

OC PIZZA

SOLUTION TECHNIQUE D'UN SYSTÈME DE GESTION DE PIZZERIA Dossier d'exploitation

Version 1.1

Auteur
Edouard LOUSSOUARN
Analyste-Programmeur

TABLE DES MATIÈRES

1 - Introduction.....	4
1.1 - Objet du document	4
1.2 - Références.....	4
2 - Pré-requis	5
2.1 - Système	5
2.1.1 - <i>Serveur de Base de données</i>	5
2.1.1.1 - Caractéristiques techniques	5
2.1.2 - <i>Serveur Web</i>	5
2.1.2.1 - Caractéristiques techniques	5
2.2 - Bases de données.....	6
2.3 - Web-services	6
3 - Procédure de déploiement	7
3.1 - Déploiement de l'application	7
3.1.1 - <i>Variables d'environnement</i>	7
3.1.2 - <i>Configuration</i>	7
3.1.3 - <i>Activation</i>	7
3.1.3.1 - Télécharger Nginx sur le serveur	7
3.1.3.2 - Création d'un fichier de configuration	7
3.1.3.3 - Servir l'application	8
3.2 - Déploiement de la base de données.....	8
3.2.1.1 - Affecter un mot de passe au super-administrateur	8
3.2.1.2 - Ajouter un serveur avec une connexion super-administrateur.....	9
3.2.1.3 - Créer la base de donnée.....	9
3.2.1.4 - Créer une table.....	11
3.2.1.5 - Ajouter les données contenues dans une table	12
4 - Procédure de démarrage / arrêt	13
4.1 - Base de données	13
4.1.1 - <i>Procédure de démarrage</i>	13
4.1.2 - <i>Procédure d'arrêt</i>	13
4.2 - Application web	13
4.2.1 - <i>Procédure de démarrage</i>	13
4.2.2 - <i>Procédure d'arrêt</i>	13
5 - Procédure de mise à jour	14
5.1 - Base de données	14
5.1.1 - <i>Modifier les données contenues dans une table</i>	14
5.2 - Application web	14
6 - Supervision/Monitoring.....	16
6.1 - Supervision de l'application web.....	16
6.1.1 - <i>Réduire les temps d'arrêt de Nginx avec l'analyseur Nginx</i>	16
6.1.2 - <i>Suivre les performances avec le moniteur Nginx d'Applications Manager</i>	16
6.1.3 - <i>Assurer les performances des applications grâce à la supervision des applications Nginx</i>	16
6.1.4 - <i>Dépanner rapidement pour une supervision efficace de Nginx</i>	16
7 - Procédure de sauvegarde et restauration	17
7.1 - Sauvegarde de la base de donnée.....	17
7.1.1 - <i>Sauvegarde quotidienne postgresql</i>	17
7.2 - Restauration de la base de donnée	17
8 - Glossaire	18

Versions

Auteur	Date	Description	Version
Edouard	01/09/2021	Création du document	0.1
Edouard	05/09/2021	Ajout du glossaire	0.2
Edouard	10/09/2021	Rélecture, mise en page et correction	1.1

1 - INTRODUCTION

1.1 - Objet du document

Le présent document constitue le dossier d'exploitation de l'application OC Pizza.

Objectif du document est de contenir les informations dont l'équipe d'exploitation a besoin pour pouvoir assurer l'exploitation en règle du système et pouvoir réagir de manière appropriée lorsqu'un problème surgit.

1.2 - Références

Pour de plus amples informations, se référer :

1. **P10_OCPizza_01_dossier_de_conception_fonctionnelle**: Dossier de conception fonctionnelle
2. **P10_OCPizza_02_Dossier_de_conception_technique** : Dossier de conception technique de l'application

2 - PRÉ-REQUIS

2.1 - Système

2.1.1 - Serveur de Base de données

Le serveur de base de données sert à stocker, à extraire et à gérer les données dans la base de données. Il permet également de gérer la mise à jour des données.

PostgreSQL* est le serveur de la base de données qui héberge la base de données.

2.1.1.1 - Caractéristiques techniques

Base de données : db_ocpizza

Version utilisée : PostgreSQL 13.1

PostgreSQL est un système de gestion de data base open-source. L'outil assure une excellente compatibilité avec SQL pour une requête relationnelle

Ce système de gestion de base de données permet de gérer nombreux types des datas avec ses fonctionnalités basiques. Les données les plus complexes peuvent très bien être traitées pour une optimisation de la performance de l'entreprise.

Très populaire, une vaste communauté de développeurs web expérimentés a contribué au développement et à l'amélioration du système, devenant l'un des plus fiables sur le marché.

PostgreSQL fonctionne selon une architecture client/serveur, il est constitué :

- d'une partie serveur qui héberge la base de données et qui est capable de traiter les requêtes des clients. Il s'agit dans le cas de PostgreSQL d'un programme résident en mémoire appelé postmaster.
- d'une partie client qui doit être installée sur toutes les machines nécessitant d'accéder au serveur de base de données.

Les clients peuvent interroger le serveur de bases de données à l'aide de requêtes SQL.

2.1.2 - Serveur Web

Un serveur Web est un serveur informatique qui héberge le contenu de l'application Web et publie celui ci en ligne sur Internet.

NGINX est le serveur web utilisé pour l'application web OC Pizza.

2.1.2.1 - Caractéristiques techniques

Version utilisée : NGINX 1.17.0

NGINX est conçu pour offrir une faible utilisation de la mémoire et une grande simultanéité. Plutôt

que de créer de nouveaux processus pour chaque requête Web, NGINX utilise une approche asynchrone et événementielle où les requêtes sont traitées dans un seul thread.

2.2 - Bases de données

Les bases de données et schéma suivants doivent être accessibles et à jour :

- Schéma OC_Pizza et PostgreSQL : version 13.1

2.3 - Web-services

Les web services suivants doivent être accessibles et à jour :

- L'API **Directions*** : Fournis les itinéraires pour les livreurs et affiche les informations sur le trafic en temps réel. **Clé d'identification** : AlzaSyAnlcq5vYqkb_8wrc_Tr8g5CJ5ZIKstq8w
- **Stripe***: c'est la plateforme de traitement des paiements qui permet de transférer de l'argent du compte bancaire du client vers le compte d'une entreprise au moyen d'une transaction par carte de crédit. C'est le moyen simple d'accepter des paiements en ligne, sans frais d'installation ni frais mensuels. Stripe assure un niveau de sécurité élevé, qui permet de recevoir vos paiements en toute fiabilité.

3 - PROCÉDURE DE DÉPLOIEMENT

3.1 - Déploiement de l'application

3.1.1 - Variables d'environnement

Voici les variables d'environnement reconnues par l'application OC PIZZA :

Nom	Obligatoire	Description
ENV	OUI	Répertoire racine de l'installation de l'application
GMAP_API_KEY	OUI	Contient la clé API nécessaire au web service GoogleMap

3.1.2 - Configuration

Voici les différents fichiers de configuration :

- *oc_pizza_config* /etc/nginx/nginx.conf qui contient la configuration générale de nginx.
- *oc_pizza_params* /etc/nginx/proxy_params qui contient les paramètres généraux de nginx proxy.
- *oc_pizza_env*/etc/nginx/site-enabled et/etc/nginx/site-available qui contient la configuration du site.

3.1.3 - Activation

3.1.3.1 - Télécharger Nginx sur le serveur

Pour télécharger Nginx sur le serveur

```
sudo apt-get install nginx
```

Pour créer le site, il faut que le fichier de configuration soit présent dans */etc/nginx/site-enabled*

3.1.3.2 - Création d'un fichier de configuration

- créer un nouveau fichier dans *sites-available* ;
- ajouter un lien symbolique dans *sites-enabled* grâce à la commande *ln*.

```
occonsulting@ocpizza:/etc/nginx/$ sudo touch sites-available/ocpizza
```

```
[sudo] password for oconsulting:
```

```
occonsulting@ocpizza:/etc/nginx/$ sudo ln -s /etc/nginx/sites-available/ocpizza /etc/nginx/sites-enabled
```

3.1.3.3 - Servir l'application

Ouvrez le document `etc/nginx/sites-available/ocpizza`

```
server {
    listen 80; server_name 178.62.117.192;
    root /home/occonsulting/ocpizza;
    location / {
        proxy_set_header Host $http_host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_redirect off;
        proxy_pass http://127.0.0.1:8000;
    }
}
```

Nginx ne recharge pas automatiquement les configurations. Il faut lui indiquer manuellement en exécutant la commande reload :

```
occonsulting@ocpizza:/etc/nginx$ sudo service nginx reload
```

3.2 - Déploiement de la base de données

On utilise **Postgres pgAdmin 4*** pour la gestion de la base de donnée Oc_Pizza.

3.2.1.1 - Affecter un mot de passe au super-administrateur

pgAdmin permet de paramétrer une connexion au serveur en local par le socket Unix. Pour cela on doit d'abord affecter un mot de passe au super-administrateur qui est l'utilisateur/développeur authentifié sur la machine de bureau. On utilise `psql`:

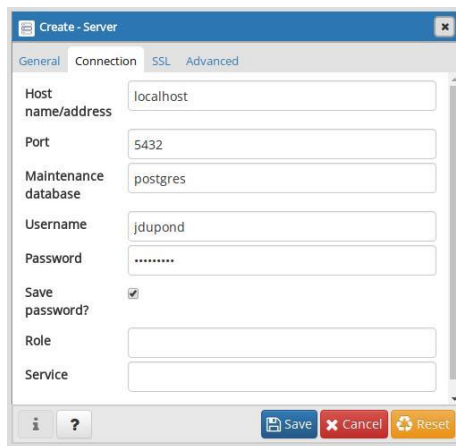
```
psql -d postgres
```

Une fois connecté, exécuter:

```
alter user ocpizza with encrypted password 'mpocpizza';
```


3.2.1.2 - Ajouter un serveur avec une connexion super-administrateur

Afficher le menu contextuel de l'objet Serveurs de développement puis sélectionner: Create / Server... Ceci affiche la fenêtre de paramétrage d'un nouveau serveur.



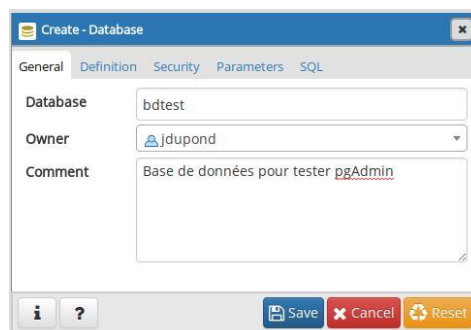
Saisir le nom du serveur dans le champ Name: « Serveur local superadmin ». Ensuite passer à l'onglet (à sélectionner en haut de la fenêtre) Connection. Dans cet onglet renseigner les valeurs des champs comme indiqué dans la copie d'écran.

A la fin de la saisie, enregistrer avec le bouton Save.

3.2.1.3 - Créer la base de donnée

Afficher le menu contextuel du serveur Serveur local superadmin puis sélectionner Create / Database..

Dans le 1er onglet qui s'affiche par défaut:

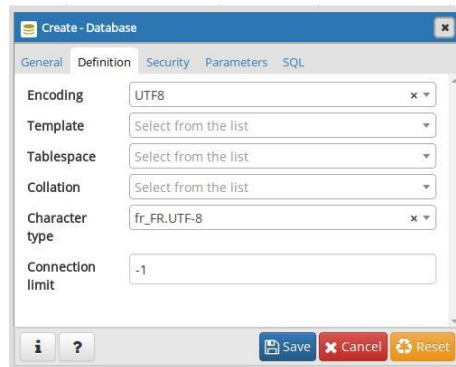


Saisir le nom de la base de données: *ocpizza_db*

Choisir le propriétaire de la BDD dans la **Combobox Owner**: *ocpizza*

Saisir un commentaire: *Base de données pour tester pgAdmin*

Activer, ensuite, le deuxième onglet **Définition**.



The screenshot shows the 'Create - Database' dialog box with the 'Definition' tab selected. The fields are as follows:

Field	Value
Encoding	UTF8
Template	Select from the list
Tablespace	Select from the list
Collation	Select from the list
Character type	fr_FR.UTF-8
Connection limit	-1

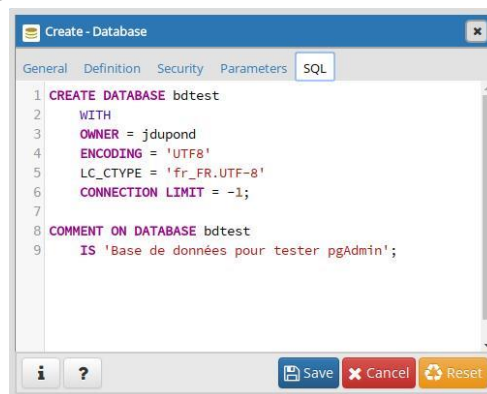
At the bottom, there are buttons for 'Save', 'Cancel', and 'Reset'.

Sélectionner dans la **Combobox Encoding**: *UTF8*

Sélectionner dans la **Combobox Character type**: *fr_FR.UTF-8*

Sélectionner, ensuite, le dernier onglet **SQL**

Après la saisie des paramètres de création de la base de données, on a la possibilité d'examiner le code SQL qui servira à sa création effective.



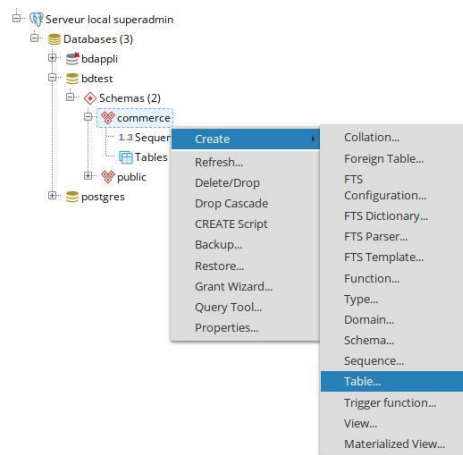
The screenshot shows the 'Create - Database' dialog box with the 'SQL' tab selected. The SQL code is as follows:

```
1 CREATE DATABASE bctest
2 WITH
3 OWNER = jdupond
4 ENCODING = 'UTF8'
5 LC_CTYPE = 'fr_FR.UTF-8'
6 CONNECTION LIMIT = -1;
7
8 COMMENT ON DATABASE bctest
9 IS 'Base de données pour tester pgAdmin';
```

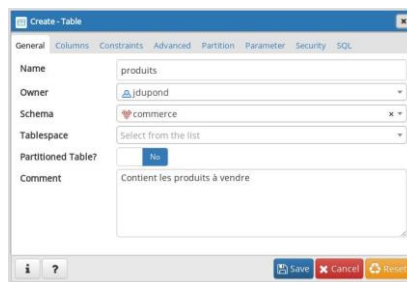
At the bottom, there are buttons for 'Save', 'Cancel', and 'Reset'.

3.2.1.4 - Créer une table

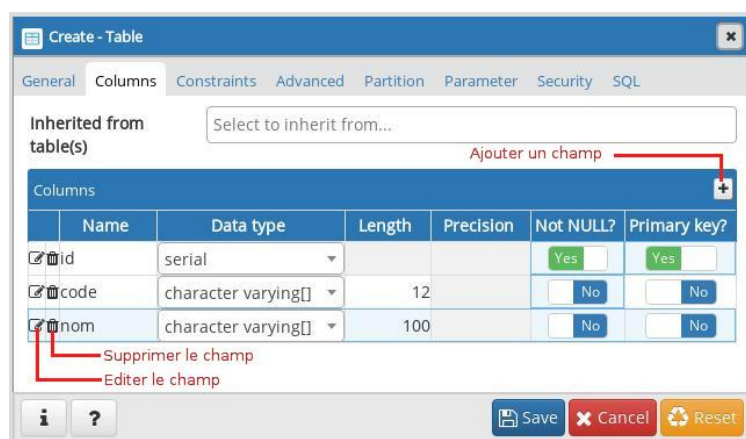
Pour créer une table dans le nouveau schéma créé, on utilise son menu contextuel: **Create / Table...**



Ce menu permet d'accéder à la fenêtre de création de table:



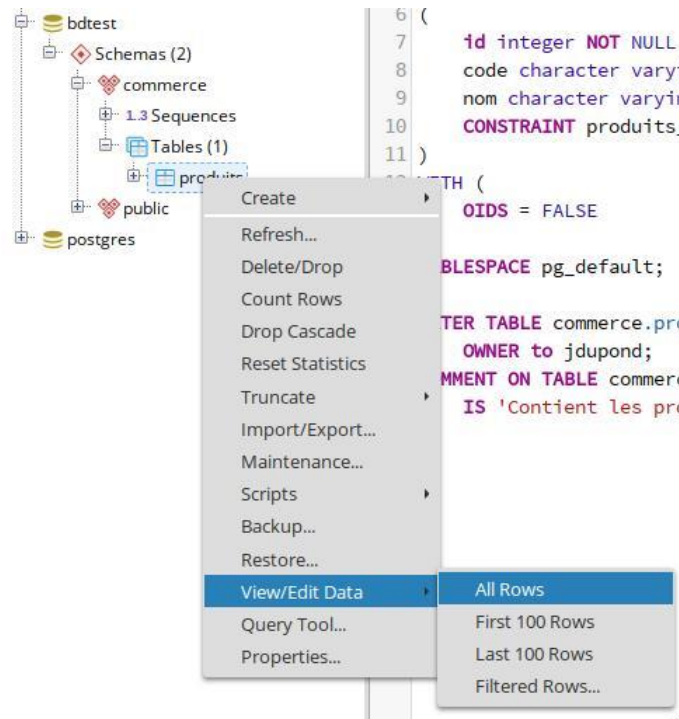
Renseigner les données de l'onglet **General** comme indiqué, puis sélectionner l'onglet **Columns** qui permet de définir les champs de la table.



S'agissant d'une **création** de table, le tableau de cet onglet est initialement **vide**. Utiliser le bouton mentionné « **Ajouter un champ** » pour ajouter les trois champs de la table. Après chaque ajout, paramétrer le champ comme indiqué puis enregistrer avec le bouton **Save** pour créer la table.

3.2.1.5 - Ajouter les données contenues dans une table

Pour ajouter les données contenues dans une table, on utilise son menu contextuel: **View/Edit Data / All Rows**. Les 3 autres menus servent quand il y a déjà une grande quantité de données.



Ce menu permet d'accéder à la zone d'édition des données de la table qui apparaît dans le volet droit de l'interface.

4 - PROCÉDURE DE DÉMARRAGE / ARRÊT

4.1 - Base de données

Pour lancer la base de données PostgreSQL l'utilisateur doit se connecter au serveur avec les identifiants fournis à la livraison.

4.1.1 - Procédure de démarrage

Pour démarrer PostgreSQL:

```
sudo systemctl start postgresql
```

4.1.2 - Procédure d'arrêt

Pour arrêter PostgreSQL:

```
sudo systemctl stop postgresql
```

4.2 - Application web

Pour lancer l'application web l'utilisateur devra se connecter au serveur NGINX avec les identifiants fournis à la livraison.

4.2.1 - Procédure de démarrage

Pour démarrer nginx:

```
sudo systemctl start nginx
```

4.2.2 - Procédure d'arrêt

Pour arrêter nginx:

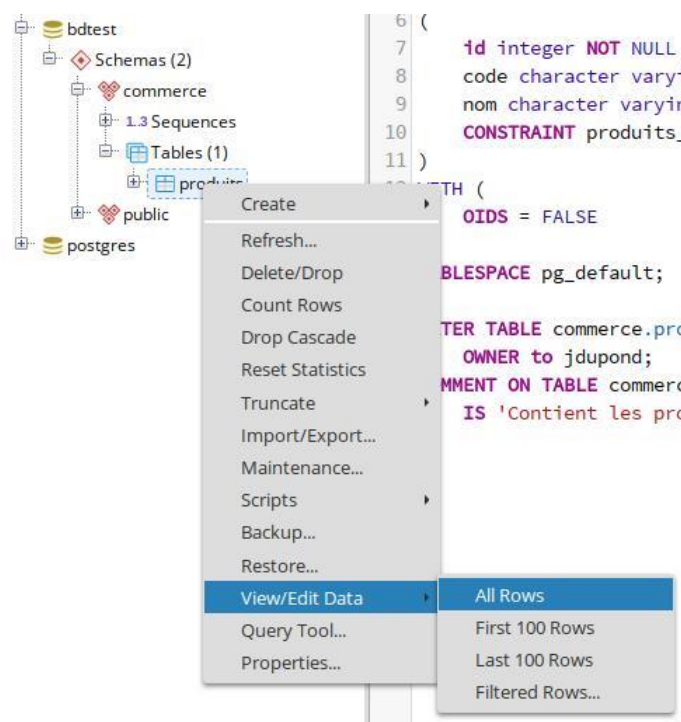
```
sudo systemctl stop nginx
```

5 - PROCÉDURE DE MISE À JOUR

5.1 - Base de données

5.1.1 - Modifier les données contenues dans une table

Pour modifier les données contenue dans une table, on utilise son menu contextuel: **View/Edit Data / All Rows**. Les 3 autres menus servent quand il y a déjà une grande quantité de données.



Ce menu permet d'accéder à la zone d'édition des données de la table qui apparaît dans le volet droit de l'interface.

5.2 - Application web

La mise à jour de l'application nécessite de la passer en maintenance. Lors de la mise en maintenance, les utilisateurs n'auront plus accès au site.

Pour mettre l'application web OC Pizza en mode maintenance. Il faut activer une variable géo par défaut et la faire générer une réponse d'état HTTP 503. Ensuite, Nginx livre la page de maintenance jusqu'à ce que nous la réactivions.

La mise en place d'une page de maintenance avec un en-tête 503 est nécessaire. Dans le fichier de configuration de l'hôte virtuel Nginx :

```
geo $maintenance_SITE {
    default off;          # Set to 'on' to enable maintenance mode
    123.123.123.120/29 off;
}

upstream maintenance {
    server maint:80;
}

server {
    listen 443 ssl http2;
    ...
    if ($maintenance_SITE = on) {
        return 503;
    }

    error_page 503 @maintenance;

    location @maintenance {
        rewrite ^(.*)$ / break;
        proxy_pass http://maintenance;
    }
    ...
}
```

6 - SUPERVISION/MONITORING

6.1 - Supervision de l'application web

Afin de tester que l'application web est toujours fonctionnelles, on utilise **Applications Manager*** comme solution de supervision.

La supervision des serveurs Nginx est impérative pour le bon fonctionnement des applications Web. L'outil de supervision Nginx d'Applications Manager fournit des capacités de surveillance Nginx complètes pour garantir des performances élevées et une disponibilité maximale.

6.1.1 - Réduire les temps d'arrêt de Nginx avec l'analyseur Nginx

- Surveiller la disponibilité du serveur Nginx.
- Prendre des mesures correctives en exécutant des scripts personnalisés pour démarrer Nginx en cas d'échec.

6.1.2 - Suivre les performances avec le moniteur Nginx d'Applications Manager

- Surveiller le temps de réponse.
- Surveiller les demandes servies par Nginx.
- Recevoir des alertes lorsque les performances sont dégradées.

6.1.3 - Assurer les performances des applications grâce à la supervision des applications Nginx

- Surveiller les connexions actives
- Surveiller les demandes en état d'attente.
- Effectuer une analyse détaillée des applications Nginx et prenez des mesures correctives lorsque davantage de demandes sont en attente pour garantir les performances de l'application.

6.1.4 - Dépanner rapidement pour une supervision efficace de Nginx

- Avec le puissant outil de gestion des pannes du système de supervision Nginx, la source des problèmes vous est signalée pour vous aider à corriger les pannes plus rapidement.
- Obtenir des graphiques de performances instantanés pour les attributs clés.
- Rapports qui peuvent être regroupés en fonction de la disponibilité, de l'état d'intégrité et du temps de connexion.
- Obtenir un aperçu des performances de votre serveur Nginx sur une période donnée

7 - PROCÉDURE DE SAUVEGARDE ET RESTAURATION

7.1 - Sauvegarde de la base de donnée

Le principe est de générer un fichier texte de commandes SQL (appelé « fichier dump »), qui, si on le renvoie au serveur, recrée une base de données identique à celle sauvegardée. PostgreSQL™ propose pour cela le programme utilitaire **pg_dump***. L'usage basique est :

```
pg_dump base_de_donnees > fichier_de_sortie
```

7.1.1 - Sauvegarde quotidienne postgresql

Nous allons créer un script permettant faire la sauvegarde de votre base données quotidiennement.

On édite le fichier **/etc/cron.daily/postgresql-Nom_BDD_db-backup** dans le quel on ajoute les éléments suivants :

```
vim /etc/cron.daily/postgresql-Nom_BDD_db-backup

#!/bin/bash

# Backup quotidien

su -c "pg_dump -F c -b -v -f /backup/database/postgresql/NOM_DE_LA_DB-$(date +"%Y-%m-%d-%H-%M-%S").dump NOM_DE_LA_DB" postgres

# Suppression des backups de plus de 10 jours

su -c "find /backup/database/postgresql/ -type f -mtime +10 -delete" postgres
```

Ensuite, on modifie les droits d'accès au fichier :

```
chmod +x /etc/cron.daily/postgresql-Nom_BDD_db-backup
```

NB: Les modifications et mise en place de sauvegarde sont automatiquement prises en compte.

7.2 - Restauration de la base de donnée

Les fichiers texte créés par **pg_dump** sont prévus pour être lus par le programme **psql**. La syntaxe générale d'une commande de restauration est

```
psql base_de_donnees < fichier_d_entree
```

8 - GLOSSAIRE

PostgreSQL	PostgreSQL est un système de gestion de base de données relationnelle orienté objet puissant et open source qui est capable de prendre en charge en toute sécurité les charges de travail de données les plus complexes.
Postgres pgAdmin 4	Postgres pgAdmin 4 permet de créer toute sorte d'objets du serveur de bases de données PostgreSQL. Ces objets peuvent être des bases de données (BDD), des schémas, des tables, des utilisateurs.. Cet outil permet également d'exécuter des requêtes SQL.
Nginx	NGINX est conçu pour offrir une faible utilisation de la mémoire et une grande simultanéité. Plutôt que de créer de nouveaux processus pour chaque requête Web, NGINX utilise une approche asynchrone et événementielle où les requêtes sont traitées dans un seul thread.
Directions	L'API Directions : Fournis les itinéraires pour les livreurs et affiche les informations sur le trafic en temps réel.
Stripe	Plateforme de traitement des paiements qui permet de transférer de l'argent du compte bancaire du client vers le compte d'une entreprise au moyen d'une transaction par carte de crédit
Applications Manager	Applications Manager est une solution d'analyse de la performance des applications qui assure un suivi détaillé du fonctionnement et de l'expérience utilisateur des applications stratégiques et des éléments d'infrastructure.
pg_dump	pg_dump est un outil de sauvegarde d'une base de données PostgreSQL™. Les sauvegardes réalisées sont cohérentes, même lors d'accès concurrents à la base de données. pg_dump ne bloque pas l'accès des autres utilisateurs (ni en lecture ni en écriture).