

1 Set Environment Variables

Edit your shell configuration (`~/.zshrc` or `~/.bashrc`) and add:

```
# Hadoop setup
export HADOOP_HOME=/Users/yourname/Downloads/hadoop-3.3.6
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin

# Hive setup
export HIVE_HOME=/usr/local/hive
export PATH=$PATH:$HIVE_HOME/bin

# Java setup (required by Hadoop/Hive)
export JAVA_HOME=$(/usr/libexec/java_home)

# Optional: Zookeeper if used
export ZOOKEEPER_HOME=/opt/homebrew/Cellar/zookeeper/3.8.1/libexec
export PATH=$PATH:$ZOOKEEPER_HOME/bin
```

Reload:

```
source ~/.zshrc
```

2 Configure `hive-site.xml`

1. Go to `$HIVE_HOME/conf/`
2. Copy template:

```
cp hive-default.xml.template hive-site.xml
```

```
/usr/local/hive/conf/hive-site.xml
```

3. Edit `hive-site.xml` to configure key properties. Example minimal setup:

```
<configuration>
```

```
<!-- Hive metastore -->
<property>
  <name>hive.metastore.warehouse.dir</name>
  <value>/Users/yourname/hive/warehouse</value>
</property>

<!-- HiveServer2 authentication (NONE for local testing) -->
<property>
  <name>hive.server2.authentication</name>
  <value>NONE</value>
</property>

<!-- Metastore DB (embedded Derby for local testing) -->
<property>
  <name>javax.jdo.option.ConnectionURL</name>
  <value>jdbc:derby:;databaseName=/Users/yourname/hive/metastore_db;create=true</value>
</property>
</configuration>
```

For production, replace Derby with MySQL/Postgres.

3 Create Hive directories

```
mkdir -p /Users/dhandapanidhandapaniyedappalli/hive/warehouse
mkdir -p /Users/dhandapanidhandapaniyedappalli/hive/metastore_db
```

- These are used for table storage and metastore DB.
-

4 Initialize the Metastore

If using **embedded Derby** (local testing):

```
schematool -initSchema -dbType derby
```

- For MySQL/Postgres: replace `-dbType derby` with `mysql` or `postgres`.
-

5 Start Hive services

- **Hive Metastore** (optional but recommended):

```
hive --service metastore &
```

- **HiveServer2** (for Beeline):

```
hiveserver2 &
```

- **Beeline** (to connect):

```
beeline -u "jdbc:hive2://localhost:10000/default"
```

6 Test Hive

```
SHOW DATABASES;  
CREATE TABLE test_table (id INT, name STRING);  
INSERT INTO test_table VALUES (1, 'Alice');  
SELECT * FROM test_table;
```

Optional Configs

- **Logging**: copy or create `log4j.properties` in `$HIVE_HOME/conf` to avoid warnings.
- **Zookeeper**: if using HA HiveServer2.
- **Metastore DB**: use MySQL/Postgres for production.

Step 5: Connect via UI

Option A: Hue (Recommended)

- Install Hue via Homebrew (experimental, may require manual setup):

```
brew install hue
```

- Start Hue:

```
hue runserver
```

- Access UI: <http://localhost:8888> → Go to Hive Query Editor

Option B: DBeaver (Third-party GUI)

- Download: <https://dbeaver.io/download/>
- Add a Hive connection:
 - **Host:** `localhost`
 - **Port:** `10000`
 - **Database:** `default`
 - **Username:** your macOS username (or Hive user)

Hive metastore in PostgreSQL

Step 1: Install PostgreSQL

If you don't have it yet:

```
brew install postgresql  
brew services start postgresql
```

Create a database + user for Hive:

```
psql postgres
```

Inside psql:

```
CREATE DATABASE hive_metastore;  
CREATE USER hiveuser WITH PASSWORD 'hivepass';  
GRANT ALL PRIVILEGES ON DATABASE hive_metastore TO hiveuser;  
\q
```

Step 2: Get PostgreSQL JDBC Driver

Hive needs the Postgres driver JAR.

```
brew install postgresql@14 # if not already
```

Or download JDBC JAR (example: `postgresql-42.7.3.jar`) and copy it to Hive lib:

```
cp ~/Downloads/postgresql-42.7.3.jar $HIVE_HOME/lib/
```

Step 3: Configure Hive Metastore

Edit `hive-site.xml` in `$HIVE_HOME/conf/`. Add these properties:

```
<configuration>
  <property>
    <name>javax.jdo.option.ConnectionURL</name>
    <value>jdbc:postgresql://localhost:5432/hive_metastore</value>
  </property>

  <property>
    <name>javax.jdo.option.ConnectionDriverName</name>
    <value>org.postgresql.Driver</value>
  </property>

  <property>
    <name>javax.jdo.option.ConnectionUserName</name>
    <value>hiveuser</value>
  </property>

  <property>
    <name>javax.jdo.option.ConnectionPassword</name>
    <value>hivepass</value>
  </property>

  <property>
    <name>datanucleus.autoCreateSchema</name>
    <value>>false</value>
  </property>

  <property>
    <name>datanucleus.fixedDatastore</name>
    <value>true</value>
  </property>
</configuration>
```

Step 4: Initialize Schema

Now initialize Hive schema in Postgres:

```
schematool -initSchema -dbType postgres
```

If success, you'll see `Schema initialization complete`.

Step 5: Start Hive Services

Start metastore:

```
hive --service metastore
```

(leave running in one terminal)

Start HiveServer2:

```
hive --service hiveserver2
```

Step 6: Connect from Beeline or DBeaver

Beeline:

```
beeline -u jdbc:hive2://localhost:10000
```

DBeaver:

- Driver: Hive2
- URL: `jdbc:hive2://localhost:10000/default`
- User: (leave blank or use `hiveuser`)
- Password: blank