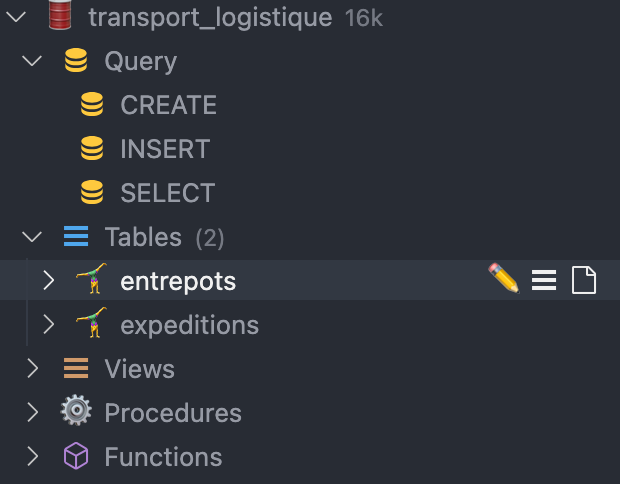
Rapport de TP : Base de données de transport

La majorité des données a été remplie à l'aide d'une extension Visual Studio Code.

**Création de la base de données**

* Créez une base de données nommée "**transport\_logistique**".

CREATE DATABASE transport\_logistique;



* Créez une table "**entrepots**" :

CREATE TABLE entrepots (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

nom\_entrepot VARCHAR(255) NOT NULL,

adresse VARCHAR(255) NOT NULL,

ville VARCHAR(255) NOT NULL,

pays VARCHAR(255) NOT NULL

);



* Créez une table "**expeditions**"

CREATE TABLE expeditions (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

date\_expedition DATE NOT NULL,

id\_entrepot\_source INT UNSIGNED NOT NULL,

id\_entrepot\_destination INT UNSIGNED NOT NULL,

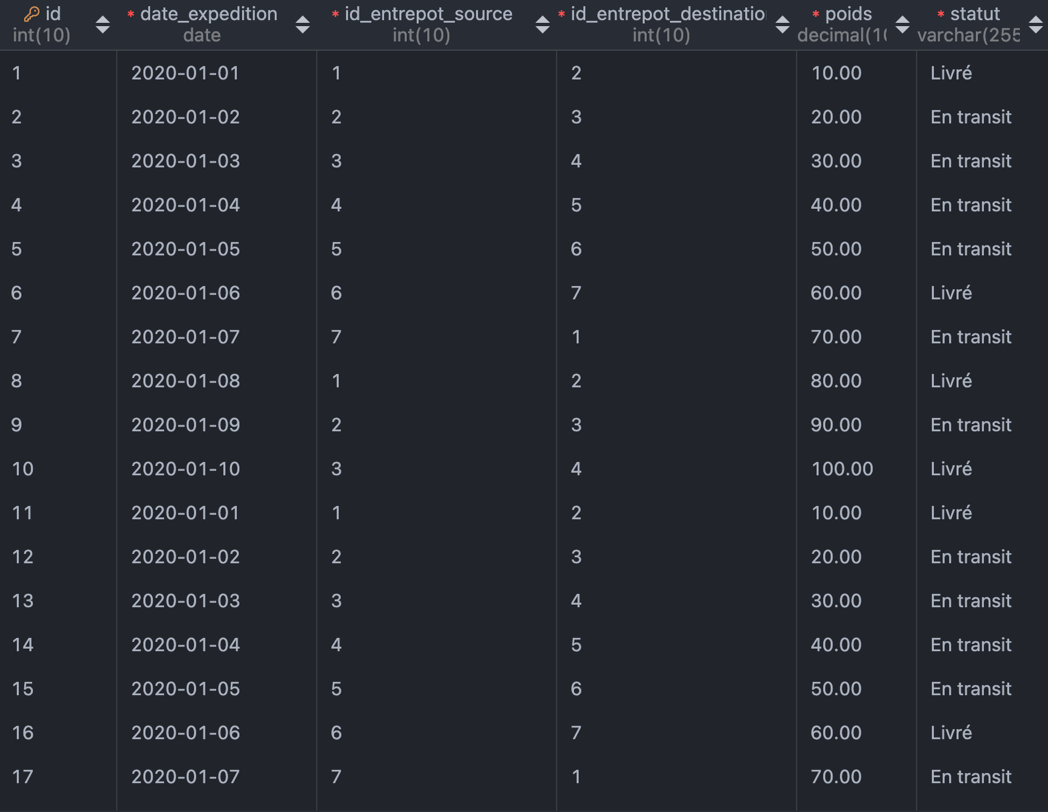
poids DECIMAL(10,2) NOT NULL,

statut VARCHAR(255) NOT NULL,

FOREIGN KEY (id\_entrepot\_source) REFERENCES entrepots(id),

FOREIGN KEY (id\_entrepot\_destination) REFERENCES entrepots(id)

);



**Ajout de données**

* Ajoutez 5 entrepôts dans différentes villes et pays.

INSERT INTO entrepots (nom\_entrepot, adresse, ville, pays)

VALUES ('Entrepot 1', '1 rue de la paix', 'Paris', 'France'),

('Entrepot 2', '2 rue de la paix', 'Madrid', 'Espagne'),

('Entrepot 3', '3 rue de la paix', 'Londres', 'Angleterre'),

('Entrepot 4', '4 rue de la paix', 'Berlin', 'Allemagne'),

('Entrepot 5', '5 rue de la paix', 'Rome', 'Italie'),



* Ajoutez 10 expéditions de différents poids, en provenance de différents entrepôts et à destination de différents entrepôts. Assurez-vous que certaines expéditions ont été livrées et d'autres sont en transit.

INSERT INTO expeditions (date\_expedition, id\_entrepot\_source, id\_entrepot\_destination, poids, statut)

VALUES ('2020-01-01', 1, 2, 10, 'Livré'),

('2020-01-02', 2, 3, 20, 'En transit'),

('2020-01-03', 3, 4, 30, 'En transit'),

('2020-01-04', 4, 5, 40, 'En transit'),

('2020-01-05', 5, 6, 50, 'En transit'),

('2020-01-06', 6, 7, 60, 'Livré'),

('2020-01-07', 7, 1, 70, 'En transit'),

('2020-01-08', 1, 2, 80, 'Livré'),

('2020-01-09', 2, 3, 90, 'En transit'),

('2020-01-10', 3, 4, 100, 'Livré');



**Requêtes de base**

* Affichez tous les entrepôts.

SELECT \* FROM entrepots;



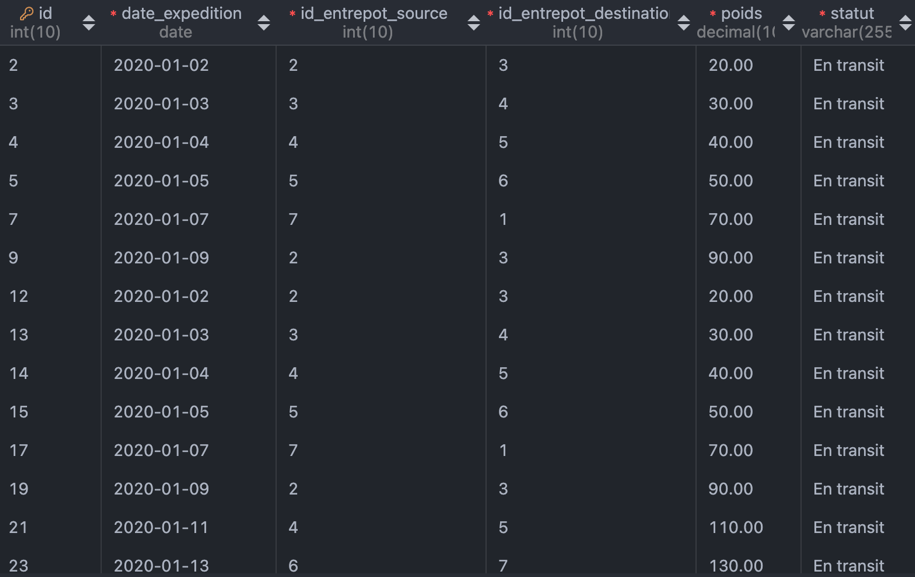
* Affichez toutes les expéditions.

SELECT \* FROM expeditions;



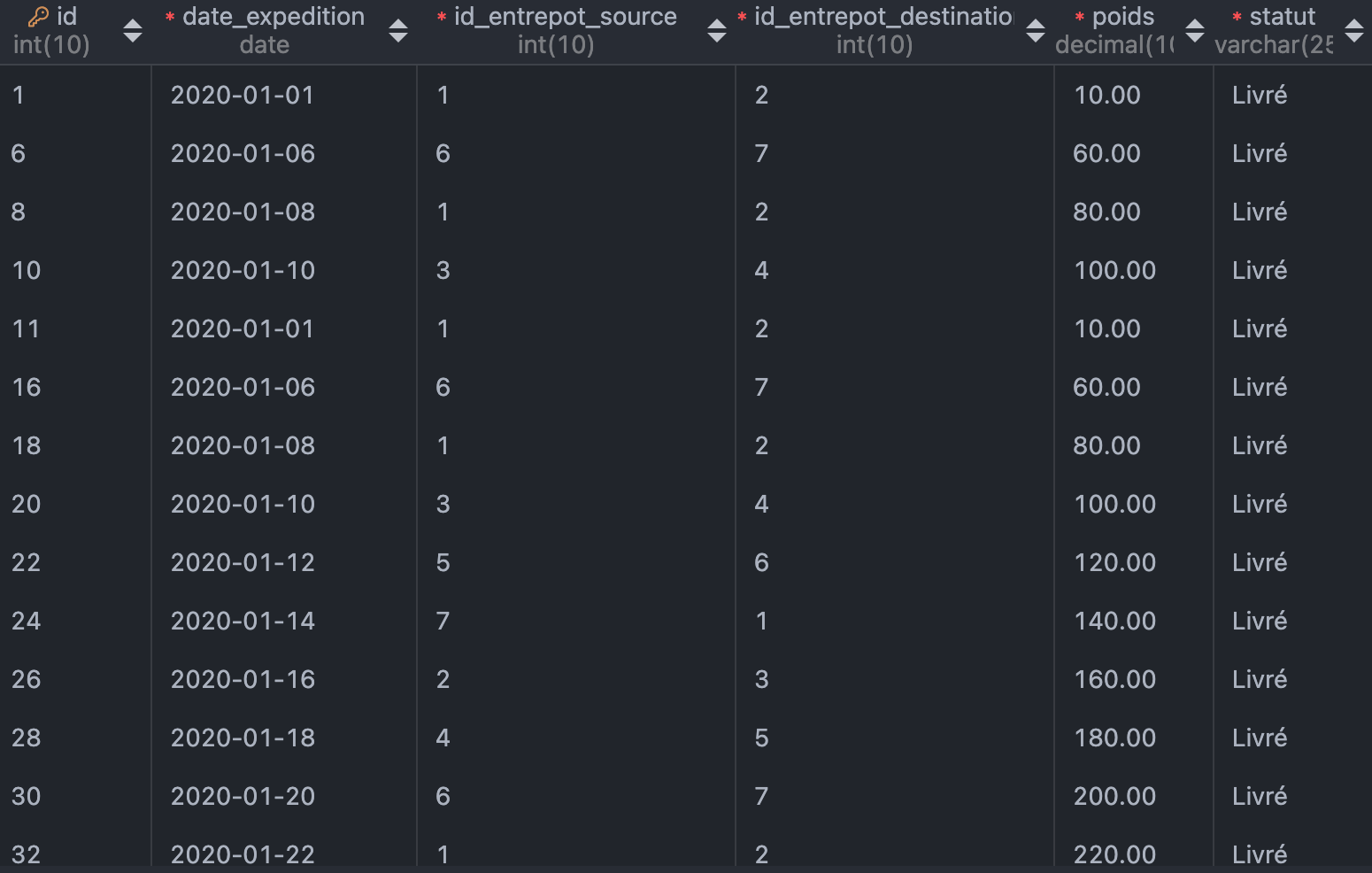
* Affichez toutes les expéditions en transit.

SELECT \* FROM expeditions WHERE statut = 'En transit';

****

* Affichez toutes les expéditions livrées.

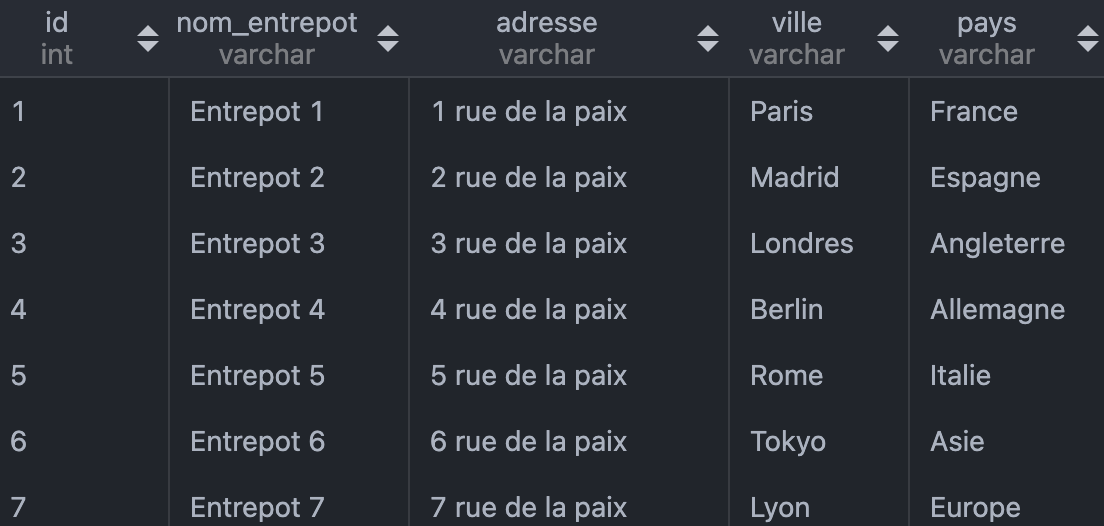
SELECT \* FROM expeditions WHERE statut = 'Livré';

****

**Requêtes avancées**

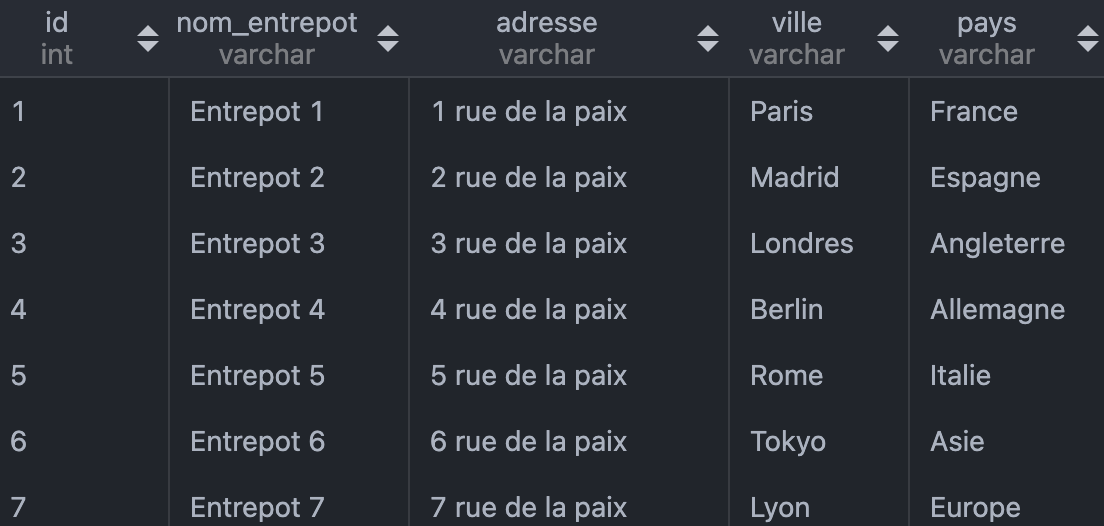
* Affichez les entrepôts qui ont envoyé au moins une expédition en transit.

SELECT \* FROM entrepots WHERE id IN (SELECT id\_entrepot\_source FROM expeditions WHERE statut = 'En transit');



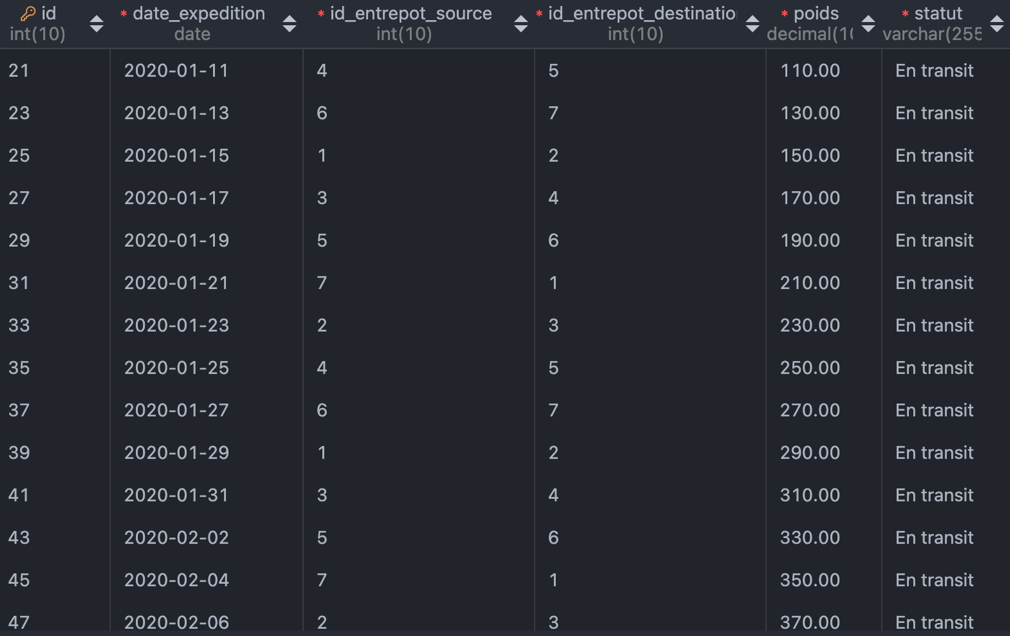
* Affichez les entrepôts qui ont reçu au moins une expédition en transit.

SELECT \* FROM entrepots WHERE id IN (SELECT id\_entrepot\_destination FROM expeditions WHERE statut = 'En transit');



* Affichez les expéditions qui ont un poids supérieur à 100 kg et qui sont en transit.

SELECT \* FROM expeditions WHERE poids > 100 AND statut = 'En transit';



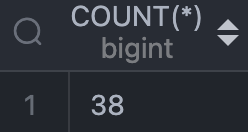
* Affichez le nombre d'expéditions envoyées par chaque entrepôt.

SELECT nom\_entrepot, COUNT(\*) FROM expeditions INNER JOIN entrepots ON expeditions.id\_entrepot\_source = entrepots.id GROUP BY id\_entrepot\_source;



* Affichez le nombre total d'expéditions en transit.

SELECT COUNT(\*) FROM expeditions WHERE statut = 'En transit';



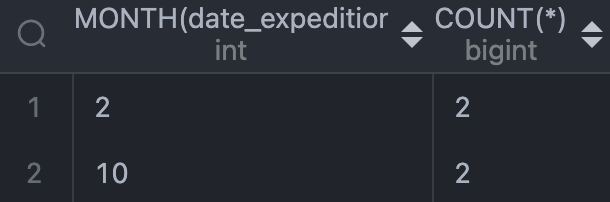
* Affichez le nombre total d'expéditions livrées.

SELECT COUNT(\*) FROM expeditions WHERE statut = 'Livré';



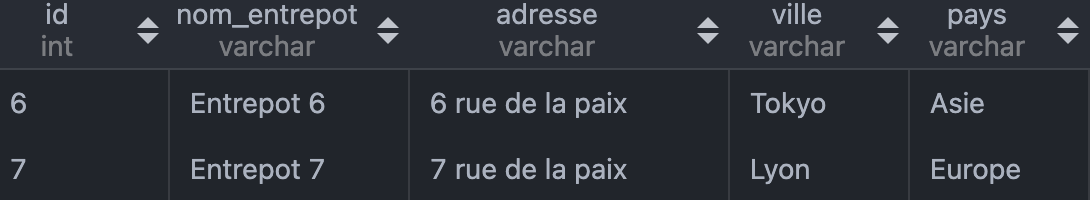
* Affichez le nombre total d'expéditions pour chaque mois de l'année en cours.

SELECT MONTH(date\_expedition), COUNT(\*) FROM expeditions WHERE YEAR(date\_expedition) = YEAR(CURDATE()) GROUP BY MONTH(date\_expedition);



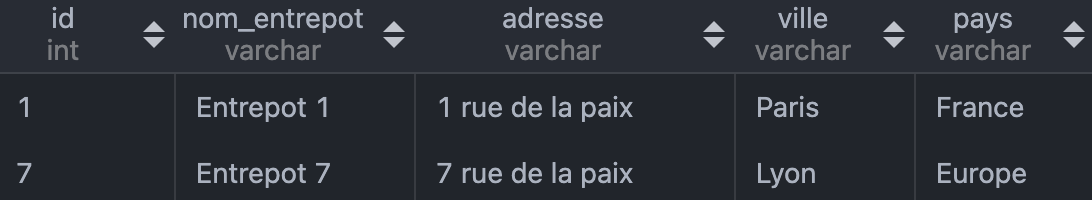
* Affichez les entrepôts qui ont envoyé des expéditions au cours des 30 derniers jours.

SELECT \* FROM entrepots WHERE id IN (SELECT id\_entrepot\_source FROM expeditions WHERE date\_expedition > DATE\_SUB(CURDATE(), INTERVAL 30 DAY));



* Affichez les entrepôts qui ont reçu des expéditions au cours des 30 derniers jours.

SELECT \* FROM entrepots WHERE id IN (SELECT id\_entrepot\_destination FROM expeditions WHERE date\_expedition > DATE\_SUB(CURDATE(), INTERVAL 30 DAY));



* Affichez les expéditions qui ont été livrées dans un délai de moins de 5 jours ouvrables.

Ajout de la date\_livraison dans la table expedition :

ALTER TABLE expeditions ADD date\_livraison DATE;

Ajout de dates de livraison aléatoire pour chaque expeditions :

UPDATE expeditions SET date\_livraison = DATE\_ADD(date\_expedition, INTERVAL FLOOR(RAND() \* 10) DAY);

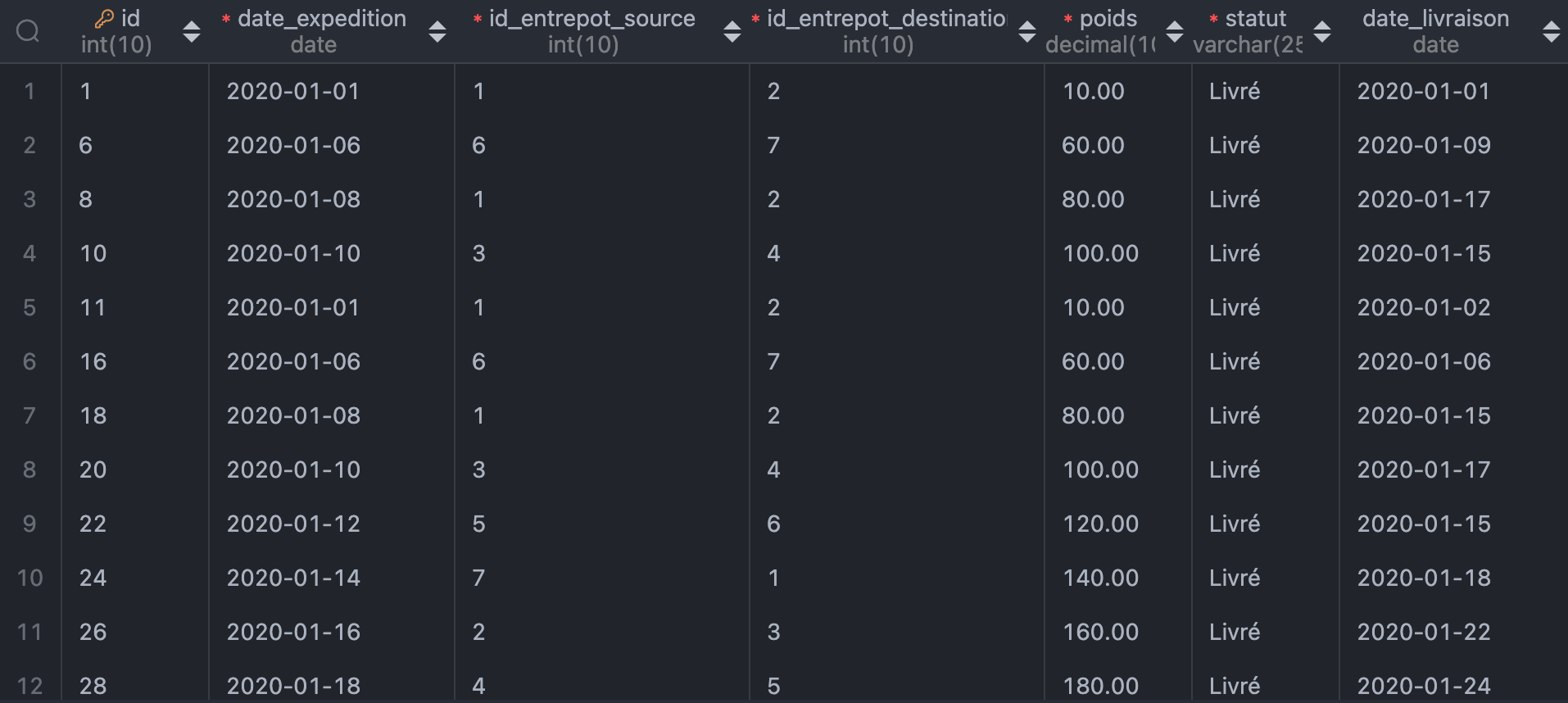
« RAND() » génère un nombre décimal aléatoire entre 0 et 1.

« \* 10 » multiplie ce nombre par 10, ce qui donne un nombre décimal aléatoire entre 0 et 10.

« FLOOR(RAND() \* 10) » arrondit ce nombre décimal aléatoire à l'entier inférieur le plus proche, ce qui donne un nombre entier aléatoire entre 0 et 9.

➤ En résumé, cette commande mettra à jour la colonne "date\_livraison" de la table "expeditions" en ajoutant un nombre aléatoire de jours (entre 0 et 9 jours) à la date d'expédition existante pour chaque enregistrement. Cela simule un délai de livraison variable pour chaque expédition.

SELECT \* FROM expeditions WHERE statut = 'Livré' AND DATEDIFF(date\_expedition, date\_livraison) < 5;



**Requêtes complexes**

* Affichez les expéditions en transit qui ont été initiées par un entrepôt situé en Europe et à destination d'un entrepôt situé en Asie.

Creation de la table pays contenant le nom du pays et le continent en plus de l’id :

CREATE TABLE pays (

id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

nom\_pays VARCHAR(255) NOT NULL,

continent VARCHAR(255) NOT NULL

);

Suppression de la colonne pays dans entrepots :

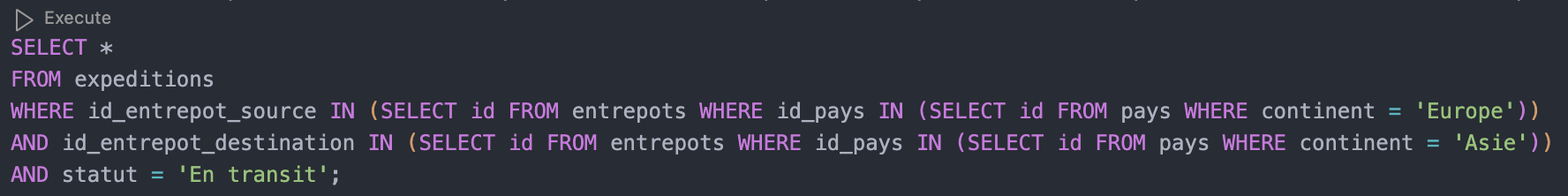
ALTER TABLE entrepots DROP COLUMN pays;

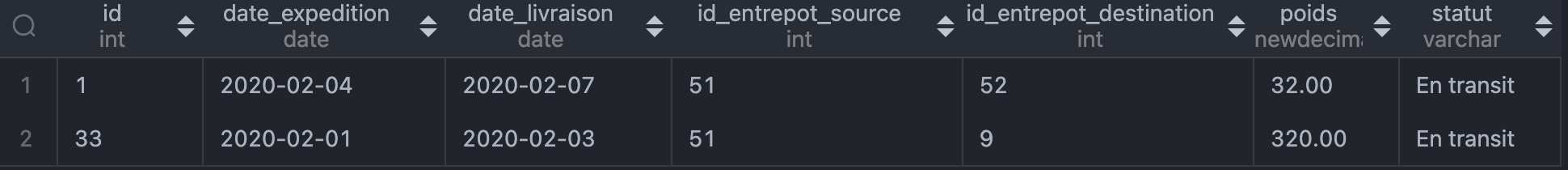
Ajout d’une nouvelle colonne id\_pays faisant référence id de pays :

ALTER TABLE entrepots ADD id\_pays INT UNSIGNED NOT NULL;

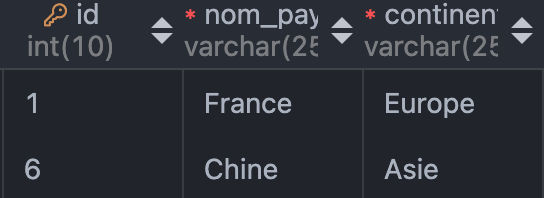
ALTER TABLE entrepots ADD FOREIGN KEY (id\_pays) REFERENCES pays(id);

Ajout de données dans la table pays

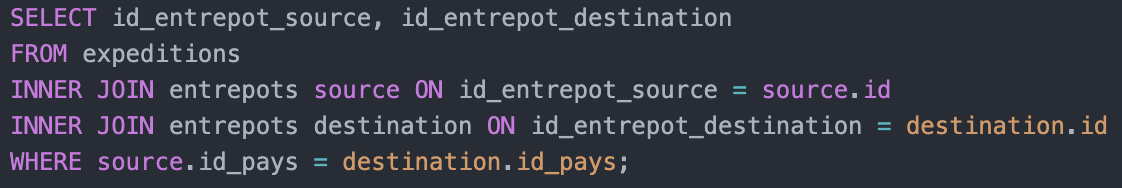




Vérification dans la table entrepôt puis dans la table pays si ils appartient bien au même continent :



* Affichez les entrepôts qui ont envoyé des expéditions à destination d'un entrepôt situé dans le même pays.

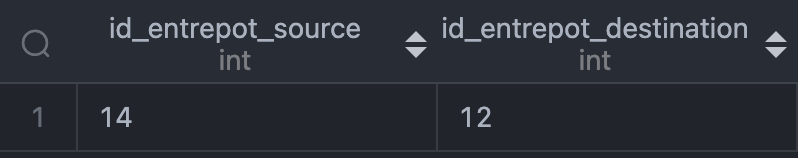


Sélectionne les colonnes id\_entrepot\_source et id\_entrepot\_destination de la table "expeditions".

Utilise des jointures internes pour lier la table "expeditions" aux tables "entrepots" deux fois, une fois pour l'entrepôt source et une fois pour l'entrepôt de destination.

Filtre les résultats pour n'inclure que les expéditions où l'entrepôt source et l'entrepôt de destination se trouvent dans le même pays, c'est-à-dire où source.id\_pays est égal à destination.id\_pays.

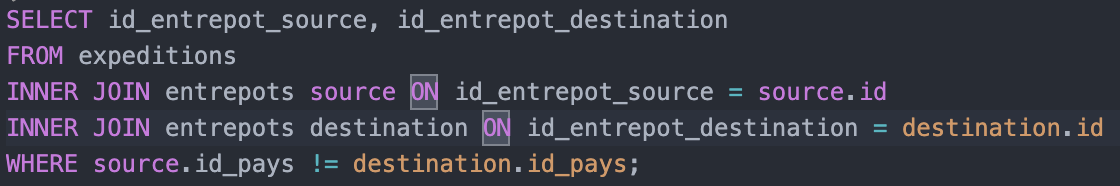
Cela renverra les paires d'entrepôts qui ont envoyé des expéditions à destination d'un entrepôt situé dans le même pays.



Vérification dans la table entrepôt :

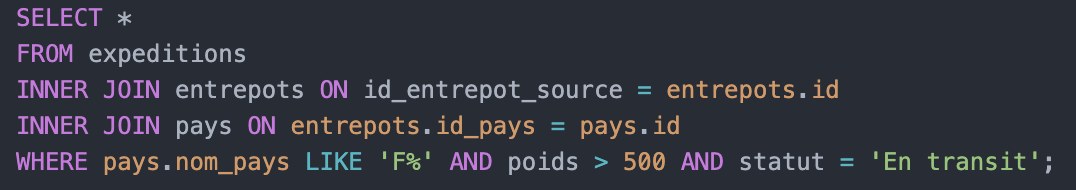


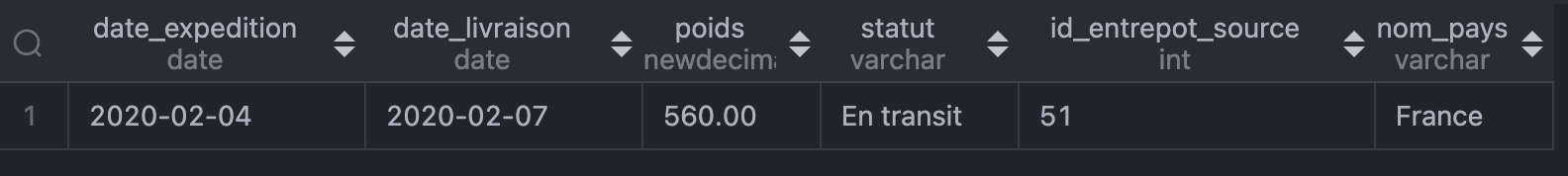
* Affichez les entrepôts qui ont envoyé des expéditions à destination d'un entrepôt situé dans un pays différent



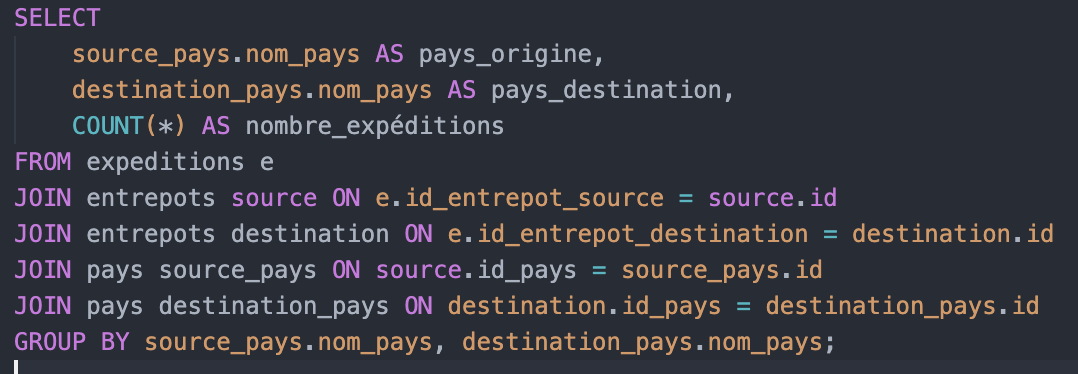


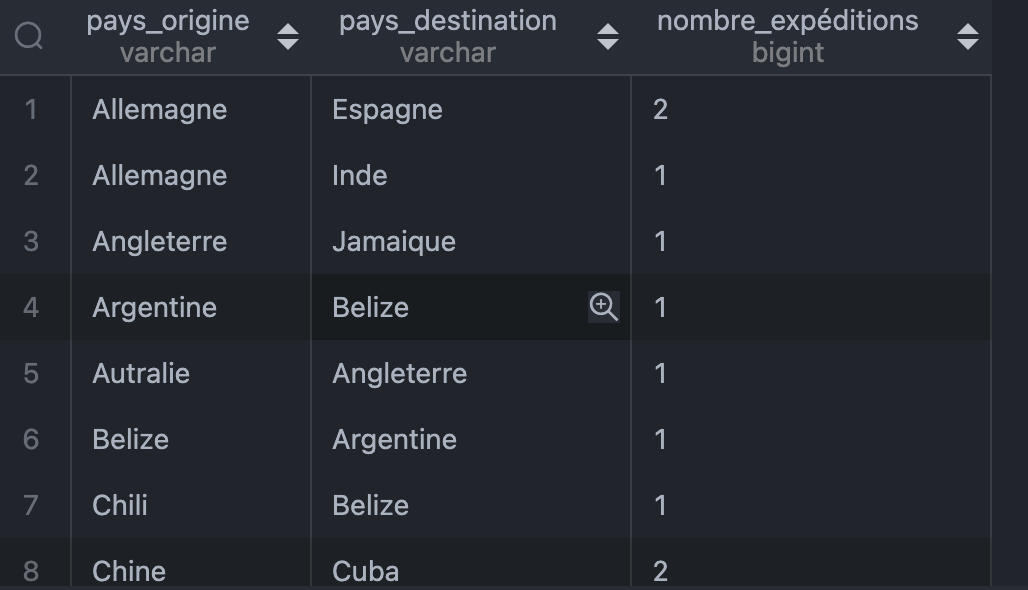
* Affichez les expéditions en transit qui ont été initiées par un entrepôt situé dans un pays dont le nom commence par la lettre "F" et qui pèsent plus de 500 kg.





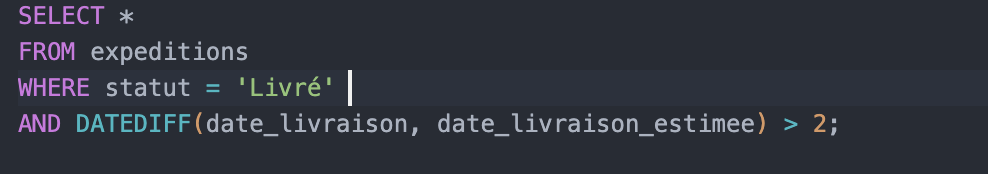
* Affichez le nombre total d'expéditions pour chaque combinaison de pays d'origine et de destination.





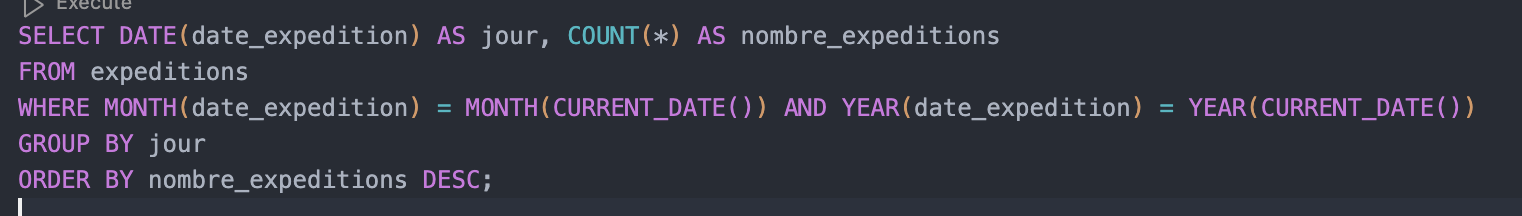
* Affichez les expéditions qui ont été livrées avec un retard de plus de 2 jours ouvrables.

Ajout d’un champ date\_livraison\_estimee dans la table expedition





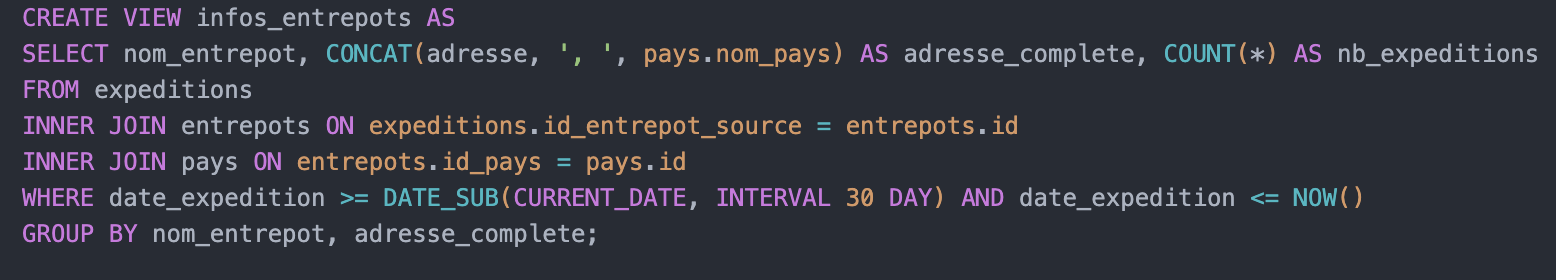
* Affichez le nombre total d'expéditions pour chaque jour du mois en cours, trié par ordre décroissant.

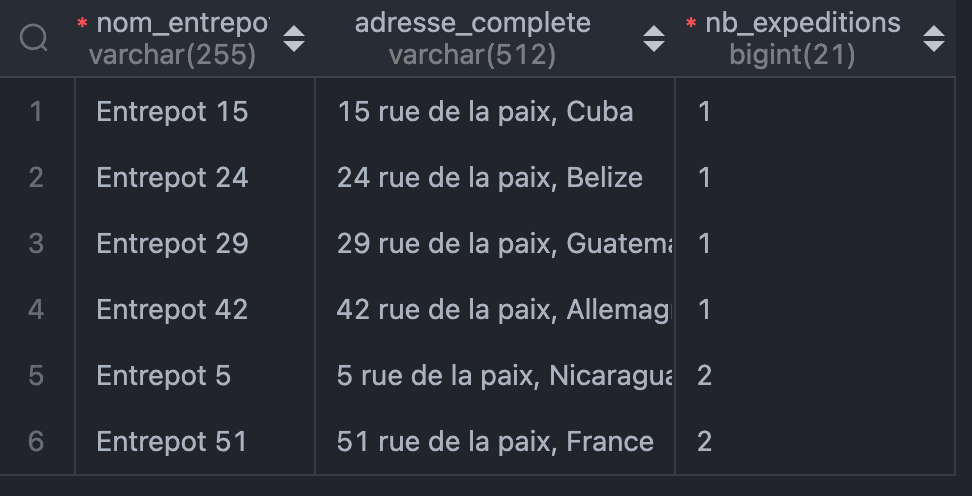




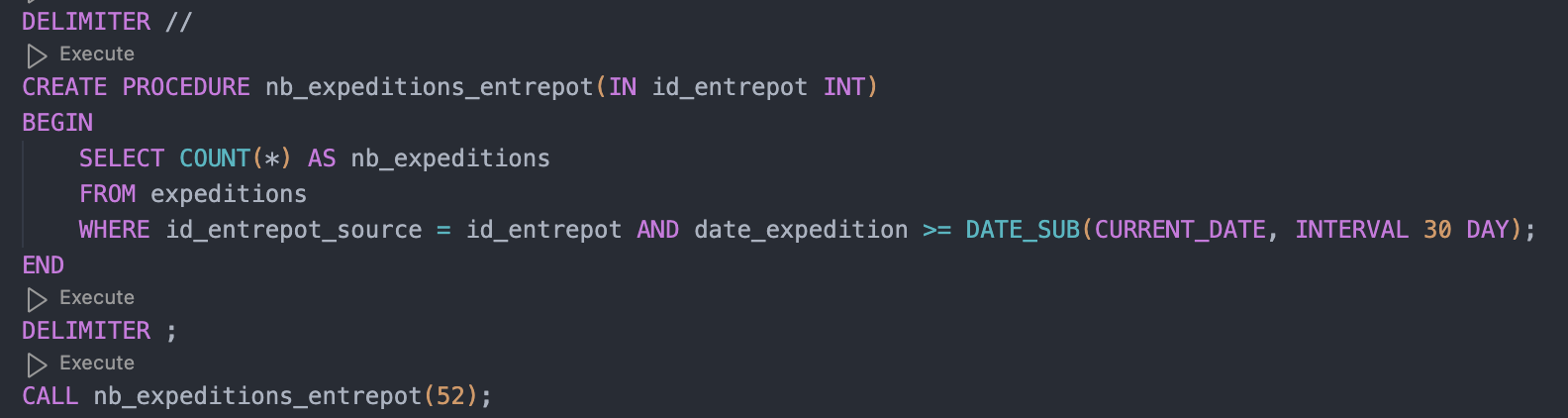
**T-SQL**

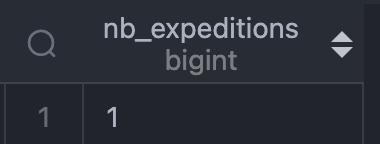
* Créez une vue qui affiche les informations suivantes pour chaque entrepôt : nom de l'entrepôt, adresse complète, nombre d'expéditions envoyées au cours des 30 derniers jours.



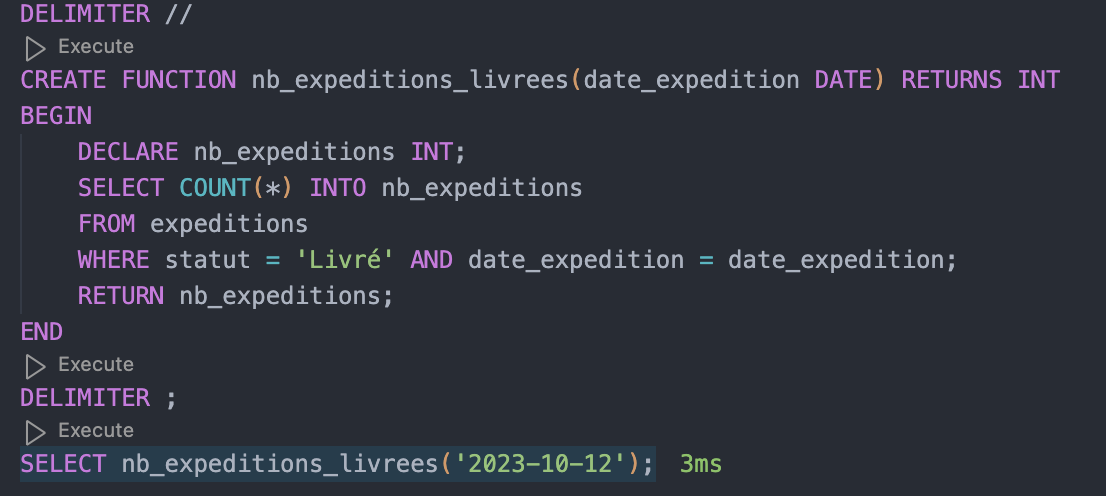


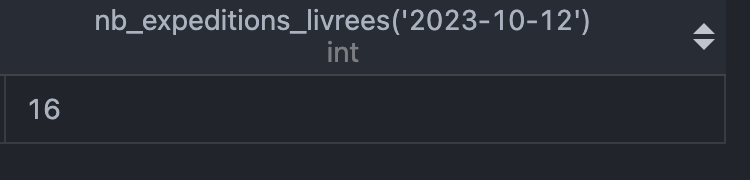
* Créez une procédure stockée qui prend en entrée l'ID d'un entrepôt et renvoie le nombre total d'expéditions envoyées par cet entrepôt au cours du dernier mois.





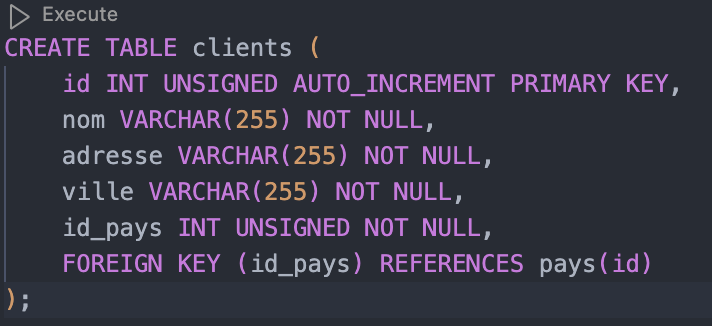
* Créez une fonction qui prend en entrée une date et renvoie le nombre total d'expéditions livrées ce jour-là.

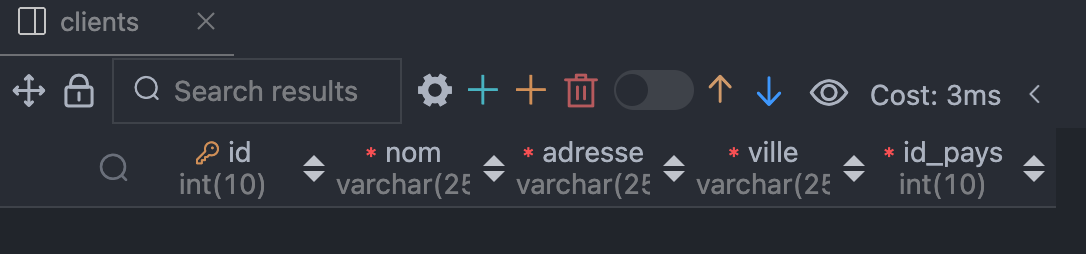




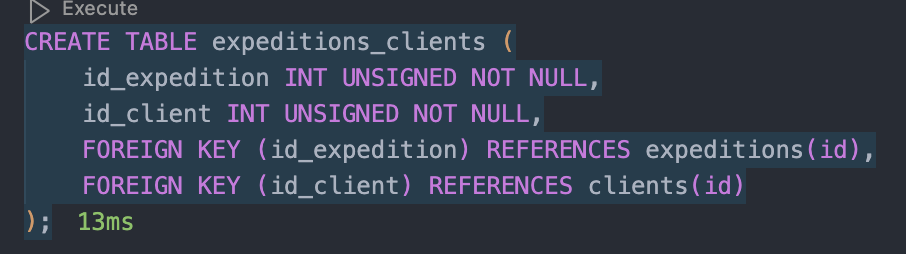
**Bonus**

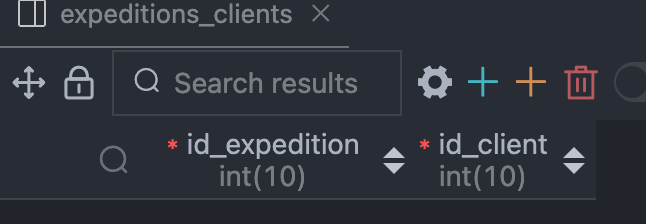
* Ajoutez une table "clients"



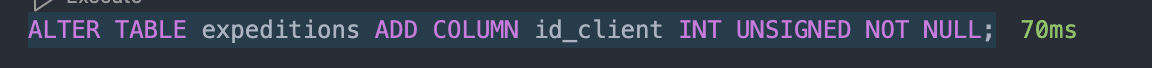


* Ajoutez une table de jointure "expeditions\_clients" contenant les colonnes suivantes :

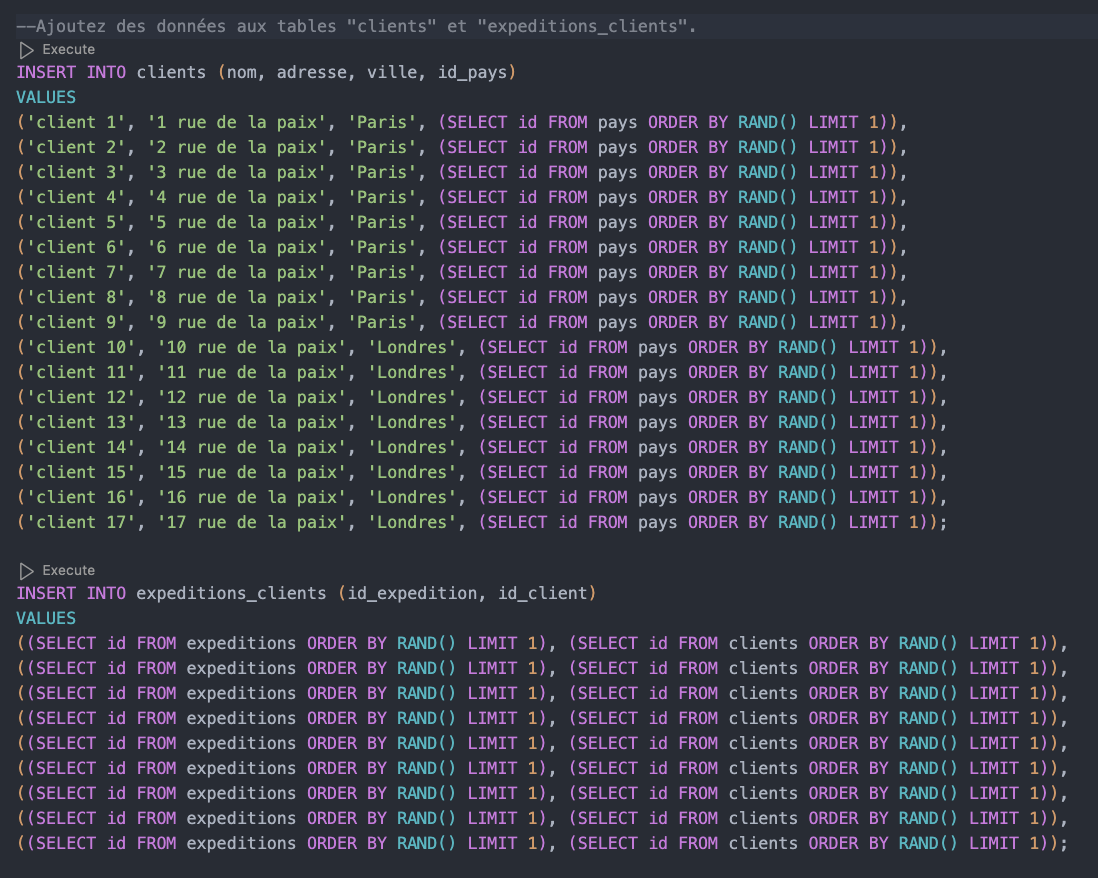




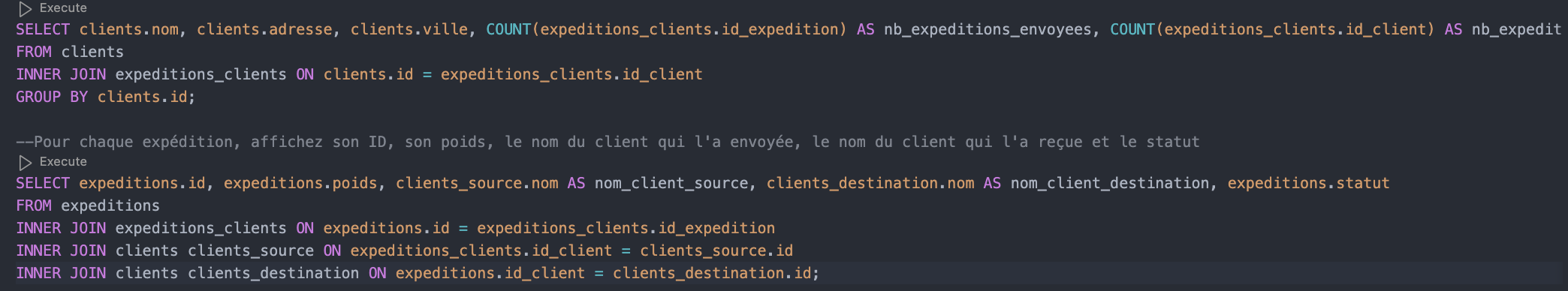
* Modifiez la table "expeditions" pour y ajouter une colonne "id\_client" (entier, clé étrangère faisant référence à la table "clients").



* Ajoutez des données aux tables "clients" et "expeditions\_clients".

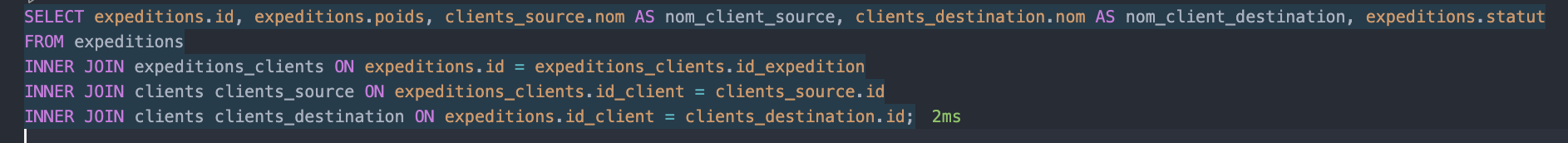


* Écrivez des requêtes pour extraire les informations suivantes :
  + Pour chaque client, affichez son nom, son adresse complète, le nombre total d'expéditions qu'il a envoyées et le nombre total d'expéditions qu'il a reçues.





* + Pour chaque expédition, affichez son ID, son poids, le nom du client qui l'a envoyée, le nom du client qui l'a reçue et le statut



▲ Fonctionne pas

Fin de TP Ven.3 nov. A 8:26PM