

---

# BIG DATA

HUE



EDUARD LARA

# 1. INTRODUCCION

---

- ❖ HUE es un producto open source aunque está bajo el paraguas de Cloudera, una de las distribuciones de Hadoop
- ❖ Es un entorno gráfico que permite trabajar de manera mucho más amigable con un entorno Hadoop, con distintas características:
- ❖ Tiene una serie de editores que nos permiten trabajar con bases de datos Hive, Impala, etc
- ❖ Dispone de una especie de editor donde se pueden crear scripts y comandos sin tener que hacerlos desde el modo en línea.
- ❖ Tiene un dashboard que permite cargar determinada información y a partir de ahí sacar gráficos y otra serie de características
- ❖ Tiene un planificador con el cual podemos lanzar determinados trabajos a través de uno de esos productos que se llama Oozie dentro de Hadoop Big Data y nos permita sobretodo ejecutar trabajos de manera automática y de forma regular
- ❖ También es un navegador de componentes: permite navegar por HDFS o acceder al metastore de Hive para ver las BD, los ficheros<sup>2</sup>

# 1. INTRODUCCION

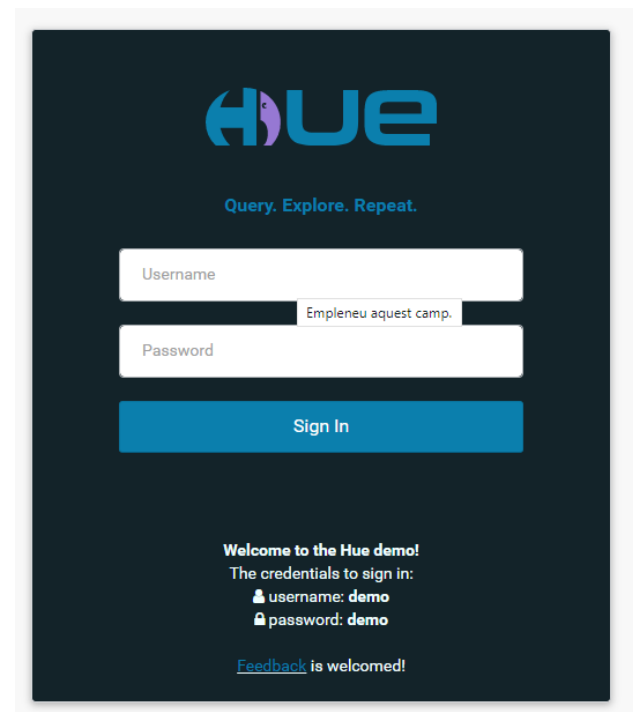
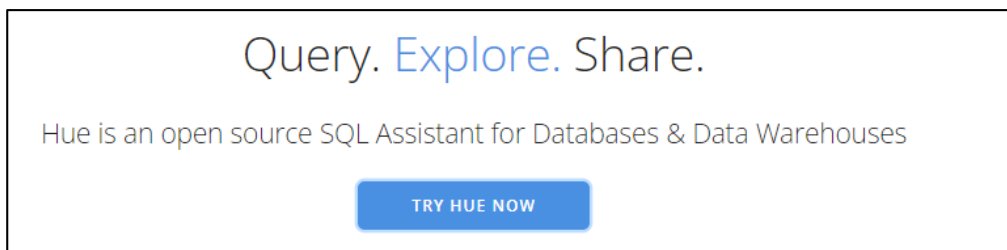
---

	HIVE	HUE
<b>Inventor / invención</b>	Hive fue lanzado por Apache Software Foundation.	Hue fue lanzado por Cloudera.
<b>Alcance / Significado</b>	Hive o HiveQL es un lenguaje de consulta analítica utilizado para procesar y recuperar datos de un almacén de datos.	Hue es una interfaz de usuario web que facilita a los usuarios interactuar con el ecosistema de Hadoop.
<b>Instalación / Configuración</b>	Hive se puede instalar o configurar mediante la interfaz de línea de comandos de un ecosistema de Hadoop.	Hue se puede instalar o configurar solo con un navegador web.
<b>Funcionalidad</b>	Hive utiliza un algoritmo de reducción de mapas para procesar y analizar los datos.	Hue proporciona un editor de interfaz de usuario web para acceder a Hive y otros lenguajes de programación.
<b>Implementación</b>	Hive se implementa y se accede a él mediante una interfaz de línea de comandos o una interfaz de interfaz de usuario web.	Hue se implementa en un navegador web para acceder a múltiples programas instalados en Cloudera.
<b>Dependencia</b>	Hive se puede incrustar en varios marcos de Hadoop.	Hue solo está disponible en el Framework Hadoop basado en Cloudera.

## 2. PASEO POR GETHUE.COM

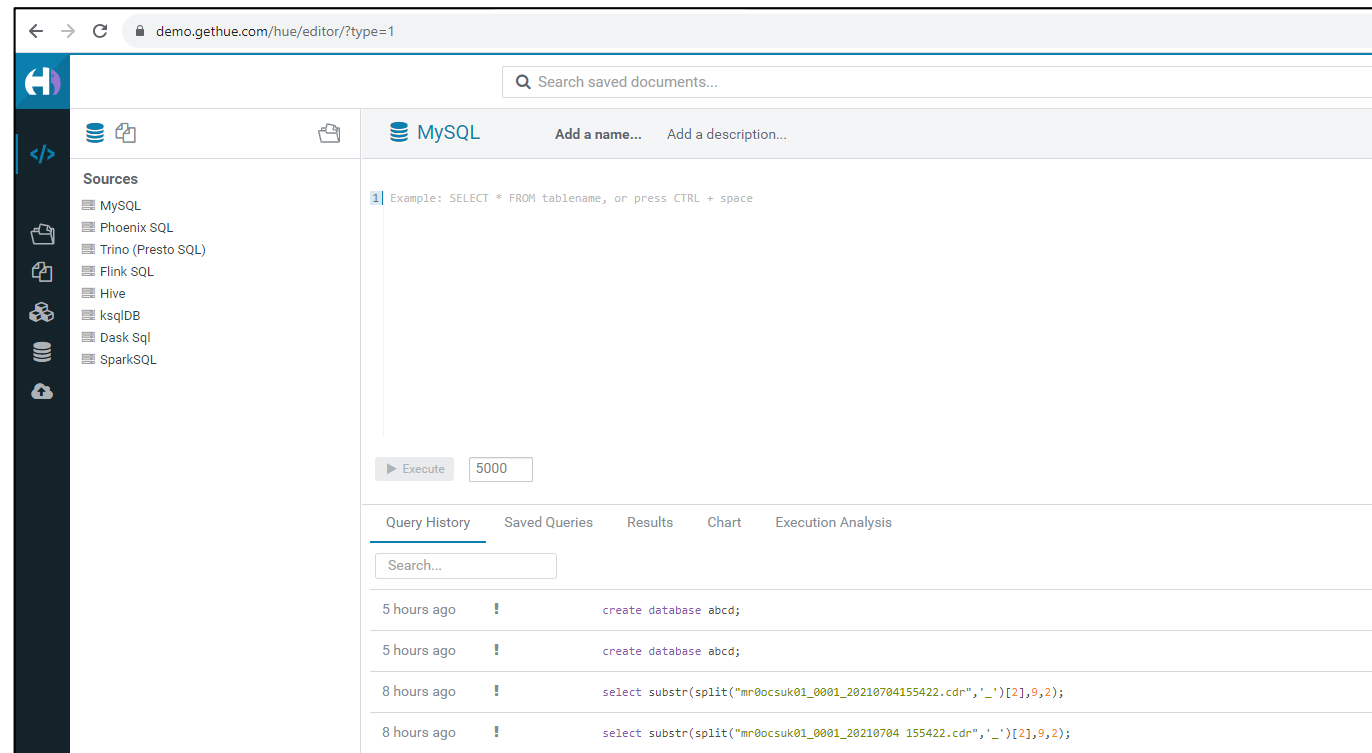
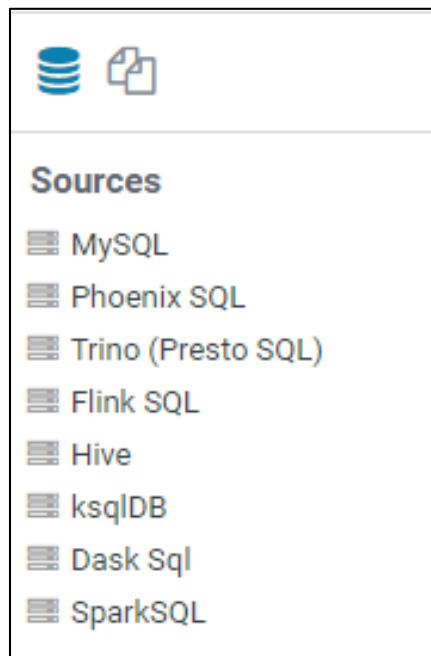
**Paso 1.** En la página web [gethue.com](https://gethue.com) podemos encontrar este producto. Se puede probar antes de instalarlo. El botón **Try Hue Now** nos lleva a un entorno en la nube ya preparado, que nos permite ver distintas funcionalidades.

Las credenciales para entrar son: usuario demo y password demo.



## 2. PASEO POR GETHUE.COM

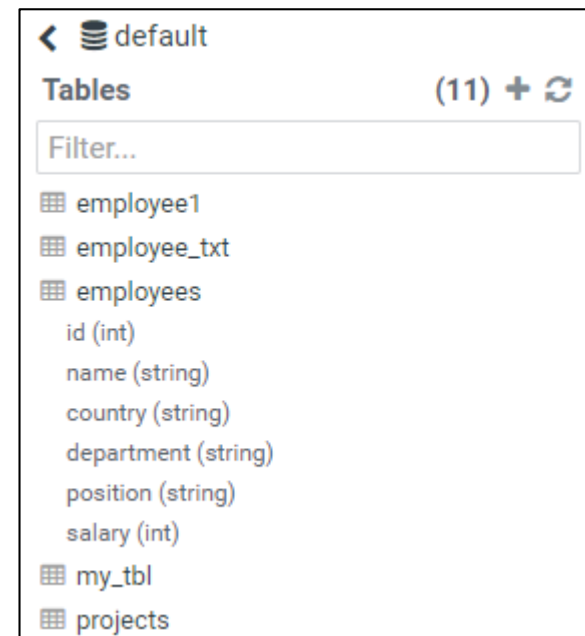
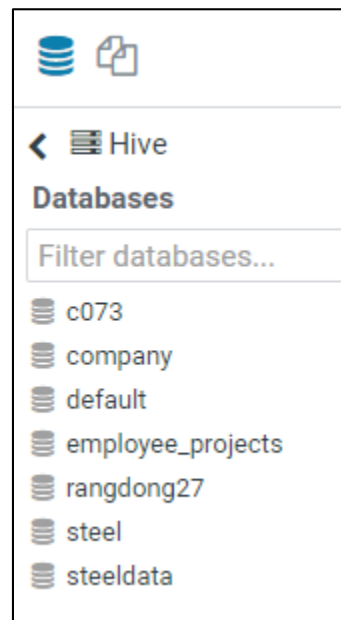
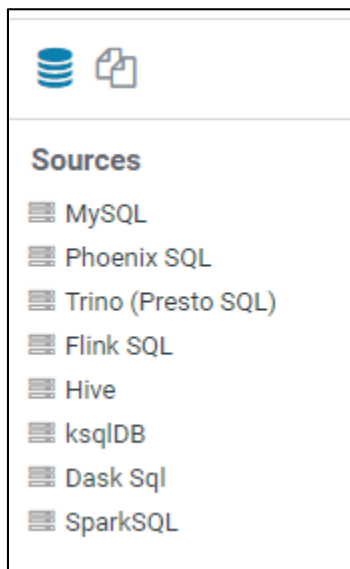
**Paso 2.** Aparece un entorno gráfico desde el que nos podemos conectar a diferentes base de datos (Mysql, Hive, SparkSQL, etc) En la parte central tenemos una especie de editor grafico con los comandos que se han ido dando



## 2. PASEO POR GETHUE.COM

**Paso 3.** En la parte izquierda podemos acceder a diferentes fuentes de base de datos (Hive, Mysql, SparkSQL, etc). Podemos hacer:

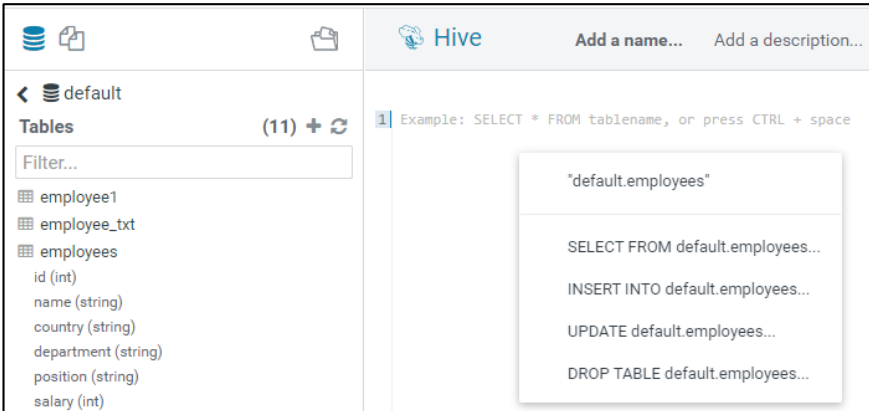
- Si seleccionamos Hive podemos ver sus databases.
- Si seleccionamos la database default, vemos sus tablas.
- Si seleccionamos la tabla employees vemos las columnas que tiene y su estructura



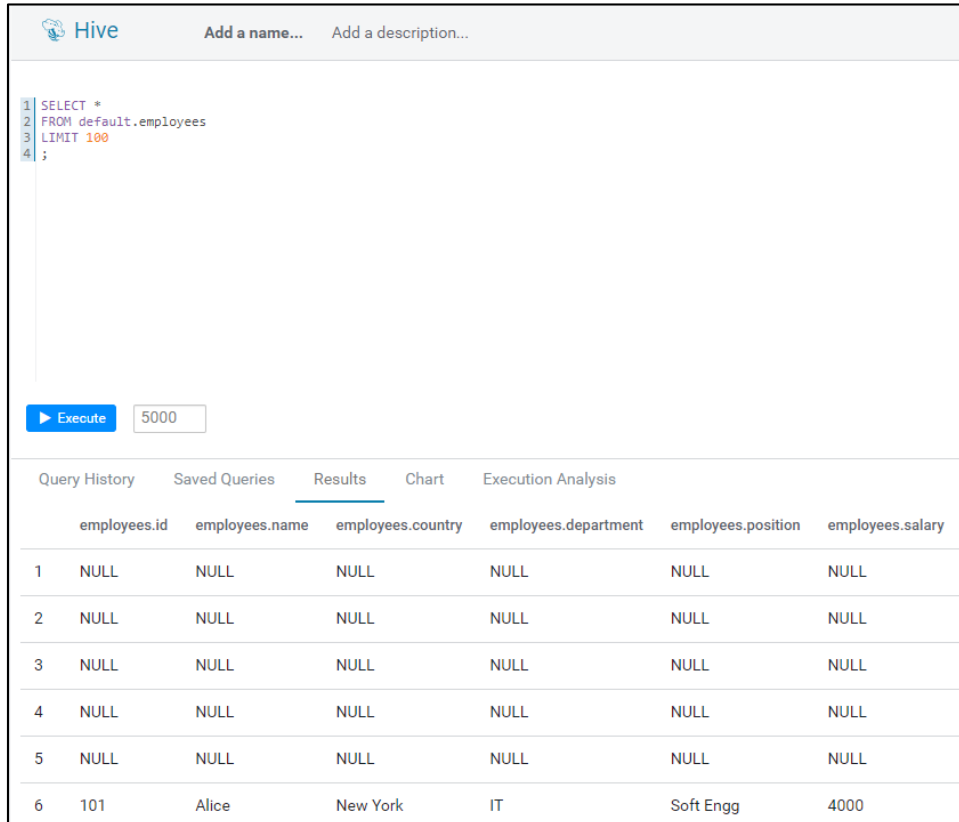
## 2. PASEO POR GETHUE.COM

**Paso 4.** Podemos poner la consulta de forma automática, arrastrando la tabla al editor y ejecutarla para ver los resultados

Desde aquí es mucho mas fácil trabajar con Hive, todo a nivel grafico, una especie de workbench para Hive



The screenshot shows the Hive web interface. On the left, there's a sidebar with a 'Tables' section showing a list of tables: employee1, employee\_txt, and employees. The 'employees' table is selected, and its schema is displayed: id (int), name (string), country (string), department (string), position (string), and salary (int). In the main editor area, there's a text input with the placeholder 'Example: SELECT \* FROM tablename, or press CTRL + space'. A context menu is open over the editor, showing options: 'default.employees', 'SELECT FROM default.employees...', 'INSERT INTO default.employees...', 'UPDATE default.employees...', and 'DROP TABLE default.employees...'.



The screenshot shows the Hive web interface with a query entered in the editor: 

```
1 SELECT *
2 FROM default.employees
3 LIMIT 100
4 ;
```

 Below the editor is an 'Execute' button and a text input for the number of rows to return, set to '5000'. Below the editor, there's a tabbed interface with 'Query History', 'Saved Queries', 'Results', 'Chart', and 'Execution Analysis'. The 'Results' tab is active, showing a table with 6 rows and 6 columns: employees.id, employees.name, employees.country, employees.department, employees.position, and employees.salary. The first 5 rows are NULL, and the 6th row contains data for an employee named Alice.

	employees.id	employees.name	employees.country	employees.department	employees.position	employees.salary
1	NULL	NULL	NULL	NULL	NULL	NULL
2	NULL	NULL	NULL	NULL	NULL	NULL
3	NULL	NULL	NULL	NULL	NULL	NULL
4	NULL	NULL	NULL	NULL	NULL	NULL
5	NULL	NULL	NULL	NULL	NULL	NULL
6	101	Alice	New York	IT	Soft Engg	4000

## 2. PASEO POR GETHUE.COM

---

- ❖ Es una herramienta que nos permite editar distintos componentes de Hadoop.
- ❖ Hue levanta un pequeño servidor web a partir del cual con un navegador puede acceder.
- ❖ En Query History están los comandos que se han ido lanzando. Cuando lo instalemos lo veremos con mas detalle
- ❖ Recordar que es un entorno de pruebas con lo cual poco se puede hacer. Solo es para mostrar porque es interesante instalar HUE
- ❖ Nos permite trabajar con un entorno Hadoop de una manera mas fácil, grafica y rápida
- ❖ Ambari es una herramienta todo terreno que también tiene este tipo de componentes
- ❖ HUE es una buena alternativa para los que no vayan a utilizar Ambari o Cloudera, puesto que permite trabajar en entornos Hadoop de una manera mucho más sencilla que la que podemos hacer con líneas de comandos



## 3. INSTALACION HUE

**Paso 1.** Descargaremos la versión 4.11 de Hue. Vamos al menú Blog y en AllCategories seleccionamos versión 4.11. Hacemos click en el boton Read Article. Al final de la pagina tenemos las diferentes formas de instalación de Hue. Descargamos la versión tarball

Category: [Version 4.11](#)


---

23 January 2023

Hue 4.11 and its new dialects and features are out!

Hi Data Explorers, The Hue team is glad to release Hue 4.11. Thanks to all the contributors! Note: Hue 4.11 is the last release to support Python 2.7. New releases will be based on Python 3.8 and higher. Here's a summary of what's new in Hue 4.11: Ability to import and query Iceberg tables from Hue Apache Iceberg is a high-performance table format and extends multifunction analytics to a petabyte scale for multi-cloud and hybrid use cases....

3 minutes read - [Version 4.11](#) / [Release](#)

 **hue-4.11.0.tgz**  
Completada — 99,5 MB

---

[Mostrar todas las descargas](#)

Go grab it and give it a spin!

- Docker

```
docker run -it -p 8888:8888 gethue/4.11.0
```

- Kubernetes :

```
helm repo add gethue https://helm.gethue.com  
helm repo update  
helm install hue gethue/hue
```

- [demo.gethue.com](#)
- [Tarball](#) or [source](#)

## 3. INSTALACION HUE

Paso 2. En Downloads descomprimos el fichero tgz

**tar xvf hue-4.11.0.tgz**

```
hadoop@nodol:~/Downloads$ ls
access_log          deslizamientos.csv  motivo_pais.csv    MyJob.java
apache-hive-3.1.3-bin.tar.gz  empleados.txt      'MyJob$MapClass.class'  tabla_deslizamientos.txt
cite75_99.txt       hue-4.11.0.tgz      'MyJob$Reduce.class'
ContarPalabras.java hadoop-3.2.4.tar.gz MyJob.class
countries.csv       hadoop-3.3.6.tar.gz
hadoop@nodol:~/Downloads$ tar xvf hue-4.11.0.tgz
```

**NOTA:** En el directorio de descarga hemos descomprimido el fichero pero aquí no hay binarios, sólo el código fuente. Se debe de compilar para generar los binarios y los componentes necesarios para ejecutarlo en nuestro entorno. No hay unos RPM o instaladores directos.

Antes de realizar la instalación, tenemos que preparar nuestro sistema con una serie de librerías necesarias

### 3. INSTALACION HUE

---

Paso 3. Instalamos una serie de paquete y librerías necesarios

<https://tutorialforlinux.com/2022/08/08/step-by-step-hue-ubuntu-22-04-installation-guide/>

```
sudo apt update -y
```

```
sudo apt install ant gcc g++
```

```
sudo apt install libkrb5-dev *libmysqlclient-dev
```

```
sudo apt install libssl-dev libsasl2-dev
```

```
sudo apt install libsasl2-modules-gssapi-mit
```

```
sudo apt install libsqlite3-dev libtidy*dev libxml2-dev libxslt-dev
```

```
sudo apt install maven libldap2-dev python2-dev
```

```
sudo apt install python-setuptools git
```

```
pip install --upgrade pip
```

### 3. INSTALACION HUE

**Paso 4.** Compilaremos e instalaremos hue. Dentro del directorio hue pondremos el comando PREFIX= y el sitio donde lo queremos instalar. Es un comando típico de Linux para compilar programas Open Source

**sudo PREFIX=/opt/hadoop make install**

Le indicamos que construya todos los componentes. Dejará el resultado en /opt/hadoop, donde generará un directorio llamado hue, con los binarios y todo lo necesario para empezar a trabajar.

```
Post-processed 'dashboard/css/search_mobile.css' as 'dashboard/css/search_mobile.d3ccel5268e8.css'
Post-processed 'oozie/css/workflow.css' as 'oozie/css/workflow.5bf771c653dc.css'
Post-processed 'debug_toolbar/css/toolbar.css' as 'debug_toolbar/css/toolbar.b35523ea26ab.css'
Post-processed 'rest_framework/css/default.css' as 'rest_framework/css/default.8d5591a6aabc.css'
Post-processed 'oozie/css/coordinator-editor.css' as 'oozie/css/coordinator-editor.a3cd65079101.css'
Post-processed 'desktop/css/bootstrap-daterangepicker.css' as 'desktop/css/bootstrap-daterangepicker.3b118749763b.css'
Post-processed 'desktop/css/jqCron.css' as 'desktop/css/jqCron.ad6fb9aledc5.css'
Post-processed 'pig/css/pig.css' as 'pig/css/pig.971ef685eb43.css'
Post-processed 'oozie/css/workflow-editor.css' as 'oozie/css/workflow-editor.8a0492fbbc38.css'
Post-processed 'desktop/css/bootstrap-spinedit.css' as 'desktop/css/bootstrap-spinedit.983726642cbb.css'
Post-processed 'rest_framework/css/font-awesome-4.0.3.css' as 'rest_framework/css/font-awesome-4.0.3.c1e1ea213abf.css'

133 static files copied to '/usr/local/hue/build/static', 1145 unmodified, 168 post-processed.
make[1]: Leaving directory '/usr/local/hue'
hadoop@nodol1:~/Downloads/hue-4.10.0$ sudo EXPORT=/opt/hadoop make install
```

### 3. INSTALACION HUE

Paso 5. Como hemos ejecutado PREFIX con sudo, la carpeta /opt/hadoop/hue pertenece a root. Debemos cambiar el usuario y el grupo a hadoop → **sudo chown -R hadoop:hadoop /opt/hadoop/hue**

```
hadoop@nodol1:/opt/hadoop$ ls -l
total 216
drwxr-xr-x  2 hadoop hadoop  4096 jul 12  2022 bin
drwxr-xr-x  3 hadoop hadoop  4096 jul 12  2022 etc
drwxrwxr-x 11 hadoop hadoop  4096 mar 18 16:44 hive
drwxr-xr-x  9 root   root    4096 mar 26 03:47 hue
drwxr-xr-x  2 hadoop hadoop  4096 jul 12  2022 include
drwxr-xr-x  3 hadoop hadoop  4096 jul 12  2022 lib
drwxr-xr-x  4 hadoop hadoop  4096 jul 12  2022 libexec
-rw-rw-r--  1 hadoop hadoop 150571 jul 12  2022 LICENSE.txt
drwxrwxr-x  3 hadoop hadoop  4096 mar 22 21:02 logs
-rw-rw-r--  1 hadoop hadoop  21932 jul 12  2022 NOTICE.txt
-rw-rw-r--  1 hadoop hadoop   1397 feb 23 19:21 README.txt
drwxr-xr-x  3 hadoop hadoop  4096 jul 12  2022 sbin
drwxr-xr-x  4 hadoop hadoop  4096 jul 12  2022 share
hadoop@nodol1:/opt/hadoop$ sudo chown -r hadoop:hadoop hue
[sudo] password for hadoop:
chown: invalid option -- 'r'
Try 'chown --help' for more information.
1 hadoop@nodol1:/opt/hadoop$ sudo chown -R hadoop:hadoop hue
hadoop@nodol1:/opt/hadoop$ ls -l
total 216
drwxr-xr-x  2 hadoop hadoop  4096 jul 12  2022 bin
drwxr-xr-x  3 hadoop hadoop  4096 jul 12  2022 etc
drwxrwxr-x 11 hadoop hadoop  4096 mar 18 16:44 hive
drwxr-xr-x  9 hadoop hadoop  4096 mar 26 03:47 hue
drwxr-xr-x  2 hadoop hadoop  4096 jul 12  2022 include
drwxr-xr-x  3 hadoop hadoop  4096 jul 12  2022 lib
drwxr-xr-x  4 hadoop hadoop  4096 jul 12  2022 libexec
-rw-rw-r--  1 hadoop hadoop 150571 jul 12  2022 LICENSE.txt
drwxrwxr-x  3 hadoop hadoop  4096 mar 22 21:02 logs
-rw-rw-r--  1 hadoop hadoop  21932 jul 12  2022 NOTICE.txt
-rw-rw-r--  1 hadoop hadoop   1397 feb 23 19:21 README.txt
drwxr-xr-x  3 hadoop hadoop  4096 jul 12  2022 sbin
drwxr-xr-x  4 hadoop hadoop  4096 jul 12  2022 share
hadoop@nodol1:/opt/hadoop$
```



## 4. CONFIGURAR Y ARRANCAR HUE

**Paso 1.** Si todo ha sido correcto, en el directorio `/opt/hadoop/hue` tenemos una serie de ficheros y directorios. Para que funcione correctamente debemos de configurar el fichero de configuración **`/opt/hadoop/hue/desktop/conf/hue.ini`**

```
hadoop@nodo1:/opt/hadoop/hue$ ls -l
total 68
drwxrwxr-x 3 hadoop hadoop 4096 mar 25 20:00 desktop
drwxr-xr-x 3 hadoop hadoop 4096 mar 24 09:47 ext
-rw-r--r-- 1 hadoop hadoop 11349 mar 22 2021 LICENSE.txt
-rw-r--r-- 1 hadoop hadoop 7043 ene 23 08:57 Makefile
-rw-rw-r-- 1 hadoop hadoop 44 mar 25 20:00 Makefile.buildvars
-rw-r--r-- 1 hadoop hadoop 9062 ene 10 12:30 Makefile.sdk
-rw-r--r-- 1 hadoop hadoop 6054 ene 10 12:30 Makefile.vars
-rw-r--r-- 1 hadoop hadoop 2192 ene 23 08:57 Makefile.vars.priv
-rw-r--r-- 1 hadoop hadoop 1445 ene 23 08:57 README
drwxrwxr-x 4 hadoop hadoop 4096 mar 25 20:00 tools
-rw-r--r-- 1 hadoop hadoop 933 ene 19 07:35 VERSION
hadoop@nodo1:/opt/hadoop/hue$
```

```
hadoop@nodo1:/opt/hadoop/hue/desktop/conf$ ls -l
total 88
-rw-r--r-- 1 hadoop hadoop 79806 ene 10 12:30 hue.ini
-rw-r--r-- 1 hadoop hadoop 1843 ago 10 2020 log4j.properties
-rw-r--r-- 1 hadoop hadoop 1873 mar 22 2021 log.conf
hadoop@nodo1:/opt/hadoop/hue/desktop/conf$
```

## 4. CONFIGURAR Y ARRANCAR HUE

---

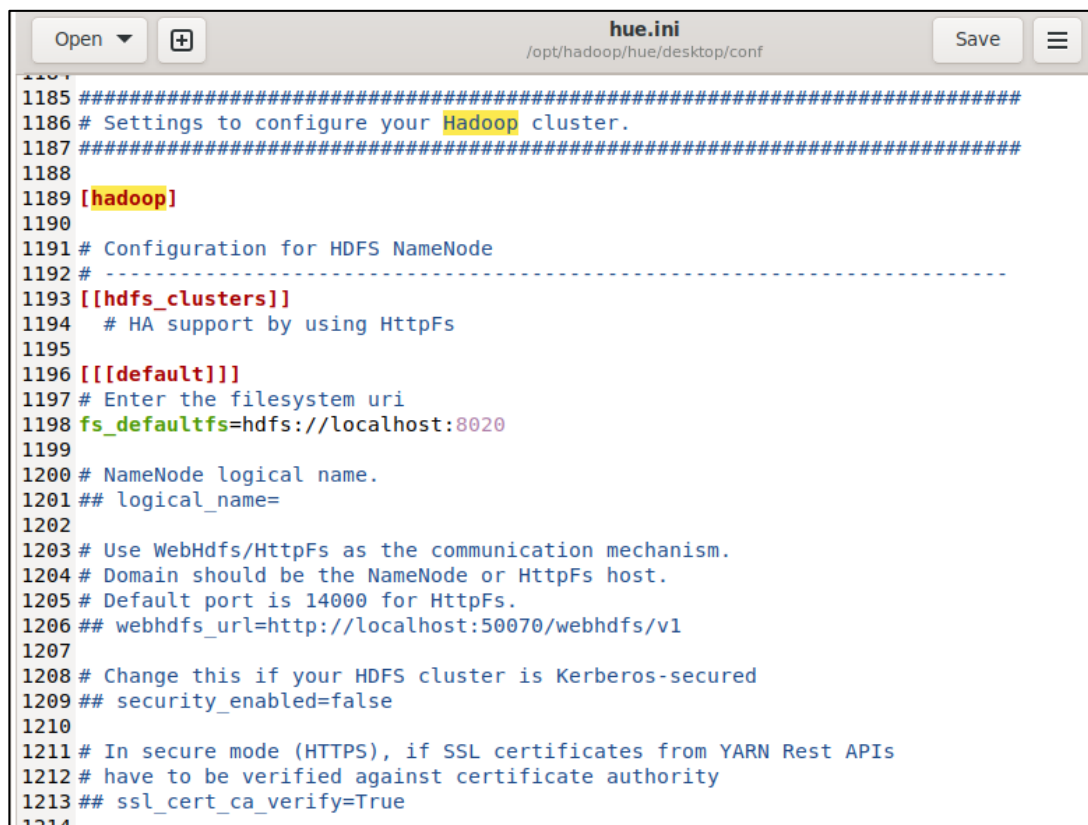
**Paso 2.** Editamos el fichero hue.ini con gedit al tener un tamaño bastante considerable. Hue es un producto que nos va a permitir funcionar contra distintos productos y hay que realizar la configuración para cada uno de ellos. Tenemos que configurarlo para 3 productos ya instalados:

- HDFS
- Yarn
- Hive

Según vayamos viendo otros productos (Pig, Hbase, o Impala) iremos ampliando las propiedades de Hue para trabajar con ellos

## 4. CONFIGURAR Y ARRANCAR HUE

**Paso 3.** Primero configuramos la parte de Hadoop. Buscamos la sección Hadoop donde se configuran sus componentes



```
1185 #####
1186 # Settings to configure your Hadoop cluster.
1187 #####
1188
1189 [hadoop]
1190
1191 # Configuration for HDFS NameNode
1192 # -----
1193 [[hdfs_clusters]]
1194 # HA support by using HttpFs
1195
1196 [[[default]]]
1197 # Enter the filesystem uri
1198 fs_defaultfs=hdfs://localhost:8020
1199
1200 # NameNode logical name.
1201 ## logical_name=
1202
1203 # Use WebHdfs/HttpFs as the communication mechanism.
1204 # Domain should be the NameNode or HttpFs host.
1205 # Default port is 14000 for HttpFs.
1206 ## webhdfs_url=http://localhost:50070/webhdfs/v1
1207
1208 # Change this if your HDFS cluster is Kerberos-secured
1209 ## security_enabled=false
1210
1211 # In secure mode (HTTPS), if SSL certificates from YARN Rest APIs
1212 # have to be verified against certificate authority
1213 ## ssl_cert_ca_verify=True
1214
```



## 4. CONFIGURAR Y ARRANCAR HUE

---

Paso 4. En la uri de HDFS pondremos nodo1:9000

```
[[[default]]]  
# Enter the filesystem uri  
fs_defaultfs=hdfs://localhost:8020
```

```
[[[default]]]  
# Enter the filesystem uri  
fs_defaultfs=hdfs://nodo1:9000
```

Estos son los valores que se indican en el fichero  
/opt/hadoop/etc/hadoop/core-site.xml

```
<!-- Put site-specific property overrides in this file. -->  
  
<configuration>  
  <property>  
    <name>fs.defaultFS</name>  
    <value>hdfs://nodo1:9000</value>  
  </property>  
</configuration>
```

## 4. CONFIGURAR Y ARRANCAR HUE

**Paso 5.** Una vez hemos tocado la parte HDFS, vamos a buscar la parte del Yarn que se llama `yarn_clusters`. Aquí dentro, debemos de cambiar la variable `resourcemanager_host` de `localhost` a `nodo1`. Vemos que el valor de `resorcemanager_port` es correcto (8032), y por lo tanto no lo tocamos.

```
# Configuration for YARN (MR2)
# -----
[[yarn_clusters]]

[[[default]]]
# Enter the host on which you are running the ResourceManager
## resourcemanager_host=localhost

# The port where the ResourceManager IPC listens on
## resourcemanager_port=8032

# Whether to submit jobs to this cluster
submit_to=True

# Resource Manager logical name (required for HA)
## logical_name=

# Change this if your YARN cluster is Kerberos-secured
## security_enabled=false

# URL of the ResourceManager API
## resourcemanager_api_url=http://localhost:8088
```

```
# Configuration for YARN (MR2)
# -----
[[yarn_clusters]]

[[[default]]]
# Enter the host on which you are running the ResourceManager
resourcemanager_host=nodo1

# The port where the ResourceManager IPC listens on
## resourcemanager_port=8032

# Whether to submit jobs to this cluster
submit_to=True

# Resource Manager logical name (required for HA)
## logical_name=

# Change this if your YARN cluster is Kerberos-secured
## security_enabled=false
```

## 4. CONFIGURAR Y ARRANCAR HUE

**Paso 6.** Vamos a la sección beeswax, que es la parte de Hive. Cambiamos la variable `hive_server_host` de `localhost` a `nodo1`. El puerto no hace falta cambiarlo porque es el 10000.

```
#####  
# Settings to configure Beeswax with Hive  
#####  
  
[beeswax]  
  
# Host where HiveServer2 is running.  
# If Kerberos security is enabled, use fully-qualified domain name (FQDN).  
## hive_server_host=localhost  
  
# Binary thrift port for HiveServer2.  
## hive_server_port=10000  
  
# Http thrift port for HiveServer2.  
## hive_server_http_port=10001  
  
# Host where LLAP is running  
## llap_server_host = localhost  
  
# LLAP binary thrift port  
## llap_server_port = 10500  
  
# LLAP HTTP Thrift port  
## llap_server_thrift_port = 10501
```

```
#####  
# Settings to configure Beeswax with Hive  
#####  
  
[beeswax]  
  
# Host where HiveServer2 is running.  
# If Kerberos security is enabled, use fully-  
hive_server_host=nodo1  
  
# Binary thrift port for HiveServer2.  
## hive_server_port=10000  
  
# Http thrift port for HiveServer2.  
## hive_server_http_port=10001  
  
# Host where LLAP is running  
## llap_server_host = localhost  
  
# LLAP binary thrift port  
## llap_server_port = 10500
```

## 4. CONFIGURAR Y ARRANCAR HUE

---

**Paso 7.** Más abajo está la parte del metastore, Impala. Por cada producto tendríamos que entrar aquí y configurarlo. Si se instala impala, se debe de configurar la parte de Hue para que se pueda conectar al Impala. Guardamos el fichero y salimos

```
#####  
# Settings to configure Metastore  
#####  
  
[metastore]  
# Flag to turn on the new version of the create table wizard.  
## enable_new_create_table=true  
  
# Flag to force all metadata calls (e.g. list tables, table or column detail  
via HiveServer2 if available instead of Impala.  
## force_hs2_metadata=false  
  
# Choose whether to show the table ERD component. Default false  
## show_table_erd=false  
  
#####  
# Settings to configure Impala  
#####  
  
[impala]  
# Host of the Impala Server (one of the Impalad)  
## server_host=localhost  
  
# Port of the Impala Server  
## server_port=21050  
  
# Endpoint of the Impala Proxy Server, for example: '/endpoint'.  
# Note that SERVER_PORT will be used when set.  
## proxy_endpoint=
```

## 4. CONFIGURAR Y ARRANCAR HUE

**Paso 8.** Ahora falta añadir una propiedad importante que tiene HDFS en el fichero `hdfs-site.xml` llamada **`dfs.webhdfs.enabled`**.

Esta propiedad permite acceder a HDFS a través de web. Es un componente de HDFS que permite hacer llamadas a través de http al entorno Web. Puesta a true, habilitamos la parte de web HDFS que es lo que utiliza HUE para conectarse vía web

```
GNU nano 6.2 /opt/hadoop/etc/hadoop/hdfs-site.xml

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>dfs.replication</name>
    <value>2</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>/datos/namenode</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>/datos/datanode</value>
  </property>
  <property>
    <name>dfs.webhdfs.enabled</name>
    <value>true</value>
  </property>
</configuration>
```

## 4. CONFIGURAR Y ARRANCAR HUE

**Paso 9.** Por ultimo modificaremos el fichero core-site.xml. Activaremos las propiedades `hadoop.proxyuser.hue.hosts` y `hadoop.proxyuser.hue.groups` poniéndole asterisco, es decir indicando que cualquier host de cualquier grupo se pueda conectar. En la vida real seguramente tendremos que ser un poco más finos, pero por ahora necesario poner asterisco

```
GNU nano 6.2 /opt/hadoop/etc/hadoop/core-site.xml

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://nodol:9000</value>
  </property>
  <property>
    <name>hadoop.proxyuser.hue.hosts</name>
    <value>*</value>
  </property>
  <property>
    <name>hadoop.proxyuser.hue.groups</name>
    <value>*</value>
  </property>
</configuration>
```

## 4. CONFIGURAR Y ARRANCAR HUE

**Paso 10.** Arrancaremos Hue para ver que funciona. Pero primero por seguridad debemos activar la interficie que hemos desactivado durante la instalación y reiniciar los servicios de yarn y hdfs:

**sudo ifconfig enp0s8 down**

**start-dfs.sh**

**start-yarn.sh**

**Paso 11.** En la carpeta **/opt/hadoop/hue/build/env/bin**, hay una serie de comandos que Hue utiliza de manera interna. El comando que nos interesa es el supervisor

```
hadoop@nodol: /opt/hadoop/hue/build/env/bin$ ls
activate          hue               odfuserfield     python2.7
activate.csh      instance_events  parquet          python-config
activate.fish     jws              parse_xsd2.py    route53
activate_this.py  kill_instance    pasteurize       rst2html4.py
asadmin           launch_instance  pip              rst2html5.py
avro              list_instances  pip2             rst2html.py
bundle_image      lss3            pip2.7           rst2latex.py
celery            mailodf         pyami_sendmail   rst2man.py
cfadmin           make_metadata.py pybabel          rst2odt_prepstyles.py
cq                mako-render     pydruid          rst2odt.py
csv2ods           markdown        pygmentize       rst2pseudoxml.py
cutil            mdexport.py     pyjwt            rst2s5.py
django-admin      merge_metadata.py pyrsa-decrypt    rst2xetex.py
django-admin.py  mturk           pyrsa-decrypt-bigfile rst2xml.py
dynamodb_dump    ndg_httpclient  pyrsa-encrypt    rstpep2html.py
dynamodb_load    nosetests       pyrsa-encrypt-bigfile s3put
easy_install      nosetests-2.7   pyrsa-keygen     sdbadmin
easy_install-2.7 odf2mht         pyrsa-priv2pub   sqlformat
elbadmin         odf2html        pyrsa-sign       supervisor
fetch_file       odf2xml         pyrsa-verify     tabulate
futarize         odfigimport     py.test          taskadmin
glacier          odflint         pytest           wheel
gunicorn         odffmeta        python           xml2odf
gunicorn_paster  odffoutline     python2
```

## 4. CONFIGURAR Y ARRANCAR HUE

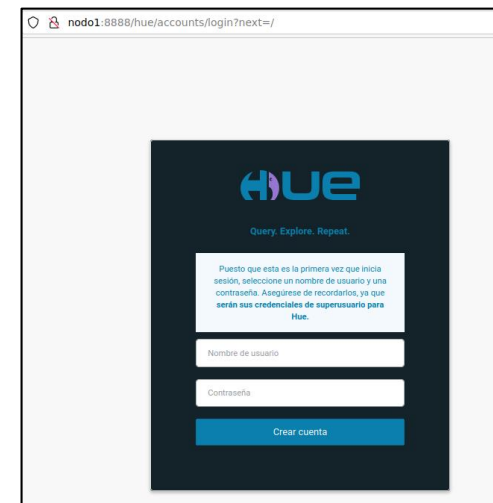
Paso 12. El comando para arrancar Hue es supervisor:  
**supervisor -d** → en modo daemon, se ejecuta en 2º plano

```
hadoop@nodo1:/opt/hadoop/hue/build/env/bin$ ./supervisor -d
hadoop@nodo1:/opt/hadoop/hue/build/env/bin$
```

Paso 13. Hacemos `ps -ef | grep supervisor`, si sale el proceso en principio es que está funcionando:

```
hadoop@nodo1:/opt/hadoop/hue/build/env/bin$ ps -ef | grep supervisor
hadoop 1138264 1346 1 06:22 ? 00:00:00 /opt/hadoop/hue/build/env/bin/python2.7 ./supervisor -d
hadoop 1140044 1850 0 06:23 pts/1 00:00:00 grep --color=auto supervisor
hadoop@nodo1:/opt/hadoop/hue/build/env/bin$
```

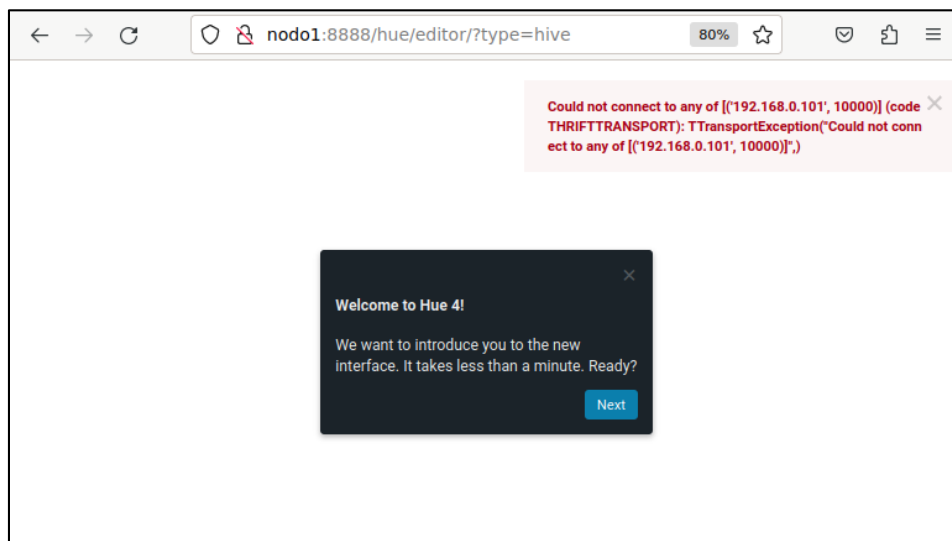
Paso 14. En un navegador ponemos **nodo1:8888** (es el valor por defecto de hue). Si todo es correcto deberíamos ver la ventana de conexión





## 5. PASEO POR HUE

**Paso 1.** En la ventana de conexión de hue ponemos admin admin y observamos un error



This message indicates that Hive Server is down or HUE Can't reach the server.

## 5. PASEO POR HUE

---