

Projet de Bases de Données Relationnelles *Executive Master IASD*

Table des matières

Enoncé	1
Travail à réaliser - Requêtes d'interrogation en SQL	1
Rendu du TP	3
Contenu des instances de la base de données	3
Résultat de la requête h) sur les instances de la base de données	5
Résultat de la requête i) sur les instances de la base de données	6
Résultat de la requête j) sur les instances de la base de données	6
Script SQL de création des relations de la base de données :.....	8

Enoncé

Le script SQL, situé à la fin du sujet (page 8), modélise une base de données permettant de gérer les incidents des lignes de métro¹

Une ligne de métro est caractérisée par un numéro de ligne, 2 terminus, un nombre d'incidents et un nombre d'heures d'incidents.

Un type d'incident est caractérisé par un identificateur et un nom (ex. « Régulation », « Malaise voyageur »).

Un incident est caractérisé par un identificateur, une date, le numéro de la ligne de métro affectée, l'identificateur du type d'incident, un nombre de fois où le type d'incident a eu lieu à cette date et le nombre d'heures impactées.

Des exemples d'instances de la base de données sont données en fin de sujet (page 3).

Travail à réaliser - Requêtes d'interrogation en SQL

Ce projet est à effectuer seul.e

Vos requêtes SQL doivent fonctionner sur les instances données dans le sujet mais également sur toutes les instances possibles. Par conséquent, vous ne pouvez donc utiliser que les informations énoncées dans l'énoncé de chaque requête et non les identificateurs dont vous auriez eu connaissance en regardant le contenu des tables.

Pour connaître le temps d'exécution d'une requête, vous pouvez utiliser la commande `EXPLAIN ANALYSE` (requête `SELECT`). Cette instruction affiche le plan d'exécution de la requête, avec à la fin de l'affichage le temps que le SGBD a mis pour optimiser et le temps pour exécuter la requête.

Sous *dbfiddle*, pour calculer le temps, il faut lancer la requête seule (pour que les résultats des requêtes précédentes ne soient pas en mémoire), de préférence plusieurs fois, pour faire une moyenne des temps.

¹ Schéma inspiré des données de <https://x.com/METROQualite/status/1610687301830709248>

Ecrivez en SQL les requêtes suivantes² :

- a) Quelles lignes de métro (en précisant son numéro) n'a eu aucun incident le 01/01/2024 ?
- b) Quels sont les noms des types d'incident associés à aucune ligne ? **Écrire 2 requêtes SQL, une avec NOT IN ou NOT EXISTS et une avec une jointure externe et comparer leur temps d'exécution.**
- c) Quelles lignes (en précisant le numéro) a fait l'objet d'incidents de type 'Régulation' OU 'Malaise voyageur' ? **Écrire 2 requêtes SQL, une avec UNION et une sans UNION et comparer leur temps d'exécution.**
- d) Quelles lignes (en précisant le numéro) a fait l'objet d'incidents de type 'Régulation' ET d'incidents de type 'Malaise' ? **Écrire 2 requêtes SQL, une avec INTERSECT et une sans INTERSECT et comparer leur temps d'exécution.**
- e) Quels types d'incidents (en précisant tous ses attributs) sont associés à toutes les lignes ?

Par exemple, sur les nuplets insérés par le script donné en fin de sujet, la requête doit renvoyer :

tid	nomincident
1	Incident technique

- f) Quelle est la moyenne des heures de perturbations par jour et par ligne (i.e. pour chaque ligne et chaque jour, la somme du nombre d'heures du jour divisé par le nombre d'incidents du jour) ?
- g) Quelles sont les lignes (en précisant leur numéro) avec le plus d'incidents ? **Écrire 3 requêtes SQL, une avec ALL, une avec MAX et une avec NOT EXISTS.**
- h) Quel est, pour chaque ligne (en précisant leur numéro), le nombre d'heures d'incidents, en numérotant les lignes de métro de la moins impactée à la plus impactée. *Un exemple de résultat de cette requête sur les instances de la base de données du sujet est donné page 5.*
- i) Quelles sont les lignes impactées par un incident à chaque date ? Votre requête devra renvoyer pour chaque date, la liste des numéros de lignes séparés par une virgule. *Un exemple de résultat de cette requête sur les instances de la base de données du sujet est donné page 6.*
- j) Quels sont le nombre d'incidents et le nombre d'heures par numéro de ligne et par type d'incidents, en calculant le total par ligne et le total pour toutes les lignes. *Un exemple de résultat de cette requête sur les instances de la base de données du sujet est donné page 6.*

² Vous pouvez utiliser des vues pour décomposer vos requêtes en sous-requêtes si besoin.

Rendu du TP

Ce TP donnera lieu à un **script SQL** contenant :

- Les requêtes d'interrogation en SQL. *Merci d'indiquer svp en commentaire l'intitulé de chaque requête.*
- Les requêtes a) à e) en Algèbre Relationnelle.

Le **script SQL devra être enregistré dans un fichier**, dont le nom devra obligatoirement être de la forme **NomPrenom.sql** (i.e. en précisant votre nom et prénom), vos requêtes en Algèbre devront être sauvegardées de préférence dans un fichier pdf ou dans un fichier de type image dont le nom devra également comporter votre nom et prénom.

Les fichiers **devront être déposés sur Moodle**.

Contenu des instances de la base de données

Un exemple d'instance (créée à partir du script situé en fin de sujet et généré par *chatGPT*) de la relation Ligne est :

numeroligne	terminus1	terminus2	nbincidents	nbheuresincidents
1	La Défense	Château de Vincennes	1	2
2	Porte Dauphine	Nation	2	3.5
3	Pont de Levallois	Gallieni	3	5
3bis	Gambetta	Porte des Lilas	1	2
4	Porte de Clignancourt	Mairie de Montrouge	2	4.5
5	Place d Italie	Bobigny - Pablo Picasso	3	6
6	Charles de Gaulle - Étoile	Nation	3	6
7	La Courneuve - 8 Mai 1945	Villejuif - Louis Aragon	2	3.5
7bis	Louis Blanc	Pré Saint-Gervais	1	2
8	Balard	Créteil - Pointe du Lac	2	4
9	Pont de Sèvres	Mairie de Montreuil	2	5.5
10	Boulogne - Pont de Saint-Cloud	Gare d Austerlitz	3	4
11	Châtelet	Mairie des Lilas	2	4
12	Front Populaire	Mairie d Issy	3	6.5
13	Saint-Denis - Université	Les Courtilles	2	5
14	Saint-Lazare	Olympiades	2	4.5
RER A	Saint-Germain-en-Laye	Boissy-Saint-Léger	3	7
RER B	Robinson	Saint-Rémy-lès-Chevreuse	2	3
RER C	Pontoise	Massy - Palaiseau	2	4
RER D	Creil	Melun	2	4.5
RER E	Hausmann - Saint-Lazare	Tournan	2	3.5

Un exemple d'instance (créée à partir du script situé en fin de sujet, généré par *chatGPT* et modifié) de la relation `TableTypeIncident` est donné sur la page suivante.

tid	nomincident
1	Incident technique
2	Grève
3	Malaise voyageur
4	Incident matériel
5	Régulation
6	Personnes sur les voies
7	Mesure de sécurité
8	Panne de train

Un exemple d'instance (créée à partir du script situé en fin de sujet généré par *chatGPT* et corrigé) de la relation `Incident` est :

id	datei	numeroligne	typeincident	nbincidents	nbheures
1	2024-01-01T00:00:00.000Z	1	1	1	2
2	2024-01-02T00:00:00.000Z	2	2	1	1.5
3	2024-01-03T00:00:00.000Z	3	3	2	3
4	2024-01-04T00:00:00.000Z	4	4	1	2.5
5	2024-01-05T00:00:00.000Z	5	5	1	1
6	2024-01-06T00:00:00.000Z	6	1	2	4
7	2024-01-07T00:00:00.000Z	7	2	1	1.5
8	2024-01-08T00:00:00.000Z	8	3	1	2
9	2024-01-09T00:00:00.000Z	9	4	1	3.5
10	2024-01-10T00:00:00.000Z	10	5	1	1
11	2024-01-11T00:00:00.000Z	11	1	1	2
12	2024-01-12T00:00:00.000Z	12	2	2	4.5
13	2024-01-13T00:00:00.000Z	13	3	1	3
14	2024-01-14T00:00:00.000Z	14	4	1	2.5
15	2024-01-15T00:00:00.000Z	RER A	5	2	5
16	2024-01-16T00:00:00.000Z	RER B	1	1	1
17	2024-01-17T00:00:00.000Z	RER C	2	1	2
18	2024-01-18T00:00:00.000Z	RER D	3	1	2.5
19	2024-01-19T00:00:00.000Z	RER E	4	1	1.5
20	2024-01-18T00:00:00.000Z	5	3	1	3
21	2024-01-17T00:00:00.000Z	10	3	1	1
22	2024-01-02T00:00:00.000Z	2	1	1	2
23	2024-01-02T00:00:00.000Z	3	1	1	2
24	2024-01-03T00:00:00.000Z	3bis	1	1	2
25	2024-01-04T00:00:00.000Z	4	1	1	2

Place du Maréchal de Lattre de
Tassigny
75775 Paris Cedex 16
dauphine.psl.eu

26	2024-01-03T00:00:00.000Z	5	1	1	2
27	2024-01-01T00:00:00.000Z	6	1	1	2
28	2024-01-01T00:00:00.000Z	7	1	1	2
29	2024-01-02T00:00:00.000Z	7bis	1	1	2
30	2024-01-04T00:00:00.000Z	8	1	1	2
31	2024-01-01T00:00:00.000Z	9	1	1	2
32	2024-01-02T00:00:00.000Z	10	1	1	2
33	2024-01-04T00:00:00.000Z	11	1	1	2
34	2024-01-02T00:00:00.000Z	12	1	1	2
35	2024-01-03T00:00:00.000Z	13	1	1	2
36	2024-01-01T00:00:00.000Z	14	1	1	2
37	2024-01-01T00:00:00.000Z	RER A	1	1	2
38	2024-01-01T00:00:00.000Z	RER B	1	1	2
39	2024-01-04T00:00:00.000Z	RER C	1	1	2
40	2024-01-04T00:00:00.000Z	RER D	1	1	2
41	2024-01-04T00:00:00.000Z	RER E	1	1	2

Résultat de la requête h) sur les instances de la base de données

Sur les nuplets insérés par le script donné en fin de sujet, la requête h) doit renvoyer :

numeroligne	nbheuresincidents	dense_rank
7bis	2	1
1	2	1
3bis	2	1
RER B	3	2
RER E	3.5	3
2	3.5	3
7	3.5	3
RER C	4	4
8	4	4
10	4	4
11	4	4
14	4.5	5
4	4.5	5
RER D	4.5	5
3	5	6
13	5	6
9	5.5	7
5	6	8

Place du Maréchal de Lattre de
Tassigny
75775 Paris Cedex 16
dauphine.psl.eu

6	6	8
12	6.5	9
RER A	7	10

Résultat de la requête i) sur les instances de la base de données

Sur les nuplets insérés par le script donné en fin de sujet, la requête i) doit renvoyer :

datei	listelignes
2024-01-01T00:00:00.000Z	1, 14, 6, 7, 9, RER A, RER B
2024-01-02T00:00:00.000Z	10, 12, 2, 2, 3, 7bis
2024-01-03T00:00:00.000Z	13, 3, 3bis, 5
2024-01-04T00:00:00.000Z	11, 4, 4, 8, RER C, RER D, RER E
2024-01-05T00:00:00.000Z	5
2024-01-06T00:00:00.000Z	6
2024-01-07T00:00:00.000Z	7
2024-01-08T00:00:00.000Z	8
2024-01-09T00:00:00.000Z	9
2024-01-10T00:00:00.000Z	10
2024-01-11T00:00:00.000Z	11
2024-01-12T00:00:00.000Z	12
2024-01-13T00:00:00.000Z	13
2024-01-14T00:00:00.000Z	14
2024-01-15T00:00:00.000Z	RER A
2024-01-16T00:00:00.000Z	RER B
2024-01-17T00:00:00.000Z	10, RER C
2024-01-18T00:00:00.000Z	5, RER D
2024-01-19T00:00:00.000Z	RER E

Résultat de la requête j) sur les instances de la base de données

Sur les nuplets insérés par le script donné en fin de sujet, la requête j) doit renvoyer :

numeroligne	nomincident	nbtotallincidents	nbtotalthours
1	Incident technique	1	2
1	Total	1	2
10	Incident technique	1	2
10	Malaise voyageur	1	1
10	Régulation	1	1
10	Total	3	4

Place du Maréchal de Lattre de
Tassigny
75775 Paris Cedex 16
dauphine.psl.eu

11	Incident technique	2	4
11	Total	2	4
12	Grève	2	4.5
12	Incident technique	1	2
12	Total	3	6.5
13	Incident technique	1	2
13	Malaise voyageur	1	3
13	Total	2	5
14	Incident matériel	1	2.5
14	Incident technique	1	2
14	Total	2	4.5
2	Grève	1	1.5
2	Incident technique	1	2
2	Total	2	3.5
3	Incident technique	1	2
3	Malaise voyageur	2	3
3	Total	3	5
3bis	Incident technique	1	2
3bis	Total	1	2
4	Incident matériel	1	2.5
4	Incident technique	1	2
4	Total	2	4.5
5	Incident technique	1	2
5	Malaise voyageur	1	3
5	Régulation	1	1
5	Total	3	6
6	Incident technique	3	6
6	Total	3	6
7	Grève	1	1.5
7	Incident technique	1	2
7	Total	2	3.5
7bis	Incident technique	1	2
7bis	Total	1	2
8	Incident technique	1	2
8	Malaise voyageur	1	2
8	Total	2	4
9	Incident matériel	1	3.5
9	Incident technique	1	2
9	Total	2	5.5
RER A	Incident technique	1	2

RER A	Régulation	2	5
RER A	Total	3	7
RER B	Incident technique	2	3
RER B	Total	2	3
RER C	Grève	1	2
RER C	Incident technique	1	2
RER C	Total	2	4
RER D	Incident technique	1	2
RER D	Malaise voyageur	1	2.5
RER D	Total	2	4.5
RER E	Incident matériel	1	1.5
RER E	Incident technique	1	2
RER E	Total	2	3.5
Total	Total	45	90

Script SQL de création des relations de la base de données :

```
CREATE TABLE TableTypeIncident
( TID serial PRIMARY KEY,
  nomIncident varchar(50) NOT NULL UNIQUE
);

CREATE TABLE Ligne
( numeroLigne varchar(5) PRIMARY KEY,
  terminus1 varchar(50) NOT NULL,
  terminus2 varchar(50) NOT NULL,
  nbIncidents integer DEFAULT 0,
  nbHeuresIncidents float DEFAULT 0,
  CONSTRAINT CK_Ligne CHECK (numeroLigne IN ('1','2','3','3bis','4','5','6','7','7bis','8','9','10','11','12','13','14','RER A','RER B','RER C','RER D','RER E')),
  CONSTRAINT UN_Ligne UNIQUE(terminus1,terminus2)
);

CREATE TABLE Incident
(ID serial PRIMARY KEY,
  dateI date NOT NULL,
  numeroLigne varchar(5) NOT NULL,
  typeIncident integer NOT NULL NOT NULL,
  nbIncidents integer NOT NULL DEFAULT 0,
  nbHeures float NOT NULL DEFAULT 0,
  CONSTRAINT UN_Incidents UNIQUE(dateI,numeroLigne,typeIncident),
  CONSTRAINT FK_typeIncident FOREIGN KEY(typeIncident) REFERENCES
TableTypeIncident(TID), CONSTRAINT FK_Incident FOREIGN KEY(numeroLigne) REFERENCES
Ligne(numeroLigne));

CREATE OR REPLACE FUNCTION FunctionTriggerLigne() RETURNS trigger AS
'DECLARE
BEGIN
  UPDATE Ligne SET nbIncidents = (SELECT SUM(nbIncidents) FROM Incident WHERE
numeroLigne=NEW.numeroLigne), nbHeuresIncidents = (SELECT SUM(nbHeures) FROM
Incident WHERE numeroLigne=NEW.numeroLigne) WHERE numeroLigne=NEW.numeroLigne;
  RETURN NEW;
END;'
LANGUAGE 'plpgsql';
```


Place du Maréchal de Lattre de
Tassigny
75775 Paris Cedex 16
dauphine.psl.eu

```
CREATE TRIGGER trigger_Ligne
AFTER INSERT ON Incident
FOR EACH ROW
EXECUTE PROCEDURE FunctionTriggerLigne();

INSERT INTO TableTypeIncident (nomIncident) VALUES
('Incident technique'),
('Grève'),
('Malaise voyageur'),
('Incident matériel'),
('Régulation'),
('Personnes sur les voies'),
('Mesure de sécurité'),
('Panne de train');

INSERT INTO Ligne (numeroLigne, terminus1, terminus2, nbIncidents,
nbHeuresIncidents) VALUES
('1', 'La Défense', 'Château de Vincennes', 5, 12.5),
('2', 'Porte Dauphine', 'Nation', 3, 8.0),
('3', 'Pont de Levallois', 'Gallieni', 4, 10.0),
('3bis', 'Gambetta', 'Porte des Lilas', 1, 2.0),
('4', 'Porte de Clignancourt', 'Mairie de Montrouge', 6, 15.0),
('5', 'Place d Italie', 'Bobigny - Pablo Picasso', 2, 5.0),
('6', 'Charles de Gaulle - Étoile', 'Nation', 4, 11.0),
('7', 'La Courneuve - 8 Mai 1945', 'Villejuif - Louis Aragon', 3, 7.5),
('7bis', 'Louis Blanc', 'Pré Saint-Gervais', 1, 3.0),
('8', 'Balard', 'Créteil - Pointe du Lac', 2, 4.5),
('9', 'Pont de Sèvres', 'Mairie de Montreuil', 3, 7.0),
('10', 'Boulogne - Pont de Saint-Cloud', 'Gare d Austerlitz', 1, 2.5),
('11', 'Châtelet', 'Mairie des Lilas', 2, 5.5),
('12', 'Front Populaire', 'Mairie d Issy', 3, 8.0),
('13', 'Saint-Denis - Université', 'Les Courtilles', 4, 9.0),
('14', 'Saint-Lazare', 'Olympiades', 2, 5.0),
('RER A', 'Saint-Germain-en-Laye', 'Boissy-Saint-Léger', 5, 12.0),
('RER B', 'Robinson', 'Saint-Rémy-lès-Chevreuse', 4, 10.0),
('RER C', 'Pontoise', 'Massy - Palaiseau', 3, 7.5),
('RER D', 'Creil', 'Melun', 4, 11.0),
('RER E', 'Haussmann - Saint-Lazare', 'Tournan', 2, 6.0);

INSERT INTO Incident (dateI, numeroLigne, typeIncident, nbIncidents, nbHeures)
VALUES ('2024-01-01', '1', 1, 1, 2.0), ('2024-01-02', '2', 2, 1, 1.5), ('2024-01-03', '3', 3, 2, 3.0), ('2024-01-04', '4', 4, 1, 2.5), ('2024-01-05', '5', 5, 1, 1.0), ('2024-01-06', '6', 1, 2, 4.0), ('2024-01-07', '7', 2, 1, 1.5), ('2024-01-08', '8', 3, 1, 2.0), ('2024-01-09', '9', 4, 1, 3.5), ('2024-01-10', '10', 5, 1, 1.0), ('2024-01-11', '11', 1, 1, 2.0), ('2024-01-12', '12', 2, 2, 4.5), ('2024-01-13', '13', 3, 1, 3.0), ('2024-01-14', '14', 4, 1, 2.5), ('2024-01-15', 'RER A', 5, 2, 5.0), ('2024-01-16', 'RER B', 1, 1, 1.0), ('2024-01-17', 'RER C', 2, 1, 2.0), ('2024-01-18', 'RER D', 3, 1, 2.5), ('2024-01-19', 'RER E', 4, 1, 1.5), ('2024-01-18', '5', 3, 1, 3.0), ('2024-01-17', '10', 3, 1, 1.0), ('2024-01-02', '2', 1, 1, 2.0), ('2024-01-02', '3', 1, 1, 2.0), ('2024-01-03', '3bis', 1, 1, 2.0), ('2024-01-04', '4', 1, 1, 2.0), ('2024-01-03', '5', 1, 1, 2.0), ('2024-01-01', '6', 1, 1, 2.0), ('2024-01-01', '7', 1, 1, 2.0), ('2024-01-02', '7bis', 1, 1, 2.0), ('2024-01-04', '8', 1, 1, 2.0), ('2024-01-01', '9', 1, 1, 2.0), ('2024-01-02', '10', 1, 1, 2.0), ('2024-01-04', '11', 1, 1, 2.0), ('2024-01-02', '12', 1, 1, 2.0), ('2024-01-03', '13', 1, 1, 2.0), ('2024-01-01', '14', 1, 1, 2.0), ('2024-01-01', 'RER A', 1, 1, 2.0), ('2024-01-01', 'RER B', 1, 1, 2.0), ('2024-01-04', 'RER C', 1, 1, 2.0), ('2024-01-04', 'RER D', 1, 1, 2.0), ('2024-01-04', 'RER E', 1, 1, 2.0);
```