

Introduction, 2eme partie.

How to access stored data:

Ici, nous verrons comment nous avons accédé aux données stockées dans la base de données spatiales et où vivent nos données. Lorsque vous travaillez avec des données basées sur des fichiers, telles qu'un fichier de forme avec une couche d'entités provenant d'une base de données, nous pouvons trouver ces données en y accédant via notre système de fichiers informatique.

Nous pouvons le déplacer, le renommer ou l'envoyer par e-mail à un collègue, etc. Mais lorsque nous travaillons avec des données spatiales avec une base de données spatiales, les données ne sont pas accessibles via le système de fichiers de l'ordinateur.

Où réside une base de données spatiale ?

Les données spatiales se trouvent quelque part sur un fichier sur votre disque dur, mais vous ne pouvez pas y accéder de manière significative autrement qu'en utilisant le logiciel de base de données. Ce qui rend difficile le partage, mais en même temps, il présente des avantages importants, notamment en matière de sécurité.

Bien que les fichiers non spatiaux liés à une carte puissent être modifiés, il serait beaucoup plus difficile de le faire avec une base de données spéciale car vous n'accédez pas directement aux données et ne savez probablement pas où elles se trouvent et ne les reconnaîtrez pas si vous trouviez parce que c'est juste un tas de fichiers avec des nombres impairs pour les noms et en plus vous avez besoin d'identifiants de connexion pour stocker les données. Cela rend vos données à la fois pirates et infailibles.

Pour accéder à la connexion à la base de données, vous avez besoin d'une chaîne contenant 5 informations :

(i) une adresse IP ou quelque chose comme un emplacement réseau nommé ou un domaine Internet et le nom d'un ordinateur sur lequel réside la base de données. Si le serveur qui vit localement sur votre ordinateur de bureau le même ordinateur où votre logiciel client écrit, le nom de l'hôte est localhost;

(ii) un numéro de port, rappelez-vous qu'un serveur est vraiment un logiciel qui attend que quelqu'un fasse une demande dans le port. C'est juste un numéro qui identifie le logiciel serveur que la requête va remplir. Dans postgresSQL, le port par défaut est 5432;

vous devez fournir vos identifiants, **(iii) un nom d'utilisateur** et un **(iv) mot de passe** pour se connecter au logiciel de base de données car rappelez-vous que les bases de données offrent beaucoup de sécurité en obligeant l'utilisateur à se connecter et chaque utilisateur peut également être limité quant à ce qu'il peut faire avec la base de données;

et enfin, vous devez fournir **(v) le nom de la base de données** à laquelle vous souhaitez accéder, car une seule instance de PostgreSQL peut avoir plusieurs bases de données et vous ne pouvez accéder qu'à une seule à la fois.

Et une fois que vous avez fourni ces informations au client, il a tout ce dont il a besoin pour établir une connexion à la base de données et vous pouvez accéder aux données via le logiciel client et faire tout ce que vous voulez tant que le client fournit cette fonctionnalité et vous avez les privilèges requis.

Qu'est-ce que SQL et pourquoi voudriez-vous l'utiliser ?

Structured Query Language, SQL, est une manière standardisée d'interagir avec une base de données. En fait, presque tout ce que vous pouvez faire avec une base de données peut être fait via une instruction de SQL. Nous n'écrivons pas toujours le code SQL directement, nous utilisons souvent une interface utilisateur graphique que nous utilisons pour interagir avec la base de données. Le langage SQL a commencé dans les années 70 et est devenu une norme ANSI officielle en 1986.

Il existe cependant de nombreuses variantes, seules les instructions de séquence les plus simples peuvent être exécutées textuellement sur différentes bases de données.

SQL est considéré comme un langage de programmation déclaratif, qui signifie essentiellement que vous dites à la base de données ce que vous voulez, puisque vous déterminez comment le faire. Ceci est différent d'un langage de programmation impératif comme Python, ou Javascript etc où vous devez dire à l'ordinateur exactement quoi faire, étape par étape. Ce qui signifie que SQL est facile à écrire pour les non-programmeurs.

Les sous-ensembles de SQL:

Le premier s'appelle **DQL, Data Query Language** et il nous permet d'interroger la base de données, donc de récupérer des informations de la base de données sous une forme utile ;

Le **DDL, Data Definition Language**, permet de créer de nouvelles tables et de définir la structure de la table, tous les différents champs, index, relations entre tables, etc. Il est principalement utilisé dans les phases de conception;

Le **DML, Data Manipulation Language**, nous permet d'ajouter de nouvelles données aux tables existantes et de modifier les valeurs dans les données.

Enfin, le **DCL, Data Control Language**, nous permet de définir des utilisateurs et de les affecter à des rôles et de créer des privilèges d'accès aux données pour les utilisateurs et les rôles. C'est ainsi que nous pouvons empêcher certains utilisateurs d'entrer ou de sortir de certaines tables ou même de certains champs.

Les Extensions du SQL:

PostgreSQL a des extensions qui permettent d'écrire des procédures dans au moins 10 langages de programmation différents, y compris C, Python, Javascript et R. Les extensions spatiales de SQL n'utilisent pas le SQL ANSI standard et sont donc moins susceptibles d'être transférables d'une base de données à une autre que standard et les instructions de suite sont composées d'expressions de causes et les clauses de prédicats font partie du langage SQL réel qui ne sont pas fonctionnellement différents de ce que l'on appelle souvent des mots-clés dans d'autres langages.

L'anatomie d'une instruction SQL:

```
SELECT nest_id, recentstat, ST_Distance(geom, home)
FROM raptor_nests
WHERE recentspec='RTHA'
ORDER BY recentstat, nest_id
```

Clauses:

SELECT, FROM, WHERE, ...

Expressions:

1+2, total*prix, ST_Buffer(geom,400)

Predicates:

True, False, NULL eg ST_Contains(geom_A, geom_B)

L'avantage de SQL est qu'il est basé sur du texte, il est lisible par l'homme, facilement stocké et réutilisé.