chacun de ces matériaux (la moyenne étant calculée pour tous les fournisseurs proposant ce matériau), en se limitant aux prix moyens strictement inférieurs à 50 euros par kilogramme est :

```

SELECT
    nom,
    AVG(prix_kg) AS moyenne
FROM
    materiaux
INNER JOIN
    prix
    ON id_mat = id_materiau
GROUP BY
    id_mat
HAVING
    moyenne < 50;

```

Je rappelle que l'instruction `GROUP BY id_mat` permet d'effectuer la moyenne des prix pour chacun des matériaux.

### Exercice 2 (Mines 2021)

Lors de la préparation d'une randonnée, une accompagnatrice doit prendre en compte les exigences des participants. Elle dispose d'informations rassemblées dans deux tables d'une base de données :

- la table **Rando** décrit les randonnées possibles ? la clef primaire entière **rid**, son nom, le niveau de difficulté du parcours (entier entre 1 et 5), le dénivelé (en mètres), la durée moyenne (en minutes) :

rid	rnom	diff	deniv	duree
1	La belle des champs	1	20	30
2	Lac de Castellane	4	650	150
3	Le tour du mont	2	200	120
4	Les crêtes de la mort	5	1200	360
5	Yukon Ho!	3	700	210
...	...	...	...	...

- la table **Participant** décrit les randonneurs ? la clef primaire entière **pid**, le nom du randonneur, son année de naissance, le niveau de difficulté maximum de ses randonnées :

pid	pnom	ne	diff_max
1	Calvin	2014	2
2	Hobbes	2015	2
3	Susie	2014	2
4	Rosalyn	2001	4
...	...	...	...

Donner une requête SQL sur cette base pour :

1. Compter le nombre de participants nés entre 1999 et 2003 inclus.
2. Calculer la durée moyenne des randonnées pour chaque niveau de difficulté.
3. Extraire le nom des participants pour lesquels la randonnée n°42 est trop difficile.
4. Extraire les clés primaires des randonnées qui ont un ou des homonymes (nom identique et clé primaire distincte), sans redondance.

Une correction

1. Pour compter le nombre de participants nés entre 1999 et 2003 inclus on peut procéder de la manière suivante :

```
SELECT
    COUNT(*)
FROM
    Participant
WHERE
    ne >= 1999 AND ne <= 2003;
```

2. La durée moyenne des randonnées pour chaque niveau de difficulté peut s'obtenir de la manière suivante :

```
SELECT
    diff,
    AVG(duree)
FROM
    Rando
GROUP BY
    diff ;
```

3. Le nom des participants pour lesquels la randonnée n°42 est trop difficile peut s'obtenir par :

```
SELECT
    pnom
FROM
    Participant
WHERE
    diff_max < (SELECT
                    diff
                FROM
                    Rando
                WHERE
                    rid=42);
```

On peut aussi utiliser la requête suivante :

```
SELECT
    pnom
FROM
    Participant,
    Rando
WHERE
    rid = 42
    AND diff_max < diff ;
```

4. Pour extraire les clés primaires des randonnées qui ont un ou des homonymes (nom identique et clé primaire distincte), sans redondance, il y a les deux possibilités suivantes.

- Avec une auto-jointure :

```
SELECT DISTINCT
    R.rid
FROM
    Rando AS Rando_clone1
INNER JOIN
    Rando AS Rando_clone2
        ON Rando_clone1.rnom=Rando_clone2.rnom
WHERE
    R.rid <> S.rid ;
```

- Avec un GROUP BY et une sous-requête :

```

SELECT DISTINCT
    rid
FROM
    Rando
WHERE
    rnom IN ( SELECT
                rnom
            FROM
                rando
            GROUP BY
                rnom
            HAVING
                COUNT(*) > 1);

```

### Exercice 3 (Centrale 2017)

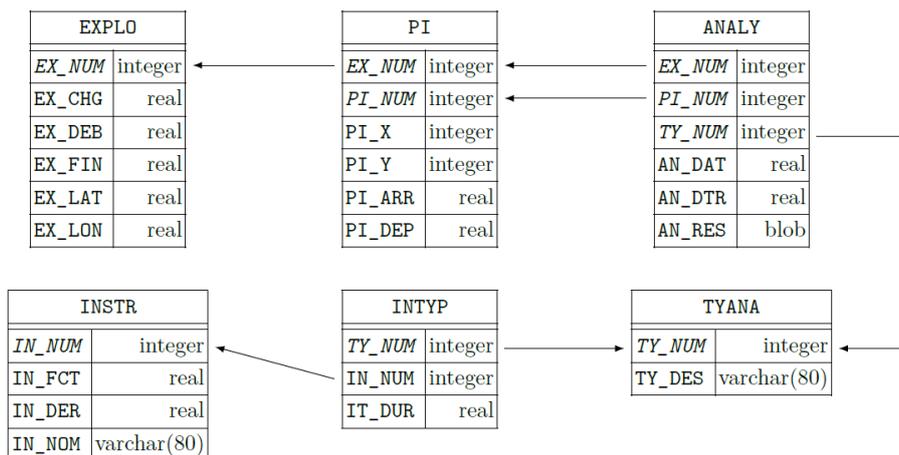
Mars Exploration Rovers (MER) est une mission de la NASA qui cherche à étudier le rôle joué par l'eau dans l'histoire de la planète Mars. Deux robots géologues, Spirit et Opportunity (figure 1), se sont posés sur cette planète, sur deux sites opposés, en janvier 2004. Leur mission est de rechercher et d'analyser différents types de roches et de sols qui peuvent contenir des indices sur la présence d'eau. Ils sont équipés de six roues et d'une suspension spécialement conçue pour leur permettre de se déplacer quelle que soit la nature du terrain rencontré. Leur cahier des charges prévoyait une durée de vie de 90 jours martiens (le jour martien est environ 40 minutes plus long que le jour terrestre). Spirit a cessé d'émettre le 22 mars 2010, soit 2210 jours martiens après son arrivée sur la planète. Début 2017, Opportunity est toujours en activité et il a parcouru plus de 44 km sur Mars.

Chaque robot est équipé de plusieurs instruments d'analyse (caméra, microscope, spectromètres) et d'un bras qui permet d'amener les instruments au plus près des roches et sols dignes d'intérêt. À partir de photographies de la surface de la planète, prises à plusieurs longueurs d'ondes par différents satellites et par le robot lui-même, les scientifiques de la NASA définissent une liste d'emplacements (points d'intérêt ou PI) où effectuer des analyses. Cette liste est transmise au robot qui doit se rendre à chaque emplacement indiqué et y effectuer les analyses prévues. Chaque robot est capable d'effectuer un certain nombre de types d'analyses géologiques correspondant aux différents instruments dont il dispose. Une fois tous les points d'intérêts visités et les résultats des analyses

Afin d'assurer son autonomie opérationnelle, le robot dispose localement des informations nécessaires à son fonctionnement quotidien. Ainsi, il enregistre la durée d'utilisation de ses différents instruments embarqués. Il connaît également les différents types d'analyses qu'il peut effectuer et, pour chacun de ces types, les instruments à utiliser. Il enregistre la prochaine exploration, c'est-à-dire les différents points d'intérêts qu'il doit visiter et pour chacun la ou les analyses qu'il doit effectuer. D'autre part, il conserve les résultats d'analyses effectuées lors de ses explorations passées. Ces résultats ne sont effacés qu'après confirmation de leur bonne transmission sur Terre. Ces différentes informations sont stockées dans une base de données relationnelle dont le modèle physique est schématisé figure suivante.

Cette base comporte les six tables suivantes :

- la table EXPLO des explorations, avec les colonnes
  - EX\_NUM numéro (entier) de l'exploration (clef primaire)
  - EX\_CHG date de transmission des points d'intérêts de cette exploration
  - EX\_DEB date de début de l'exploration (NULL si l'exploration n'est pas encore commencée)
  - EX\_FIN date de fin de l'exploration (NULL si l'exploration n'est pas encore terminée)
  - EX\_LAT latitude (en degrés décimaux) du point de coordonnées (0, 0) de la zone d'exploration
  - EX\_LON longitude (en degrés décimaux) du point de coordonnées (0, 0) de la zone d'exploration



- la table PI des points d'intérêts, de clef primaire (EX\_NUM, PI\_NUM), avec les colonnes
  - EX\_NUM numéro de l'exploration à laquelle appartient le point d'intérêt
  - PI\_NUM numéro du point d'intérêt dans l'exploration (au sein d'une exploration les PI sont numérotés en séquence en commençant à 0, ce numéro n'a pas de rapport avec l'ordre dans lequel les PI sont explorés par le robot)
  - PI\_X l'abscisse du point d'intérêt dans la zone d'exploration (entier positif en millimètres)
  - PI\_Y l'ordonnée du point d'intérêt dans la zone d'exploration (entier positif en millimètres)
  - PI\_ARR date d'arrivée du robot au point d'intérêt (NULL si ce point n'a pas encore été visité)
  - PI\_DEP date à laquelle le robot a quitté le point d'intérêt (NULL si ce point n'a pas encore été exploré ou si la visite est en cours)
- la table INSTR des instruments embarqués, avec les colonnes
  - IN\_NUM le numéro (entier) de l'instrument (clef primaire)
  - IN\_FCT la durée pendant lequel l'instrument a déjà été utilisé depuis l'arrivée sur la planète (nombre décimal : fraction de jour martien)
  - IN\_DER la date de la dernière utilisation de l'instrument
  - IN\_NOM nom de l'instrument
- la table TYANA des types d'analyses à effectuer, avec les colonnes
  - TY\_NUM le numéro de référence (entier) du type d'analyse (clef primaire)
  - TY\_DES le nom du type d'analyse
- la table INTYP des instruments utilisés pour un type d'analyse, de clef primaire (TY\_NUM, IN\_NUM), avec les colonnes
  - TY\_NUM le numéro de référence (entier) du type d'analyse
  - IN\_NUM le numéro (entier) de l'instrument
  - IT\_DUR la durée standard d'utilisation de l'instrument dans ce type d'analyse (nombre décimal : fraction de jour martien)
- la table ANALY indiquant pour chaque point d'intérêt les types d'analyses à effectuer ou effectuées, de clef primaire (EX\_NUM, PI\_NUM, TY\_NUM) et avec les colonnes
  - EX\_NUM numéro de l'exploration à laquelle appartient le point d'intérêt
  - PI\_NUM numéro du point d'intérêt dans l'exploration
  - TY\_NUM type de l'analyse
  - AN\_DAT date de l'analyse (NULL si l'analyse n'a pas été effectuée)
  - AN\_DTR date de transmission sur Terre des résultats de l'analyse (NULL si l'analyse n'a pas été transmise)
  - AN\_RES résultat de l'analyse (donnée opaque dont la signification dépend du type d'analyse)

Toutes les dates sont stockées sous forme d'un nombre décimal correspondant au nombre de jours martiens depuis l'arrivée du robot sur la planète.

1. Écrire une requête SQL qui donne le numéro de l'exploration en cours, s'il y en a une.
2. Écrire une requête SQL qui donne, pour une exploration dont on connaît le numéro, la liste des points d'intérêts de cette exploration avec leurs coordonnées.
3. Écrire une requête SQL qui donne la surface, en mètres carrés, de chaque zone déjà explorée par le robot. La zone d'exploration est définie comme le plus petit rectangle qui englobe l'ensemble des points d'intérêts de l'exploration et dont les bords sont parallèles aux axes de référence (axes des abscisse et des ordonnées).
4. Quelle est la surface maximale d'une zone d'exploration que peut stocker cette base de données ?
5. Écrire une requête SQL qui donne, pour l'exploration en cours, le nombre de fois où chaque instrument doit être utilisé et sa durée d'utilisation théorique (en jours martiens) pour la totalité de l'exploration.

### Correction

Le sujet manipule des valeurs NULL, hors-programme pour nous. Une valeur NULL représente un champ vide, non-rempli. La condition `NULL = NULL` teste si un champ non rempli vaut un autre champ non rempli. Pour SQL il y a trois valeurs de vérité : `true`, `false` et `unknown`.

La condition `NULL = NULL` renvoie `unknown`. On peut en fait utiliser `IS NULL` et `IS NOT NULL` qui permettent de vérifier si une valeur vaut NULL, ce qui renverra `true` ou `false`.

1. L'exploration en cours a une date de début, mais pas de date de fin. Une requête qui donne le numéro de l'exploration en cours, s'il y en a une, est :

```
SELECT
    EX_NUM
FROM
    EXPLO
WHERE
    EX_DEB IS NOT NULL
    AND EX_FIN IS NULL;
```

2. Une requête qui donne, pour une exploration dont on connaît le numéro `number_x`, la liste des points d'intérêts de cette exploration avec leurs coordonnées peut-être :

```
SELECT
    PI_NUM,
    PI_X,
    PI_Y
FROM
    PI
WHERE
    EX_NUM = number_x;
```

3. A l'aide d'un petit calcul d'aire (faire un dessin...), une requête qui donne la surface, en mètres carrés, de chaque zone déjà explorée par le robot peut-être :

```
SELECT
    EX_NUM,
    (MAX(PI_X)-MIN(PI_X)) * (MAX(PI_Y)-MIN(PI_Y)) AS aire
    #Calcul l'aire d'après l'indication de l'énoncé.
FROM
    EXPLO
INNER JOIN
```

```

        PI
        ON EXPL0.EX_NUM=PI.EX_NUM
WHERE
        EX_FIN IS NOT NULL
        #La zone est déjà explorée
GROUP BY EX_NUM;

```

4. **Question dégueulasse.** D'après le rapport du jury, "la question 4 nécessitait de prendre des initiatives en définissant le nombre de bits de codage des entiers pour en déduire la surface maximale d'exploration enregistrable."

Comme les coordonnées sont positives, une exploration a tous ses points dans un carré dont les extrémités d'u diagonale sont les points  $(0, 0)$  et  $(Max, Max)$  où  $Max$  est l'entier maximal représentable. L'aire de ce carré est  $\mathcal{A} = Max^2$ .

En supposant que les entiers sont de 64 bits, alors  $Max = 2^{64} - 1$ . On obtient la superficie maximale en élevant au carré, sans oublier l'unité ( $m^2$ ).

5. Une requête SQL qui donne, pour l'exploration en cours, le nombre de fois où chaque instrument doit être utilisé et sa durée d'utilisation théorique (en jours martiens) pour la totalité de l'exploration, peut-être :

```

SELECT
        IN_NUM,
        COUNT(*) AS NB_UTILISATIONS,
        SUM(IT_DUR) AS DUREE
FROM
        EXPL0
JOIN
        ANALY
        ON EXPL0.EX_NUM = ANALY.EX_NUM
JOIN
        INTYP
        ON ANALY.TY_NUM = INTYP.TY_NUM
WHERE
        EX_DEB IS NOT NULL
        AND EX_FIN IS NULL
GROUP BY
        IN_NUM;

```