

MATIÈRE : FOUILLE DE DONNÉES

GUIDE D'INSTALLATION DOCKER ET CONFIGURATION HADOOP/HIVE

Prérequis :

Outils et versions :

- Apache Hadoop Version : 2.7.2
- Apache hive 2.3.2
- Docker
- Git
- Java Version 1.8
- Windows 10
- Windows subsystem for linux

Présentation :

Hive traduit des requêtes écrites en HivQL(un dialecte de SQL influencé par MySQL) en un workflow de jobs MapReduce, puis soumet ces jobs sur le cluster Hadoop. Il ne s'agit donc pas d'une base de données, mais d'une couche d'abstraction au-dessus du framework MapReduce.

Installation Docker :

Etape 1 : Télécharger Docker

Lien :

- Windows/Mac : Téléchargez Docker Desktop depuis <https://www.docker.com/products/docker-desktop>
- Linux : Suivez les instructions d'installation spécifiques à votre distribution

Vérifiez que Docker est correctement installé :

Etape 2 : Ouvrez le terminal de commande de votre système d'exploitation et assurez-vous que Docker est correctement installé en saisissant la commande suivante :

Docker --version

```
Administrateur : Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https://aka.ms/PSWindows

PS C:\Users\ffode> Docker --version
Docker version 28.1.1, build 4eba377
PS C:\Users\ffode>
```

Etape 3 : Clonez le référentiel Docker Hadoop sur votre ordinateur en utilisant la commande suivante :

Git clone <https://github.com/Yanlou/docker-hive>

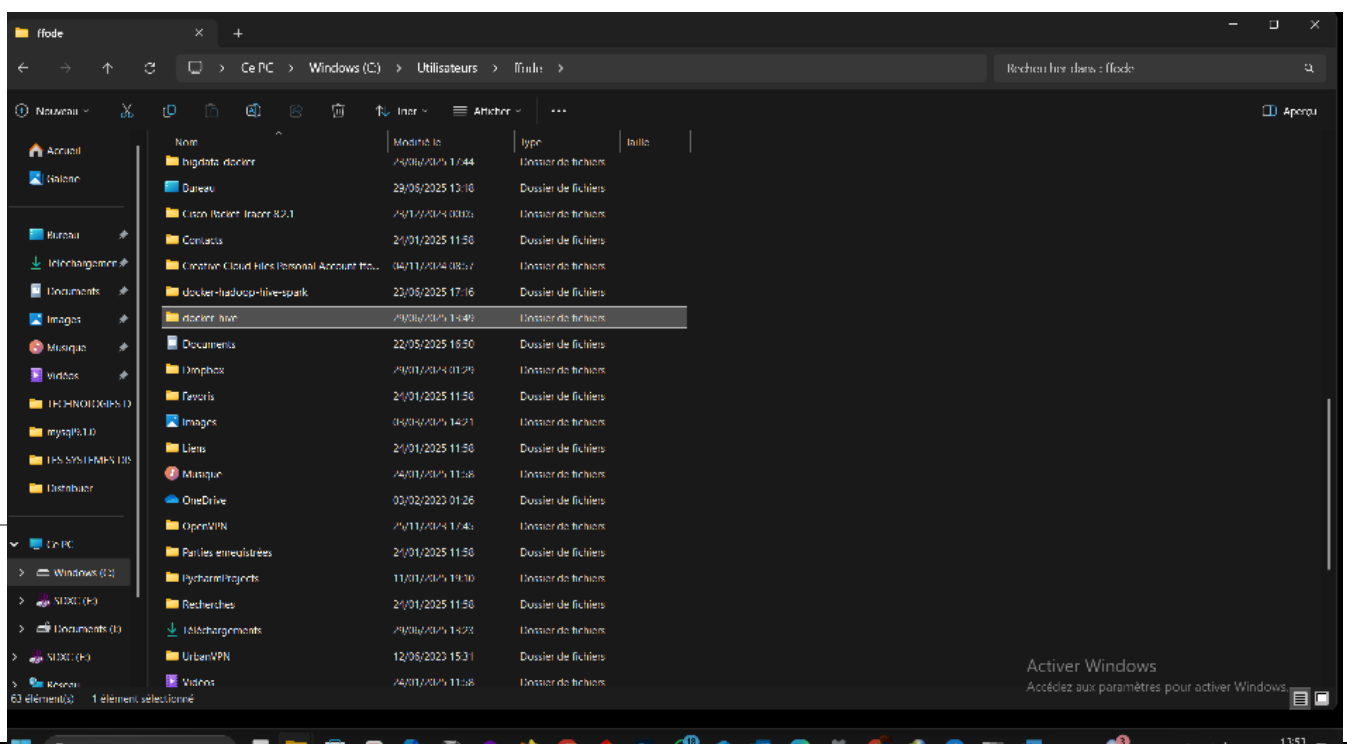
Resultat attendu:

```
Administrateur : Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https://aka.ms/PSWindows

PS C:\Users\ffode> Docker --version
Docker version 28.1.1, build 4eba377
PS C:\Users\ffode> Git clone https://github.com/Yanlou/docker-hive
Cloning into 'docker-hive'...
remote: Enumerating objects: 127, done.
remote: Total 127 (delta 0), reused 0 (delta 0), pack-reused 127 (from 1)
Receiving objects: 100% (127/127), 30.45 KiB | 842.00 KiB/s, done.
Resolving deltas: 100% (67/67), done.
PS C:\Users\ffode>
```

Retrouve le dossier dans le répertoire (docker-hive)

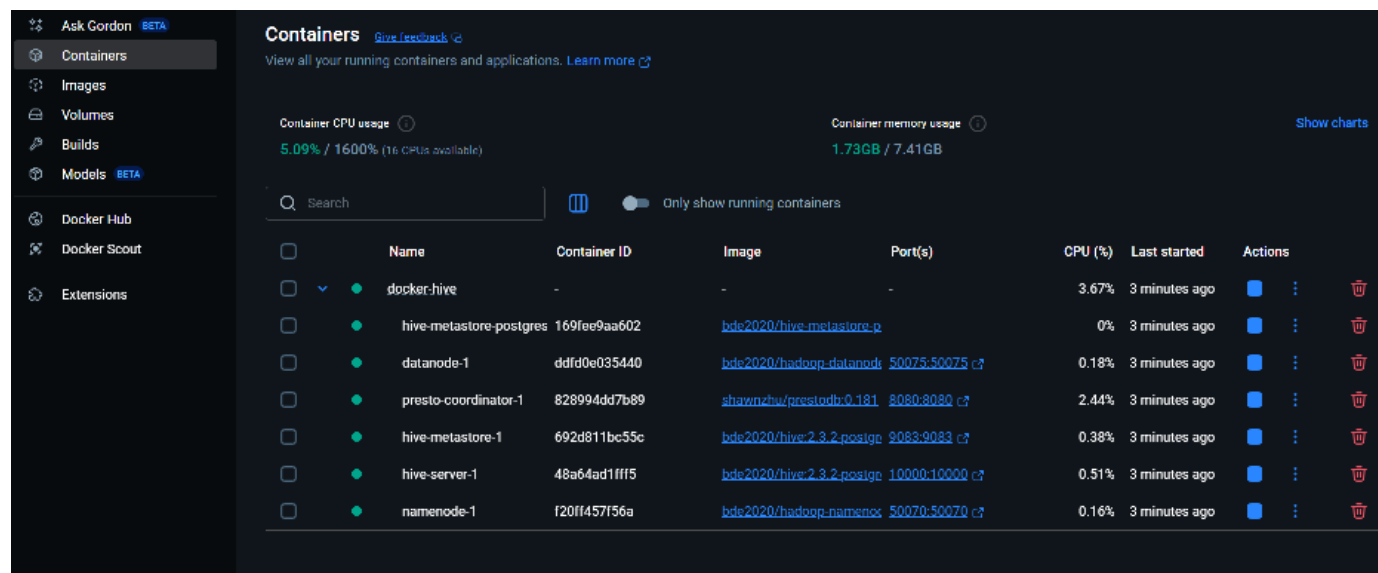


Docker-hive : Il s'agit d'un conteneur Docker pour Apache hive 2.3.2. Il est basé sur <https://github.com/Yanlou/docker-hive>, donc veuillez vérifier là-bas pour les configurations hadoop. Cela déploie Hive et lance un hiveserver2 sur le port 10000. Le metastore fonctionne avec une connexion à une base de données Postgresql. La configuration Hive est effectuée avec les variables HIVE_SITE_CONF_(voir hadoop-hive.env un exemple).

Pour exécuter hive avec le metastore PostgreSQL :

Docker-compose up -d

```
PS C:\Users\ffode\docker-hive> Docker-compose up -d
time="2025-06-29T15:27:49Z" level=warning msg="C:\\Users\\ffode\\docker-hive\\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 7/7
 ✓ Network docker-hive_default          Created           0.0s
 ✓ Container docker-hive-hive-server-1   Started           0.9s
 ✓ Container docker-hive-namenode-1      Started           0.9s
 ✓ Container docker-hive-hive-metastore-postgresql-1 Started...        0.6s
 ✓ Container docker-hive-datanode-1      Started           0.9s
 ✓ Container docker-hive-hive-metastore-1 Started           0.9s
 ✓ Container docker-hive-presto-coordinator-1 Started          0.7s
PS C:\Users\ffode\docker-hive>
```



The screenshot shows the Docker Desktop interface. On the left is a sidebar with navigation options: Ask Gordon, Containers (selected), Images, Volumes, Builds, Models, Docker Hub, Docker Scout, and Extensions. The main panel is titled 'Containers' and shows a summary of container usage: 5.09% / 1600% CPU and 1.73GB / 7.41GB memory. Below this is a table of running containers.

	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	docker-hive	-	-	-	3.67%	3 minutes ago	
<input type="checkbox"/>	hive-metastore-postgres	169f6e9aa602	bde2020/hive-metastore-p		0%	3 minutes ago	
<input type="checkbox"/>	datanode-1	ddfd0e035440	bde2020/hadoop-datanode	50075:50075 c?	0.18%	3 minutes ago	
<input type="checkbox"/>	presto-coordinator-1	828994dd7b89	shawnzhu/presto-dc0.181	8080:8080 c?	2.44%	3 minutes ago	
<input type="checkbox"/>	hive-metastore-1	692d811bc55c	bde2020/hive:2.3.2-postgre	9083:9083 c?	0.38%	3 minutes ago	
<input type="checkbox"/>	hive-server-1	48a64ad1fff5	bde2020/hive:2.3.2-postgre	10000:10000 c?	0.51%	3 minutes ago	
<input type="checkbox"/>	namenode-1	f20ff457f56a	bde2020/hadoop-namenode	50070:50070 c?	0.16%	3 minutes ago	

Au niveau de Docker Destop :

Voici la description des conteneurs inclus dans Docker-Hive :

- « **hive-metastore-postgresql-1** » : ce conteneur contient le service de métastore Hive avec PostgreSQL comme base de données. Le métastore stocke les métadonnées des tables, des partitions et des bases de données Hive.
- « **hive-server-1** » : ce conteneur contient le serveur Hive, qui fournit une interface de requête SQL pour interagir avec les données stockées dans le cluster Hive.
- « **namenode-1** » et « **datanode-1** » : ces deux conteneurs contiennent les services de stockage de données de base du système de fichiers Hadoop (HDFS). Le nœud de nom (« namenode ») gère le système de fichiers et les métadonnées, tandis que les nœuds de données (« datanodes ») stockent les données elles-mêmes.
- « **presto-coordinator-1** » : ce conteneur contient le service de coordination Presto, qui permet l'exécution de requêtes ad hoc sur des données stockées dans le cluster Hive (ainsi que d'autres sources de données).
- « **hive-metastore-1** » : ce conteneur contient un deuxième service de métastore Hive pour la haute disponibilité.

Télécharger le fichier purchases.txt sur le lien : <https://raw.githubusercontent.com/Yanlou/udacity-hadoop-course/master/Datasets/purchases.txt.gz>

Après le télécharger , extrait le fichier et placé dans un répertoire accessible pour pouvoir charger.

Copier le fichier purchases.txt dans le conteneur hive-server-1

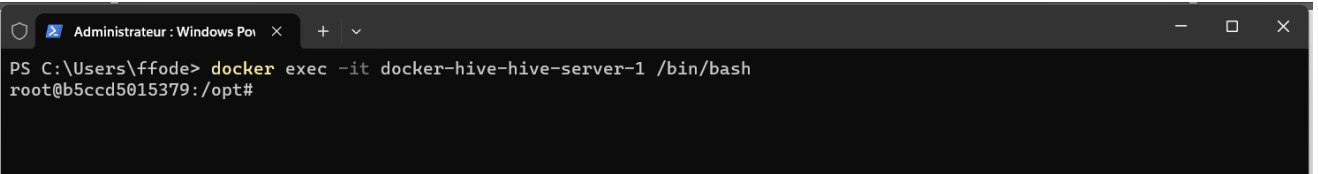
`docker cp i:/purchases.txt docker-hive-hive-server-1:/opt/purchases.txt`

```
PS C:\Users\ffode> docker cp i:/purchases.txt docker-hive-hive-server-1:/opt/purchases.txt
Successfully copied 211MB to docker-hive-hive-server-1:/opt/purchases.txt
```

Premier pas avec Hive (charger des données dans hive purchases.txt) :

Déployez le serveur hive :

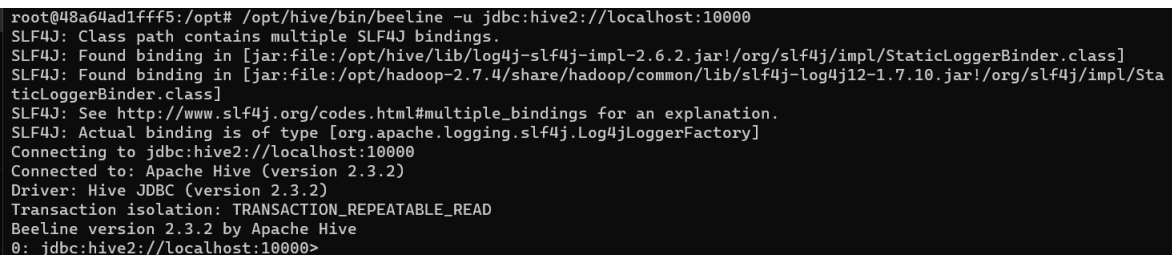
`docker exec -it docker-hive-hive-server-1 /bin/bash`



```
Administrateur : Windows Poi x + v
PS C:\Users\ffode> docker exec -it docker-hive-hive-server-1 /bin/bash
root@b5ccd5015379:/opt#
```

Lancer le client beeline : Il se connecte à HiveServer2 via JDBC. Il exécute des requêtes HiveQL. Il remplace l'ancien hive CLI qui se connectait directement au metastore (ce qui est maintenant déconseillé).

`/opt/hive/bin/beeline -u jdbc:hive2://localhost:10000`



```
root@48a64ad1fff5:/opt# /opt/hive/bin/beeline -u jdbc:hive2://localhost:10000
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/hive/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/hadoop-2.7.4/share/hadoop/common/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Connecting to jdbc:hive2://localhost:10000
Connected to: Apache Hive (version 2.3.2)
Driver: Hive JDBC (version 2.3.2)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 2.3.2 by Apache Hive
0: jdbc:hive2://localhost:10000>
```

La structure du fichier purchases.txt est de la forme suivante :

```
date temps magasin produit cout paiement
```

CRÉATION DE LA TABLE CONFORME A LA STRUCTURE CI HAUT.

CREATE TABLE purchases (

 `date` STRING,

 `time` STRING,

 store STRING,

 product STRING,

```
cost DOUBLE,  
payment STRING  
)  
ROW FORMAT DELIMITED  
FIELDS TERMINATED BY ','  
STORED AS TEXTFILE;
```

```
0: jdbc:hive2://localhost:10000> CREATE TABLE purchases (  
.....> 'date' STRING,  
.....> 'time' STRING,  
.....> store STRING,  
.....> product STRING,  
.....> cost DOUBLE,  
.....> payment STRING  
.....> )  
.....> ROW FORMAT DELIMITED  
.....> FIELDS TERMINATED BY ','  
.....> STORED AS TEXTFILE;  
No rows affected (1.322 seconds)  
0: jdbc:hive2://localhost:10000>
```

Charger les données du purchases.txt dans notre nouvelle table :

```
LOAD DATA LOCAL INPATH '/opt/purchases.txt' INTO TABLE purchases;
```

```
0: jdbc:hive2://localhost:10000> LOAD DATA LOCAL INPATH '/opt/purchases.txt' INTO TABLE purchases;  
No rows affected (1.83 seconds)  
0: jdbc:hive2://localhost:10000>
```

1) Obtenir le cout total de tous les achats

```
SELECT SUM(cost) AS total_cost FROM purchases;
```

2) Obtenir le nombre d'achats effectués dans chaque magasin

```
- SELECT store, COUNT(*) AS n_achats FROM purchases GROUP BY store;
```

3) Obtenir le cout total des achats effectués pour chaque produit

- SELECT product, SUM(cost) AS total FROM purchases GROUP BY product;

4) Obtenir le cout total des achats effectués pour chaque produit dans chaque magasin

- SELECT product, SUM(cost) AS total FROM purchases GROUP BY product;

5) Obtenir les 10 produits les plus vendus, classés par ordre décroissant

```
SELECT product, COUNT(*) AS total_sales
FROM purchases
GROUP BY product
ORDER BY total_sales DESC
LIMIT 10;
```

6) Obtenir la liste des magasins et leur cout total de ventes triés par ordre décroissant

```
SELECT store, SUM(cost) AS total_sales FROM purchases
GROUP BY store
ORDER BY total_sales DESC;
```

7) Obtenir le nombre de ventes effectuées chaque jour

```
SELECT TO_DATE(date) AS sale_day, COUNT(*) AS nb_ventes
FROM purchases
GROUP BY TO_DATE(date)
ORDER BY sale_day;
```

8) Obtenir la moyenne des couts des ventes par produit

```
SELECT product, AVG(cost) AS avg_cost
FROM purchases
GROUP BY product;
```

9) Obtenir les achats effectués par mode paiement et leur cout total

```
SELECT payment_method, SUM(cost) AS total_cost
FROM purchases
GROUP BY payment_method;
```

10) Obtenir les produits vendus le plus souvent le lundi

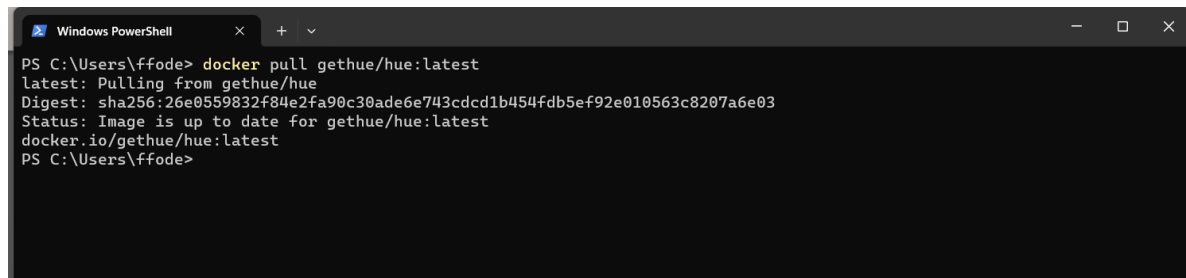
```
SELECT product, COUNT(*) AS nb_ventes
FROM purchases
WHERE dayofweek(date) = 2 - 1 = dimanche, 2 = lundi, etc.
GROUP BY product
ORDER BY nb_ventes DESC;
```

7. INTERFACE WEB HUE POUR HIVE (OPTIONNEL)

installation de hue

Télécharger et démarrer Hue

docker pull gethue/hue:latest



```
Windows PowerShell
PS C:\Users\ffode> docker pull gethue/hue:latest
latest: Pulling from gethue/hue
Digest: sha256:26e0559832f84e2fa90c30ade6e743cdcd1b454fdb5ef92e010563c8207a6e03
Status: Image is up to date for gethue/hue:latest
docker.io/gethue/hue:latest
PS C:\Users\ffode>
```

#Lancer le conteneur

docker run -it -p 8888:8888 gethue/hue:latest bash


```
Windows PowerShell
PS C:\Users\ffode> docker run -it -p 8888:8888 gethue/hue:latest bash
hue@2048115894e7:/usr/share/hue$
```

configuration de hue # À l'intérieur du conteneur

Hue

Démarrer le serveur Hue

./build/env/bin/hue runserver_plus 0.0.0.0:8888

```
hue@2048115894e7:/usr/share/hue$ ./build/env/bin/hue runserver_plus 0.0.0.0:8888
[29/Jun/2025 16:14:06 ] settings INFO Welcome to Hue 4.11.0
[29/Jun/2025 09:14:06 -0700] backend WARNING mozilla_django_oidc module not found
[29/Jun/2025 09:14:08 -0700] apps INFO AXES: BEGIN LOG
[29/Jun/2025 09:14:08 -0700] apps INFO AXES: Using django-axes version 5.13.0
[29/Jun/2025 09:14:08 -0700] apps INFO AXES: blocking by IP only.
CommandError: Werkzeug is required to use runserver_plus. Please visit http://werkzeug.pocoo.org/ or install via pip. (
pip install Werkzeug)
hue@2048115894e7:/usr/share/hue$ |
```

source build/env/bin/activate

```
hue@2048115894e7:/usr/share/hue$ source build/env/bin/activate
(env) hue@2048115894e7:/usr/share/hue$
```

pip install Werkzeug

```
(env) hue@2048115894e7:/usr/share/hue$ pip install Werkzeug
WARNING: The directory '/home/hue/.cache/pip' or its parent directory is not owned or is not writable by the current use
r. The cache has been disabled. Check the permissions and owner of that directory. If executing pip with sudo, you shoul
d use sudo's -H flag.
Collecting Werkzeug
  Obtaining dependency information for Werkzeug from https://files.pythonhosted.org/packages/6c/69/05837f91dfe42109203ff
a3e488214ff86a6d68b2ed6c167da6cdc42349b/werkzeug-3.0.6-py3-none-any.whl.metadata
  Downloading werkzeug-3.0.6-py3-none-any.whl.metadata (3.7 kB)
Requirement already satisfied: MarkupSafe>=2.1.1 in ./build/env/lib/python3.8/site-packages (from Werkzeug) (2.1.5)
  Downloading werkzeug-3.0.6-py3-none-any.whl (227 kB)
    228.0/228.0 kB 94.3 kB/s eta 0:00:00
DEPRECATION: celery 4.4.7 has a non-standard dependency specifier pytz>dev. pip 23.3 will enforce this behaviour change.
A possible replacement is to upgrade to a newer version of celery or contact the author to suggest that they release a
version with a conforming dependency specifiers. Discussion can be found at https://github.com/pypa/pip/issues/12063
Installing collected packages: Werkzeug
Successfully installed Werkzeug-3.0.6

[notice] A new release of pip is available: 23.2.1 -> 25.0.1
[notice] To update, run: pip install --upgrade pip
(env) hue@2048115894e7:/usr/share/hue$
```

Ensuite :

./build/env/bin/hue runserver_plus 0.0.0.0:8888

```
(env) hue@2048115894e7:/usr/share/hue$ ./build/env/bin/hue runserver_plus 0.0.0.0:8888
[29/Jun/2025 16:18:02 ] settings INFO Welcome to Hue 4.11.0
[29/Jun/2025 09:18:02 -0700] backend WARNING mozilla_django_oidc module not found
[29/Jun/2025 09:18:03 -0700] apps INFO AXES: BEGIN LOG
[29/Jun/2025 09:18:03 -0700] apps INFO AXES: Using django-axes version 5.13.0
[29/Jun/2025 09:18:03 -0700] apps INFO AXES: blocking by IP only.
[29/Jun/2025 09:18:03 -0700] jdbc WARNING Failed to import py4j
[29/Jun/2025 09:18:03 -0700] api3 WARNING simple_salesforce module not found
[29/Jun/2025 09:18:04 -0700] schemas INFO Include schema from 'file:///usr/share/hue/build/env/lib/python3.8/site-packages/xmlschema/schemas/XSD_1.1/xsd11-extra.xsd'
[29/Jun/2025 09:18:04 -0700] urls WARNING.djangosaml2 module not found
[29/Jun/2025 09:18:05 -0700] middleware INFO Unloading MimeTypeJSFileFixStreamingMiddleware
[29/Jun/2025 09:18:05 -0700] middleware INFO Unloading CacheControlMiddleware
[29/Jun/2025 09:18:05 -0700] middleware INFO Unloading HueRemoteUserMiddleware
[29/Jun/2025 09:18:05 -0700] middleware INFO Unloading SpnegoMiddleware
[29/Jun/2025 09:18:05 -0700] middleware INFO Unloading ProxyMiddleware
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:8888
* Running on http://172.17.0.2:8888
Press CTRL+C to quit
* Restarting with stat
[29/Jun/2025 09:18:05 -0700] settings INFO Welcome to Hue 4.11.0
[29/Jun/2025 09:18:06 -0700] backend WARNING mozilla_django_oidc module not found
```

L'interface Hue sera accessible sur <http://localhost:8888>

8. COMMANDES UTILES POUR LA GESTION DES CONTENEURS

gestion des conteneurs

Voir les conteneurs en cours d'exécution

docker ps

Arrêter un conteneur docker stop nom_conteneur /id_conteneur

Redémarrer un conteneur

docker start nom_conteneur/id_conteneur

Voir les logs d'un conteneur

docker logs nom_conteneur /id_conteneur

Supprimer un conteneur

`docker rm nom_conteneur/id_conteneur`

sauvegarde et restauration

Créer une image depuis un conteneur modifié

`docker commit nom_conteneur/id_conteneur my-image:latest`

Exporter les données docker `exec hive-server tar czf /tmp/`

`backup.tar.gz /opt/hive/warehouse docker cp hive-server:/tmp/`

`backup.tar.gz ./backup.tar.gz`

CONSEILS ET BONNES PRATIQUES

1. ****Persistance des données**** : Utilisez des volumes Docker pour persister vos données

2. ****Monitoring**** : Surveillez l'utilisation des ressources avec ``docker stats``

3. ****Sécurité**** : Ne pas exposer les ports en production sans authentification

4. ****Performance**** : Allouez suffisamment de mémoire au conteneur pour Hadoop/Hive

DÉPANNAGE problèmes courants

- ****Erreur de connexion Beeline**** : Vérifiez que les services Hive sont démarrés

- ****Manque de mémoire**** : Augmentez la mémoire allouée à Docker

****Problèmes de permissions**** : Vérifiez les droits sur les fichiers de données commandes de diagnostic

Vérifier l'état des services dans le conteneur

`docker exec hive-server jps`

Voir les logs de Hive `docker exec hive-server cat /tmp/hive.log`