

Installation de Mariadb

1 Installation

Commençons par installer le serveur **MariaDB** (sur votre serveur principal) :

```
# installation
dnf module -y install mariadb:10.3

# lancement et activation
systemctl enable --now mariadb

# sécuriser (choisir le mot de passe 'hello')
mysql_secure_installation

# se connecter au serveur (-p pour donner le mot de passe)
mysql -p
```

Une fois connecté, avec les ordres ci-dessous, créez une BD, configurez les droits :

```
CREATE DATABASE moviesdb ;

GRANT ALL PRIVILEGES ON moviesdb.* TO moviesuser@localhost
IDENTIFIED BY 'moviespass' WITH GRANT OPTION;
```

Vous pouvez maintenant vous connecter avec cet utilisateur :

```
mysql -u moviesuser -p
```

Une fois connecté

```
USE moviesdb;

CREATE TABLE test_table (id int, name varchar(50), primary key (id));

INSERT INTO test_table(id, name) values(1, "CentOS");

SELECT * FROM test_table;
```

2 Utiliser cette base de données

Téléchargez une application Spring-boot qui travaille, par défaut, avec une base de données en mémoire et testez-là (en créant un tunnel `ssh` à partir de votre poste DOSI `ssh -L9000:localhost:8081 VM`) :

```
wget https://jean-luc-massat.pedaweb.univ-amu.fr/ens/asr/spring-app.zip
unzip spring-app.zip
java -jar spring-app.war
```

Nous allons maintenant lancer cette application en changeant les paramètres Spring-Boot afin d'utiliser la nouvelle base de données **MariaDB** :

Script spring-app.sh à préparer

```
java \  
-Dspring.datasource.driver-class-name=com.mysql.jdbc.Driver \  
-Dspring.datasource.url=jdbc:mysql://localhost/moviesdb \  
-Dspring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect \  
-Dspring.jpa.hibernate.ddl-auto=update \  
-Dspring.datasource.username=moviesuser \  
-Dspring.datasource.password=moviespass \  
-jar spring-app.war
```

Utilisez l'application et vérifiez que les tables sont correctement créées et remplies dans la base `moviesdb`.