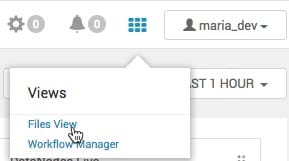
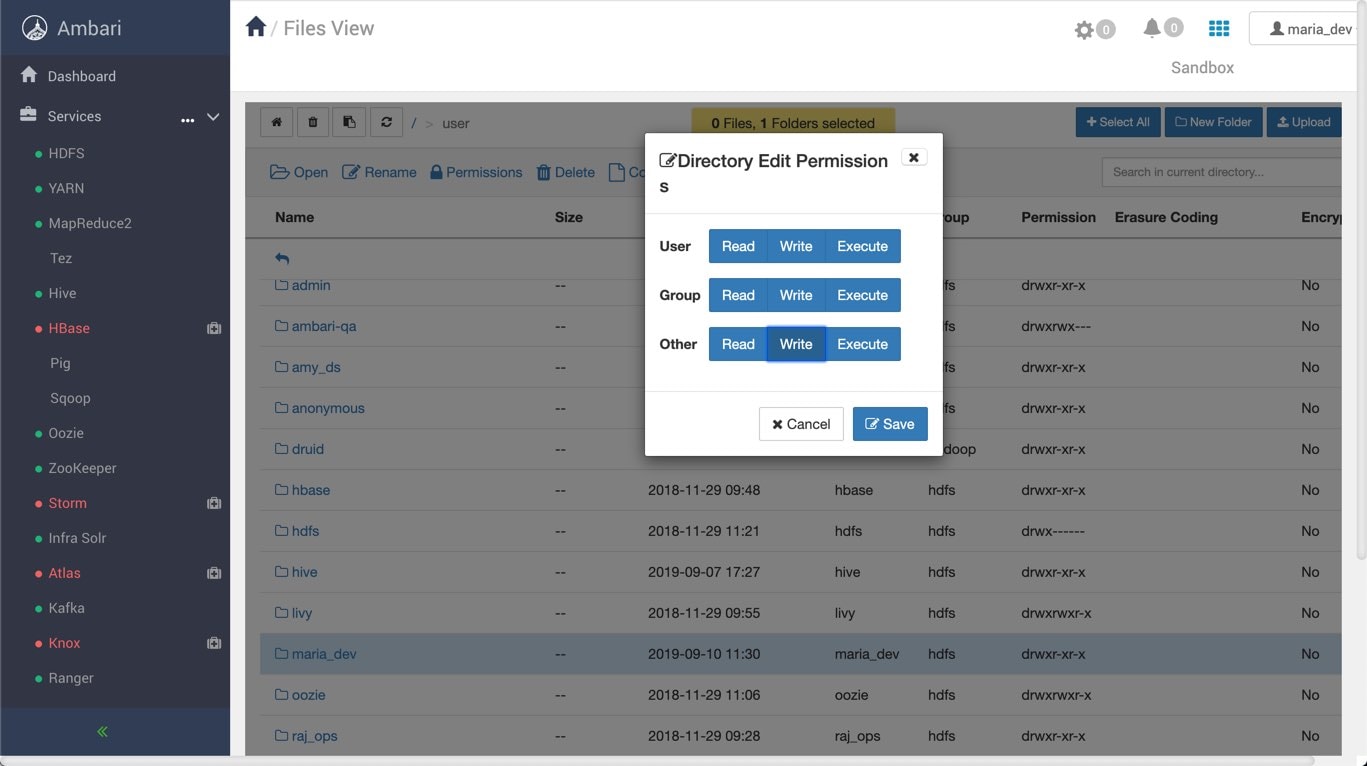
Apuntes HUE

**Cargar los archivos de datos**

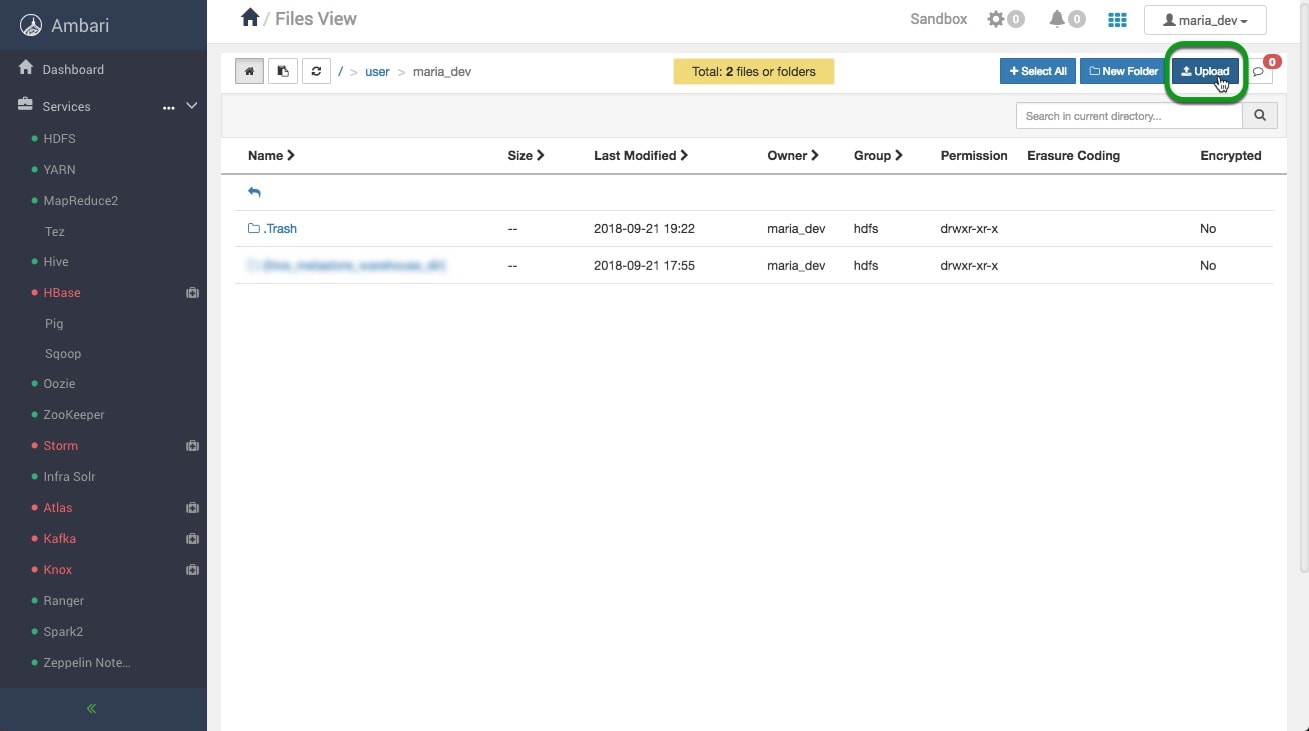
Comenzar seleccionando del HDFS el menú Files view en la parte superior.



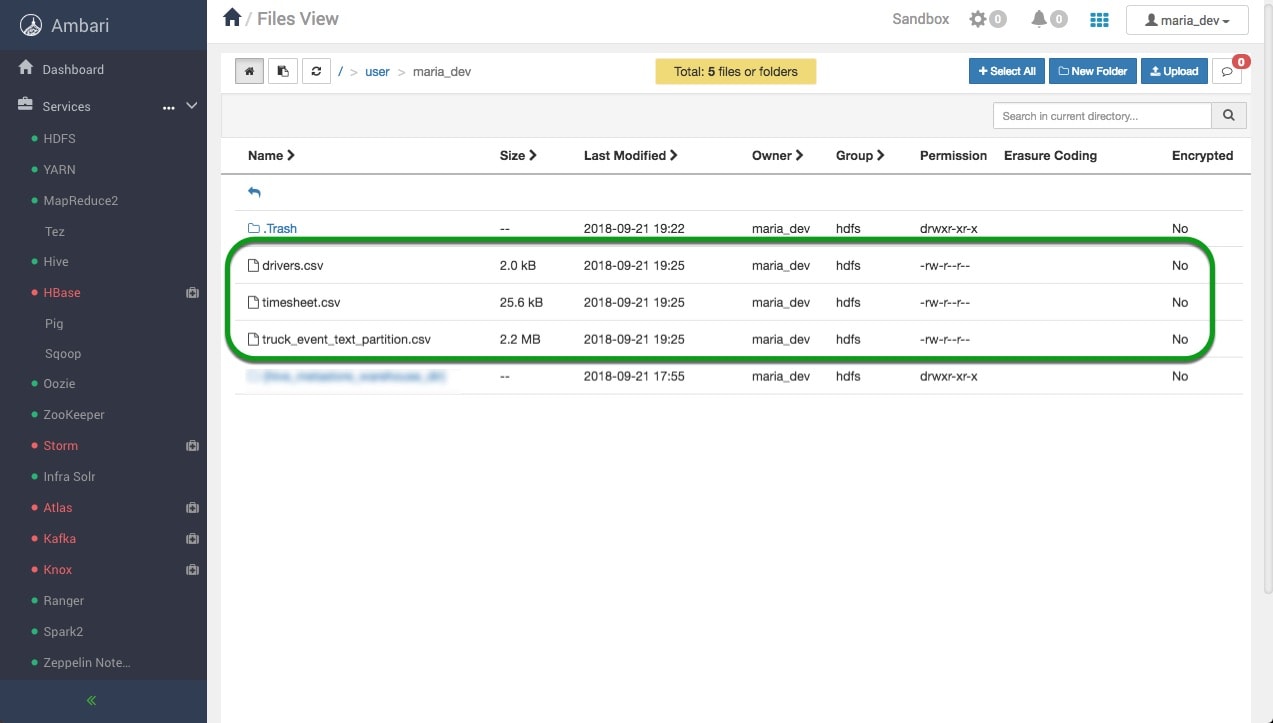
Configurar los permisos de la carpeta /user/nuestro\_usuario para leer, escribir, ejecutar:



Navegar a /user/nuestro\_usuario y hacer clic en el botón **Upload** para seleccionar los archivos que se quieran cargar en el entorno.

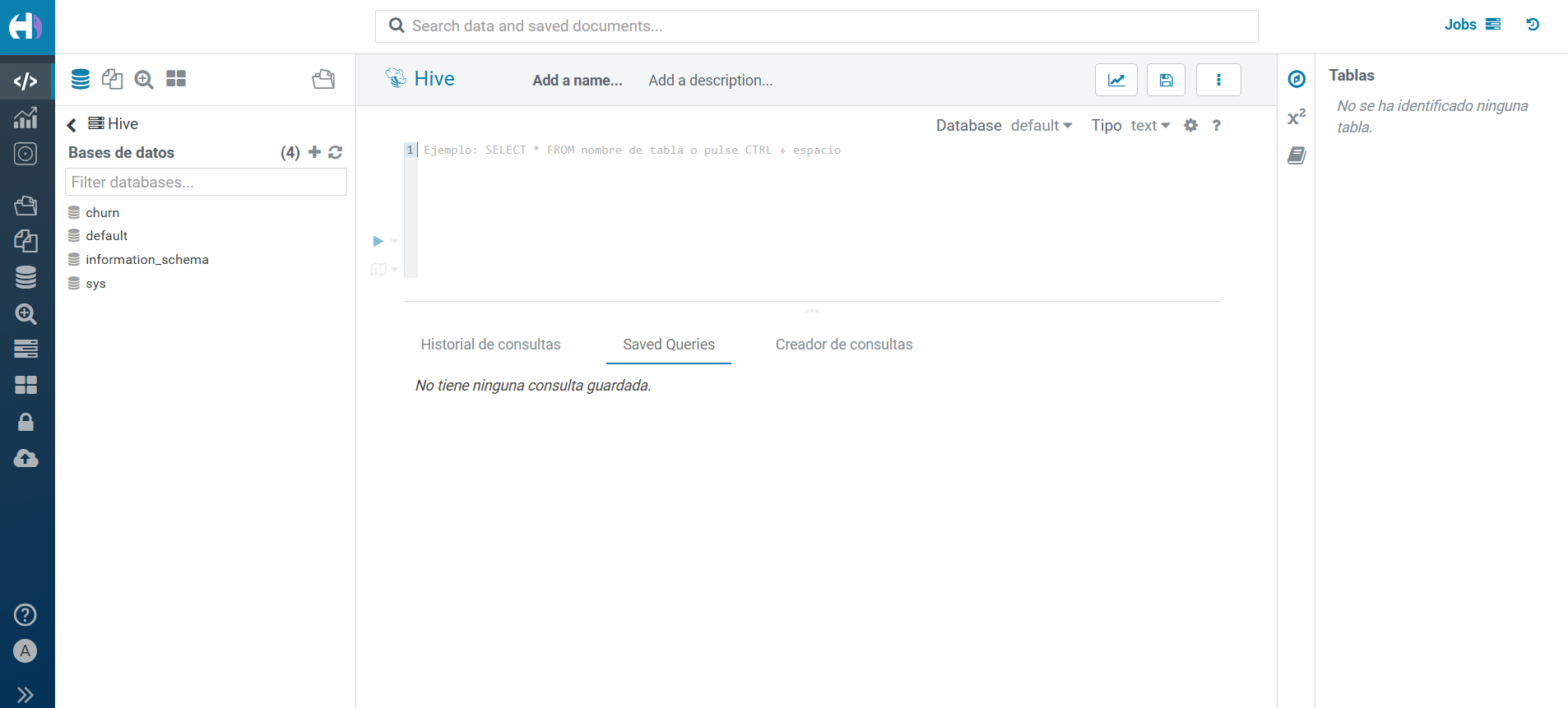


Hacer clic en el botón **browse** para abrir el cuadro de diálogo. Navegar y seleccionar hasta el archivo drivers.csv en el disco local y hacer clic en **open**. Hacer lo mismo con los demás.

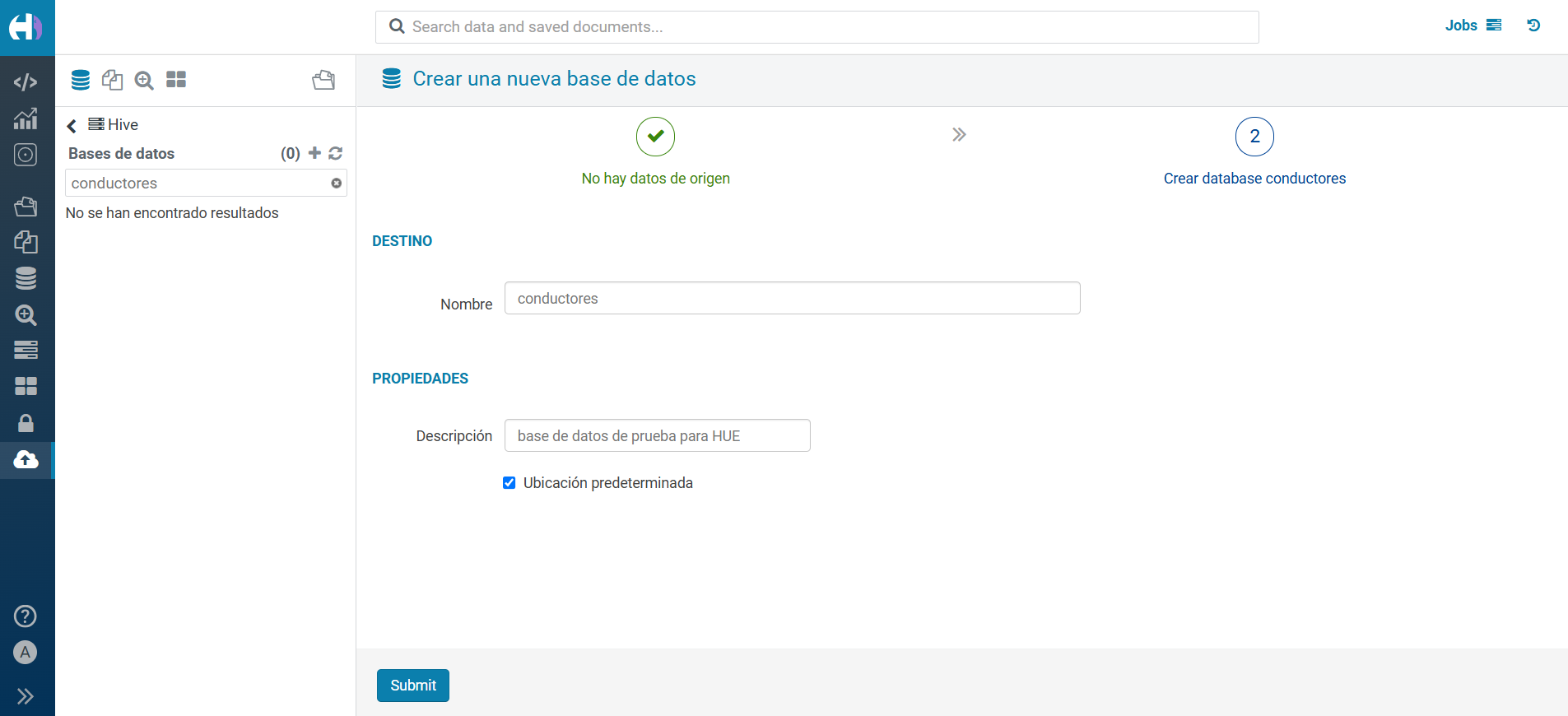


[**Iniciar HUE**](https://www.cloudera.com/tutorials/how-to-process-data-with-apache-hive.html#start-das)

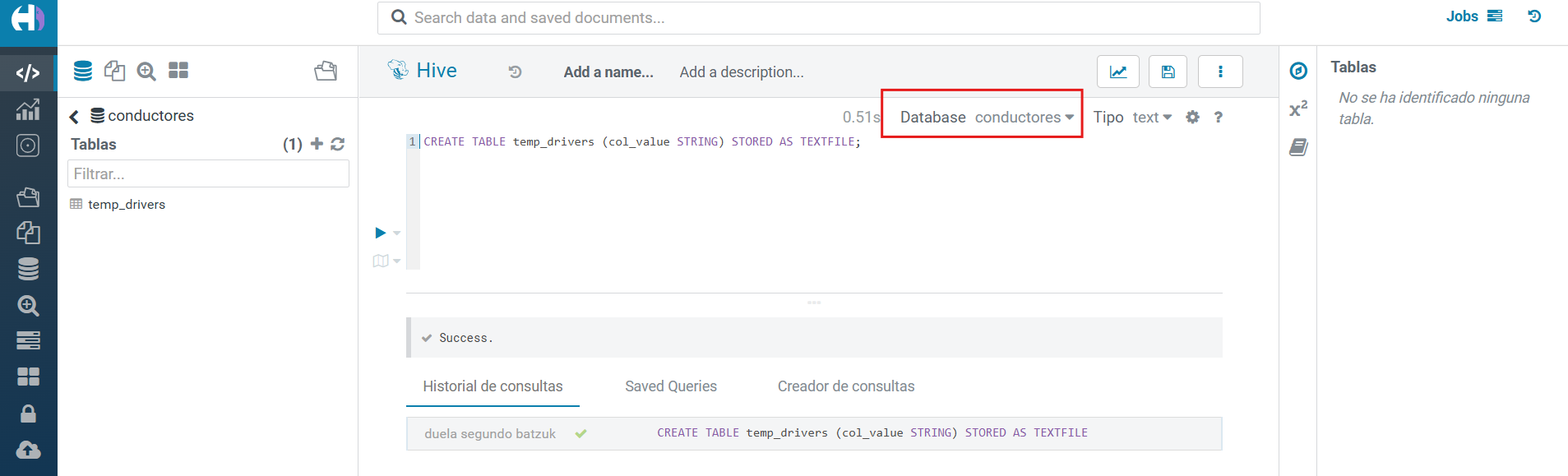
Abrir la UI de Hive en <http://hadoop-admin1:8888/hue/editor/?type=hive>. HUE proporciona una interfaz de usuario para el sistema de almacenamiento de datos de Hive para Hadoop.



**Crear una bbdd nueva**

****

[**Crear una tabla temp\_drivers**](https://www.cloudera.com/tutorials/how-to-process-data-with-apache-hive.html#create-table-temp-drivers)

****

La primera tarea que haremos será crear una tabla para almacenar los datos. Escribiremos la consulta en el **Query Editor**. Una vez que haya escrito la consulta, presione el botón **Execute** de la parte inferior.

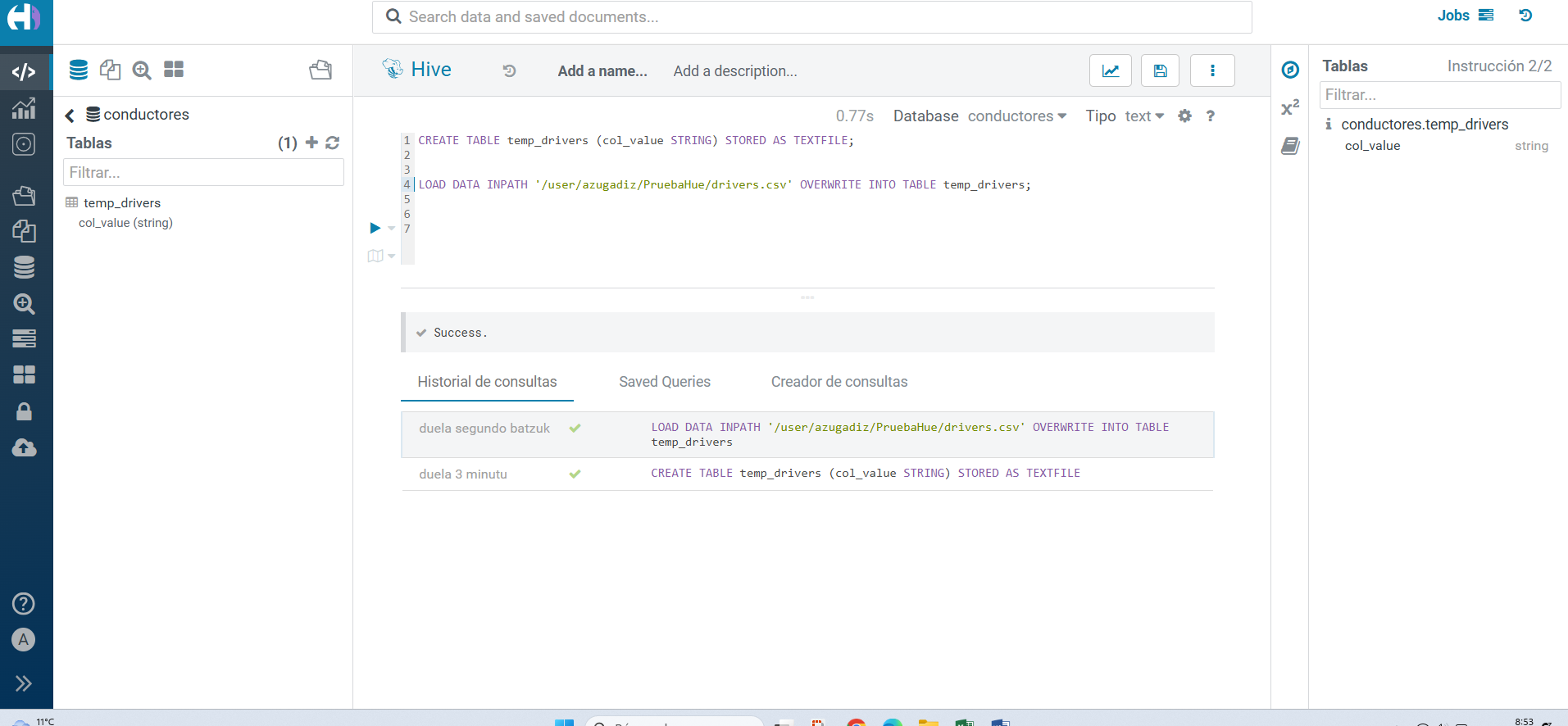
CREATE TABLE temp\_drivers (col\_value STRING) STORED AS TEXTFILE

La consulta no devuelve ningún resultado porque en este punto acabamos de crear una tabla vacía y no hemos copiado ningún dato en ella. Una vez que se ha ejecutado la consulta, podemos actualizar el apartado Database volviendo a seleccionar sobre **Database**. Nos aparecerá la nueva tabla (temp\_drivers).

[**Crear una consulta para llenar la tabla Hive temp\_drivers con datos drivers.csv**](https://www.cloudera.com/tutorials/how-to-process-data-with-apache-hive.html#create-query-to-populate-hive-table-temp-drivers-with-driverscsv-data)

La siguiente línea de código cargará el archivo de datos drivers.csv en la tabla temp\_drivers.

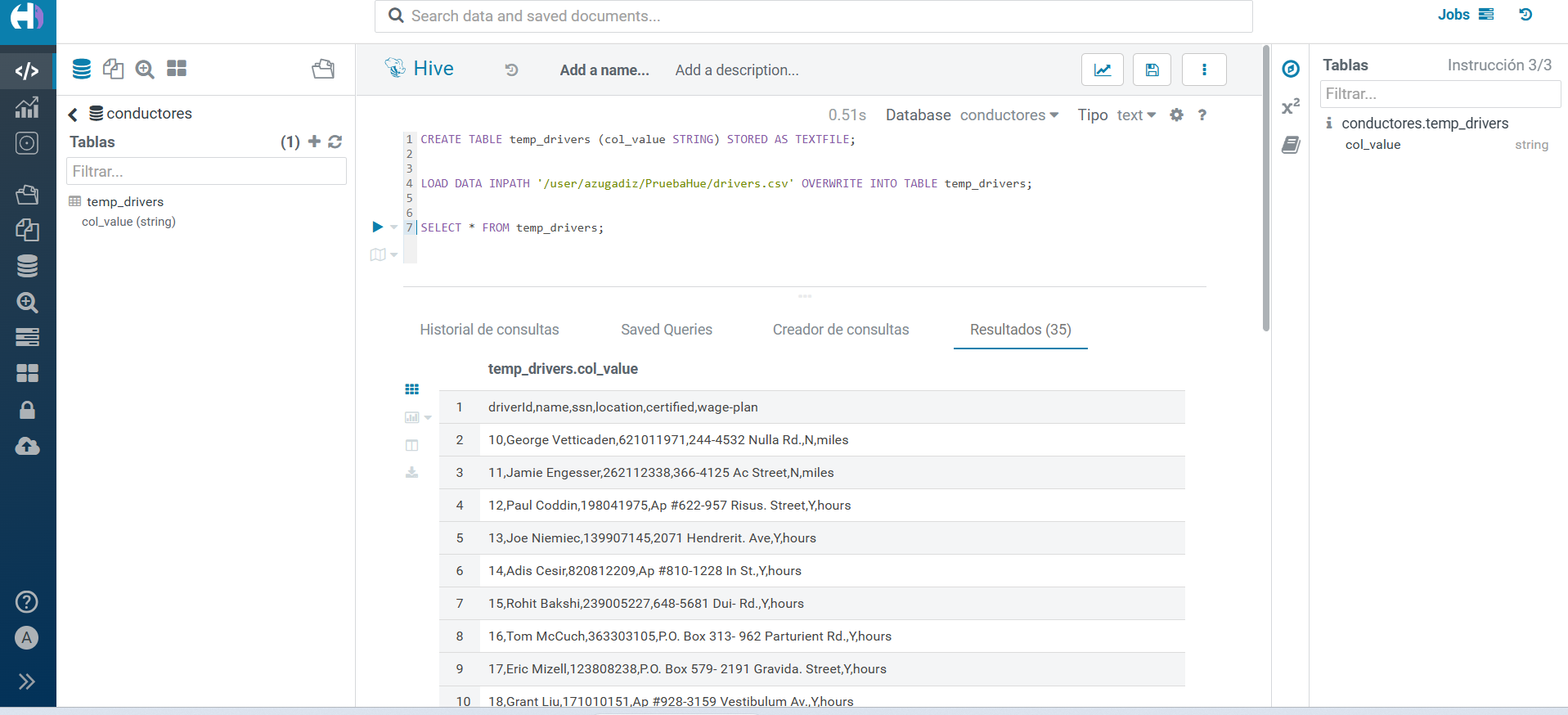
LOAD DATA INPATH '/user/nuestro\_usuario/drivers.csv' OVERWRITE INTO TABLE temp\_drivers



Después de ejecutar LOAD DATA, podemos ver que la tabla temp\_drivers se completó con datos de drivers.csv. Tened en cuenta que Hive ha **consumido** el archivo de datos drivers.csv durante este paso. Si miramos en el File Browser veremos que drivers.csv ya no existe.

Echemos un vistazo rápido a lo que está almacenado en nuestra tabla temporal:

SELECT \* FROM temp\_drivers

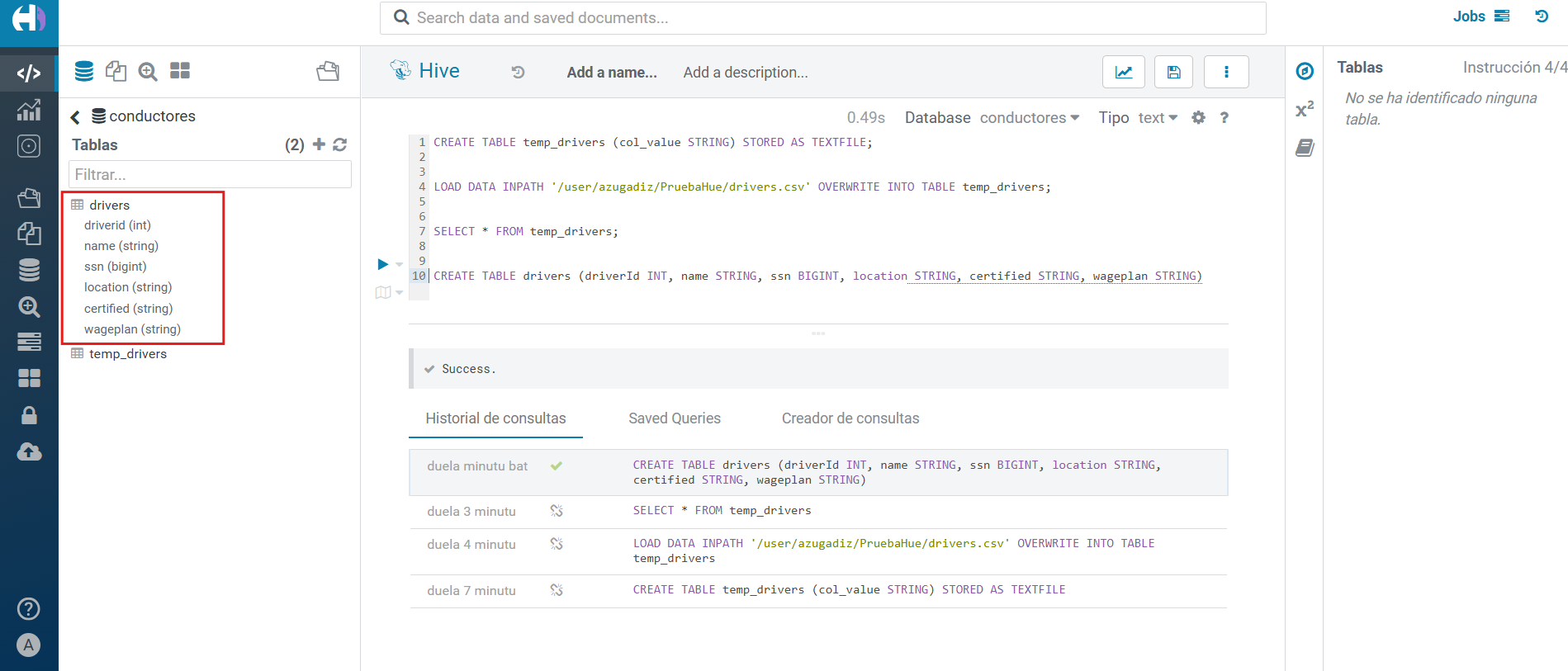


### [Crear controladores de tabla](https://www.cloudera.com/tutorials/how-to-process-data-with-apache-hive.html#create-table-drivers)

Ahora que hemos leído los datos, podemos empezar a trabajar con ellos. Lo siguiente que queremos hacer es, extraer los datos. Primero escribiremos una consulta para crear una nueva tabla llamada **drivers** como contener los datos. Esa tabla tendrá seis columnas: driverId, name, ssn, location, certified y wage-plan.

CREATE TABLE drivers (driverId INT, name STRING, ssn BIGINT, location STRING, certified STRING, wageplan STRING)

### Crear la consulta para extraer datos de temp\_drivers y almacenarlos en controladores

A continuación, extraemos los datos que queremos de temp\_drivers y los copiamos en las tablas drivers. Lo haremos con la función regexp. Para hacer esto, vamos a construir una consulta de varias líneas, una por cada columna. 

Cada llamada a regexp\_extract va a extraer una de los campos: driverId, name, ssn, location, certified y wage-plan que contiene la tabla temp\_drivers.

insert overwrite table drivers

SELECT

regexp\_extract(col\_value, '^(?:([^,]\*),?){1}', 1) driverId,

regexp\_extract(col\_value, '^(?:([^,]\*),?){2}', 1) name,

regexp\_extract(col\_value, '^(?:([^,]\*),?){3}', 1) ssn,

regexp\_extract(col\_value, '^(?:([^,]\*),?){4}', 1) location,

regexp\_extract(col\_value, '^(?:([^,]\*),?){5}', 1) certified,

regexp\_extract(col\_value, '^(?:([^,]\*),?){6}', 1) wageplan

from temp\_drivers

**Tened cuidado con el patrón, no hay espacios!!!**

Después de cargar los datos, visualizamos la tabla drivers:

SELECT \* FROM drivers

### Crear las tablas temp\_timesheet y timesheet de la misma forma

Del mismo modo, tenemos que crear una tabla llamada temp\_timesheet y luego cargar el archivo timesheet.csv.

Escribid las siguientes consultas una por una:

CREATE TABLE temp\_timesheet (col\_value string) STORED AS TEXTFILE

LOAD DATA INPATH '/user/maria\_dev/timesheet.csv' OVERWRITE INTO TABLE temp\_timesheet

Los datos se verán de la siguiente manera:

SELECT \* FROM temp\_timesheet LIMIT 10

Ahora crear la tabla timesheet usando la siguiente consulta:

CREATE TABLE timesheet (driverId INT, week INT, hours\_logged INT , miles\_logged INT)

Insertar los datos en la tabla timesheet desde la tabla temp\_timesheet regexp\_extract.

insert overwrite table timesheet

SELECT

regexp\_extract(col\_value, '^(?:([^,]\*),?){1}', 1) driverId,

regexp\_extract(col\_value, '^(?:([^,]\*),?){2}', 1) week,

regexp\_extract(col\_value, '^(?:([^,]\*),?){3}', 1) hours\_logged,

regexp\_extract(col\_value, '^(?:([^,]\*),?){4}', 1) miles\_logged

from temp\_timesheet

Los datos se verán de la siguiente manera:

SELECT \* FROM timesheet LIMIT 10