

Requêtage SQL Avancé - 2023/2024 - Examen final - Briefing projet

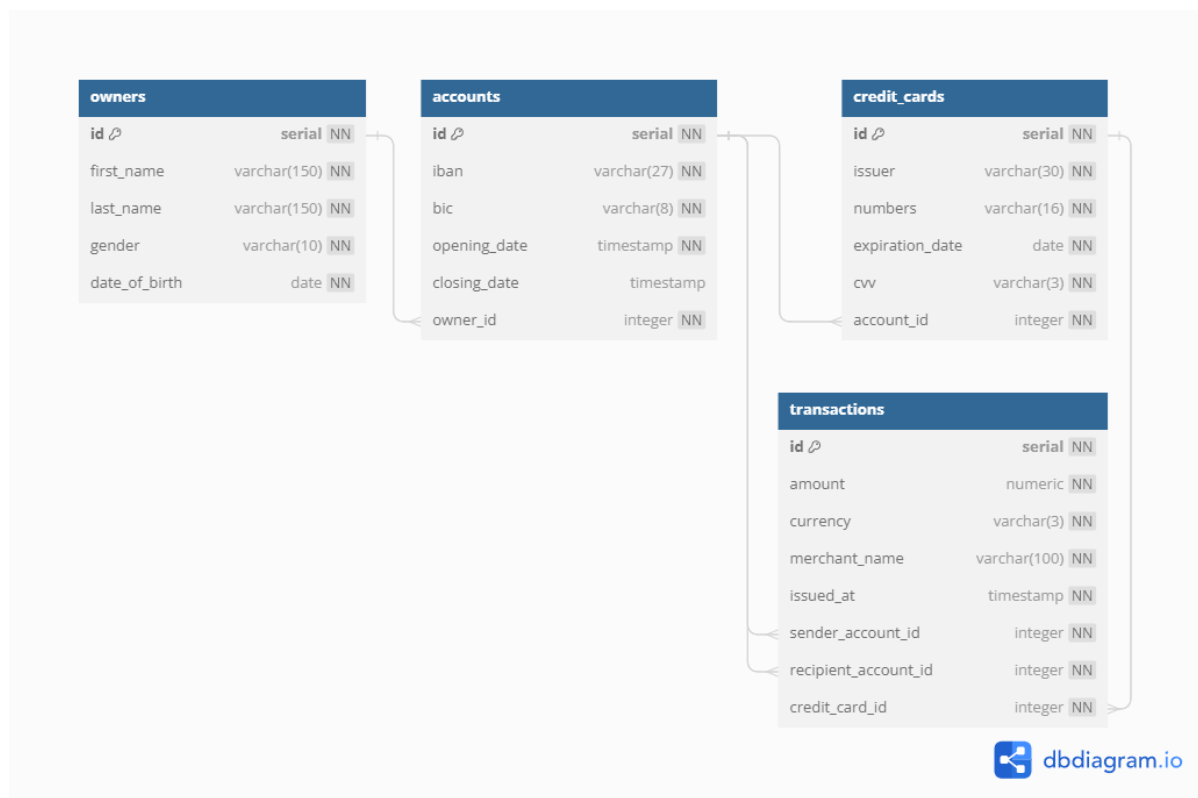
Données fournies

Fichiers

- `docker-compose.yaml` : configuration Docker Compose pour démarrer facilement un serveur PostgreSQL 16.0 ;
- `import-transactions.sql` : script permettant de créer les tables et d'ajouter les données nécessaires à la réalisation du contrôle continu ;
- `account-owners.csv` : fichier CSV contenant des titulaires de compte en banque.
- `accounts.csv` : fichier CSV contenant des comptes en banque.
- `credit-cards.csv` : fichier CSV contenant des cartes bancaires.
- `transactions.csv` : fichier CSV contenant des transactions bancaires.

N'hésitez pas à consulter les fichiers fournis pour plus de détails.

Base de données



- Une table `owners` contenant les colonnes suivantes :
 - `id` : l'identifiant du titulaire du compte ;
 - `first_name` : le prénom du titulaire ;
 - `last_name` : le nom de famille du titulaire ;
 - `gender` : le genre du titulaire (`male` ou `female`) ;
 - `date_of_birth` : la date de naissance du titulaire.

- Une table `accounts` contenant les colonnes suivantes :
 - `id` : l'identifiant du compte bancaire ;
 - `iban` : l'IBAN ([International Bank Account Number](#)) du compte ;
 - `bic` : le code BIC de la banque ;
 - `opening_date` : la date d'ouverture du compte ;
 - `closing_date` : la date de clôture du compte, s'il a été clôturé ;
 - `owner_id` : l'identifiant du titulaire de ce compte bancaire.
- Une table `credit_cards` contenant les colonnes suivantes :
 - `id` : l'identifiant de la carte bancaire ;
 - `issuer` : le réseau de carte bancaire (Visa ou MasterCard) ;
 - `numbers` : le numéro de la carte bancaire ;
 - `expiration_date` : la date d'expiration de la carte bancaire ;
 - `cvv` : le code de vérification de la carte bancaire pour les achats en ligne ;
 - `account_id` : l'identifiant du compte bancaire auquel est rattaché cette carte.
- Une table `transactions` contenant les colonnes suivantes :
 - `id` : l'identifiant de la transaction ;
 - `amount` : le montant de la transaction ;
 - `currency` : la devise de la transaction (`EUR` pour euro) ;
 - `merchant_name` : la raison sociale (nom) de la société destinataire de la transaction ;
 - `issued_at` : la date d'opération de la transaction ;
 - `sender_account_id` : l'identifiant du compte du **payeur** de la transaction (celui qui envoie l'argent) ;
 - `recipient_account_id` : l'identifiant du compte du **bénéficiaire** de la transaction (celui qui reçoit l'argent) ;
 - `credit_card_id` : l'identifiant de la carte bancaire qui a permis de réaliser la transaction.

Informations supplémentaires

- Il existe deux types d'**opération** bancaire, matérialisées par une **transaction** : un **crédit** lorsqu'on reçoit de l'argent, et un **débit** lorsqu'on envoie de l'argent.
- Une personne qui envoie de l'argent via une transaction est appelée **payeur**.
- À l'inverse, une personne qui reçoit de l'argent via une transaction est appelée **bénéficiaire**.

Mise en place de l'examen

Démarrer le serveur PostgreSQL en exécutant la commande suivante dans un terminal :

```
docker compose up
```

Dans un second terminal, ouvrir un shell dans le conteneur PostgreSQL :

```
docker compose exec db bash
```

Une fois le shell lancé, se placer dans le dossier de projet :

```
cd /sql
```

Puis, exécuter la commande suivante pour ouvrir une session avec le client `psql` :

```
psql -U postgres
```

Enfin, importer les données en utilisant la commande `psql` suivante :

```
\i datasets/import-transactions.sql
```

Vous êtes maintenant prêt pour réaliser l'examen.