Apuntes

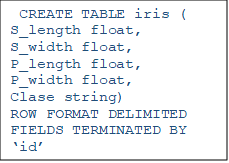
* Es una infraestructura para el almacenaje y consulta de datos basada en Hadoop
* Hadoop proporciona escalabilidad masiva y con capacidades de tolerancia a fallos para el procesamiento y almacenamiento de datos.
* Hive está diseñado para facilitar la consulta y el análisis de grandes volúmenes de datos
* Proporciona un sencillo lenguaje de consulta llamado HiveQL (o HQL), basado en SQL-92
* Las consultas expresadas en HQL se traducen a trabajos MapReduce que se ejecutan en Hadoop
* CREATE DATABASE my\_db; Para crear una base de datos
* USE my\_db; para usarla
* DROP DATABASE my\_db; para borrarla

Definiciones

* Bases de datos: espacio de nombres que agrupa tablas y otras unidades de datos
* Tablas: unidades de datos homogéneas que comparten un mismo esquema
* Particiones: cada tabla puede tener una o más claves de particionado que determinan cómo se almacenan los datos. Optimizan consultas
* Buckets (o Clusters): los datos dentro de cada partición pueden, a su vez, dividirse en buckets basados en el valor de una función de dispersión sobre alguna columna de la tabla. Optimizan joins.

Ejercicios

* Para entrar en Hive: hive
* Modifica la propiedad correspondiente para mostrar por pantalla las cabeceras de las tablas set hive.cli.print.header=true;
* Crear la tabla iris con estos campos:



* Crear un directorio y meter el dataset en hdfs:

Hadoop fs -mkdir /user/Cloudera/hive

Hadoop fs -put /home/Cloudera/ejercicios/ejercicios\_hive/iris\_completo.txt /user/Cloudera/hive

* Almacenar los datos en la tabla: Load data local inpath ‘/home/cloudera/ejercicios/hive/iris\_completo.txt’into table iris
* Mostrar las 5 primeras filas de la tabla iris: Select \* from iris limit 5;
* Utilizando funciones de ventanas, seleccionar la clase, p\_length, s\_length, p\_width, el número de valores distintos de p\_length en todo el dataset, el valor máximo de s\_length por clase y la media de p\_width por clase, ordenado por clase y s\_length de manera descendente.

select clase, p\_length, s\_length, p\_width, count(p\_length) over (partition by p\_length) as pl\_ct, max(s\_length) over (partition by clase) as sl\_ct, avg(p\_width) over (partition by clase) as sl\_av from iris order by clase,s\_length desc;