

Hadoop HBase y Hive

Elana Sanz Espada
IABD - BigData

Índice

Cluster	4
Creación	4
Visualizar sistema de archivos HDFS	9
Dataset de Kaggle	12
Subir archivos a Hadoop	13
Comandos CRUD HBase y Hive	14
HBase	14
Create	14
Reed	14
Update	15
Delete	15
Hive	16
Create	16
Reed	17
Update	17
Delete	17
Consultas	18
Estudiar los datos	18
Crear las tablas	19
Cargar los datos de los ficheros en las tablas	22
Crear diferentes consultas sobre las tablas	27
Bibliografía	29

Cluster

Creación

Primero seleccionamos las aplicaciones Hue, Hadoop, HBase y Hive.

Versión de Amazon EMR | Información

Una versión contiene un conjunto de aplicaciones que se puede instalar en el clúster.

emr-6.14.0

Paquete de aplicaciones

Spark
Interactive

Core
Hadoop

Flink

HBase

Presto

Trino

Custom

☐ Flink 1.17.1
 ☐ HCatalog 3.1.3
 ☒ Hue 4.11.0
 ☐ Livy 0.7.1
 ☐ Phoenix 5.1.3
 ☐ Spark 3.4.1
 ☐ Tez 0.10.2
 ☐ ZooKeeper 3.5.10

☐ Ganglia 3.7.2
 ☒ Hadoop 3.3.3
 ☐ JupyterEnterpriseGateway 2.6.0
 ☐ MXNet 1.9.1
 ☐ Pig 0.17.0
 ☐ Sqoop 1.4.7
 ☐ Trino 422

☒ HBase 2.4.17
 ☒ Hive 3.1.3
 ☐ JupyterHub 1.5.0
 ☐ Oozie 5.2.1
 ☐ Presto 0.281
 ☐ TensorFlow 2.11.0
 ☐ Zeppelin 0.10.1

Configuración del Catálogo de datos de AWS Glue

Utilice el Catálogo de datos de AWS Glue para proporcionar un meta-almacén externo a la aplicación.

☒ Usar para metadatos de la tabla de Hive

Configuración de almacenamiento de HBase

Elija la capa de almacenamiento para los datos almacenados en HBase. La opción HDFS utiliza la ubicación predeterminada de HBase para el directorio raíz.

☒ Sistema de archivos distribuido de Hadoop (HDFS)

▼

Aprovisionamiento y escalado de clústeres - obligatorio

Información

Elija cómo Amazon EMR debe dimensionar su clúster.

Elija una opción

Establecer el tamaño del clúster manualmente

Utilice esta opción si conoce los patrones de la carga de trabajo de antemano.

Utilizar escalado administrado por EMR

Supervise las métricas clave de la carga de trabajo de modo que EMR pueda optimizar el tamaño del clúster y la utilización de los recursos.

Utilizar el escalamiento automático personalizado

Para escalar mediante programación los nodos principales y los nodos de tarea, cree políticas de escalamiento automático personalizadas.

Configuración de aprovisionamiento

Establezca el tamaño del principal y tarea grupos de instancias. Amazon EMR intenta aprovisionar esta capacidad al lanzar el clúster.

Nombre	Tipo de instancia	Tamaño de instancia(s)	Utilizar la opción de compra de spot
Central	m5.xlarge	<div>3</div>	<input type="checkbox"/>
Tarea - 1	m5.xlarge	<div>1</div>	<input type="checkbox"/>

Redes - obligatorio

Información

Elija la configuración de red que determina la forma en que usted y otras entidades se comunican con su clúster.

Virtual Private Cloud (VPC)

Información

vpc-0f4b1437f3dbdbbc8

Examinar

Crear VPC

Subred

Información

subnet-03bc80d12d9a609b0

Examinar

Crear subred

Grupos de seguridad de EC2 (firewall)

Aviso de cambio

Hemos actualizado los nombres de algunos grupos de seguridad para utilizar un lenguaje más inclusivo. Por ejemplo, los grupos que incluían términos como "maestro" y "esclavo" ahora utilizan en su lugar los términos "principal" y "central".

Nodo principal

Grupos de seguridad administrados de EMR

EMR actualizará automáticamente el grupo seleccionado.

ElasticMapReduce-Primary

sg-01f0a9c5cb94d1d75

Grupos de seguridad adicionales - opcional

Seleccione hasta 4 grupos de seguridad adicionales.

Elegir grupos de seguridad adicionales

Nodos principales y de tareas

Grupos de seguridad administrados de EMR

EMR actualizará automáticamente el grupo seleccionado.

ElasticMapReduce-Core

sg-002dba5c83c5645bd

Grupos de seguridad adicionales - opcional

Seleccione hasta 4 grupos de seguridad adicionales.

Elegir grupos de seguridad adicionales

▼ Terminación del clúster y reemplazo de nodos Información

Elija la configuración de terminación y proteja su clúster contra un apagado accidental.

Opción de terminación

- ☒ Terminar manualmente el clúster
- ☐ Terminar automáticamente el clúster después de que finalice el último paso
- ☐ Terminar el clúster después del tiempo de inactividad (recomendado)

☐ Use la protección contra la terminación

Protege al clúster para evitar una terminación accidental. Si está activada, deberá primero desactivar la protección para terminar el clúster. Recomendamos activar la protección frente a terminaciones para los clústeres de larga duración.

Reemplazo de nodos en mal estado - *novedad* | Información

☒ Activar


Amazon EMR detiene correctamente los procesos en los nodos en mal estado para minimizar la pérdida de datos y las interrupciones del trabajo. Reemplaza rápidamente los nodos en mal estado por nuevas instancias de EC2 para que sus trabajos funcionen sin problemas.

☐ Desactivar

Amazon EMR agrega los nodos en mal estado a una lista de denegación mientras los mantiene en el clúster, lo que le permite tener acceso continuo para solucionar problemas.

▼ Registros de clúster Información

Elija dónde y cómo almacenar los archivos de registro.

-  Archivamos automáticamente los archivos de registro en Amazon S3. Puede especificar una ubicación de S3 propia o utilizar la ubicación de S3 predeterminada para Amazon EMR. La ubicación de registro predeterminada se completa previamente en el campo **Ubicación de Amazon S3**.

- ☒ Publicar registros específicos del clúster en Amazon S3

Ubicación de Amazon S3



Ver 

Explorar S3

Formato: utilizar s3://bucket/prefix

- ☐ Cifrar los registros específicos del clúster


▼ Configuración de seguridad y par de claves de EC2 Información


Elija una configuración de seguridad o cree una nueva que pueda reutilizar con otros clústeres.

Configuración de seguridad

Seleccione la configuración del servicio de cifrado, autenticación, autorización y metadatos de instancia del clúster.





Examinar 

Crear configuración de seguridad 

Par de claves de Amazon EC2 para el protocolo SSH al clúster | Información

Examinar

Crear par de claves 

-  No ha especificado una clave EC2. Si está fuera de una VPN y desea habilitar SSH o utilizar el asistente Hue SQL con este clúster, debe escribir una clave EC2.

Crear par de claves Información

Par de claves
Un par de claves, compuesto por una clave privada y una clave pública, es un conjunto de credenciales de seguridad que se utilizan para demostrar su identidad cuando se conecta a una instancia.

Nombre

El nombre puede incluir hasta 255 caracteres ASCII. No puede incluir espacios al principio ni al final.

Tipo de par de claves Información
☒ RSA ☐ ED25519

Formato de archivo de clave privada
☒ .pem
Para usar con OpenSSH
☐ .ppk
Para usar con PuTTY

Etiquetas: *opcional*
No hay etiquetas asociadas a este recurso.
[Agregar nueva etiqueta](#)
Puede agregar hasta 50 etiquetas más.

[Cancelar](#) [Crear par de claves](#)

▼ Configuración de seguridad y par de claves de EC2 Información
Elija una configuración de seguridad o cree una nueva que pueda reutilizar con otros clústeres.

Configuración de seguridad
Seleccione la configuración del servicio de cifrado, autenticación, autorización y metadatos de instancia del clúster.
 [Examinar](#) [Crear configuración de seguridad](#)

Par de claves de Amazon EC2 para el protocolo SSH al clúster Información
 [Examinar](#) [Crear par de claves](#)

8

Visualizar sistema de archivos HDFS

Para poder visualizar el sistema de archivos de HDFS, hay que cambiar en `hdfs-site.xml` y para esto necesitamos conectarnos por ssh

2_6_proyecto_Elena Se ha actualizado hace 1 minuto [Terminar](#) [Clonar en AWS CLI](#) [Clonar](#)

▼ Resumen	Aplicaciones	Administración de clústeres	Estado y hora
Información del clúster ID del clúster j-38WLVFRSEHSWH Configuración del clúster Grupos de instancias Capacidad 1 Primary (Principal) 3 Principat 1 Tarea	Versión de Amazon EMR emr-6.14.0 Aplicaciones instaladas HBase 2.4.17, Hadoop 3.3.3, Hive 3.1.3, Hue 4.11.0	Destino del registro en Amazon S3 aws-logs-476116739942-us-east-1/elasticmapreduce IU de aplicación persistente Servidor de línea de tiempo de YARN UI de Tez DNS público del nodo principal ec2-44-202-28-222.compute-1.amazonaws.com Conectarse al nodo principal mediante SSH Conectarse al nodo principal mediante SSM	Estado ⌚ Esperando Hora de creación 25 de noviembre de 2024 16:13 (UTC+01:00) Tiempo transcurrido 39 minutos, 2 segundos

Conectarse al nodo principal mediante SSH ✕

Puede conectarse al nodo principal de Amazon EMR mediante SSH para realizar acciones como ejecutar consultas interactivas, examinar archivos de registro, enviar comandos de Linux y ver interfaces Web alojadas en clústeres de Amazon EMR. [Más información](#)

Windows | **Mac/Linux**

1. Abra una ventana de terminal. En Mac OS X, elija Applications (Aplicaciones) > Utilities (Utilidades) > Terminal. En otras distribuciones de Linux, el terminal suele encontrarse en Applications (Aplicaciones) > Accessories (Accesorios) > Terminal.
2. Para establecer una conexión con el nodo principal, escriba el siguiente comando. Sustituya `~/proyecto2_6.pem` por la ubicación y el nombre de archivo del archivo de clave privada (.pem) que utilizó para lanzar el clúster.

```
ssh -i ~/proyecto2_6.pem hadoop@ec2-44-202-28-222.compute-1.amazonaws.com
```
3. Escriba Yes (Sí) para descartar la advertencia de seguridad.

[Ver interfaces Web alojadas en clústeres de Amazon EMR](#)

Cerrar

Antes de conectar por ssh hay que cambiar los permisos del .pem para que únicamente el usuario tenga poder sobre el fichero con el siguiente comando

```
chmod 600 proyecto2_6.pem
```

Conectamos por ssh

```
ssh -i ~/proyecto2_6.pem hadoop@ec2-44-202-28-222.compute-1.amazonaws.com
```



```
(base) iabd@dm2-14:~/Descargas$ chmod 600 proyecto2_6.pem
(base) iabd@dm2-14:~/Descargas$ ssh -i ~/Descargas/proyecto2_6.pem hadoop@ec2-44-202-28-222.compute-1.amazonaws.com
Last login: Mon Nov 25 16:08:56 2024

#
_#_      Amazon Linux 2
_#_      AL2 End of Life is 2025-06-30.
_#_      A newer version of Amazon Linux is available!
_#_      Amazon Linux 2023, GA and supported until 2028-03-15.
_#_      https://aws.amazon.com/linux/amazon-linux-2023/

4 package(s) needed for security, out of 7 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMM      MMMMMMM RRRRRRRRRRRRRR
E::::::::::::::::::::E M::::::::M      M::::::::M R::::::::::::R
EE::::::::EEEEEEEE::::E M::::::::M      M::::::::M R::::RRRRR::::R
E::::E      EEEEE M::::::::M      M::::::::M RR::::R      R::::R
E::::E      M::::::::M M::::M M::::M M::::M      R::::R      R::::R
E::::EEEEEEEE M::::M M::::M M::::M M::::M      R::RRRRR::::R
E::::::::::::E M::::M M::::M M::::M M::::M      R::::::::RR
E::::EEEEEEEE M::::M M::::M M::::M M::::M      R::RRRRR::::R
E::::E      M::::M M::::M M::::M M::::M      R::::R      R::::R
E::::E      EEEEE M::::M      MMM      M::::M      R::::R      R::::R
EE::::EEEEEEEE::::E M::::M      M::::M      R::::R      R::::R
E::::::::::::E M::::M      M::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM      MMMMMMM RRRRRRR      RRRRRR
```

Cambiamos la propiedad `dfs.webhdfs.enabled` a `true`:

```
sudo nano /etc/hadoop/conf/hdfs-site.xml
```

```
<property>
  <name>dfs.webhdfs.enabled</name>
  <value>true</value>
</property>
```

```
<!-- Configurations for large cluster -->
<property>
  <name>dfs.webhdfs.enabled</name>
  <value>true</value>
</property>
```

Reiniciar el servicio hdfs

```
sudo systemctl restart hadoop-hdfs-namenode
```

```
[hadoop@ip-172-31-88-209 ~]$ sudo systemctl restart hadoop-hdfs-namenode
[hadoop@ip-172-31-88-209 ~]$
```

Podemos acceder a las aplicaciones como Hive o HBase desde los enlaces en la pestaña Aplicaciones

< | Propiedades | Acciones de arranque | Instancias (hardware) | Pasos | **Aplicaciones** | Configuraciones | Monitorización | Eventos | >

Interfaces de usuario de aplicaciones Información
Las aplicaciones instaladas en el clúster de Amazon EMR publican interfaces de usuario (IU) como sitios web. Puede utilizarlas para supervisar la actividad del clúster.

☒ IU de la aplicación en el clúster

Las IU en el clúster solo están disponibles mientras se está ejecutando el clúster. Utilice los siguientes enlaces para comenzar. Para obtener acceso a todas las IU de la aplicación, configure el túnel de SSH.

☐ IU de aplicación persistente

Las IU persistentes no requieren el túnel de SSH, ya que se alojan fuera del clúster y están disponibles durante 30 días después de que finalice la aplicación.

IU de la aplicación activas
Estas IU de aplicaciones en clúster están disponibles sin el túnel de SSH.
IU de la aplicación [?](#)

No hay ninguna IU de la aplicación activa
No hay ninguna IU de la aplicación activa que mostrar

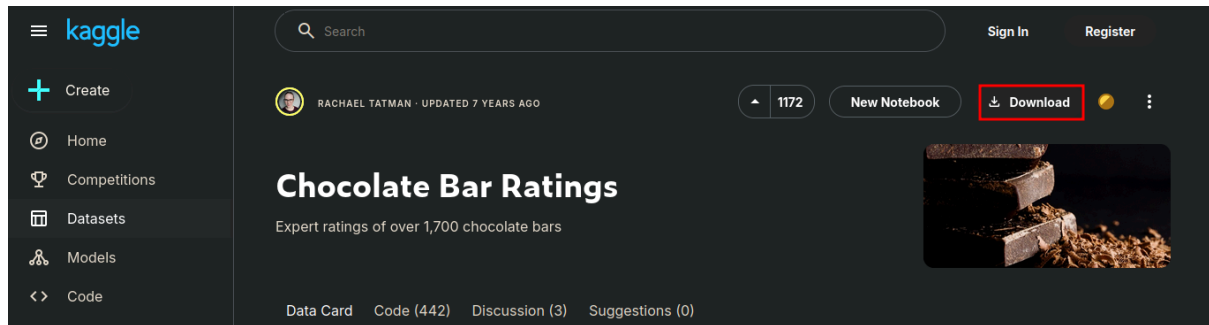
IU de la aplicación en el nodo principal
Estas requieren que el túnel de SSH esté habilitado.
[Habilitar una conexión SSH](#)

Aplicación	URL de la IU ?
Administrador de recursos	http://ec2-44-201-242-145.compute-1.amazonaws.com:8088/
HBase	http://ec2-44-201-242-145.compute-1.amazonaws.com:16010/
Nodo del nombre de HDFS	http://ec2-44-201-242-145.compute-1.amazonaws.com:9870/
Tonalidad	http://ec2-44-201-242-145.compute-1.amazonaws.com:8888/

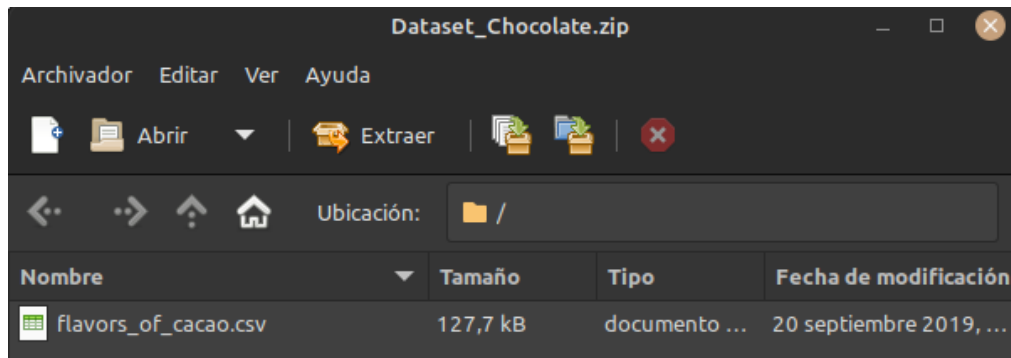
IU de aplicaciones en los nodos principales y de tareas

Dataset de Kaggle

En <https://www.kaggle.com/datasets> buscamos un dataset que nos guste.



Descargamos el .zip y lo descomprimos



Subir archivos a Hadoop

Para pasar el .csv de nuestra maquina al cluster usamos el comando scp

```
scp -i ~/Descargas/proyecto2_6.pem
/home/iabd/Escritorio/iabd/bd/proyecto/ignore/flavors_of_cacao.cshadoop@ec2-44-201-2
42-145.compute-1.amazonaws.com:/home/hadoop
```

```
^C(base) iabd@dm2-14:~$ scp -i ~/Descargas/proyecto2_6.pem /home/iabd/Escritorio/iabd/bd/proyecto/ignore/flavors_of_cacao.cs
hadoop@ec2-44-201-242-145.compute-1.amazonaws.com:/home/hadoopop
flavors_of_cacao.csv 100% 125KB 314.0KB/s 00:00
(base) iabd@dm2-14:~$
```

```
[hadoop@ip-172-31-88-209 ~]$ ls -l /home/hadoop
total 128
-rw-rw-r-- 1 hadoop hadoop 127723 nov 27 18:31 flavors_of_cacao.csv
```

Subimos el .csv a la carpeta proyecto en hadoop.

```
hdfs dfs -mkdir /proyecto
```

```
[hadoop@ip-172-31-88-209 ~]$ hdfs dfs -mkdir /proyecto
[hadoop@ip-172-31-88-209 ~]$ hdfs dfs -ls /
Found 5 items
drwxr-xr-x - hdfs hdfsadmingroup 0 2024-11-27 17:36 /apps
drwxr-xr-x - hadoop hdfsadmingroup 0 2024-11-27 17:45 /proyecto
drwxrwxrwt - hdfs hdfsadmingroup 0 2024-11-27 17:38 /tmp
drwxr-xr-x - hdfs hdfsadmingroup 0 2024-11-27 17:36 /user
drwxr-xr-x - hdfs hdfsadmingroup 0 2024-11-27 17:36 /var
```

```
hdfs dfs -put /home/hadoop/flavors_of_cacao.csv /proyecto
```

```
[hadoop@ip-172-31-88-209 ~]$ hdfs dfs -put /home/hadoop/flavors_of_cacao.csv /proyecto
[hadoop@ip-172-31-88-209 ~]$ hdfs dfs -ls /proyecto
Found 1 items
-rw-r--r-- 1 hadoop hdfsadmingroup 127723 2024-11-27 18:50 /proyecto/flavors_of_cacao.csv
```

Comandos CRUD HBase y Hive

HBase

HBase es una base de datos NoSQL de tipo clave-valor o columnar distribuida que ejecuta sobre HDFS (Hadoop Distributed File System). Proporciona una capa para acceder y actualizar los ficheros. Aporta lecturas de datos muy rápidas y está pensada para arquitecturas y sistemas de datos que escriben una vez y leen muchas veces. Eso es debido a que cuando se escriben ficheros en HDFS no se pueden modificar, pero sí acceder a ellos. En HBase los datos se dividen en tablas.

Create

Para crear tablas usamos **create** seguido del nombre de la tabla entre comillas, una coma y los nombres de las columnas entre comillas separados por comas.

```
create 'nombre_tabla', 'nombre_familia_columnas1', 'nombre_familia_columnas2'  
  
create 'alumnos', 'datos_personales', 'datos_escolares'
```

Para comprobar si se ha creado bien podemos usar **list** que nos listará las tablas creadas

Para insertar registros utilizamos **put** seguido del nombre de la tabla, el id de la fila, nombre familia de la columna : nombre de la columna y el valor. Esto se repetirá por cada columna de la fila.

```
put 'nombre_tabla', 'id_fila', 'familia_columna:nombre_columna', 'valor'  
  
put 'alumnos', '1', 'datos_personales:nombre', 'Elena'  
put 'alumnos', '1', 'datos_personales:apellido', 'Sanz'  
put 'alumnos', '1', 'datos_escolares:ciclo', 'IABD'
```

Read

Para obtener los datos empleamos **get** 'nombre_tabla', 'id_fila', {COLUMN='familia_colmna:nombre_columna'}. En caso de no añadir la sentencia entre corchetes devolverá toda la informacion de la fila

```
get 'nombre_tabla', 'id_fila', {COLUMN=> 'familia_columna:nombre_columna'}  
  
get 'alumnos', '1'  
get 'alumnos', '1', {COLUMN=> 'datos_personales:nombre'}
```

El comando **scan** 'nombre_tabla' sacará toda información de la tabla

```
scan 'nombre_tabla'
```

```
scan 'alumnos'
```

Update

Modificar datos ya existentes se hace igual que al insertar.

```
put 'nombre_tabla','id_fila', 'familia_columna', 'nombre_columna','valor'
```

```
put 'alumnos','1','datos_personales:apellido','Espada'
```

Delete

Para borrar tablas es necesario primero deshabilitarlas con **disable** 'nombre_tabla' y luego borrarlas mediante **drop** 'nombre_tabla'

```
disable 'clientes'
```

```
drop 'clientes'
```

Podemos usar **exist** para comprobar si se ha borrado correctamente

```
exist 'nombre_tabla'
```

Para borrar filas enteras usamos **deleteall** 'nombre_tabla' , 'id_fila'.

```
deleteall 'nombre_tabla','id_fila'
```

```
deleteall 'alumnos','1'
```

Si queremos borrar un registro específico de la fila usamos **delete** 'nombre_tabla', 'id_fila', 'familia_columna', 'nombre_columna','timestamp'

```
delete 'nombre_tabla','id_fila', 'familia_columna:nombre_columna',timestamp
```

```
delete 'alumnos','1','datos_personales:apellido',1417521848375
```

Hive

Hive es una plataforma que se utiliza para desarrollar tipo SQL scripts para hacer operaciones de MapReduce.

Create

Para crear tablas usamos **create table** el nombre de la tabla y entre paréntesis los nombres de las columnas seguidas del tipo de dato separadas por comas. Si añadimos **if not exists** entre create table y el nombre de la tabla, solo creará la tabla si no existe ninguna con el mismo nombre.

```
create table if not exists nombre_tabla (columna1 tipo dato, columna2 tipo dato);

create table if not exists alumnos(nombre String, edad int);
```

Con **show tables** podemos ver las tablas creadas para asegurarnos que se ha creado

```
SHOW TABLES;
```

Insertar una fila lo haremos mediante **insert into table** nombre_tabla y los valores de las columnas entre paréntesis. Puedes añadir tantas filas como paréntesis separados por comas haya

```
INSERT INTO TABLE tablename VALUES values_row;

INSERT INTO TABLE students VALUES ('Elena', 22),('Xabier', 23)
```

Para insertar varios registros a la vez crearemos un archivo .txt que contenga los datos que queremos insertar. Utilizaremos **load data** para introducir los datos del fichero en la base de datos. Para ello le especificaremos si se trata de un fichero local y la ruta del fichero

```
LOAD DATA LOCAL INPATH 'ruta_del fichero.txt';
```

```
fichero_con_datos.txt
```

```
1 Elena Sanz 22
2 Xabier Guerrero 23
3 Irune Guinea 23
```

Reed

Para obtener los datos construimos la sentencia ***select*** en la que iniciamos las columnas que buscamos y en qué tabla. También podemos filtrar al añadir ***where***.

```
SELECT nombre_columna FROM nombre_tabla WHERE condicion
```

```
# nombre de los alumnos con edad mayor a 10
```

```
select nombre from alumnos where edad>10;
```

Update

Modificar datos ya existentes se hace con ***update*** indicando la tabla, la columna y el valor nuevo seguido de una condición que identifique qué registros cambiar. Sin la condición ***where*** cambiará todos los valores de la columna por el nuevo.

```
UPDATE nombre_tabla SET nombre_columna= valor WHERE condicion;
```

```
update alumnos set edad=23 where id=1
```

Delete

Para borrar tablas emplearemos ***drop table*** nombre_tabla. *If exists* la borra si existe

```
DROP TABLE IF EXISTS table_name;
```

Para borrar filas usamos ***delete*** especificando el nombre de la tabla y la condición que identifique las filas a borrar. Importante definir bien la condición porque se borrarán todas las filas que la cumplan.

```
delete nombre_tabla where condicion
```

```
delete alumnos where id=1
```


Consultas

Estudiar los datos

	A	B	C	D	E	F	G	H	I
1	Company (Maker-if known)	Specific Bean Origin or Bar Name	REF	Review Date	Cocoa Percent	Company Location	Rating	Bean Type	Broad Bean Origin
2	A. Morin	Agua Grande		1876	2016 63%	France	3.75		Sao Tome
3	A. Morin	Kpime		1676	2015 70%	France	2.75		Togo
4	A. Morin	Atsane		1676	2015 70%	France		3	Togo
5	A. Morin	Akata		1680	2015 70%	France	3.5		Togo
6	A. Morin	Quilla		1704	2015 70%	France	3.5		Peru
7	A. Morin	Carenero		1315	2014 70%	France	2.75	Criollo	Venezuela
8	A. Morin	Cuba		1315	2014 70%	France	3.5		Cuba
9	A. Morin	Sur del Lago		1315	2014 70%	France	3.5	Criollo	Venezuela
10	A. Morin	Puerto Cabello		1319	2014 70%	France	3.75	Criollo	Venezuela
11	A. Morin	Pablino		1319	2014 70%	France		4	Peru
12	A. Morin	Panama		1011	2013 70%	France	2.75		Panama
13	A. Morin	Madagascar		1011	2013 70%	France		3 Criollo	Madagascar
14	A. Morin	Brazil		1011	2013 70%	France	3.25		Brazil
15	A. Morin	Equateur		1011	2013 70%	France	3.75		Ecuador
16	A. Morin	Colombie		1015	2013 70%	France	2.75		Colombia
17	A. Morin	Birmanie		1015	2013 70%	France		3	Burma
18	A. Morin	Papua New Guinea		1015	2013 70%	France	3.25		Papua New Guinea
19	A. Morin	Chuao		1015	2013 70%	France		4 Trinitario	Venezuela
20	A. Morin	Piura		1019	2013 70%	France	3.25		Peru
21	A. Morin	Chanchamayo Province		1019	2013 70%	France	3.5		Peru
22	A. Morin	Chanchamayo Province		1019	2013 63%	France		4	Peru

Vistos estos datos vamos a necesitar las siguientes columnas:

A todas las columnas hay que cambiarlas el nombre porque tienen saltos de línea, espacios o paréntesis

Columna	Tipo	Cambios
Company(Maker-if known)	String	company_maker
Specific Bean Origin or Bar Name	String	bean_origin
REF	int	ref
Review Date	Date/int	review_date
Cocoa Percent	int/float	cocoa_percent ; quitar % ; dividir entre 100
Company Location	String	company_location
Bean Type	String	bean_type ; sustituir nulos por ns
Broad Bean Origin	String	broad_bean_origin ; sustituir nulos por moda
Rating	float	rating

En HBase las columnas se agrupan por familias así que distribuiremos nuestras tal que así: datos_vaina

- Todos los datos que tengan que ver con las vainas de cacao
- bean_origin, bean_type, broad_bean_origin

datos_tableta

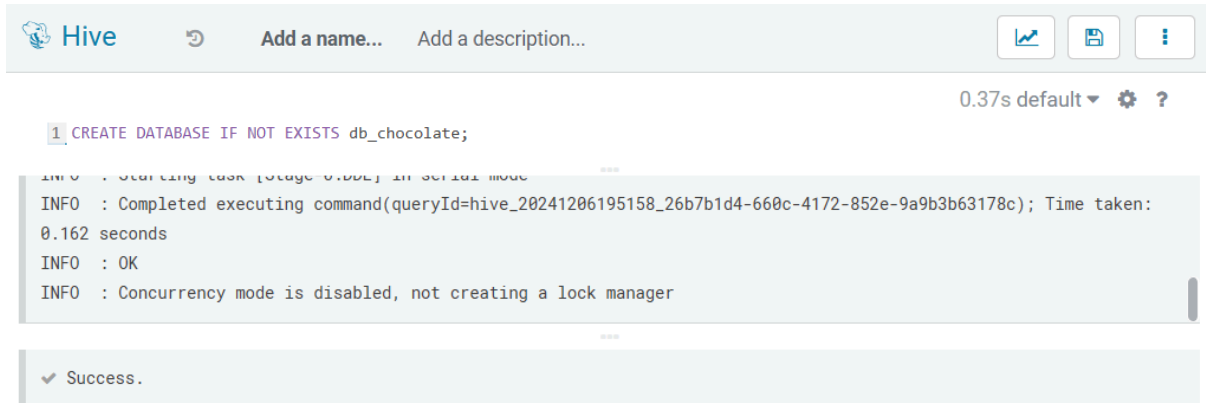
- Todos los datos relacionados con la tableta de chocolate
- company_maker, ref, review_date, cocoa_percent, company_location, rating

Crear las tablas

Hive

En caso querer crear una base de datos nueva, usaremos el siguiente comando

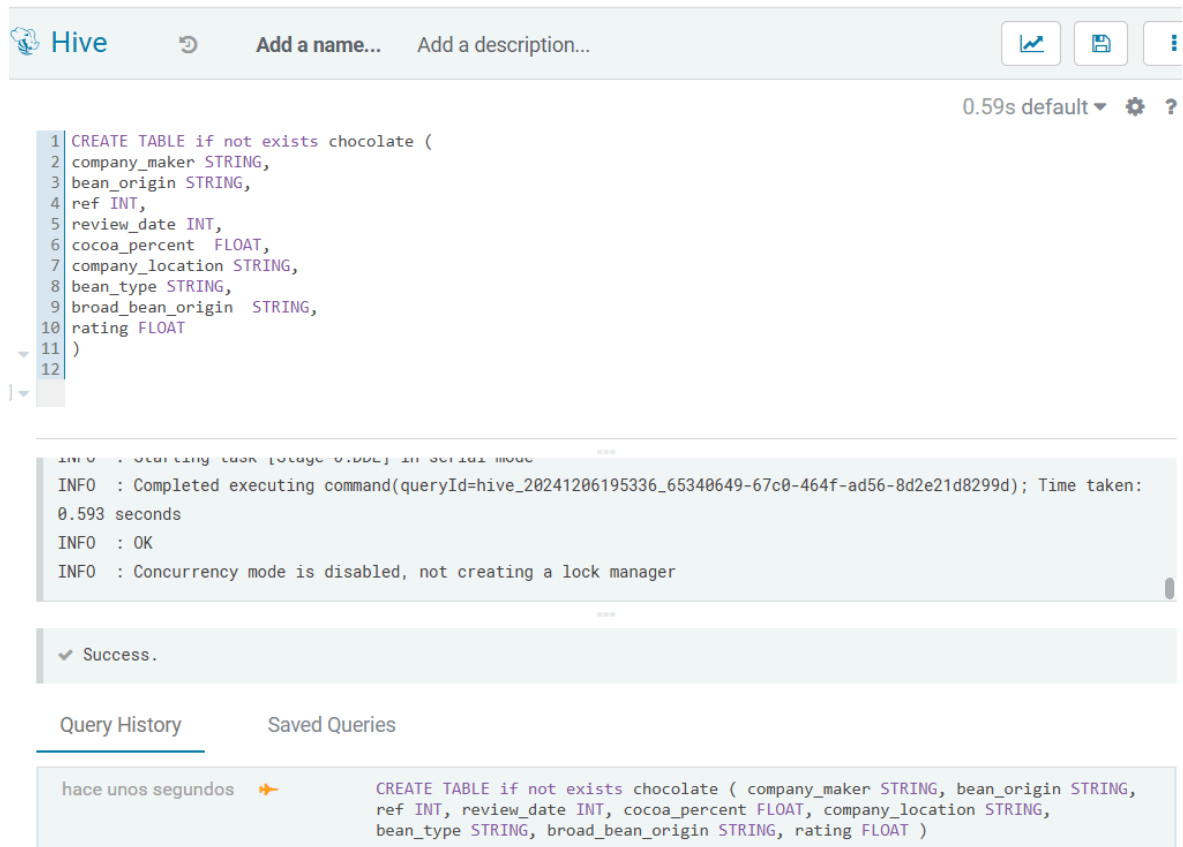
```
CREATE DATABASE IF NOT EXISTS db_chocolate;
```



En nuestro caso usaremos la base de datos default para nuestra tabla y consultas

Comando para crear la tabla chocolate

```
create table if not exists chocolate (  
  company_maker STRING,  
  bean_origin STRING,  
  ref INT,  
  review_date INT,  
  cocoa_percent FLOAT,  
  company_location STRING,  
  bean_type STRING,  
  broad_bean_origin STRING,  
  rating FLOAT  
)
```



The Hive interface shows a SQL command being executed in the editor. The command is a CREATE TABLE statement for a table named 'chocolate'. The command is as follows:

```
1 CREATE TABLE if not exists chocolate (  
2   company_maker STRING,  
3   bean_origin STRING,  
4   ref INT,  
5   review_date INT,  
6   cocoa_percent FLOAT,  
7   company_location STRING,  
8   bean_type STRING,  
9   broad_bean_origin STRING,  
10  rating FLOAT  
11 )  
12
```

The execution log shows the following information:

```
INFO : Starting task [stage-0.000] in serial mode  
INFO : Completed executing command(queryId=hive_20241206195336_65340649-67c0-464f-ad56-8d2e21d8299d); Time taken:  
0.593 seconds  
INFO : OK  
INFO : Concurrency mode is disabled, not creating a lock manager
```

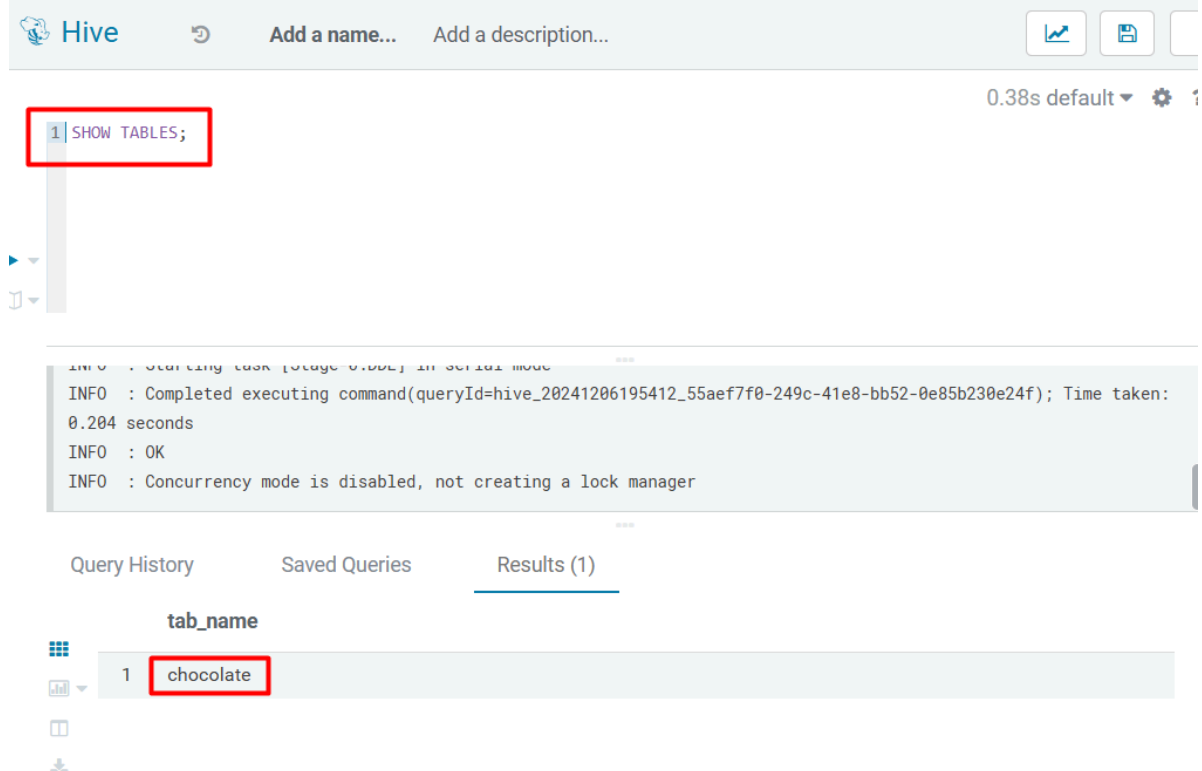
A success message is displayed: "Success."

The Query History tab shows the executed command:

```
hace unos segundos CREATE TABLE if not exists chocolate ( company_maker STRING, bean_origin STRING,  
ref INT, review_date INT, cocoa_percent FLOAT, company_location STRING,  
bean_type STRING, broad_bean_origin STRING, rating FLOAT )
```

Usamos show tables para comprobar si se ha creado

SHOW TABLES;



The Hive interface shows the execution of the command 'SHOW TABLES;'. The command is entered in the editor and highlighted with a red box:

```
1 SHOW TABLES;
```

The execution log shows the following information:

```
INFO : Starting task [stage-0.000] in serial mode  
INFO : Completed executing command(queryId=hive_20241206195412_55aef7f0-249c-41e8-bb52-0e85b230e24f); Time taken:  
0.204 seconds  
INFO : OK  
INFO : Concurrency mode is disabled, not creating a lock manager
```

The Results (1) tab shows the output of the command:

tab_name
1 chocolate

HBase

Para crear la tabla vamos a usar la consola

```
hbase shell
```

Comando para crear la tabla

```
create 'chocolate','datos_vaina','datos_tableta'
```

```
hbase:002:0> create 'chocolate','datos_vaina','datos_tableta'  
Created table chocolate  
Took 1.0756 seconds
```

```
=> Hbase::Table - chocolate
```

```
hbase:003:0> □
```

Comprobamos que se ha creado con exist

```
exists 'chocolate'
```

```
hbase:006:0> exists 'chocolate'  
Table chocolate does exist
```

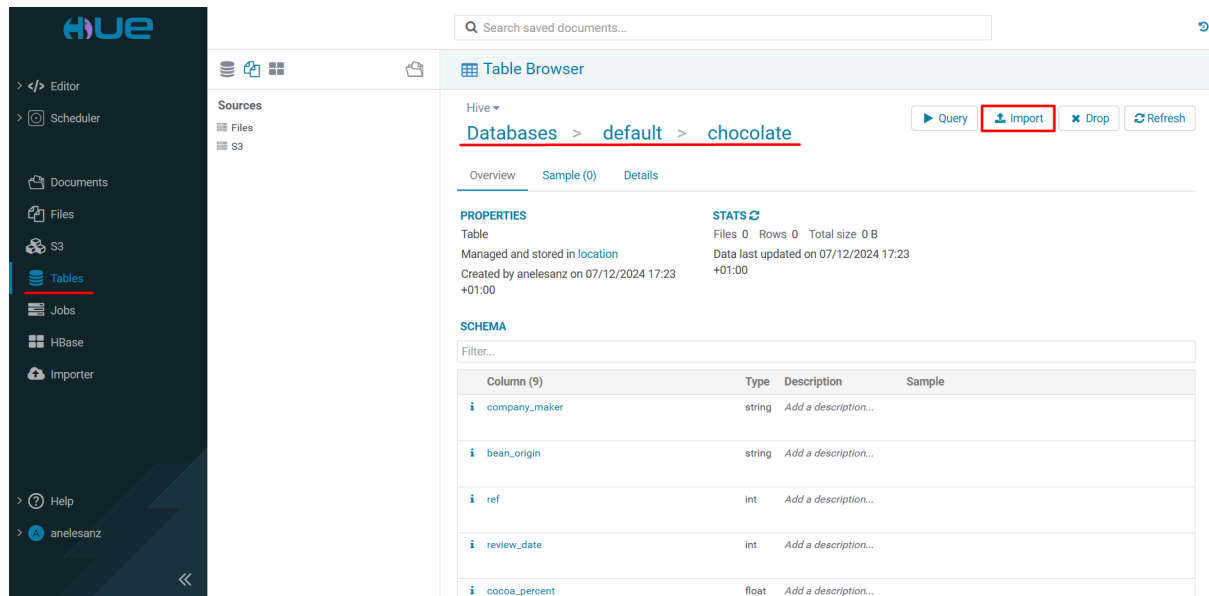
```
Took 0.0988 seconds
```

```
=> true
```

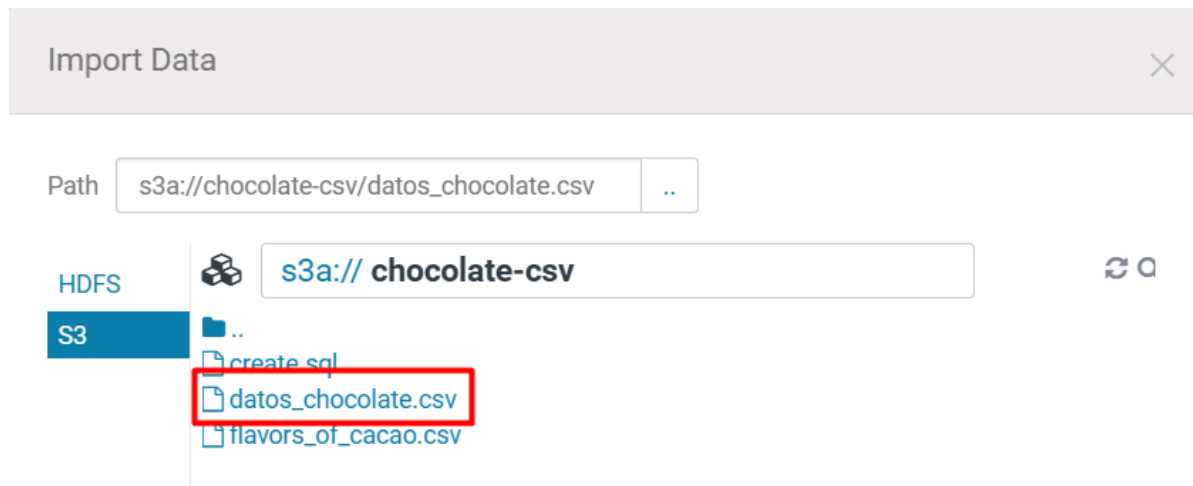
Cargar los datos de los ficheros en las tablas

Hive


Tenemos muchos datos así que vamos a cargarlos a la vez desde un archivo `datos_chocolate.csv` que tenemos en un bucket a nuestra tabla ya creada. Desde el apartado `tables` seleccionamos nuestra tabla `chocolate` en la base de datos `default` y hacemos click en `import`.




Seleccionamos el archivo `datos_chocolate.csv` del bucket.






Hacemos una búsqueda rápida para asegurarnos que haya cargado bien los datos.


Hive




Execute and watch

Add a description...

0.33s default ▾ ⚙ ?

1 | `SELECT * FROM default.chocolate LIMIT 100;`

 ▾
 ▾





```

INFO : Completed executing command(queryId=hive_20241207163938_9a9081ba-1b0f-472c-b8
1b-dd5f7027fcd1); Time taken: 0.0 seconds
INFO : OK
INFO : Concurrency mode is disabled, not creating a lock manager

```

Query History
Saved Queries
Results (100+)

chocolate.company_maker


 ▾



1	company_maker,bean_origin,ref,review_date,cocoa_percent,company_location,rating,
2	A. Morin,Agua Grande,1876,2016,0.63,France,3.75,NULL,Sao Tome
3	A. Morin,Kpime,1676,2015,0.7,France,2.75,NULL,Togo
4	A. Morin,Atsane,1676,2015,0.7,France,3.0,NULL,Togo
5	A. Morin,Akata,1680,2015,0.7,France,3.5,NULL,Togo
6	A. Morin,Quilla,1704,2015,0.7,France,3.5,NULL,Peru
7	A. Morin,Carenero,1315,2014,0.7,France,2.75,Criollo,Venezuela

HBase

Continuamos con la filosofía anterior de no meter todos los datos a mano y para eso vamos a hacer un programa en python con la librería *happybase*.

Primero que todo la instalamos con pip

```
pip install happybase
```

Necesitamos editar *hbase-site.xml* para habilitar Thrift y especificar el puerto que va a usar.

```
<property>
  <name>hbase.thrift.enabled</name>
  <value>true</value>
</property>
<property>
  <name>hbase.thrift.port</name>
  <value>9090</value>
</property>
```

Permitimos el acceso del puerto en el cluster

Bloquear el acceso público [Información](#)

El bloqueo del acceso público de Amazon EMR impide el lanzamiento de un clúster cuando está asociado a regla IPv6 ::/0 (acceso público) en un puerto, a menos que el puerto se especifique explícitamente como una excepción

Configuración del bloqueo del acceso público

Bloquear el acceso público

✓ Activado

Excepciones de rango de puertos

Un clúster se puede lanzar con reglas del grupo de seguridad que permiten el tráfico entrante desde todas las direcciones IP predeterminada por SSH.

22

8020

8888

9090

9870

10000

18080

50010

Reglas de entrada (12)

Administrar etiquetas Editar reglas de entrada

Buscar

<input type="checkbox"/>	Name	ID de la regla del gr...	Versión de IP	Tipo	Protocolo	Intervalo de puer
<input type="checkbox"/>	-	sgr-084afe4a496940af8	-	Todos los ICMP IPv4	ICMP	Todo
<input type="checkbox"/>	-	sgr-035e9405802a0a65d	IPv4	SSH	TCP	22
<input type="checkbox"/>	-	sgr-060e344d9a5a409cd	-	Todos los ICMP IPv4	ICMP	Todo
<input type="checkbox"/>	-	sgr-08d742503166828ec	IPv4	TCP personalizado	TCP	8888
<input type="checkbox"/>	-	sgr-0d4fe683f11be8282	-	Todos los UDP	UDP	0 - 65535
<input type="checkbox"/>	-	sgr-029c5afa6ee07b6be	-	Todos los TCP	TCP	0 - 65535
<input type="checkbox"/>	-	sgr-090f4a29ea05d2796	IPv4	TCP personalizado	TCP	18080
<input type="checkbox"/>	-	sgr-044981f6db0e5bb06	IPv4	TCP personalizado	TCP	9090
<input type="checkbox"/>	-	sgr-0569fdcc5d320fce2	IPv4	TCP personalizado	TCP	8088
<input type="checkbox"/>	-	sgr-0f7676d4075bb67de	IPv4	TCP personalizado	TCP	9870
<input type="checkbox"/>	-	sgr-0e1bbde1831653195	-	Todos los UDP	UDP	0 - 65535
<input type="checkbox"/>	-	sgr-047d52aee9847c710	-	Todos los TCP	TCP	0 - 65535

Ejecutamos nuestro programa en python

```
import happybase
import csv

# Conexión con HBase
connection = happybase.Connection('ec2-52-201-227-30.compute-1.amazonaws.com',
port=9090)
table_choco = connection.table('chocolate')

# Leer el CSV y cargar los datos
with open('./datos_chocolate.csv','r') as csvfile:
    reader = csv.DictReader(csvfile)
    id = 0
    for row in reader:
        table_choco.put(id,
            {
                'datos_vaina:bean_origin': row['bean_origin'],
                'datos_vaina:bean_type': row['bean_type'],
                'datos_vaina:broad_bean_origin': row['broad_bean_origin'],
                'datos_tableta:company_maker': row['company_maker'],
                'datos_tableta:fer': row['ref'],
                'datos_tableta:review_date': row['review_date'],
                'datos_tableta:cocoa_percent': row['cocoa_percent'],
                'datos_tableta:company_location': row['company_location'],
                'datos_tableta:rating': row['rating'],
            }
        )
        id += 1
```


Usamos scan para comprobar que se han introducido

```
scan 'chocolate';
```

```
hbase:022:0> scan 'chocolate'
ROW                                COLUMN+CELL
1                                  column=datos_tableta:bean origin, timestamp=2024-12-07T18:34:29.423, value=Agua Grande
1                                  column=datos_tableta:cocoa_percent, timestamp=2024-12-07T18:36:34.336, value=0.67
1                                  column=datos_tableta:company_location, timestamp=2024-12-07T18:31:18.048, value=France
1                                  column=datos_tableta:company_maker, timestamp=2024-12-07T18:27:08.387, value=A. Morin
1                                  column=datos_tableta:fer, timestamp=2024-12-07T18:30:10.699, value=1676
1                                  column=datos_tableta:reviwe date, timestamp=2024-12-07T18:30:30.716, value=2016
1                                  column=datos_vaina:broad bean origin, timestamp=2024-12-07T18:32:11.137, value=Sao Tome
1                                  column=datos_vaina:broad bean type, timestamp=2024-12-07T18:32:44.232, value=Criollo
2                                  column=datos_tableta:bean origin, timestamp=2024-12-07T18:34:11.817, value=Kpime
2                                  column=datos_tableta:cocoa_percent, timestamp=2024-12-07T18:36:42.769, value=0.7
2                                  column=datos_tableta:company_maker, timestamp=2024-12-07T18:33:44.979, value=A. Morin
2                                  column=datos_tableta:fer, timestamp=2024-12-07T18:35:11.091, value=1876
2                                  column=datos_tableta:reviwe date, timestamp=2024-12-07T18:36:03.601, value=2015
```

Crear diferentes consultas sobre las tablas

Hive

No se porqué no me deja hacer consultas más allá de selects sin condiciones pero las consultas serían las siguientes:

- Total de filas

```
SELECT count(*) FROM chocolate;
```

- Media de rating por compañía

```
SELECT company_maker, avg(rating) FROM chocolate GROUP BY company_maker;
```

- Los orígenes de las vainas

```
SELECT distinct(broad_bean_origin) FROM chocolate;
```

- Cantidad de filas por tipo de vaina

```
SELECT count(*) FROM chocolate GROUP BY bean_type;
```

- Tabletas con porcentaje de cacao mayor a 70% ordenadas por referencia

```
SELECT count(*) FROM chocolate WHERE cocoa_percent >=0.7 ASC ref;
```

- Cambiar los tipos de vaina de las que tengas Sao Tome como origen

```
UPDATE chocolate SET bean_type = 'Criollo' WHERE bean_origin = 'Sao Tome';
```

HBase

- Mostrar todos los datos

```
scan 'chocolate'
```

```
hbase:022:0> scan 'chocolate'
ROW                                COLUMN+CELL
1                                  column=datos_tableta:bean_origin, timestamp=2024-12-07T18:34:29.423, value=Agua Grande
1                                  column=datos_tableta:cocoa_percent, timestamp=2024-12-07T18:36:34.336, value=0.67
1                                  column=datos_tableta:company_location, timestamp=2024-12-07T18:31:18.048, value=France
1                                  column=datos_tableta:company_maker, timestamp=2024-12-07T18:27:08.387, value=A. Morin
1                                  column=datos_tableta:fer, timestamp=2024-12-07T18:30:10.699, value=1676
1                                  column=datos_tableta:reviwe_date, timestamp=2024-12-07T18:30:30.716, value=2016
1                                  column=datos_vaina:broad_bean_origin, timestamp=2024-12-07T18:32:11.137, value=Sao Tome
1                                  column=datos_vaina:broad_bean_type, timestamp=2024-12-07T18:32:44.232, value=Criollo
2                                  column=datos_tableta:bean_origin, timestamp=2024-12-07T18:34:11.817, value=Kpime
2                                  column=datos_tableta:cocoa_percent, timestamp=2024-12-07T18:36:42.769, value=0.7
2                                  column=datos_tableta:company_maker, timestamp=2024-12-07T18:33:44.979, value=A. Morin
2                                  column=datos_tableta:fer, timestamp=2024-12-07T18:35:11.091, value=1876
2                                  column=datos_tableta:reviwe_date, timestamp=2024-12-07T18:36:03.601, value=2015
```

- Mostrar sólo los datos de una fila

```
get 'chocolate','1'
```

```
hbase:023:0> get 'chocolate','1'
COLUMN                                CELL
datos_tableta:bean_origin              timestamp=2024-12-07T18:34:29.423, value=Agua Grande
datos_tableta:cocoa_percent            timestamp=2024-12-07T18:36:34.336, value=0.67
datos_tableta:company_location          timestamp=2024-12-07T18:31:18.048, value=France
datos_tableta:company_maker            timestamp=2024-12-07T18:27:08.387, value=A. Morin
datos_tableta:fer                      timestamp=2024-12-07T18:30:10.699, value=1676
datos_tableta:reviwe_date               timestamp=2024-12-07T18:30:30.716, value=2016
datos_vaina:broad_bean_origin           timestamp=2024-12-07T18:32:11.137, value=Sao Tome
datos_vaina:broad_bean_type            timestamp=2024-12-07T18:32:44.232, value=Criollo
1 row(s)
```

- Modificar la fila 1 para que company_maker sea Elena

```
put 'chocolate','datos_tableta:company_maker'
```

```
hbase:032:0> put 'chocolate','1','datos_tableta:company_maker','Elena'
Took 0.0044 seconds
```

- Obtener la columna company_maker de la fila 1

```
get 'chocolate','1',{COLUMN=>'datos_tableta:company_maker'}
```

```
hbase:033:0> get 'chocolate','1',{COLUMN=>'datos_tableta:company_maker'}
COLUMN                                CELL
datos_tableta:company_maker            timestamp=2024-12-07T18:46:34.579, value=Elena
1 row(s)
Took 0.0077 seconds
hbase:034:0> []
```

Bibliografía

- <https://oscarfmdc.medium.com/apache-hbase-introducci%C3%B3n-aprenderbigdata-com-d29f1b7bbdc7>
- https://www.tutorialspoint.com/hbase/hbase_create_table.htm
- https://www.tutorialspoint.com/es/hive/hive_introduction.htm
- https://docs.cloudera.com/runtime/7.2.18/using-hiveql/topics/hive_update_data_in_a_hive_table.html
- <https://stackoverflow.com/questions/17425492/hive-insert-query-like-sql>
- <https://stackoverflow.com/questions/17810537/how-to-delete-and-update-a-record-in-hive>