

Travaux pratiques : Remplissage d'un entrepôt de données à l'aide de PostgreSQL



Temps estimé nécessaire : 1 heure

Objectif du laboratoire :

Le laboratoire est conçu pour fournir une expérience pratique de la création et de la gestion d'une base de données de production à l'aide de PostgreSQL dans l'IDE Cloud IBM Skills Network Labs (SN Labs). Vous apprendrez à lancer une instance de serveur PostgreSQL, à utiliser l'interface utilisateur graphique (GUI) pgAdmin pour les opérations de base de données et à exécuter des tâches essentielles telles que la création d'une base de données, la conception de tables et le chargement de données. Le laboratoire se concentre sur la construction d'une base de données de base en guidant les apprenants tout au long du processus de configuration d'une base de données de « production » et de son remplissage avec des données suivant une conception de schéma en étoile.

Avantages de l'apprentissage en laboratoire :

Participer à ce laboratoire offre des avantages significatifs aux apprenants qui cherchent à approfondir leur compréhension des systèmes de gestion de bases de données, en particulier PostgreSQL. En travaillant dans le laboratoire, vous acquerez des compétences pratiques en SQL, en création de bases de données, en conception de tables et en manipulation de données, qui sont essentielles pour les rôles dans l'ingénierie des données, l'administration de bases de données et la science des données. L'approche pratique aide à consolider les connaissances des schémas de base de données et des requêtes SQL, améliorant ainsi la capacité de l'apprenant à gérer et analyser efficacement les données dans des scénarios réels. De plus, la familiarité avec des outils tels que pgAdmin et l'environnement Cloud IDE ajoute une expérience précieuse à votre ensemble de compétences, vous préparant à des projets et des tâches de base de données avancés.

Logiciel utilisé dans ce laboratoire

Pour réaliser ce laboratoire, vous utiliserez le service de base de données relationnelle [PostgreSQL Database](#) disponible dans le cadre d'IBM Skills Network Labs (SN Labs) Cloud IDE. SN Labs est un environnement de laboratoire virtuel utilisé dans ce cours.



Base de données utilisée dans ce laboratoire

La base de données de production est utilisée dans ce laboratoire.
La base de données de production contient :

- Client Dim
- Mois sombre
- Facturation

Objectifs

Dans ce laboratoire, vous allez :

- Créez une base de données et des tables liées à la production dans une instance PostgreSQL.
- Remplissez l'entrepôt de données de production en chargeant les tables à partir de scripts.

Structure du laboratoire

Dans ce laboratoire, vous effectuerez plusieurs tâches dans lesquelles vous apprendrez à créer des tables et à charger des données dans le service de base de données PostgreSQL à l'aide de l'outil d'interface utilisateur graphique (GUI) pgAdmin.

Données utilisées dans ce laboratoire

Voici les fichiers de données SQL utilisés dans ce laboratoire.

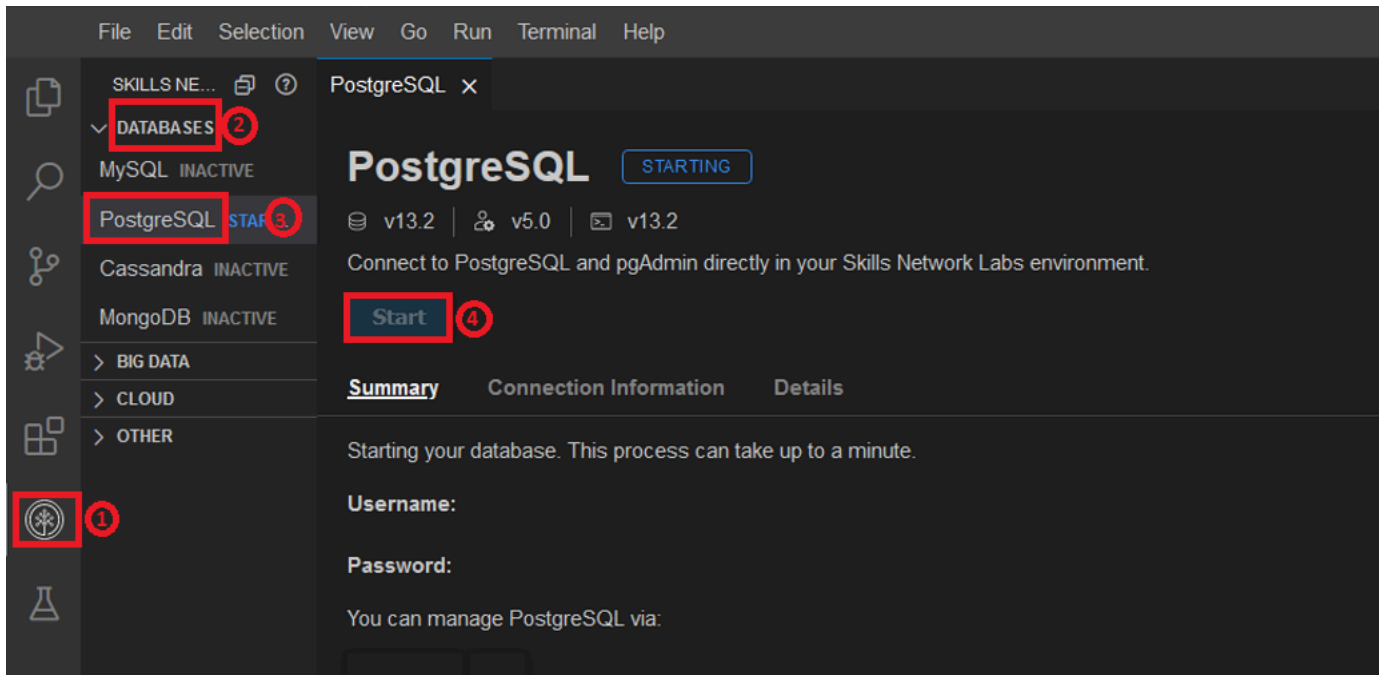
La base de données de production contient :

- [Client Dim](#)
- [Mois sombre](#)
- [Facturation](#)
- [Schéma en étoile](#)

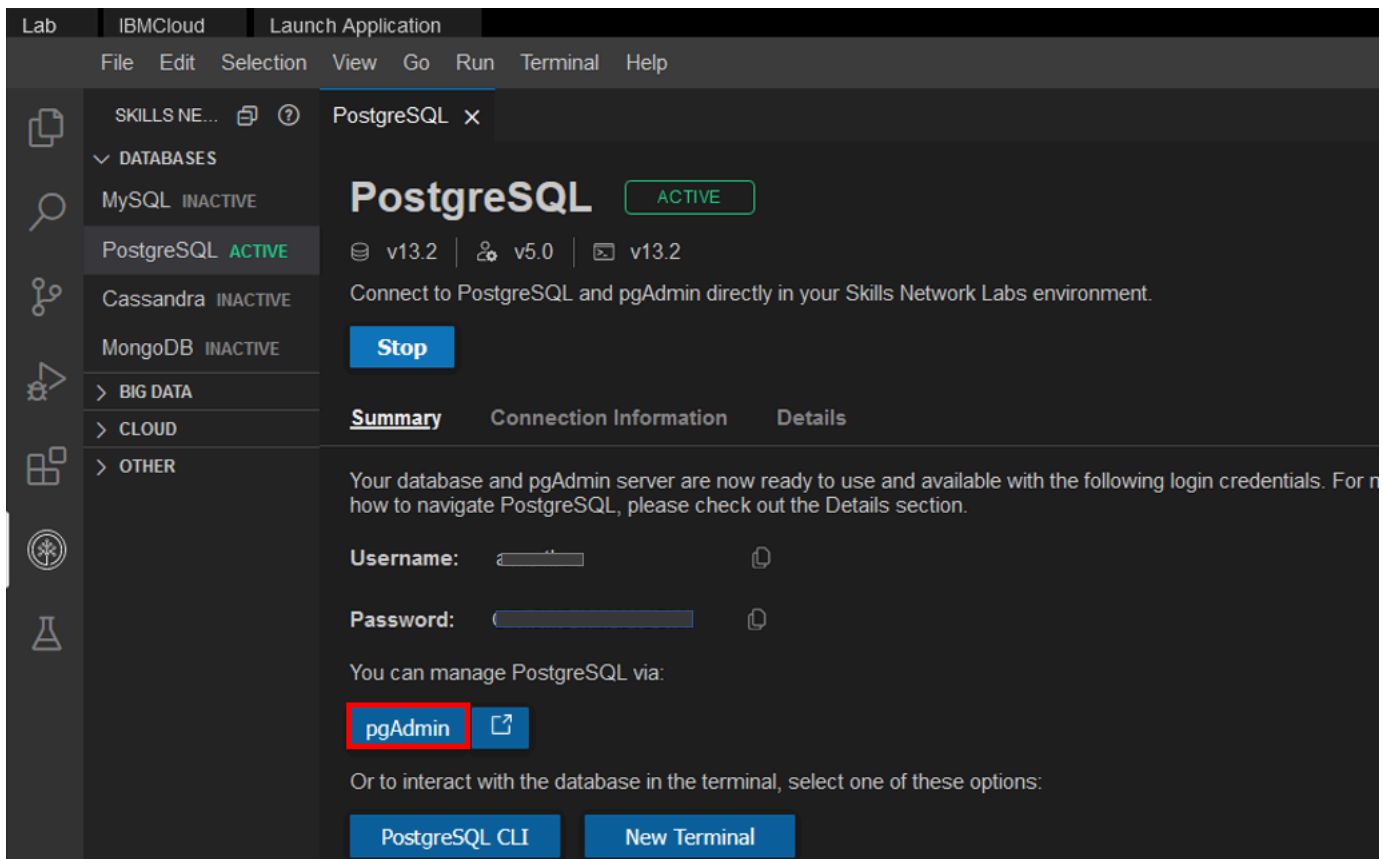
Tâche A : Créer une base de données

Tout d'abord, pour créer une base de données sur une instance de serveur PostgreSQL, vous devez d'abord lancer une instance de serveur PostgreSQL sur Cloud IDE et ouvrir l'interface utilisateur graphique pgAdmin.

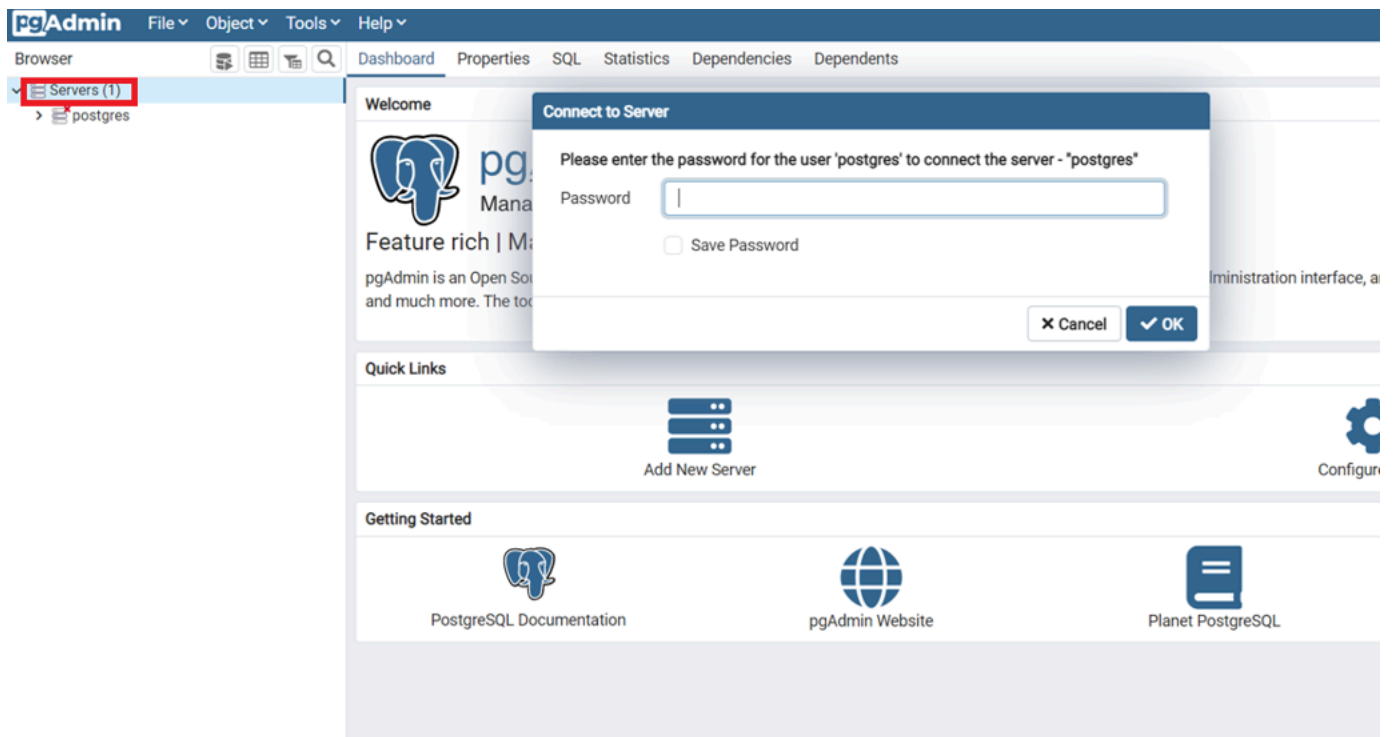
1. Cliquez sur le bouton d'extension Skills Network sur le côté gauche de la fenêtre.
2. Ouvrez le menu déroulant **BASES DE DONNÉES**.
3. Cliquez sur **PostgreSQL**
4. Cliquez sur le bouton **Démarrer**. PostgreSQL peut prendre quelques instants pour démarrer.



5. Ensuite, ouvrez l'interface utilisateur graphique pgAdmin en cliquant sur le bouton **pgAdmin** dans l'interface Cloud IDE.

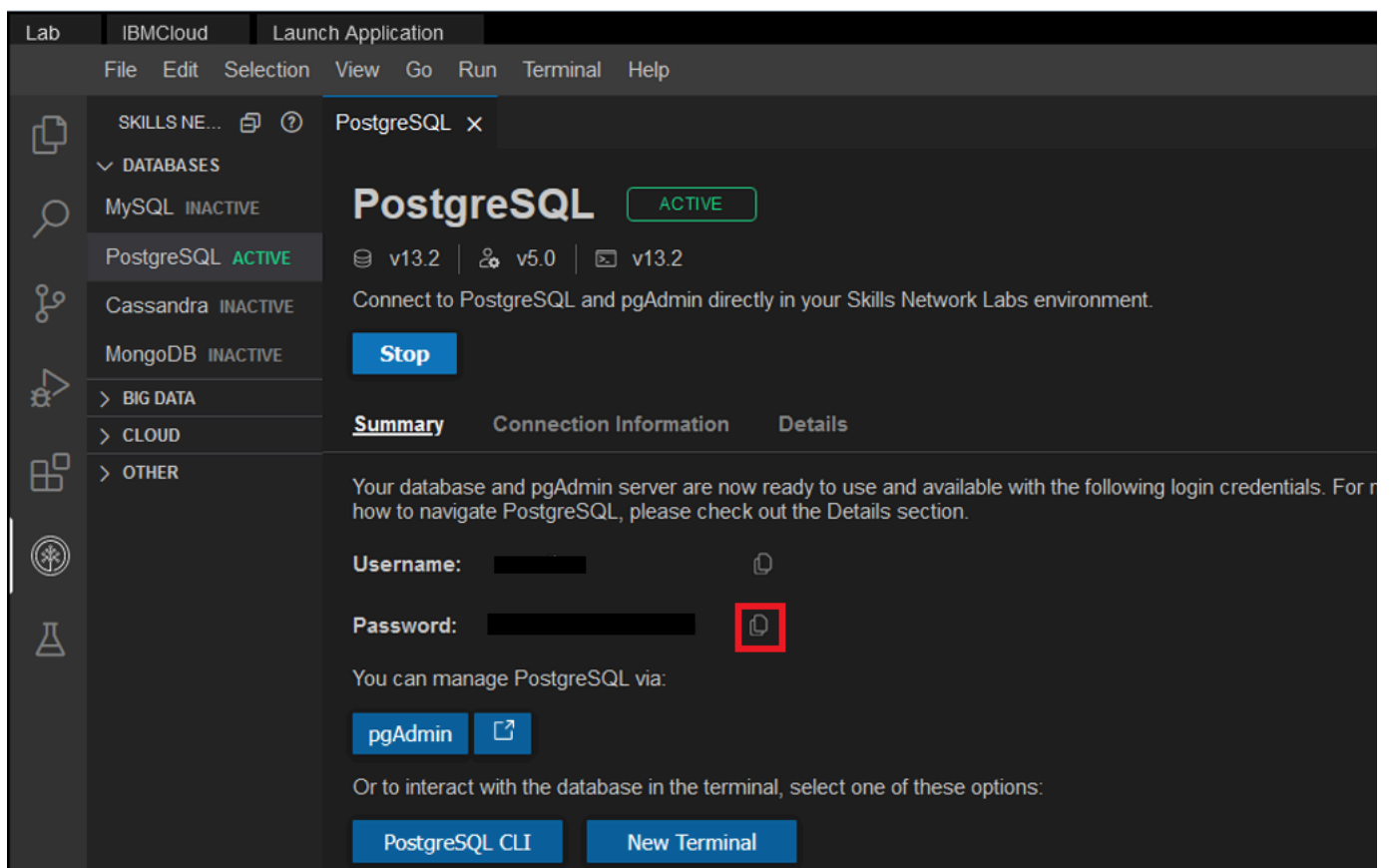


6. Une fois l'interface graphique de pgAdmin ouverte, cliquez sur l'onglet **Serveurs** sur le côté gauche de la page. Vous serez invité à saisir un mot de passe.

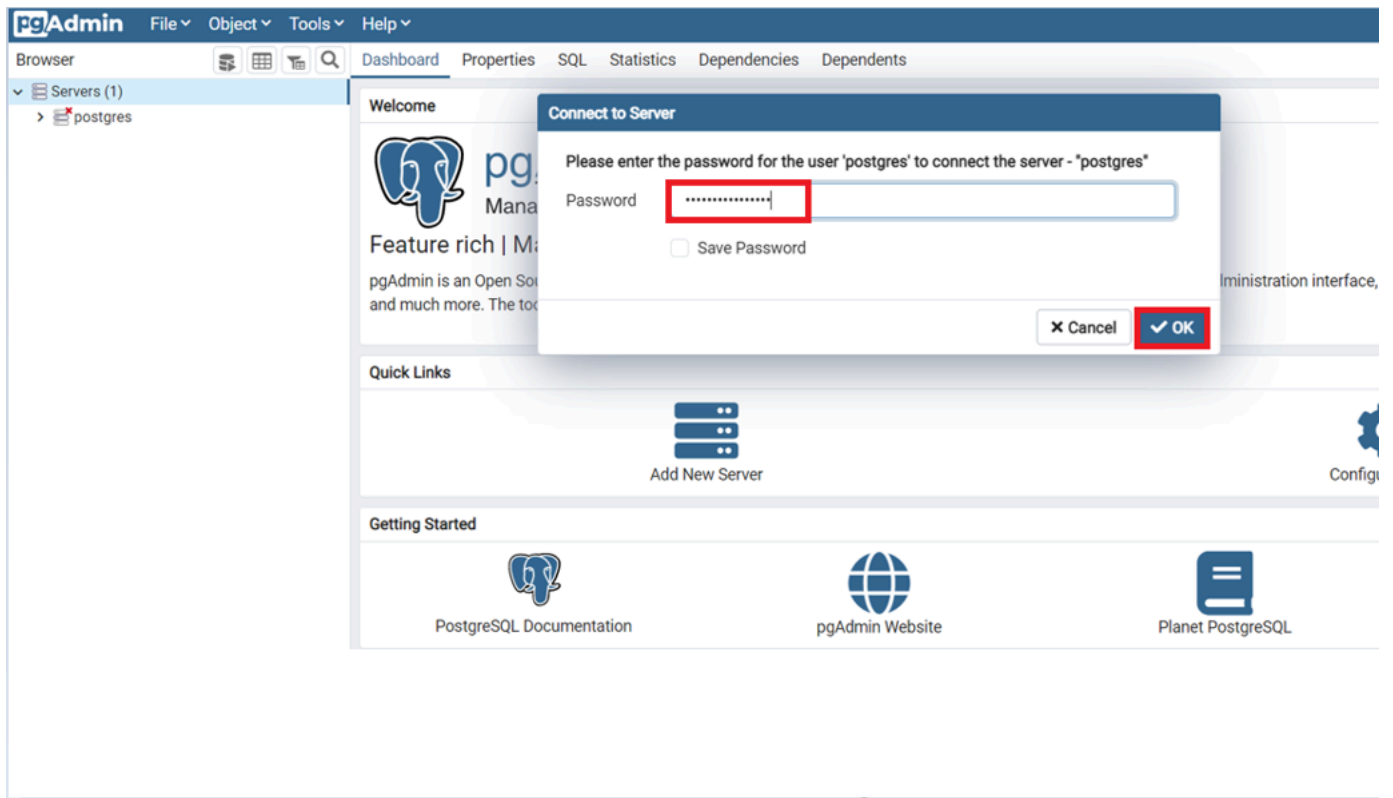


7. Pour récupérer votre mot de passe, cliquez sur l'onglet **PostgreSQL** en haut de l'interface.

8. Cliquez sur l'icône **Copier** à droite de votre mot de passe pour copier le mot de passe de session dans votre presse-papiers.

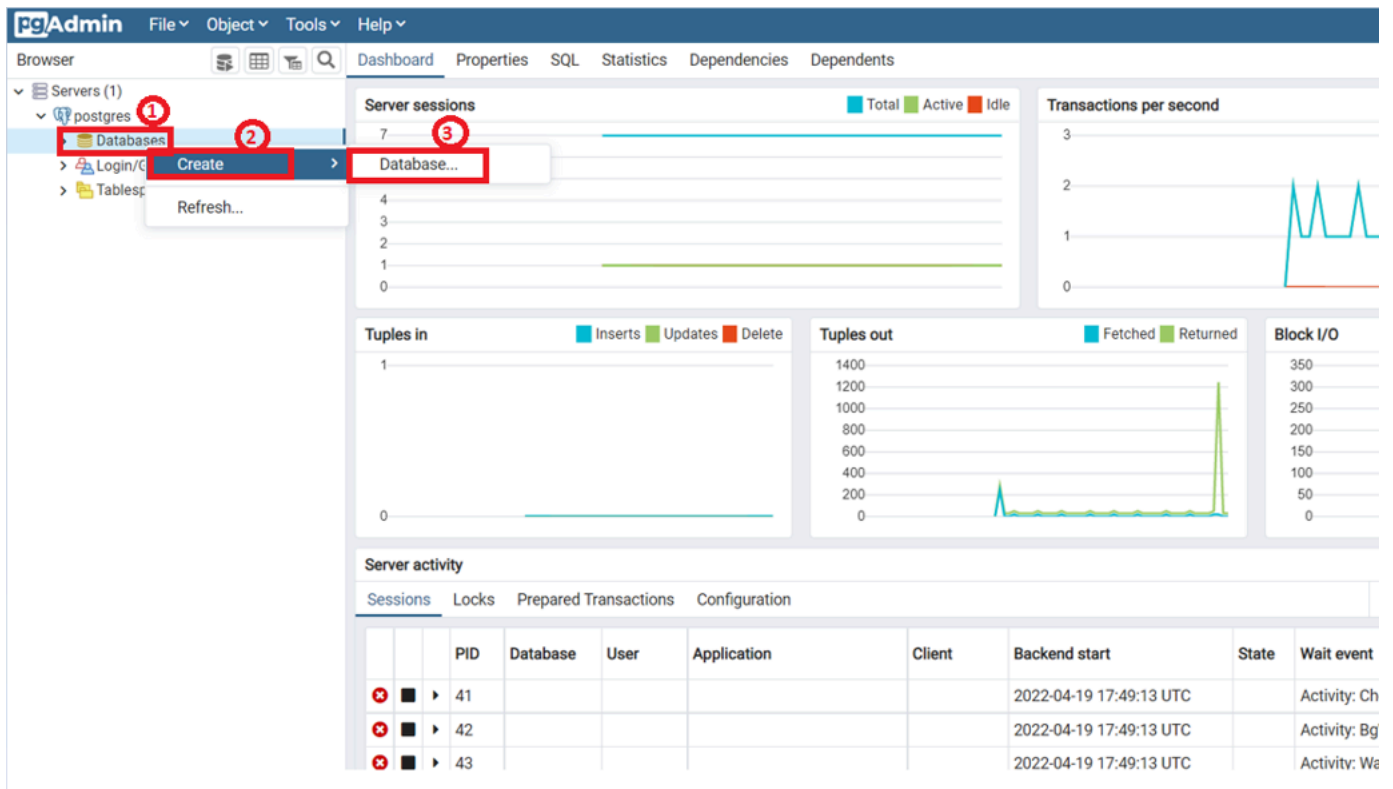


9. Revenez à l'onglet **pgAdmin** et collez votre mot de passe, puis cliquez sur OK.

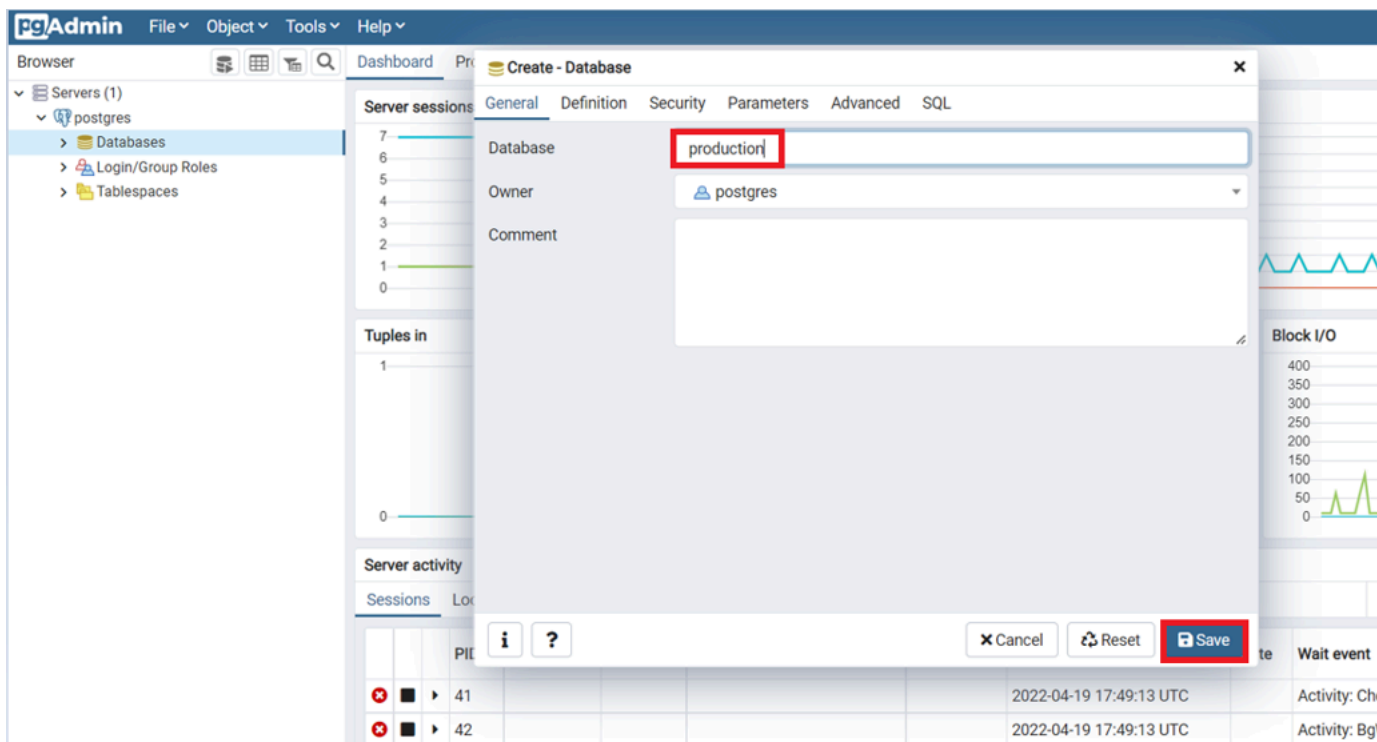


10. Vous pourrez alors accéder à l'outil GUI pgAdmin.

11. Dans l'arborescence de gauche, faites un clic droit sur **Bases de données** > **Créer** > **Base de données**.



Dans la zone Base de données, saisissez **Production** comme nom pour votre nouvelle base de données, puis cliquez sur **Enregistrer**. Passez à la tâche B.

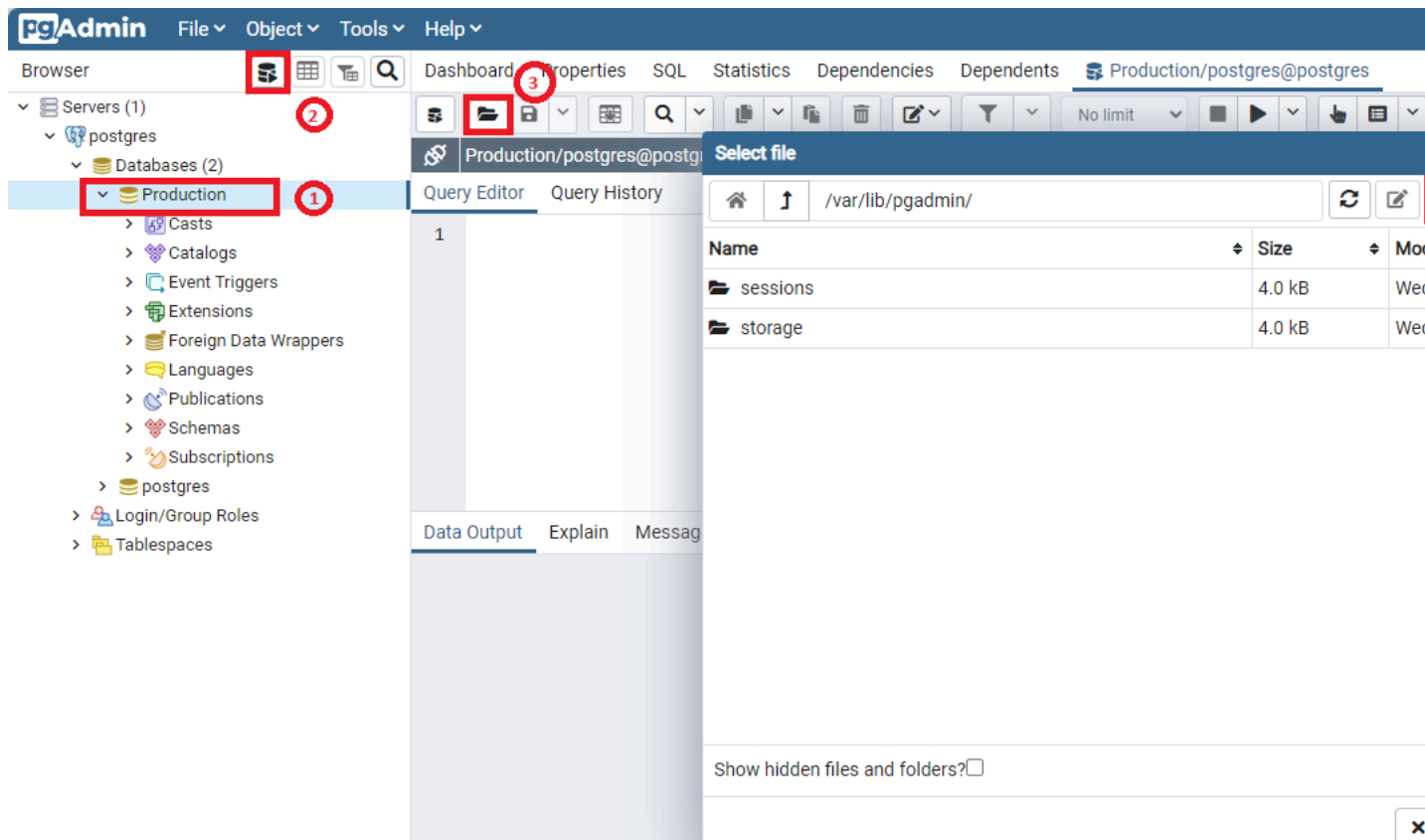


Tâche B : Créer des tableaux

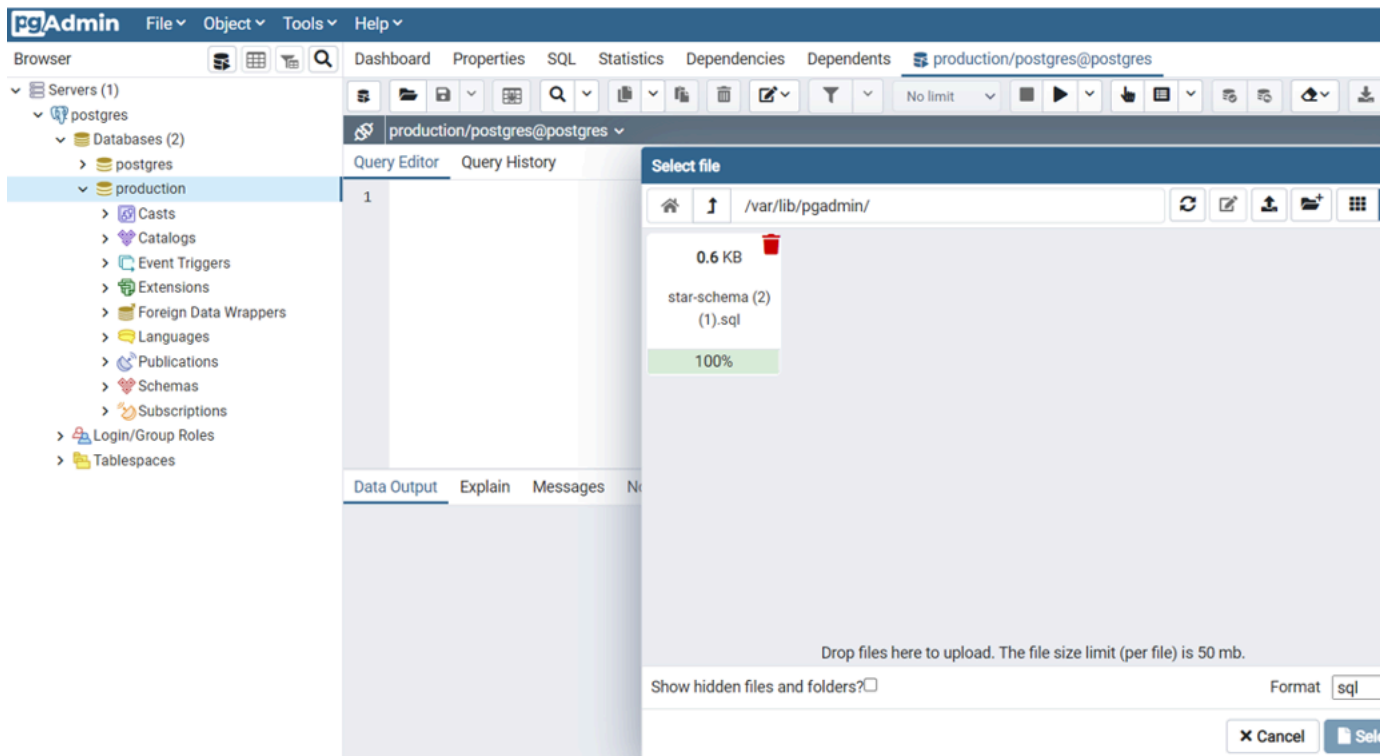
Maintenant que votre service PostgreSQL est actif et que vous avez créé la **base de données de production** à l'aide de pgAdmin, créons quelques tables pour remplir la base de données et stocker les données que nous souhaitons éventuellement y télécharger.

1. Cliquez sur la base de données de production et, en haut de la page, accédez à **l'outil de requête**, puis cliquez sur **Ouvrir un fichier**. Une nouvelle page s'affiche ensuite, intitulée **Sélectionner un fichier**. Cliquez sur l'icône **Télécharger**, comme indiqué dans la capture d'écran.

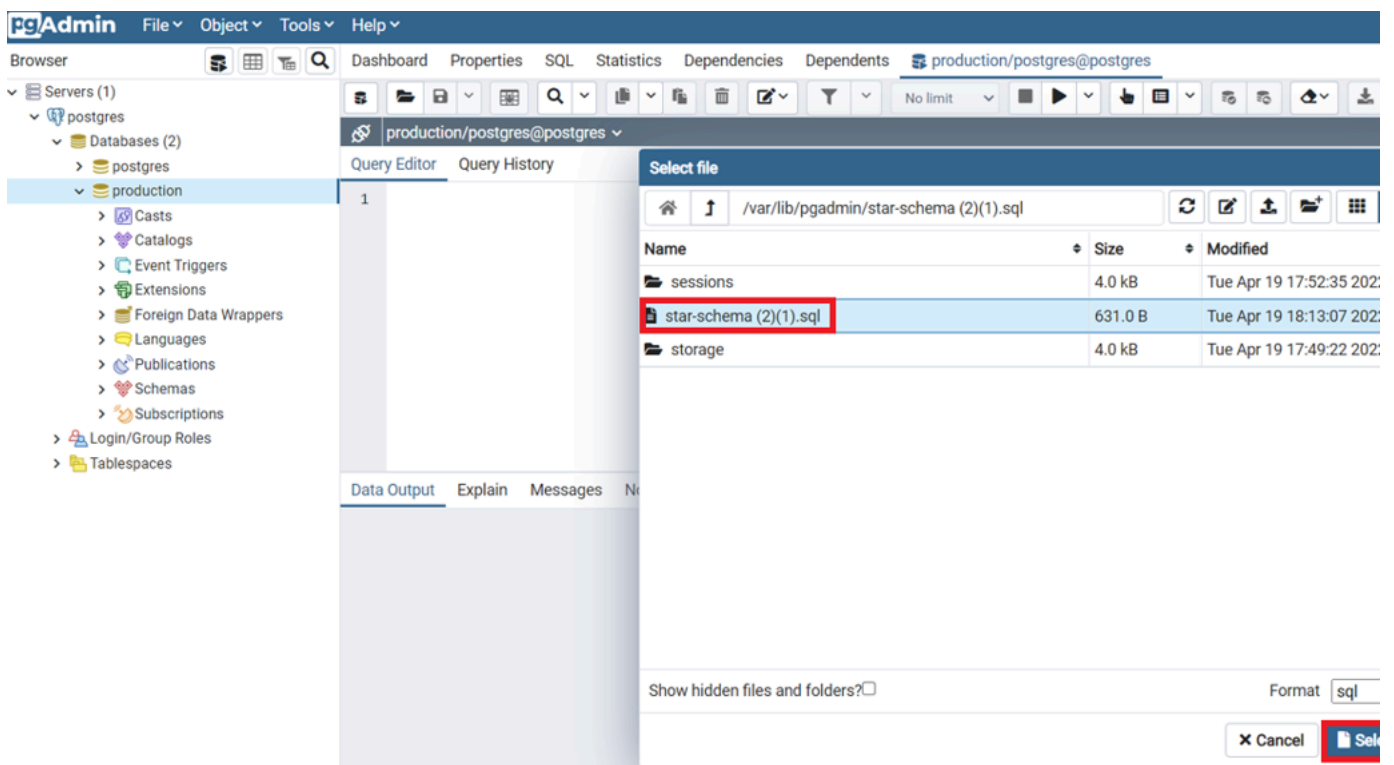
Remarque : assurez-vous de télécharger les fichiers vers ce chemin : `/var/lib/pgadmin/`



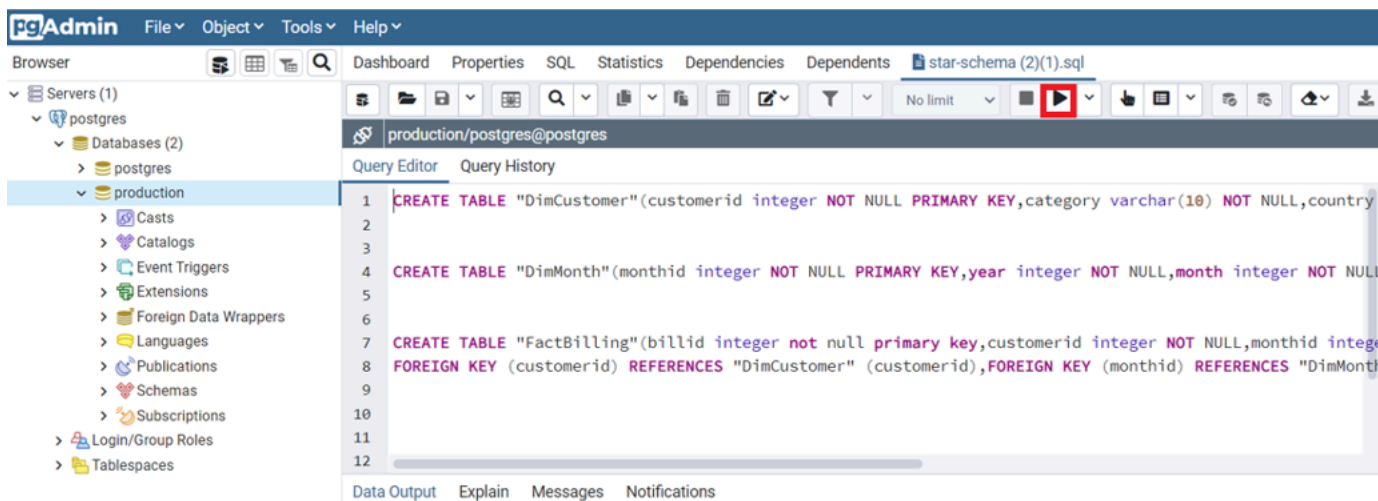
2. Dans la nouvelle page vierge qui apparaît, faites glisser et déposez le fichier **star-schema.sql** à l'intérieur de la page vierge. Une fois le fichier **star-schema.sql** chargé avec succès, cliquez sur l'icône **X** sur le côté droit de la page comme indiqué dans la capture d'écran.



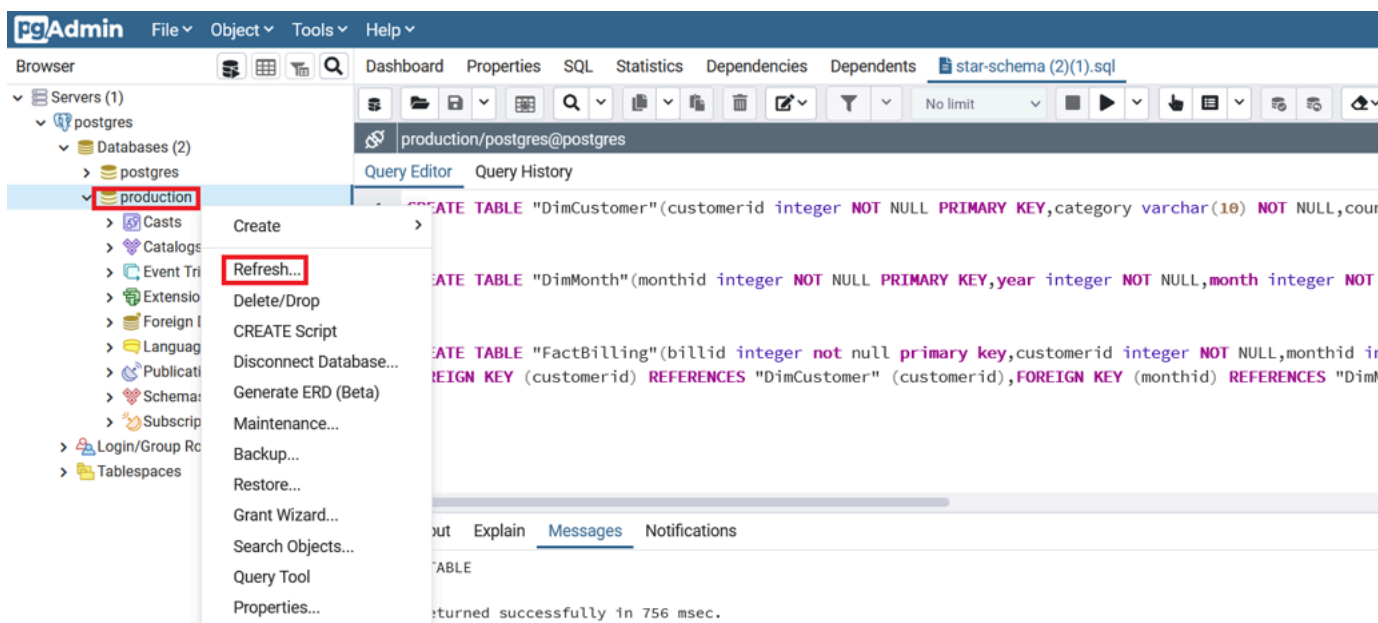
3. Une fois que vous avez cliqué sur l'icône **X**, une nouvelle page apparaît avec le fichier **star-schema.sql**. Sélectionnez le fichier **star-schema.sql** dans la liste et cliquez sur l'onglet **Sélectionner**.



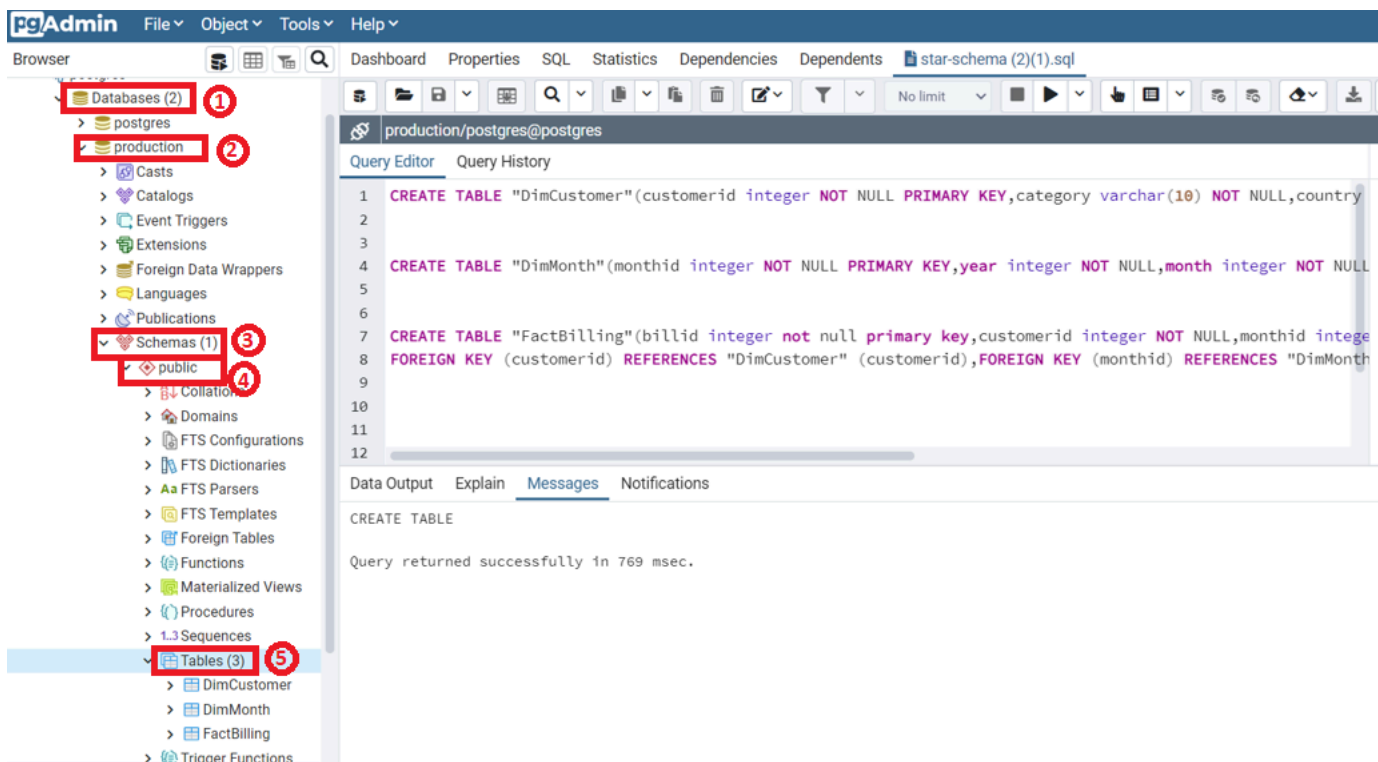
4. Une fois le fichier ouvert, cliquez sur l'option **Exécuter** pour exécuter le fichier **star-schema.sql**.



5. Ensuite, faites un clic droit sur la **base de données de production** et cliquez sur l'option **Actualiser** dans la liste déroulante.

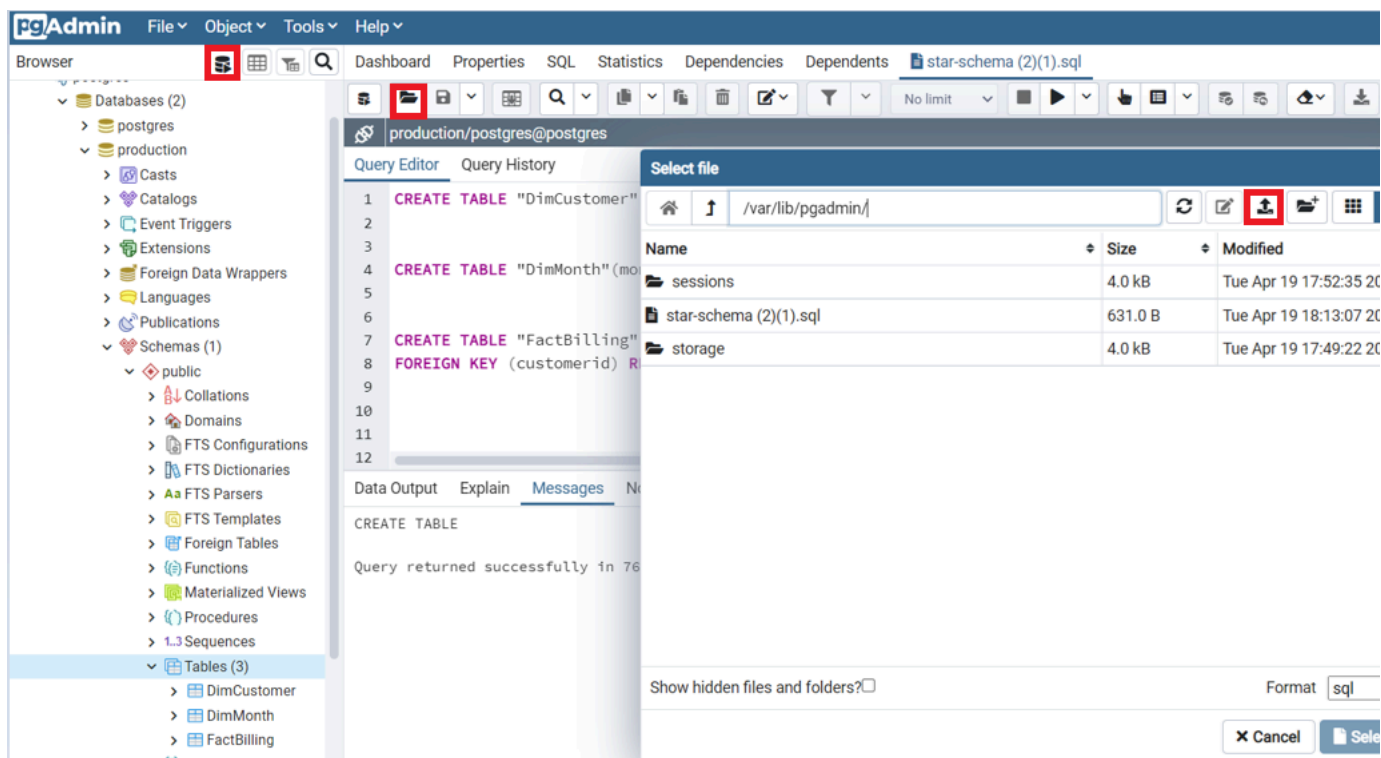


Une fois la base de données actualisée, les 3 tables (DimCustomer, DimMonth, FactBilling) sont créées sous **Bases de données > Production > Schémas > Public > Tables**.



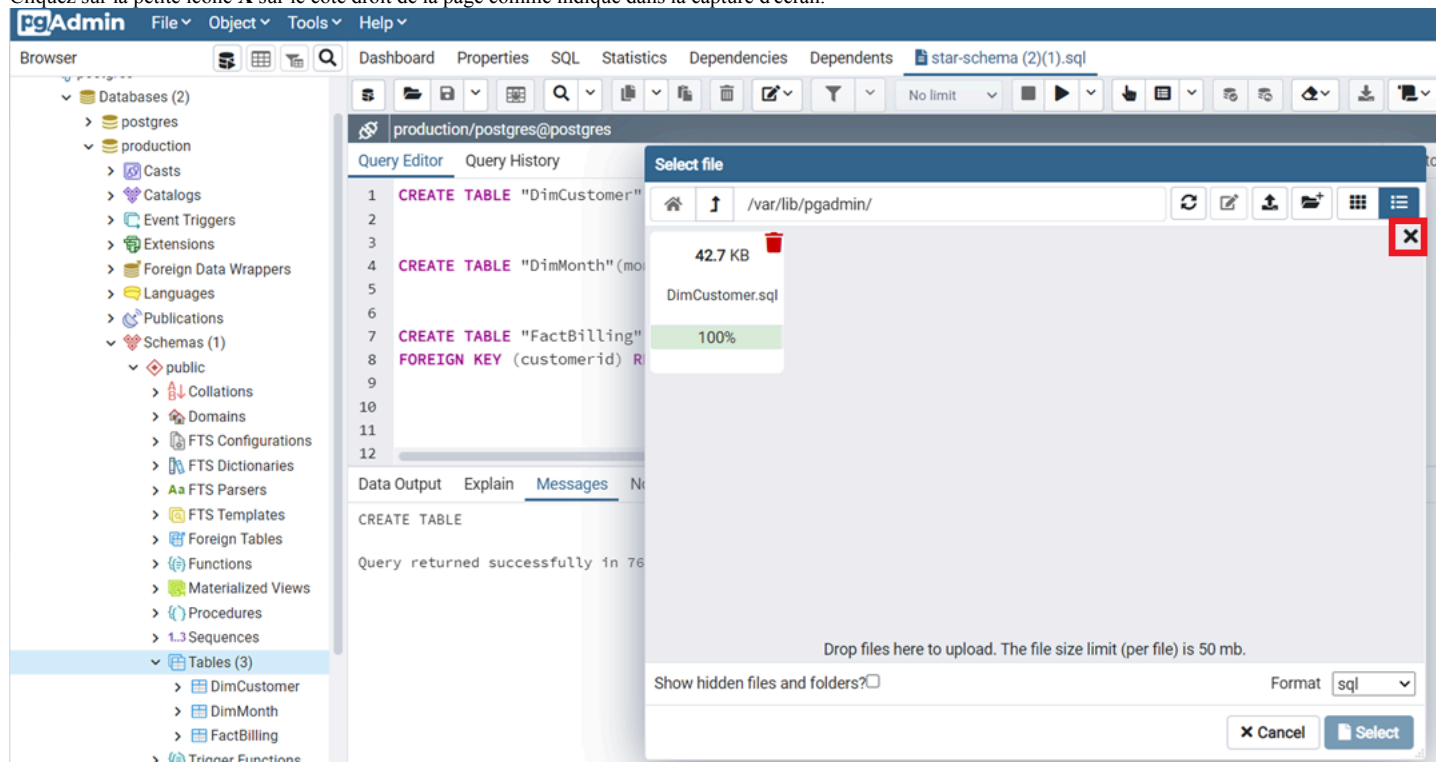
Tâche C : Charger les tables

1. Cliquez sur l'**outil de requête**, puis sur **Ouvrir** le fichier et cliquez sur l'icône **Télécharger**.

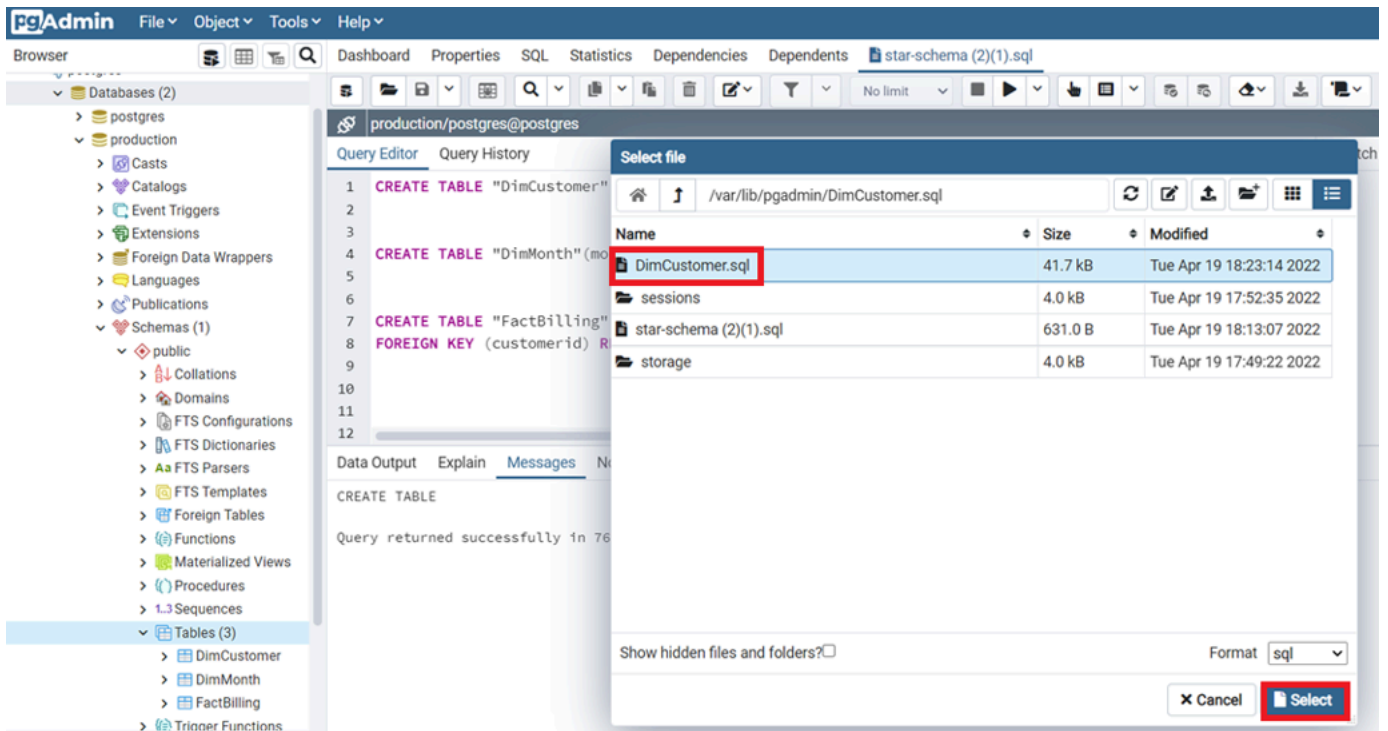


2. Dans la nouvelle page vierge qui apparaît, faites glisser et déposez le fichier **DimCustomer.sql** à l'intérieur de la page vierge. Une fois le fichier **DimCustomer.sql** chargé avec succès.

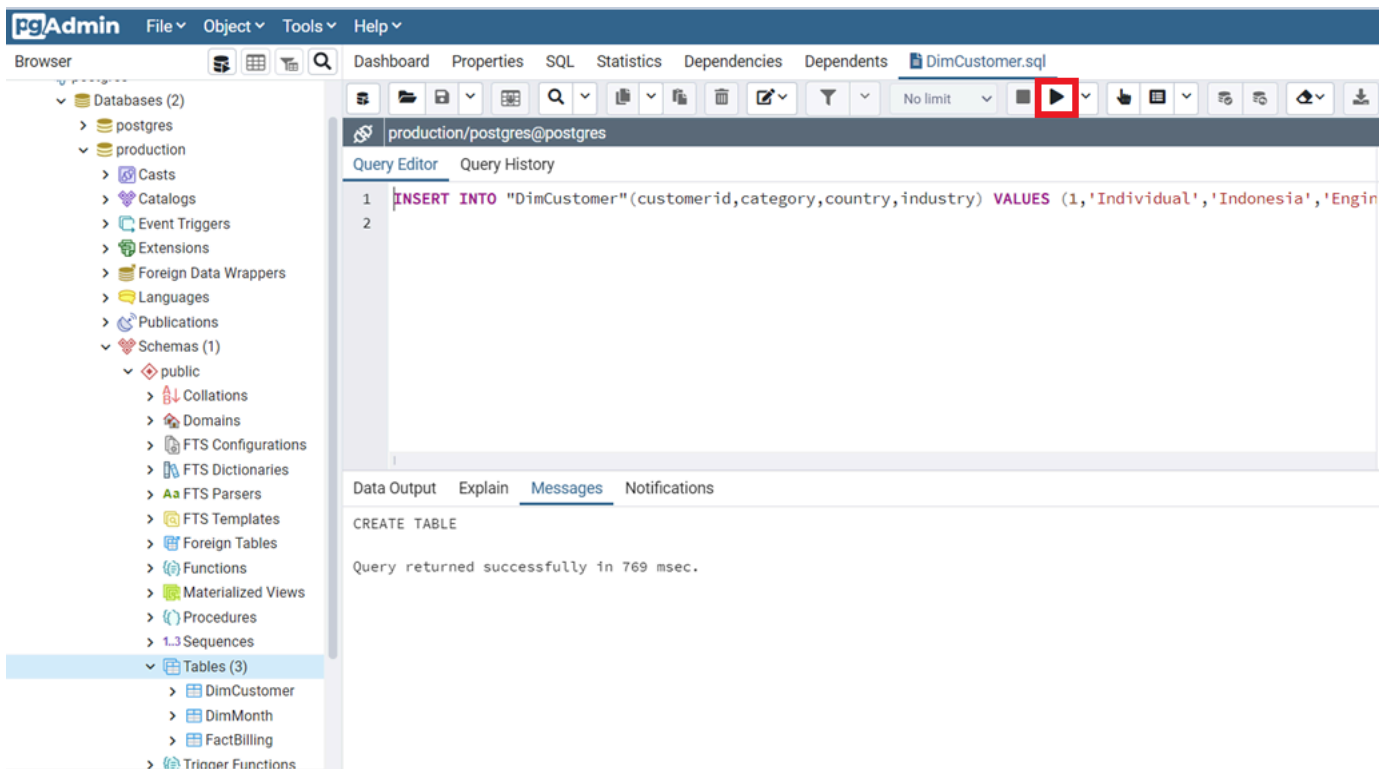
Cliquez sur la petite icône **X** sur le côté droit de la page comme indiqué dans la capture d'écran.



3. Une fois que vous avez cliqué sur l'icône **X**, une nouvelle page apparaît avec le fichier **DimCustomer.sql**. Sélectionnez le fichier **DimCustomer.sql** dans la liste et cliquez sur l'onglet **Sélectionner**.



4. Une fois le fichier ouvert, cliquez sur l'option **Exécuter** pour exécuter le fichier **DimCustomer.sql**.



Remarque : répétez les étapes indiquées dans la tâche C pour télécharger les fichiers SQL restants afin d'insérer des données dans **DimMonth** et **FactBilling**.

5. Exécutons la commande ci-dessous sur l'outil PostgreSQL.

```
select count(*) from public."DimMonth";
```

Vous devriez voir un résultat comme celui montré dans l'image ci-dessous.

The screenshot shows the pgAdmin interface. On the left, the 'Browsers' pane shows a tree view of the database structure. The 'production/postgres@postgres' connection is selected, and the 'Tables (3)' folder is expanded, showing 'DimCustomer', 'DimMonth', and 'FactBilling'. The main window is the 'Query Editor' for the 'production/postgres@postgres' connection. It contains a SQL query: `select count(*) from public.\"DimMonth\";`. Below the query editor, the 'Data Output' tab is active, displaying the result of the query. The result is a single row with two columns: 'count' and 'bigint', and a value of 132. A green message at the bottom right of the interface states: 'Successfully run. Total query'.

Nous vous encourageons à exécuter davantage de requêtes SQL.

Exercices pratiques

Problème 1 : À l'aide de l'outil PostgreSQL, recherchez le nombre de lignes dans la table FactBilling

▼ Cliquez ici pour un indice

Utilisez l'instruction select avec la fonction count sur la table FactBilling.

▼ Cliquez ici pour la solution

```
select count(*) from public.\"FactBilling\";
```

Problème 2 : À l'aide de l'outil PostgreSQL, créez une vue matérialisée simple nommée avg_customer_bill avec les champs customerid et averagebillamount.

► Cliquez ici pour un indice

▼ Cliquez ici pour la solution

```
CREATE MATERIALIZED VIEW avg_customer_bill (customerid, averagebillamount) AS
(select customerid, avg(billedamount)
from public.\"FactBilling\"
group by customerid
);
```

Cliquez sur le bouton **Exécuter tout** pour exécuter l'instruction. Vous devriez voir le statut **Succès** dans la section **Résultat**.

Problème 3 : Actualiser les vues matérialisées nouvellement créées

► Cliquez ici pour un indice

▼ Cliquez ici pour la solution

```
REFRESH MATERIALIZED VIEW avg_customer_bill;
```

Problème 4 : À l'aide des vues matérialisées nouvellement créées, recherchez les clients dont la facturation moyenne est supérieure à 11 000.

► Cliquez ici pour un indice

► Cliquez ici pour la solution

Félicitations ! Vous avez terminé avec succès le laboratoire Remplissage d'un entrepôt de données.

Auteur

Amrutha Rao

© IBM Corporation 2022. Tous droits réservés.