



INTERFACE MACHINE-MACHINE

GROUPE 12

Participants

-NTCHUIGWA YOUNGUI Victoire

-WELLAN Sergine (chef projet)

-MBILONGO TENE Danya-Maylis Boriska

- TCHIADEU TCHIADEU Fred Junior

Pour l'interface machine -machine il était question pour nous de créer une relation entre l'interface python et la base de donnée pour cela ,nous aurions pu utiliser beaucoup de modules python disponibles tels que :

Psycopg2

pg8000

py-postgresql

PyGreSQL

ocpgdb

bpgsql

SQLAlchemy

Note : Tous les modules ci-dessus adhèrent à la spécification Python Database API v2.0 (PEP 249). Cette API est conçue pour encourager et maintenir la similarité entre les modules de base de données Python pour accéder aux bases de données. En d'autres termes, la syntaxe, la méthode et la façon d'accéder à la base de données sont les mêmes dans tous les modules ci-dessus.

Nous nous en tenons à Psycopg2 parce que c'est sans doute le module le plus populaire et le plus stable pour travailler avec PostgreSQL. Nous utilisons également Psycopg2 pour travailler avec PostgreSQL pour les raisons suivantes.

Il est utilisé dans la plupart des frameworks Python et Postgres.

Il est également activement maintenu et supporte la version principale de Python, c'est-à-dire Python 3 et Python 2.

Il est thread-safe et conçu pour les applications fortement multi-threadées. Notez que les threads peuvent partager les connexions.

Ce guide Python PostgreSQL se concentre principalement sur les points suivants

Installation de Psycopg2 et utilisation de son API pour accéder à la base de données PostgreSQL

Effectuer l'insertion, la récupération, la mise à jour et la suppression de données par le biais d'une application Python.

Ensuite, il couvrira la gestion des transactions PostgreSQL, le pooling de connexion et les techniques de gestion des erreurs pour développer des programmes Python robustes avec PostgreSQL.

Plongeons-y.

Installer Psycopg2 à l'aide de la commande pip

Vous devez installer la version actuelle de Psycopg2 (2.8.6) sur votre machine pour utiliser PostgreSQL depuis Python. Ce module est disponible sur pypi.org.

En utilisant la commande pip suivante, vous pouvez installer Psycopg2 sur n'importe quel système d'exploitation, y compris Windows, macOS, Linux, Unix et Ubuntu.

pip install psycopg2

Vous pouvez également installer une version spécifique à l'aide de la commande suivante.

pip install psycopg2=2.8.6

Si vous êtes confronté à une erreur de pip install du type "**connection error : [SSL : CERTIFICATE_VERIFY_FAILED] certificate verify failed (_ssl.c:598)**". Vous pouvez résoudre cette erreur en définissant pypi.org et files.pythonhosted.org comme hôtes de confiance. Si vous rencontrez une erreur lors de l'installation de pip, essayez de suivre la commande suivante.

```
python -m pip install --trusted-host pypi.org --trusted-host  
files.pythonhosted.org --trusted-host pypi.python.org psycopg2
```

Le module actuel psycopg2 supporte :

Python version 2.7, et les versions de Python 3 de 3.4 à 3.8.

Les versions du serveur PostgreSQL de 7.4 à 12

Version de la bibliothèque client PostgreSQL à partir de 9.1

Vérifier l'installation de Psycopg2

Vous devriez obtenir les messages suivants après avoir exécuté la commande ci-dessus.

Récupération de psycopg2

Téléchargement de psycopg2-2.8.6

Installation des paquets collectés : psycopg2

Installation réussie de psycopg2-2.8.6

Veuillez utiliser la commande suivante pour installer Psycopg2 si vous utilisez anaconda.

```
conda install -c anaconda psycopg2
```

Connexion à la base de données Python PostgreSQL

Dans cette section, nous allons apprendre à nous connecter à PostgreSQL via Python en utilisant Psycopg2.

Arguments nécessaires pour connecter la base de données PostgreSQL depuis Python

Vous devez connaître les détails suivants du serveur PostgreSQL pour effectuer la connexion.

Nom d'utilisateur : Le nom d'utilisateur que vous utilisez pour travailler avec PostgreSQL, Le nom d'utilisateur par défaut pour la base de données PostgreSQL est Postgres.

Password : Le mot de passe est donné par l'utilisateur au moment de l'installation de PostgreSQL.

Nom d'hôte : C'est le nom du serveur ou l'adresse IP sur lequel PostgreSQL tourne. Si vous travaillez sur localhost, alors vous pouvez utiliser localhost, ou son IP, c'est-à-dire **127.0.0.0**.

Nom de la base de données : Nom de la base de données à laquelle vous voulez vous connecter. Ici, nous utilisons la base de données nommée "postgres_db".

Comment se connecter à PostgreSQL en Python

Installer le module Psycopg2

Installez et importez le module psycopg2. Importez en utilisant une instruction `import psycopg2` afin de pouvoir utiliser les méthodes de ce module pour communiquer avec la base de données PostgreSQL.

Utilisez la méthode connect()

Utilisez la méthode `psycopg2.connect()` avec les arguments requis pour connecter MySQL. Elle renvoie un objet `Connection` si la connexion a été établie avec succès.

Utilisez la méthode cursor()

Créez un objet curseur en utilisant l'objet de connexion renvoyé par la méthode `connect` pour exécuter des requêtes PostgreSQL depuis Python.



Utiliser la méthode `execute()`

Les méthodes `execute()` exécutent la requête SQL et renvoient le résultat.

Extraire le résultat en utilisant `fetchall()`

Utilisez `cursor.fetchall()` ou `fetchone()` ou `fetchmany()` pour lire le résultat de la requête.

Fermer les objets curseur et connexion

Utilisez la méthode `cursor.close()` et `connection.close()` pour fermer les connexions PostgreSQL à la fin de votre travail.