**UNED**

**Ingeniería Técnica de sistemas**

**Curso 2017/2018**

**Sistemas de Bases de Datos**

**Prácticas obligatorias**

Realizado por:

Alfonso Martín Murillo

DNI: 11.820.406-Q

e-mail: alfonsomartinmurillo@gmail.com

Teléfono móvil: 601058048

**INDICE**

[Introducción 3](#_Toc503034150)

[Entorno de desarrollo, trabajo y ejecución 3](#_Toc503034151)

[Contenido del ZIP y ejecución de la aplicación 3](#_Toc503034152)

[Respuesta a las cuestiones planteadas en el enunciado de la práctica 4](#_Toc503034153)

[Ejercicio 1. Preprocesado del fichero con datos bioinformáticos 4](#_Toc503034154)

[Ejercicio 2. Media máxima de GSM19023 para valores entre 0 y 10 5](#_Toc503034155)

[Ejercicio 3. Media total de GSD19025 5](#_Toc503034156)

[Ejercicio 4. Creación de tablas HIVE 6](#_Toc503034157)

[Ejercicio 5. Creación de consultas HIVE 9](#_Toc503034158)

[Valoración personal sobre la práctica 11](#_Toc503034159)

# Introducción

Este documento representa la memoria a elaborar en la práctica de Sistemas de Bases de Datos del grado en Ingeniería Informática

# Entorno de desarrollo, trabajo y ejecución

Los datos de ejecución mostrados en esta memoria, así como el código desarrollado y la elaboración de la memoria se han realizado sobre un ordenador MACBOOK AIR INTEL CORE 2 DUO 2,4 GHZ modelo 2008, con 4 GB de memoria RAM.

* Ordenador Macbook Air 2013. Intel CORE I5 a 1,3 ghz con 8 GB de Memoria.
* Sistema de virtualización Virtual box. Se ha trabajado con la máquina virtual de CLOUDERA suministrada por el equipo docente.

# Contenido del ZIP y ejecución de la aplicación

El contenido de la práctica se encuentra empaquetado en un fichero ZIP de nombre madrid-martinmurilloalfonso.zip. A continuación se detalla el contenido de dicho fichero:

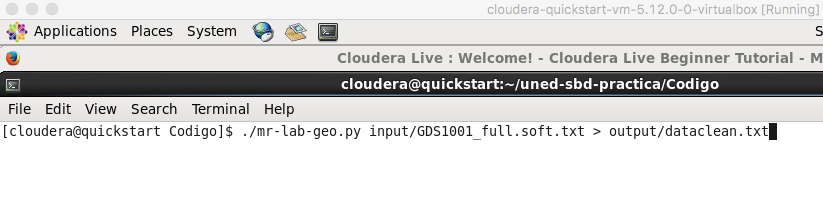
|  |  |
| --- | --- |
| mr-lab-geo.py | Fichero Python que se encarga de realizar el primer pre-procesado. Generar el fichero DATACLEAN.TXT a partir de **GSD1001\_full.soft.txt** |
| mr-lab-mayor-avg.py | Fichero python que se encarga de generar los resultados solicitados en el ejercicio dos, tomando como datos de entrada los datos generados en el apartado 1 |
| mr-lab-media-total.py | Fichero Python que se encarga de generar los resultados solicitados en el ejercicio tres, tomando como datos de entrada los datos generados en el apartado 1 |
| Uned-sbbdd-practica-v1.doc |  |
| Uned-sbbdd-practica-v1.pdf |  |

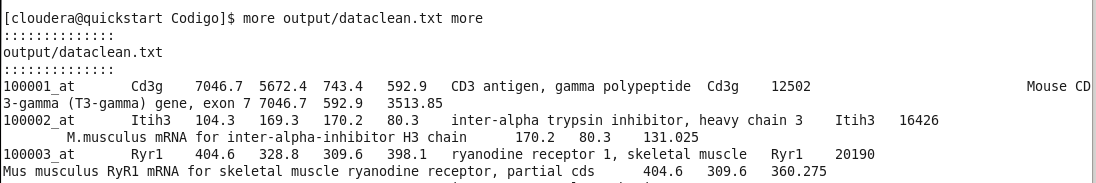
# Respuesta a las cuestiones planteadas en el enunciado de la práctica

## Ejercicio 1. Preprocesado del fichero con datos bioinformáticos

Partiendo del fichero de datos de entrada **GSD1001\_full.soft.txt** he desarrollado el programa en PYTHON denominado **mr-lab-geo.py** el cual tiene como finalidad generar la salida solicitada por el primer ejercicio, esto es, quedarnos con las primeras 13 columnas y generar tres nuevas (max,min,avg) resultado de calcular el valor máximo, mínimo y medio de las columnas " gsm19023", "gsd19024", "gsd19025" y "gsd19026" para cada fila, respectivamente.

Se adjunta captura de pantalla de una llamada a mi programa mr-lab-geo.py, donde tomando como parámetro el fichero GSD1001\_full.soft.txt redirijo la salida a un fichero denominado dataclean.txt.

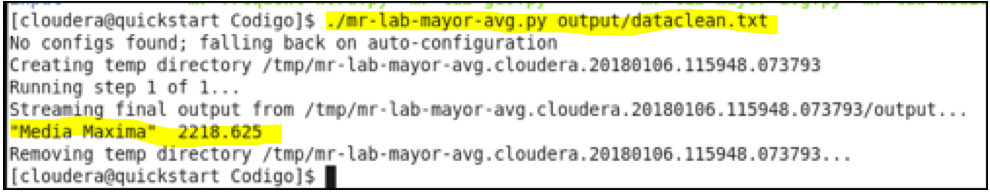
 Como comprobación posterior se puede realizar un CAT o MORE al fichero resultante para comprobar su contenido.



El resultado de este primer programa se va a utilizar como entrada para los dos programas siguientes.

## Ejercicio 2. Media máxima de GSM19023 para valores entre 0 y 10

Partiendo del fichero **DATACLEAN.TXT** generado en el apartado 1 he desarrollado un programa en Python denominado **mr-lab-mayor-avg.py** el cual, utilizando internamente la librería mrjob genera como salida el mayor “avg” de aquellas filas cuyo valor de "gsm19023" esté entre 100 y 1000.

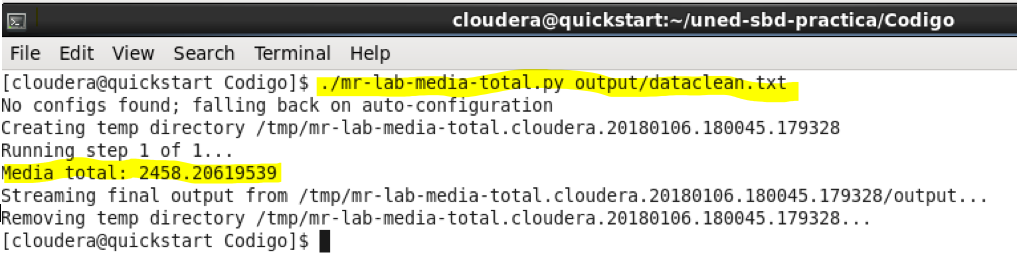


El programa consiste básicamente en la definición de un mapper que preprocesa y filtra la información de entrada, y un reducer que se encarga de generar el valor máximo (MAX) para el campo average.

El programa genera en la salida la media máxima.

## Ejercicio 3. Media total de GSD19025

Partiendo del fichero DATACLEAN.TXT generado en el apartado 1 he desarrollado un programa en Python denominado mr-lab-media-total.py el cual, utilizando internamente la librería mrjob genera como salida la media total del campo gsd19025.

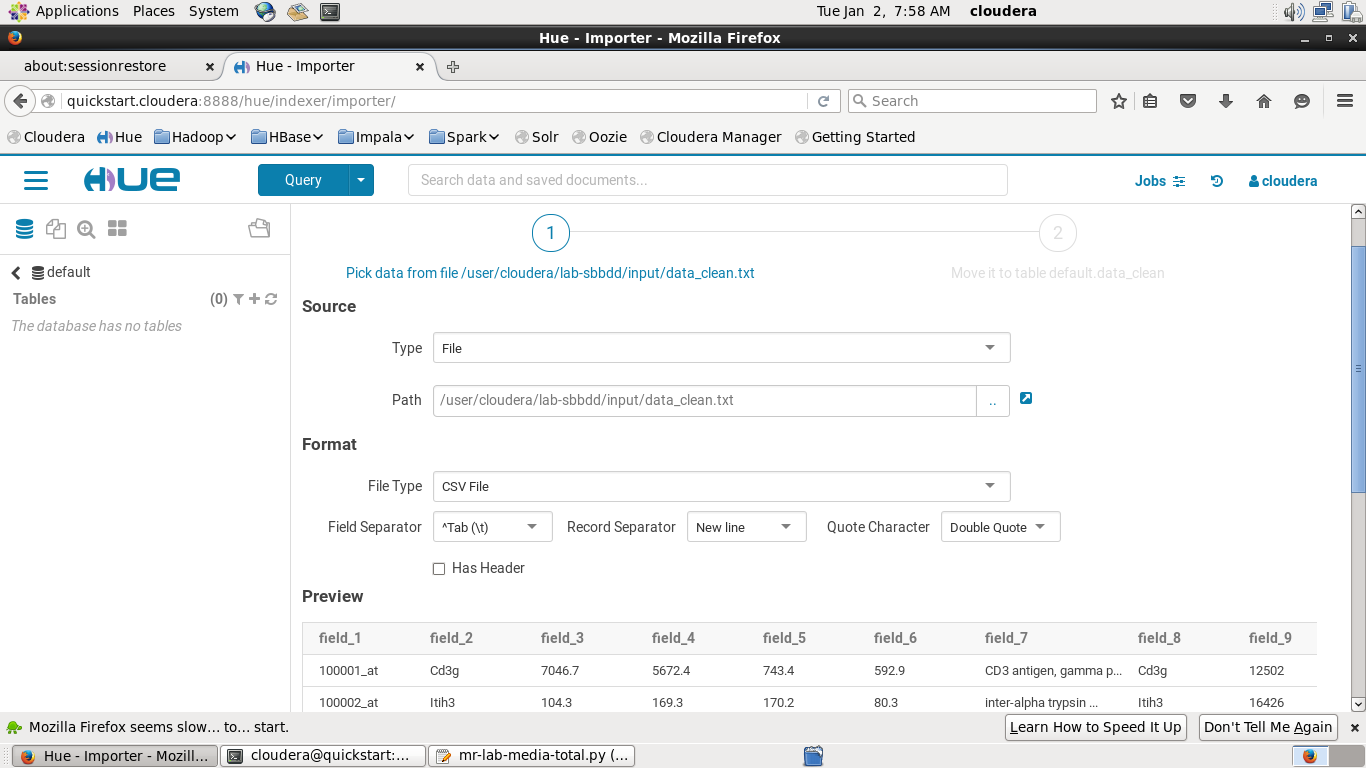


Al no disponer de la función AVERAGE la única complicación está en realizar dicha media, para lo cual debemos de sumar todos los elementos, calcular el nº total de los mismos y finalmente realizar la división correspondiente.

## Ejercicio 4. Creación de tablas HIVE

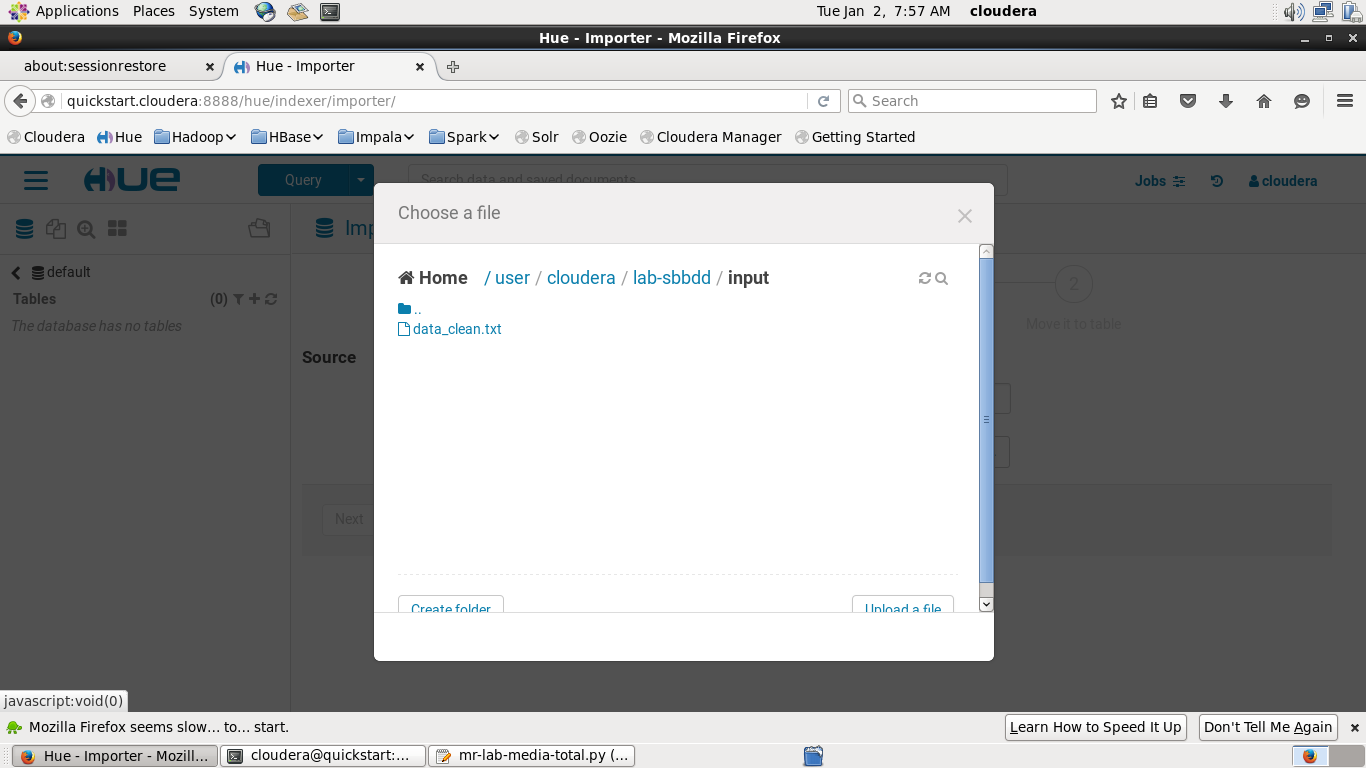
El proceso de creación de tablas HIVE lo he realizado directamente desde las utilidades existentes en HUE. Al entrar en HUE disponemos en la parte izquierda de un pequeño interfaz que nos permite crear o seleccionar una base de datos. En mi caso he seleccionado la base de datos existente por defecto (“default”).

Al hacer click sobre ella el sistema me ofrece opciones para la creación de tablas (el botón de +).



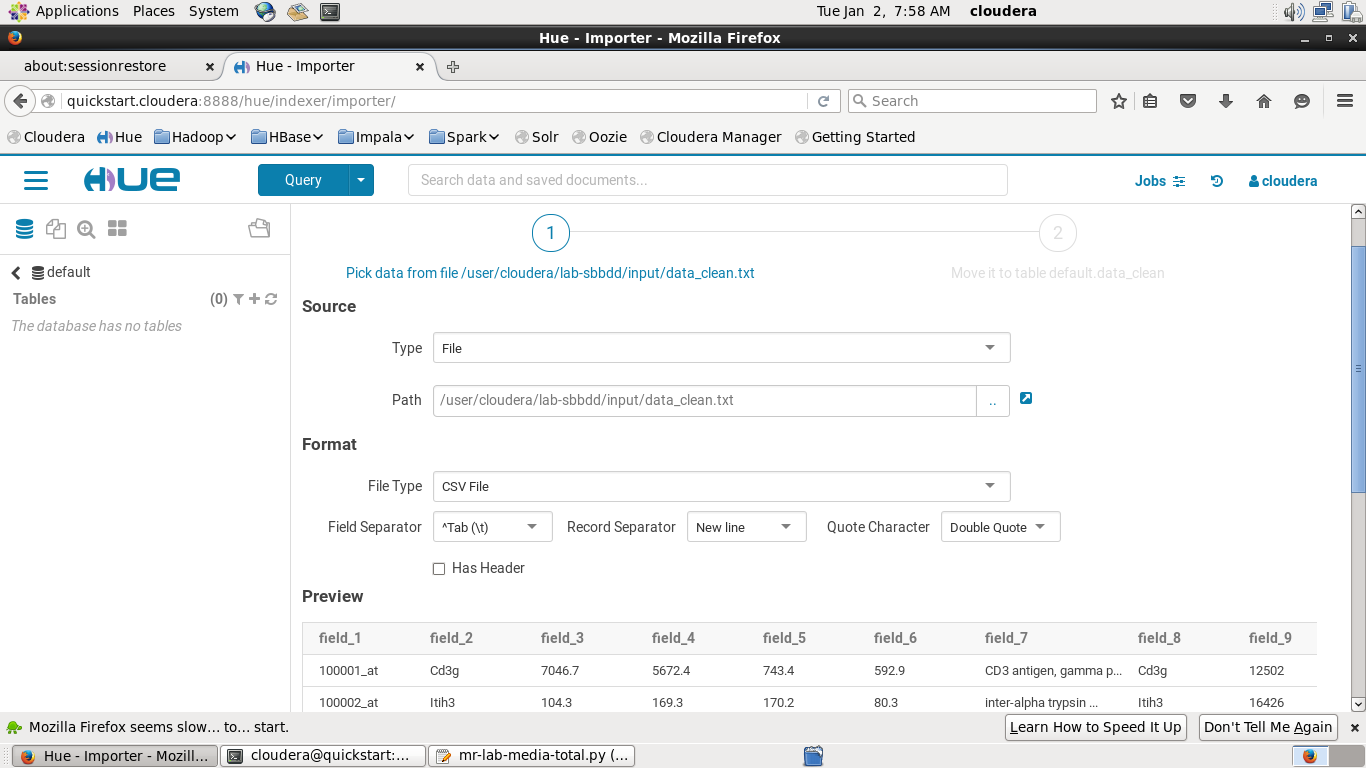
Mediante este interfaz he creado las dos tablas solicitadas (datacleantableinternal y datacleantableexternal).

Desde el interfaz solicitado, lo primero que he hecho es **seleccionar el fichero** que contiene la información que queremos almacenar en forma de tabla. Para ello he pulsado en el botón de selección de ficheros del cuadro de texto PATH.

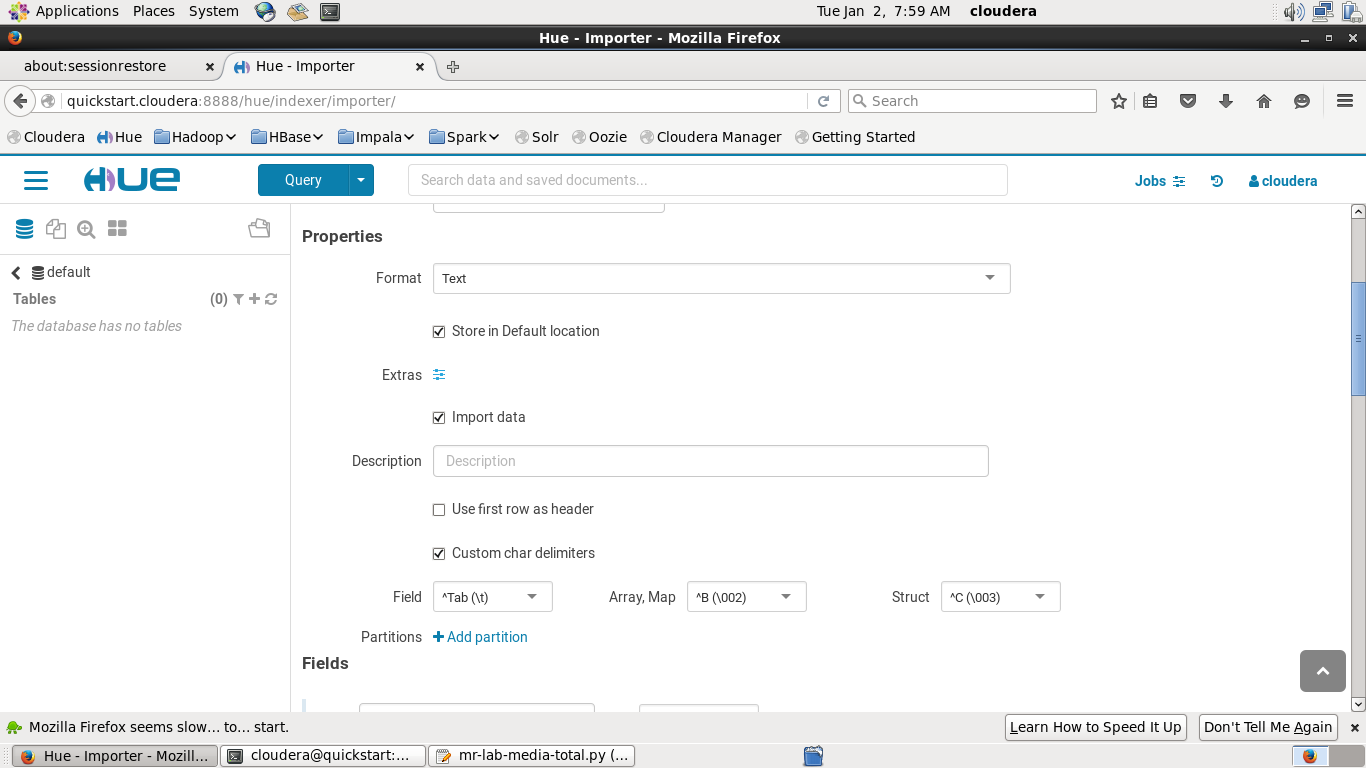


He accedido a una ventana, la cual me permite, no solo seleccionar el fichero, si no realizar la subida de ficheros locales a Hadoop, mediante el botón de upload.

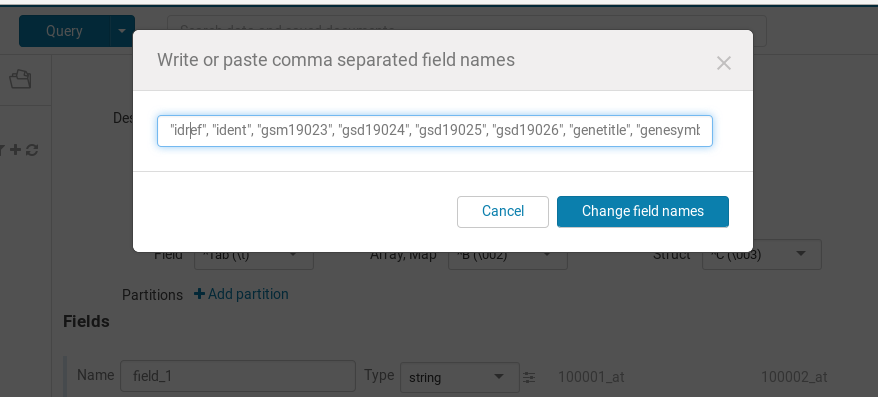
Una vez seleccionado el fichero, el sistema identifica los diferentes campos y realiza un “preview” de la información.



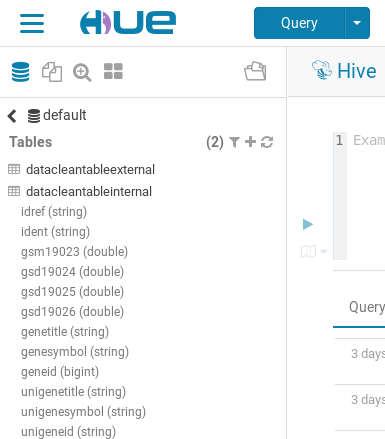
Continuando con el proceso de creación hacia abajo vemos un check denominado “Store in Default Location”. Este check es el que va a definir si queremos que la tabla sea interna (check activado) o no (check desactivado). En el caso de tabla interna, el check de Import Data nos permite establecer si queremos que la tabla se rellene con datos o no.



Para renombrar los campos de la tabla, si posicionamos el ratón encima y un poco a la derecha de fields, nos aparece un botón de edición, donde podemos incluir separados por , cómo queremos que se denominen los campos, ahorrándonos de esta manera su introducción manual por teclado.



Una vez hemos cumplimentado toda la información pulsamos el botón de SUBMIT con lo que se produce la creación de las tablas.



## Ejercicio 5. Creación de consultas HIVE

Una vez que disponemos de las tablas creadas podemos acceder al editor de consultas HIVE y elaborar y ejecutar las consultas solicitadas.

**Lo primero que se desea conseguir es implementar una consulta SQL que resuelva el apartado 2, devolviendo todas las columnas de la tabla.** Para ello elaboro la siguiente query SQL que utilizando una subconsulta me permite obtener, por un lado la media, y por otro, la lista detallada de campos del registro resultante.

select datacleantableexternal.\* from default.datacleantableexternal

where datacleantableexternal.average in

(select max(average) as max\_average from datacleantableexternal sc where gsm19023>=100 and gsm19023<=1000);

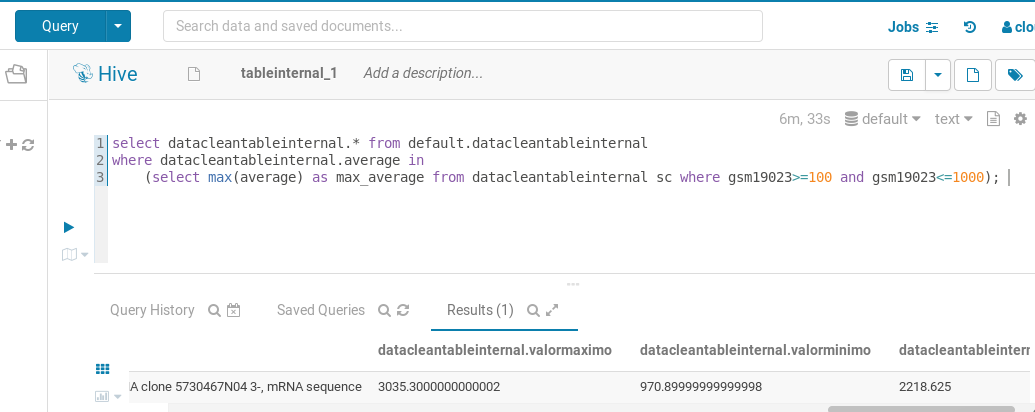
La consulta es similar tanto para la tabla interna como la externa, únicamente cambiando la definición de la misma

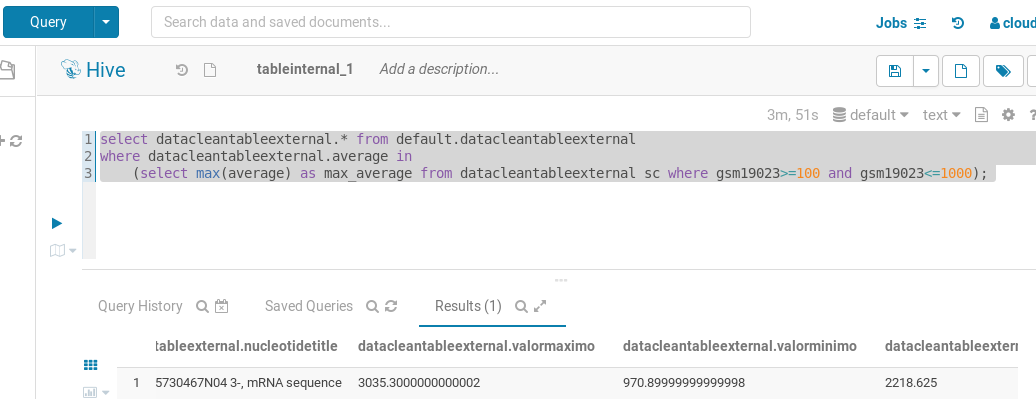
select datacleantableinternal.\* from default.datacleantableinternal

where datacleantableinternal.average in

(select max(average) as max\_average from datacleantableinternal sc where gsm19023>=100 and gsm19023<=1000);

A continuación, acompaño dos capturas de pantalla, donde se puede comprobar que el resultado, tanto en un caso como en el otro es similar, y además, similar a los datos obtenidos por el programa en python del apartado 2.





**Lo segundo que queremos hacer es elaborar la media del campo “gsd19025”.** En este caso la consulta es bastante sencilla.

select avg(gsd19025) from datacleantableinternal

select avg(gsd19025) from datacleantableexternal

A continuación, acompaño dos capturas de pantalla, donde se pueden comprobar que los resultados son similares, y similares al ofrecido por el programa en Python del apartado 3 (2458,206).

|  |  |
| --- | --- |
|  |  |

# Valoración personal sobre la práctica

|  |  |
| --- | --- |
| **Puntos fuertes**   * Buena preparación por parte del equipo docente. * Los videos han sido muy ilustrativos y me han servido bastante para prepararme la práctica. * La documentación aportada ha estado bien elaborada. * Buen “feedback” y respuesta en los foros. | **Puntos débiles**   * La utilización de mrjob resulta un poco difusa, y se asume un conocimiento de determinadas cuestiones de Python (yields) que de no tener experiencias previas te hacen dedicarle bastante tiempo. |