



M03-DC02 Cassandra

EJERCICIO EVALUABLE Cassandra

Programa: Máster Executive en Big Data, Cloud & Analytics

Periodo académico: 2019 – 2020

Autor/es: CARLOS ALFONSEL JAÉN

1. ENUNCIADO

Se requiere la entrega de un documento PDF con las soluciones y comentarios propuestos de cada ejercicio.

1. Descarga el documento de Moodle correspondiente a la solución del modelado de **KillrVideo**, planteado para intentar realizarlo en casa. ¿Existen diferencias con lo que tenías hecho?. Comenta las diferencias.
2. Ejecuta **Cassandra** en local, con Docker, en la nube o simplemente plantea las siguientes preguntas en un documento de texto. Si vas a desplegar Cassandra de alguna forma, ejecuta los comandos cql incluidos en el documento "killr_video_schema.cql" y después los comandos incluidos en "killr_video_insert_values.cql". Haz uso de la consola con el siguiente comando: cqlsh -file="full_file_path", donde "full_file_path" es la ruta completa a los dos archivos mencionados en tu local.
3. Lanza la consola de **cqlsh** y haz una consulta que indique los keyspaces que hay en Cassandra. ¿Aparece **killrvideo**? Si no aparece "killrvideo" como keyspace, algo ha salido mal y tendrás que repetir el paso dos. Activa "killrvideo" como keyspace para trabajar.
4. Ejecuta una query que devuelva todas las filas de la tabla users. Ejecuta una query que devuelva los valores de firstname y lastname de la tabla users para un userid cualquiera de los obtenidos en la consulta anterior. Repite lo mismo para la tabla videos con la columna tags para un videoid cualquiera.
5. Al haber generado un índice múltiple (y un clustering ordenando por fecha y videoid) en la tabla user_videos, podemos hacer queries muy interesantes. Ejecuta una query que devuelva name, videoid y added_date de la tabla user_videos de todos los vídeos que haya añadido el userid 9761d3d7-7fdb-4269-9988-6cf4e188678, ordenados por fecha de adición de forma descendiente. ¿Cuál es el último vídeo añadido?.
6. Busca los vídeos que haya añadido el mismo usuario de la query anterior después del 6 de junio de 2013 (introduce la fecha en formato string 'YYYY-MM-DD'). Busca los vídeos que haya subido el usuario entre el 15 de mayo de 2013 y el 1 de julio de 2013. Realiza la misma query pero ordenando resultados por fecha de adición.

7. Selecciona los campos correspondientes al rating (rating_counter y rating_total) de un vídeo cualquiera. Divide para obtener la media de ratings. ¿Qué media tiene?.

Selecciona el identificador de vídeo y tagged_date de la tabla videos_bytag cuando el campo tag sea igual a 'lol'.

8. ¿La consulta anterior se podría realizar si no hubiéramos metido la columna "tag" como índice?. Indica que opciones tendríamos si no hubiéramos introducido ese índice.

2. SOLUCIONES

En primer lugar, quiero comentar el proceso de instalación en local de **Apache Cassandra**, ya que debido a las dificultades inherentes me resultó imposible la realización de los ejercicios prácticos durante la clase.

Dada la imposibilidad de instalar **Apache Cassandra** en un entorno Windows puesto que la distribución compilada **DataStax** no está disponible, se ha optado por una instalación en una máquina virtual **Ubuntu 16.04** (<https://www.osboxes.org/ubuntu/>) para Oracle VM VirtualBox.

Dicha máquina virtual dispone de **Python 2.7** de forma nativa, pero no así del entorno de desarrollo Java 8, por lo que ha habido que proceder a su instalación. Siguiendo las indicaciones encontradas en <https://www.vultr.com/docs/how-to-install-apache-cassandra-3-11-x-on-ubuntu-16-04-lts>, se ha conseguido finalmente disponer del entorno para instalar y ejecutar **Apache Cassandra** (ver pantallazos adjuntos).

a) Instalar **OpenJDK JRE 8**:

```
osboxes@osboxes:~$ sudo apt install openjdk-8-jre -y
osboxes@osboxes:~$ java -version
```

```
openjdk version "1.8.0_242"
OpenJDK Runtime Environment (build 1.8.0_242-8u242-b08-
0ubuntu3~16.04-b08)
OpenJDK 64-Bit Server VM (build 25.242-b08, mixed mode)
```

Adicionalmente se ha creado la variable de entorno **\$JAVA_HOME**:

```
osboxes@osboxes:~$ echo "JAVA_HOME=$(readlink -f /usr/bin/java | sed "s:bin/java:::)" | sudo tee -a /etc/profile
osboxes@osboxes:~$ source /etc/profile
osboxes@osboxes:~$ echo $JAVA_HOME
```

b) Se comprueba la versión instalada de **Python**:

```
osboxes@osboxes:~$ python -V
```

Python 2.7.12

c) Se instala la última distribución estable de **Apache Cassandra**:

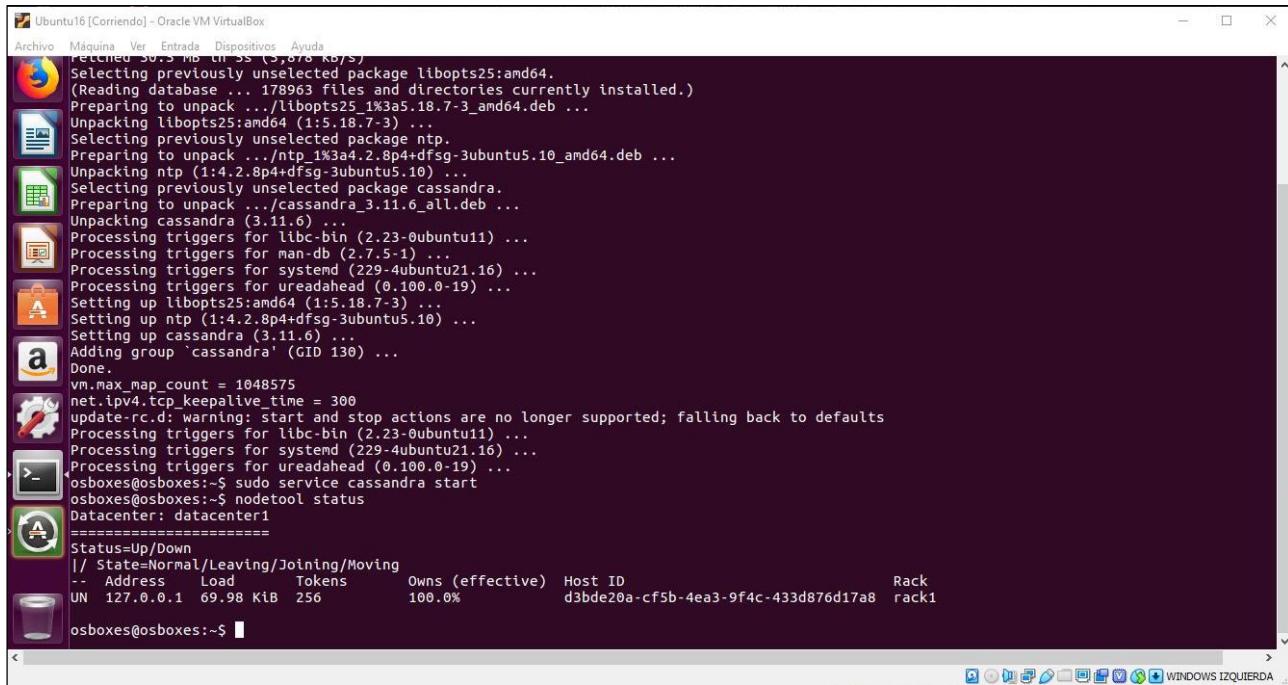
```
osboxes@osboxes:~$ echo "deb
http://www.apache.org/dist/cassandra/debian 311x main" | sudo
tee -a /etc/apt/sources.list.d/cassandra.sources.list
osboxes@osboxes:~$ curl
https://www.apache.org/dist/cassandra/KEYS | sudo apt-key add -
osboxes@osboxes:~$ sudo apt-get update
```

Se ejecuta el siguiente comando para añadir la clave pública de **Apache Cassandra A278B781FE4B2BDA** según se indica en el procedimiento seguido al haberse producido un error en la actualización:

```
osboxes@osboxes:~$ sudo apt-key adv --keyserver pool.sks-
keyservers.net --recv-key A278B781FE4B2BDA
osboxes@osboxes:~$ sudo apt-get update
```

```
osboxes@osboxes:~$ sudo apt-get install cassandra
osboxes@osboxes:~$ sudo service cassandra start
osboxes@osboxes:~$ nodetool status
```

Datacenter: datacenter1						
=====						
Status=Up/Down						
/ State=Normal/Leaving/Joining/Moving						
--	Address	Load	Tokens	Owns (effective)	Host ID	Rack
UN	127.0.0.1	69.98 KiB	256	100.0%	d3bde20a-cf5b-4ea3-9f4c-433d876d17a8	rack1



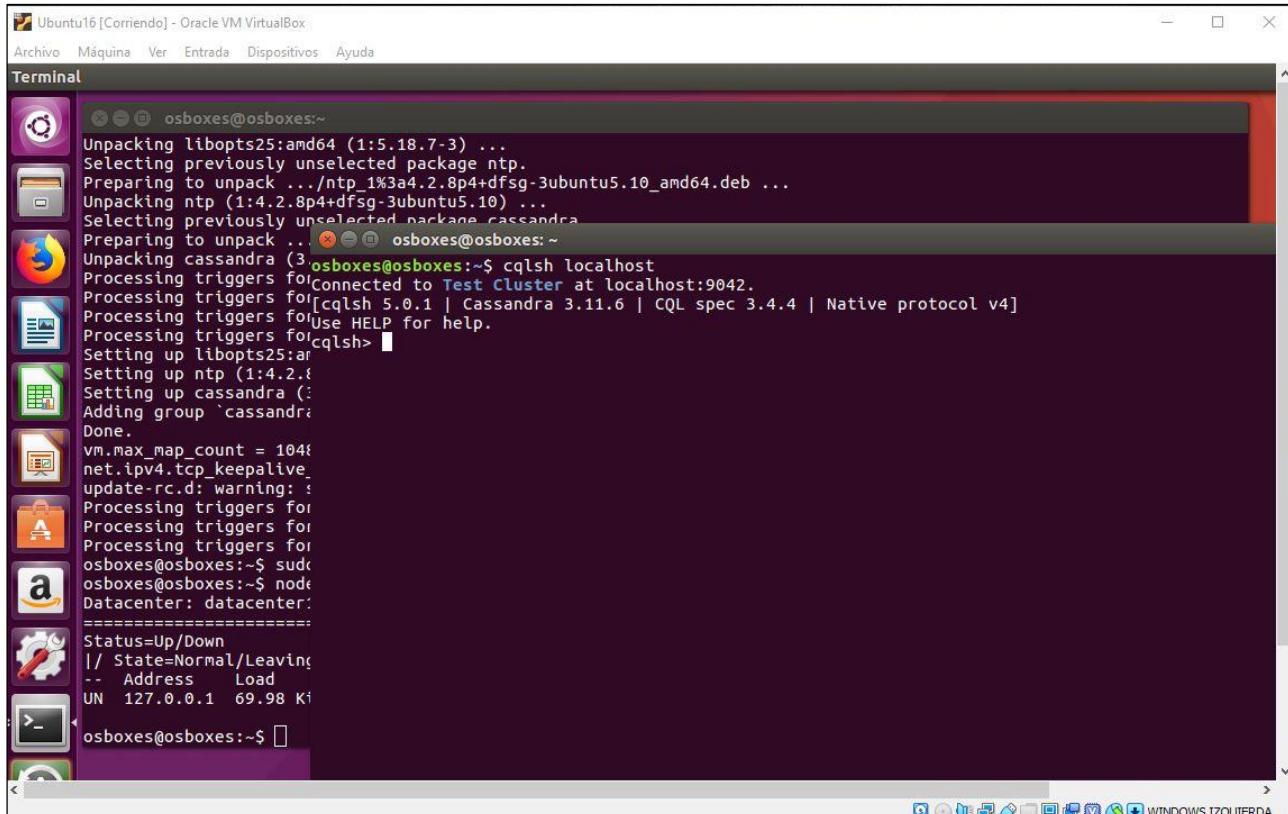
```

Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Selecting previously unselected package libopts25:amd64.
(Reading database ... 178963 files and directories currently installed.)
Preparing to unpack .../libopts25_1%3a5.18.7-3_amd64.deb ...
Unpacking libopts25:amd64 (1:5.18.7-3) ...
Selecting previously unselected package ntp.
Preparing to unpack .../ntp_1%3a4.2.8p4+dfsg-3ubuntu5.10_amd64.deb ...
Unpacking ntp (1:4.2.8p4+dfsg-3ubuntu5.10) ...
Selecting previously unselected package cassandra.
Preparing to unpack .../cassandra_3.11.6_all.deb ...
Unpacking cassandra (3.11.6) ...
Processing triggers for libc-bin (2.23-0ubuntu11) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for systemd (229-4ubuntu21.16) ...
Processing triggers for ureadahead (0.100.0-19) ...
Setting up libopts25:amd64 (1:5.18.7-3) ...
Setting up ntp (1:4.2.8p4+dfsg-3ubuntu5.10) ...
Setting up cassandra (3.11.6) ...
Adding group 'cassandra' (GID 130) ...
Done.
vm.max_map_count = 1048575
net.ipv4.tcp_keepalive_time = 300
update-rc.d: warning: start and stop actions are no longer supported; falling back to defaults
Processing triggers for libc-bin (2.23-0ubuntu11) ...
Processing triggers for systemd (229-4ubuntu21.16) ...
Processing triggers for ureadahead (0.100.0-19) ...
osboxes@osboxes:~$ sudo service cassandra start
osboxes@osboxes:~$ nodetool status
Datacenter: datacenter1
=====
Status=Up/Down
-- Address Load Tokens Owns (effective) Host ID Rack
UN 127.0.0.1 69.98 KiB 256 100.0% d3bde20a-cf5b-4ea3-9f4c-433d876d17a8 rack1
osboxes@osboxes:~$ 

```

Para interactuar con **Apache Cassandra** usamos la interfaz **cqlsh**:

```
osboxes@osboxes:~$ cqlsh localhost
```



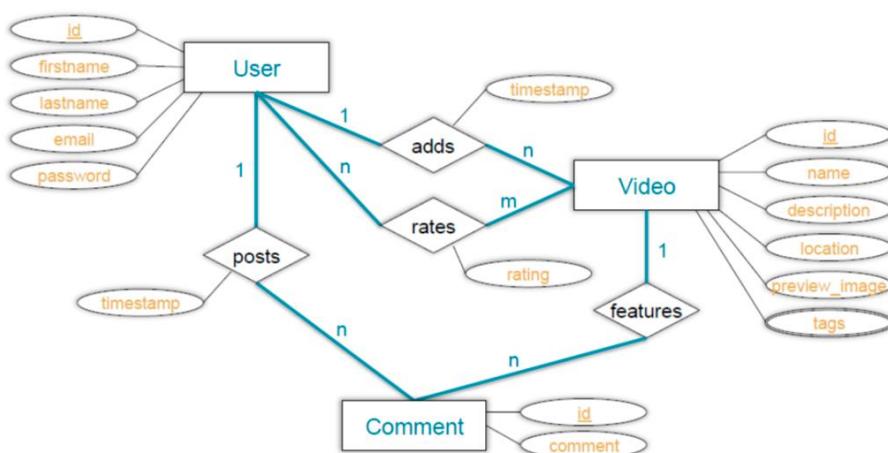
```

Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Terminal
osboxes@osboxes:~$ 
Selecting previously unselected package libopts25:amd64 (1:5.18.7-3) ...
Preparing to unpack .../libopts25_1%3a5.18.7-3_amd64.deb ...
Unpacking libopts25:amd64 (1:5.18.7-3) ...
Selecting previously unselected package ntp.
Preparing to unpack .../ntp_1%3a4.2.8p4+dfsg-3ubuntu5.10_amd64.deb ...
Unpacking ntp (1:4.2.8p4+dfsg-3ubuntu5.10) ...
Selecting previously unselected package cassandra.
Preparing to unpack .../cassandra_3.11.6_all.deb ...
Unpacking cassandra (3.11.6) ...
Processing triggers for libc-bin (2.23-0ubuntu11) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for systemd (229-4ubuntu21.16) ...
Processing triggers for ureadahead (0.100.0-19) ...
osboxes@osboxes:~$ cqlsh localhost
Processing triggers for [cqlsh 5.0.1 | Cassandra 3.11.6 | CQL spec 3.4.4 | Native protocol v4]
Processing triggers for Use HELP for help.
Processing triggers for [cqlsh>
Setting up libopts25:am
Setting up ntp (1:4.2.8
Setting up cassandra (:
Adding group 'cassandra
Done.
vm.max_map_count = 1048
net.ipv4.tcp_keepalive_
update-rc.d: warning: :
Processing triggers for
Processing triggers for
Processing triggers for
osboxes@osboxes:~$ sudo
osboxes@osboxes:~$ node
Datacenter: datacenter:
=====
Status=Up/Down
-- State=Normal/Leaving
-- Address Load
UN 127.0.0.1 69.98 Ki
osboxes@osboxes:~$ 

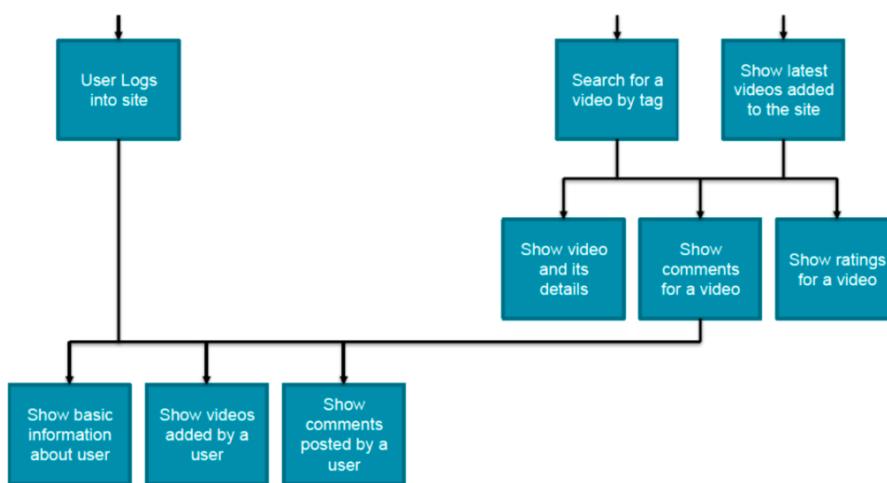
```

2.1. DESCARGA EL DOCUMENTO DE MOODLE CORRESPONDIENTE A LA SOLUCIÓN DEL MODELADO DE KillrVideo, PLANTEADO PARA INTENTAR REALIZARLO EN CASA. ¿EXISTEN DIFERENCIAS CON LO QUE TENÍAS HECHO?. COMENTA LAS DIFERENCIAS.

Una vez ya instalado **Cassandra**, se va a trabajar con el modelado de datos y su implementación a través de CQL. Para ello, se trabajará con la base de datos **KillrVideo**. **KillrVideo** es un site para compartir vídeos a través de Internet parecido a Youtube. Los usuarios pueden ver vídeos, puntuarlos, poner comentarios, etc. o buscar vídeo por una etiqueta o tag.



Para poder modelar con **Cassandra** es necesario conocer con anterioridad los workflows de las aplicaciones y cómo los usuarios accederán a los datos, ya que es necesario conocer a priori las consultas.



Una vez definidos los workflows es necesario traducirlos a las consultas que se deben realizar:

- Buscar usuario por email.
- Buscar usuario por id.
- Buscar comentarios de un vídeo (primero los últimos).
- Buscar comentarios de un usuario (primero los últimos).
- Buscar ratings por vídeo.
- Buscar vídeo por tag.
- Buscar vídeos por fecha (primero los últimos).
- Buscar vídeo por id.
- Buscar vídeos por usuario (primero los últimos).

Para los workflows establecidos y sus correspondientes consultas, establece cuál sería el modelado de datos que mejor se ajuste a las características de la aplicación **KillrVideo**.

2.2. EJECUTA Cassandra EN LOCAL. EJECUTA LOS COMANDOS CQL

INCLUIDOS EN EL DOCUMENTO "killr_video_schema.cql" Y DESPUÉS LOS COMANDOS INCLUIDOS EN "killr_video_insert_values.cql". HAZ USO DE LA CONSOLA CON EL SIGUIENTE COMANDO: cqlsh --file="full_file_path", DONDE "full_file_path" ES LA RUTA COMPLETA A LOS DOS ARCHIVOS MENCIONADOS EN TU LOCAL.

```
osboxes@osboxes:~$ cqlsh --file="/home/osboxes/cassandra/killr_video_schema.cql"
```

Warnings :

```
SASI indexes are experimental and are not recommended for production use.
```

Warnings :

```
Materialized views are experimental and are not recommended for production use.
```

```
osboxes@osboxes:~$ cqlsh --file="/home/osboxes/cassandra/killr_video_insert_values.cql"
```

2.3. LANZA LA CONSOLA DE cqlsh Y HAZ UNA CONSULTA QUE INDIQUE LOS KEYSPACES QUE HAY EN Cassandra. ¿APARECE KillrVideo?. ACTIVA KillrVideo COMO KEYSPACE PARA TRABAJAR.



```
osboxes@osboxes:~$ cqlsh localhost
Connected to Test Cluster at localhost:9042.
[cqlsh 5.0.1 | Cassandra 3.11.6 | CQL spec 3.4.4 | Native
protocol v4]
Use HELP for help.
```

```
cqlsh> SELECT * FROM system_schema.keyspaces;
```

```
cqlsh> use killrvideo;
```

```
cqlsh:killrvideo>
```

The screenshot shows a terminal window titled "Ubuntu16 [Corriendo] - Oracle VM VirtualBox". The window displays the following text:

```
osboxes@osboxes:~$ cqlsh --file="/home/osboxes/cassandra/killr_video_insert_values.cql"
osboxes@osboxes:~$ cqlsh localhost
Connected to Test Cluster at localhost:9042.
[cqlsh 5.0.1 | Cassandra 3.11.6 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
cqlsh> SELECT * FROM system_schema.keyspaces;
```

keyspace_name	durable_writes	replication
killrvideo	True	{'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
system_auth	True	{'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '1'}
system_schema	True	{'class': 'org.apache.cassandra.locator.LocalStrategy'}
system_distributed	True	{'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
system	True	{'class': 'org.apache.cassandra.locator.LocalStrategy'}
system_traces	True	{'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '2'}

(6 rows)

```
cqlsh> use killrvideo;
cqlsh:killrvideo> SELECT * FROM system_schema.keyspaces;
```

keyspace_name	durable_writes	replication
killrvideo	True	{'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
system_auth	True	{'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '1'}
system_schema	True	{'class': 'org.apache.cassandra.locator.LocalStrategy'}
system_distributed	True	{'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
system	True	{'class': 'org.apache.cassandra.locator.LocalStrategy'}
system_traces	True	{'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '2'}

(6 rows)

```
cqlsh:killrvideo>
```

2.4. EJECUTA UNA QUERY QUE DEVUELVA TODAS LAS FILAS DE LA TABLA users. EJECUTA UNA QUERY QUE DEVUELVA LOS VALORES DE firstname Y lastname DE LA TABLA users PARA UN userid CUALQUIERA DE LOS OBTENIDOS EN LA CONSULTA ANTERIOR. REPITE LO MISMO PARA LA TABLA videos CON LA COLUMNA TAGS PARA UN videoid CUALQUIERA.

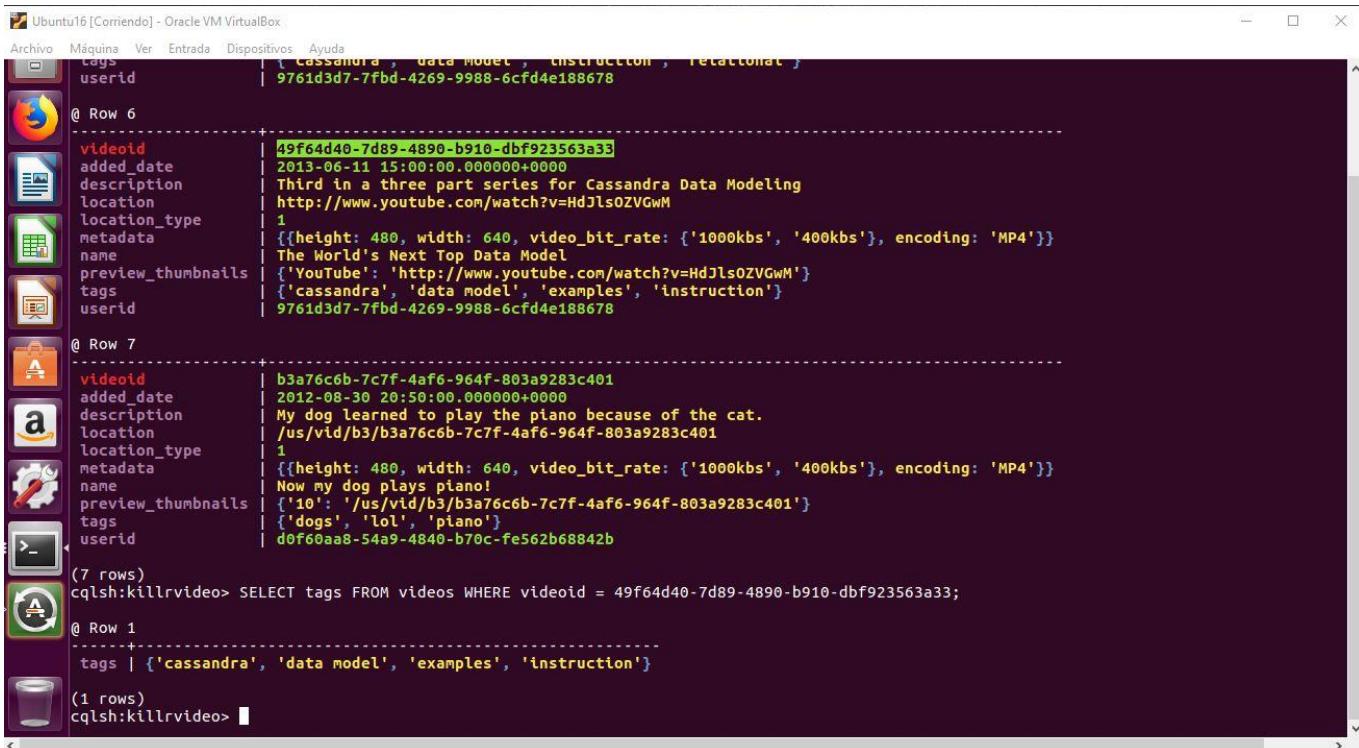
```
cqlsh:killrvideo> SELECT * FROM users;
```

```
Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Connected to test_cluster at localhost:9042.
[cqlsh 5.0.1 | Cassandra 3.1.6 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
cqlsh> SELECT * FROM system_schema.keyspaces;
  keyspace_name | durable_writes | replication
  killrvideo     |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
  system_auth    |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '1'}
  system_schema   |      True      | {'class': 'org.apache.cassandra.locator.LocalStrategy'}
  system_distributed | True | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
  system          |      True      | {'class': 'org.apache.cassandra.locator.LocalStrategy'}
  system_traces   |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '2'}
(6 rows)
cqlsh> use killrvideo;
cqlsh:killrvideo> SELECT * FROM system_schema.keyspaces;
  keyspace_name | durable_writes | replication
  killrvideo     |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
  system_auth    |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '1'}
  system_schema   |      True      | {'class': 'org.apache.cassandra.locator.LocalStrategy'}
  system_distributed | True | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
  system          |      True      | {'class': 'org.apache.cassandra.locator.LocalStrategy'}
  system_traces   |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '2'}
(6 rows)
cqlsh:killrvideo> SELECT * FROM users;
  userid          | created_date           | email            | firstname | lastname
522b1fe2-2e36-4cef-a667-cd4237d08b89 | 2011-06-20 17:50:00.000000+0000 | cdate@relational.com | Chris     | Date
d0f60aa8-54a9-4840-b70c-fe562b68842b | 2011-06-01 12:00:00.000000+0000 | tcodd@relational.com | Ted       | Codd
9761d3d7-7fbdb-4269-9988-6cf4e188678 | 2011-06-20 17:50:00.000000+0000 | patrick@datastax.com | Patrick   | McFadin
(3 rows)
cqlsh:killrvideo>
```

```
cqlsh:killrvideo> SELECT firstname, lastname FROM users WHERE
userid = 9761d3d7-7fbdb-4269-9988-6cf4e188678;
```

```
Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
  killrvideo     |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
  system_auth    |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '1'}
  system_schema   |      True      | {'class': 'org.apache.cassandra.locator.LocalStrategy'}
  system_distributed | True | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
  system          |      True      | {'class': 'org.apache.cassandra.locator.LocalStrategy'}
  system_traces   |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '2'}
(6 rows)
cqlsh> use killrvideo;
cqlsh:killrvideo> SELECT * FROM system_schema.keyspaces;
  keyspace_name | durable_writes | replication
  killrvideo     |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
  system_auth    |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '1'}
  system_schema   |      True      | {'class': 'org.apache.cassandra.locator.LocalStrategy'}
  system_distributed | True | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '3'}
  system          |      True      | {'class': 'org.apache.cassandra.locator.LocalStrategy'}
  system_traces   |      True      | {'class': 'org.apache.cassandra.locator.SimpleStrategy', 'replication_factor': '2'}
(6 rows)
cqlsh:killrvideo> SELECT * FROM users;
  userid          | created_date           | email            | firstname | lastname
522b1fe2-2e36-4cef-a667-cd4237d08b89 | 2011-06-20 17:50:00.000000+0000 | cdate@relational.com | Chris     | Date
d0f60aa8-54a9-4840-b70c-fe562b68842b | 2011-06-01 12:00:00.000000+0000 | tcodd@relational.com | Ted       | Codd
9761d3d7-7fbdb-4269-9988-6cf4e188678 | 2011-06-20 17:50:00.000000+0000 | patrick@datastax.com | Patrick   | McFadin
(3 rows)
cqlsh:killrvideo> SELECT firstname, lastname FROM users WHERE userid = 9761d3d7-7fbdb-4269-9988-6cf4e188678;
  firstname | lastname
  Patrick   | McFadin
(1 rows)
cqlsh:killrvideo>
```

```
cqlsh:killrvideo> SELECT tags FROM videos WHERE videoid =
49f64d40-7d89-4890-b910-dbf923563a33;
```



	videoid	added_date	description	location	location_type	metadata	name	preview_thumbnails	tags	userid
@ Row 6	49f64d40-7d89-4890-b910-dbf923563a33	2013-06-11 15:00:00.000000+0000	Third in a three part series for Cassandra Data Modeling	http://www.youtube.com/watch?v=HdJlsOZVGwM	1	{}{height: 480, width: 640, video_bit_rate: ['1000kbs', '400kbs'], encoding: 'MP4'}}	The World's Next Top Data Model	{'YouTube': 'http://www.youtube.com/watch?v=HdJlsOZVGwM'}	{'cassandra', 'data model', 'examples', 'instruction'}	9761d3d7-7fb7-4269-9988-6cf4e188678
@ Row 7	b3a76c6b-7c7f-4af6-964f-803a9283c401	2012-08-30 20:50:00.000000+0000	My dog learned to play the piano because of the cat.	/us/vld/b3/b3a76c6b-7c7f-4af6-964f-803a9283c401	1	{}{height: 480, width: 640, video_bit_rate: ['1000kbs', '400kbs'], encoding: 'MP4'}}	Now my dog plays piano!	{'10': '/us/vld/b3/b3a76c6b-7c7f-4af6-964f-803a9283c401'}	{'dogs', 'lol', 'piano'}	d0f60aa8-54a9-4840-b70c-fe562b68842b

(7 rows)

```
cqlsh:killrvideo> SELECT tags FROM videos WHERE videoid = 49f64d40-7d89-4890-b910-dbf923563a33;
```

@ Row 1

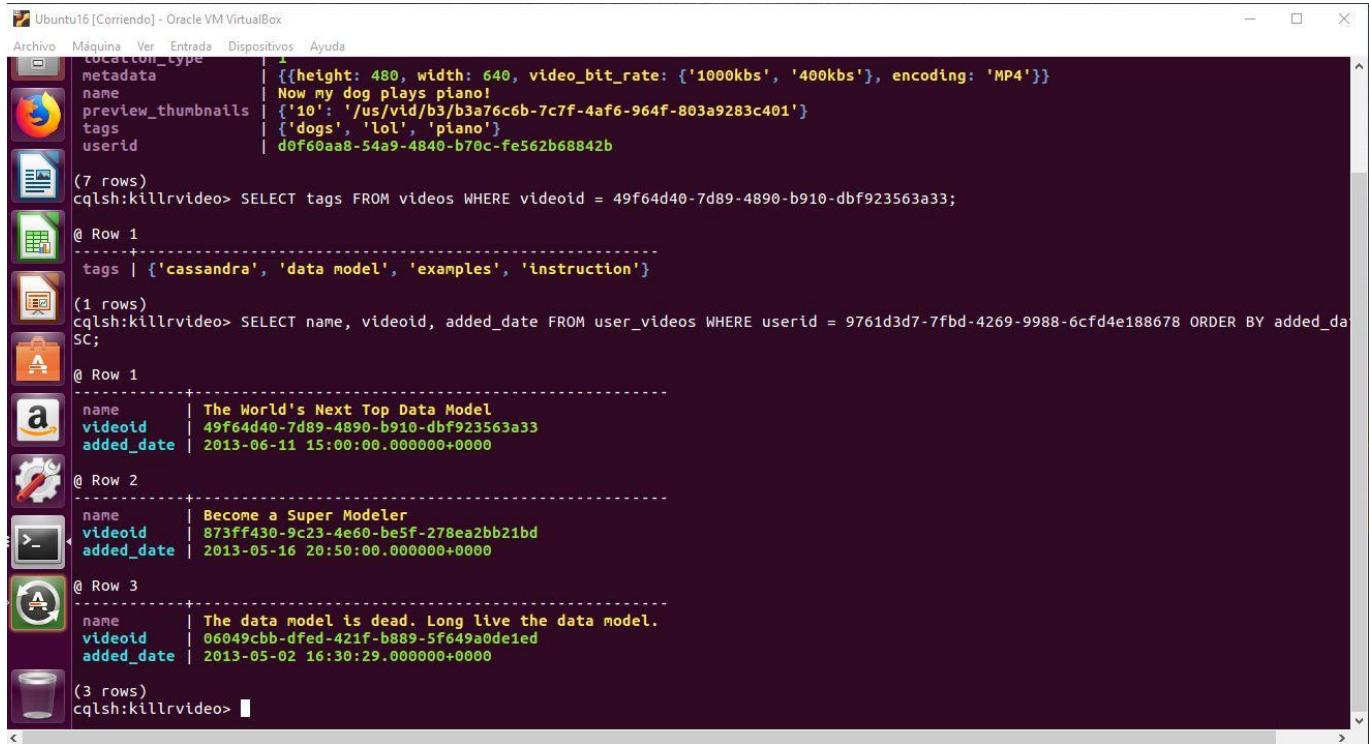
tags
{'cassandra', 'data model', 'examples', 'instruction'}

(1 rows)

```
cqlsh:killrvideo>
```

2.5. AL HABER GENERADO UN ÍNDICE MÚLTIPLE (Y UN CLUSTERING ORDENANDO POR FECHA Y videoid) EN LA TABLA user_videos, PODEMOS HACER QUERIES MUY INTERESANTES. EJECUTA UNA QUERY QUE DEVUELVA name, videoid Y added_date DE LA TABLA user_videos DE TODOS LOS VÍDEOS QUE HAYA AÑADIDO EL userid **9761d3d7-7fb7-4269-9988-6cf4e188678, ORDENADOS POR FECHA DE ADICIÓN DE FORMA DESCENDIENTE. ¿CUÁL ES EL ÚLTIMO VÍDEO AÑADIDO?.**

```
cqlsh:killrvideo> SELECT name, videoid, added_date FROM
user_videos WHERE userid = 9761d3d7-7fb7-4269-9988-6cf4e188678
ORDER BY added_date DESC;
```



```

Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
tucartoon_type | 1
metadata | {{height: 480, width: 640, video_bit_rate: ['1000kbs', '400kbs'], encoding: 'MP4'}}}
name | Now my dog plays piano!
preview_thumbnails | ['10': '/us/vid/b3/b3a76c6b-7c7f-4af6-964f-803a9283c401']
tags | ['dogs', 'lol', 'piano']
userid | d0f60aa8-54a9-4840-b70c-fe562b68842b
(7 rows)
cqlsh:killrvideo> SELECT tags FROM videos WHERE videoid = 49f64d40-7d89-4890-b910-dbf923563a33;
@ Row 1
-----+
tags | ['cassandra', 'data model', 'examples', 'instruction']
(1 rows)
cqlsh:killrvideo> SELECT name, videoid, added_date FROM user_videos WHERE userid = 9761d3d7-7fdbd-4269-9988-6cf4e188678 ORDER BY added_da
SC;
@ Row 1
-----+
name | The World's Next Top Data Model
videoid | 49f64d40-7d89-4890-b910-dbf923563a33
added_date | 2013-06-11 15:00:00.000000+0000
@ Row 2
-----+
name | Become a Super Modeler
videoid | 873ff430-9c23-4e60-be5f-278ea2bb21bd
added_date | 2013-05-16 20:50:00.000000+0000
@ Row 3
-----+
name | The data model is dead. Long live the data model.
videoid | 06049ccb-dfed-421f-b889-5f649a0de1ed
added_date | 2013-05-02 16:30:29.000000+0000
(3 rows)
cqlsh:killrvideo> ■

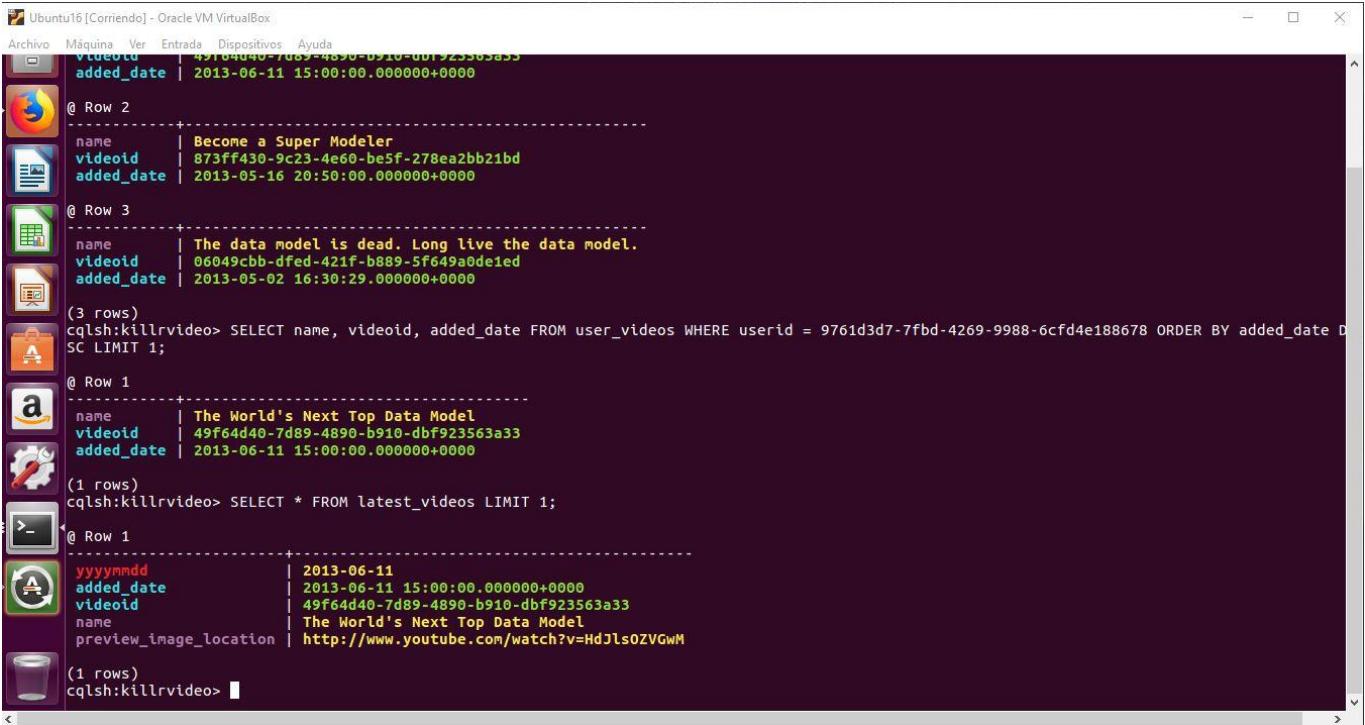
```

Para ver el último vídeo añadido del usuario indicado se puede añadir LIMIT 1:

```
cqlsh:killrvideo> SELECT name, videoid, added_date FROM
user_videos WHERE userid = 9761d3d7-7fdbd-4269-9988-6cf4e188678
ORDER BY added_date DESC LIMIT 1;
```

Señalar que el resultado es el mismo si se hace la siguiente consulta (ver captura de pantalla a continuación) a la tabla **latest_videos**:

```
cqlsh:killrvideo> SELECT * FROM latest_videos LIMIT 1;
```



```

Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Videos | 49f64d40-7d89-4890-b910-dbf923563a33
added_date | 2013-06-11 15:00:00.000000+0000

@ Row 2
-----+
name | Become a Super Modeler
videoid | 873ff430-9c23-4e60-be5f-278ea2bb21bd
added_date | 2013-05-16 20:50:00.000000+0000

@ Row 3
-----+
name | The data model is dead. Long live the data model.
videoid | 06049cbb-dfed-421f-b889-5f649a0de1ed
added_date | 2013-05-02 16:30:29.000000+0000

(3 rows)
cqlsh:killrvideo> SELECT name, videoid, added_date FROM user_videos WHERE userid = 9761d3d7-7fdbd-4269-9988-6cf4e188678 ORDER BY added_date DESC LIMIT 1;

@ Row 1
-----+
name | The World's Next Top Data Model
videoid | 49f64d40-7d89-4890-b910-dbf923563a33
added_date | 2013-06-11 15:00:00.000000+0000

(1 rows)
cqlsh:killrvideo> SELECT * FROM latest_videos LIMIT 1;

@ Row 1
-----+
yyyy-mm-dd | 2013-06-11
added_date | 2013-06-11 15:00:00.000000+0000
videoid | 49f64d40-7d89-4890-b910-dbf923563a33
name | The World's Next Top Data Model
preview_image_location | http://www.youtube.com/watch?v=HdJlsOZVGwM

(1 rows)
cqlsh:killrvideo>

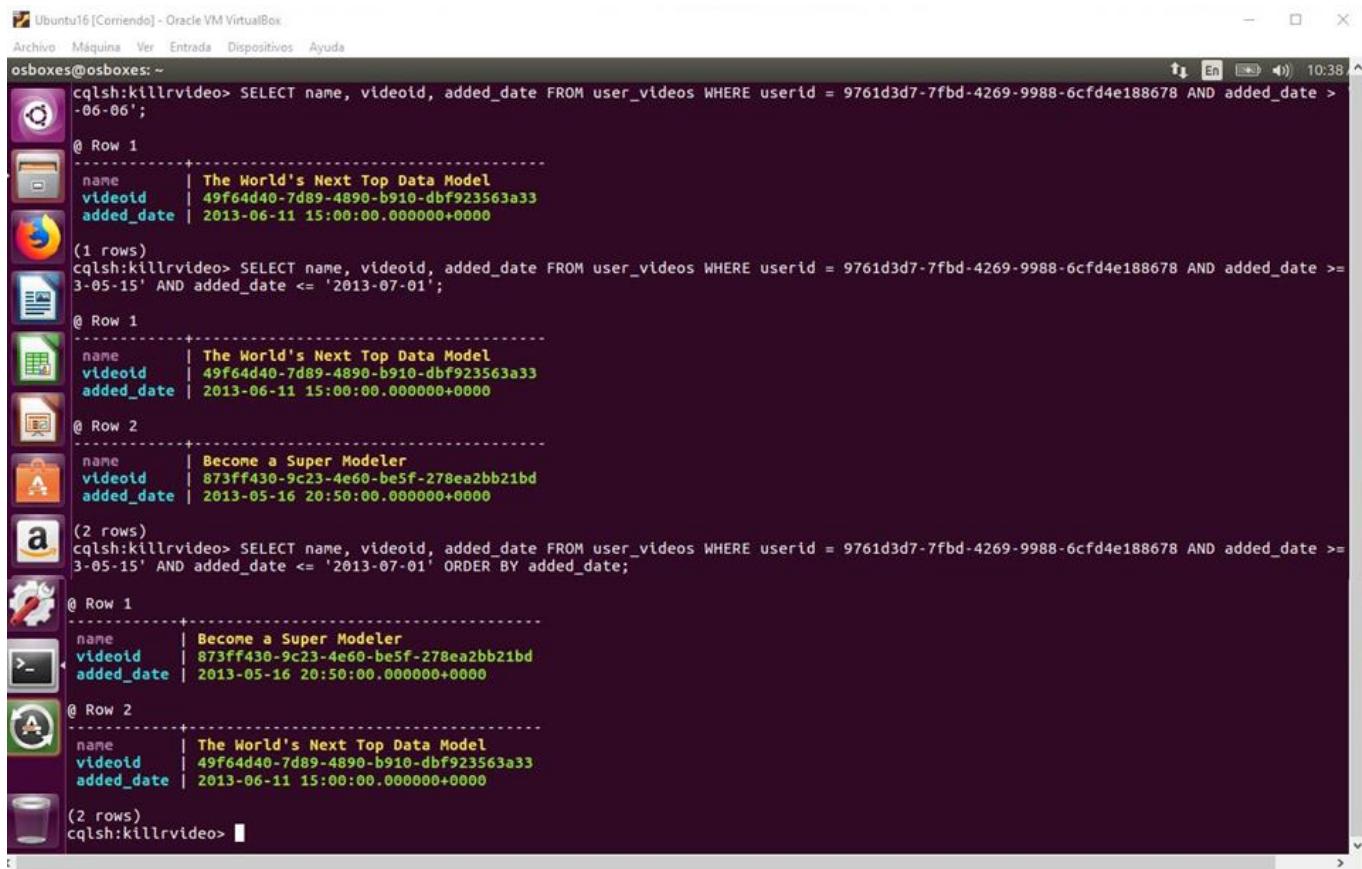
```

2.6. BUSCA LOS VÍDEOS QUE HAYA AÑADIDO EL MISMO USUARIO DE LA QUERY ANTERIOR DESPUÉS DEL 6 DE JUNIO DE 2013 (INTRODUCE LA FECHA EN FORMATO STRING 'YYYY-MM-DD'). BUSCA LOS VÍDEOS QUE HAYA SUBIDO EL USUARIO ENTRE EL 15 DE MAYO DE 2013 Y EL 1 DE JULIO DE 2013. REALIZA LA MISMA QUERY PERO ORDENANDO RESULTADOS POR FECHA DE ADICIÓN.

```
cqlsh:killrvideo> SELECT name, videoid, added_date FROM
user_videos WHERE userid = 9761d3d7-7fdbd-4269-9988-6cf4e188678
AND added_date > '2013-06-06';
```

```
cqlsh:killrvideo> SELECT name, videoid, added_date FROM
user_videos WHERE userid = 9761d3d7-7fdbd-4269-9988-6cf4e188678
AND added_date >= '2013-05-15' AND added_date >= '2013-07-01';
```

```
cqlsh:killrvideo> SELECT name, videoid, added_date FROM
user_videos WHERE userid = 9761d3d7-7fdbd-4269-9988-6cf4e188678
AND added_date >= '2013-05-15' AND added_date >= '2013-07-01'
ORDER BY added_date;
```



```

Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayudas
osboxes@osboxes: ~
cqlsh:killrvideo> SELECT name, videoid, added_date FROM user_videos WHERE userid = 9761d3d7-7fdb-4269-9988-6cf4e188678 AND added_date >= '2013-06-06';
@ Row 1
+-----+
| name      | The World's Next Top Data Model
| videoid   | 49f64d40-7d89-4890-b910-db923563a33
| added_date| 2013-06-11 15:00:00.000000+0000
+-----+
(1 rows)
cqlsh:killrvideo> SELECT name, videoid, added_date FROM user_videos WHERE userid = 9761d3d7-7fdb-4269-9988-6cf4e188678 AND added_date >= '2013-05-15' AND added_date <= '2013-07-01';
@ Row 1
+-----+
| name      | The World's Next Top Data Model
| videoid   | 49f64d40-7d89-4890-b910-db923563a33
| added_date| 2013-06-11 15:00:00.000000+0000
+-----+
@ Row 2
+-----+
| name      | Become a Super Modeler
| videoid   | 873ff430-9c23-4e60-be5f-278ea2bb21bd
| added_date| 2013-05-16 20:50:00.000000+0000
+-----+
(2 rows)
cqlsh:killrvideo> SELECT name, videoid, added_date FROM user_videos WHERE userid = 9761d3d7-7fdb-4269-9988-6cf4e188678 AND added_date >= '2013-05-15' AND added_date <= '2013-07-01' ORDER BY added_date;
@ Row 1
+-----+
| name      | Become a Super Modeler
| videoid   | 873ff430-9c23-4e60-be5f-278ea2bb21bd
| added_date| 2013-05-16 20:50:00.000000+0000
+-----+
@ Row 2
+-----+
| name      | The World's Next Top Data Model
| videoid   | 49f64d40-7d89-4890-b910-db923563a33
| added_date| 2013-06-11 15:00:00.000000+0000
+-----+
(2 rows)
cqlsh:killrvideo>
    
```

2.7. SELECCIONA LOS CAMPOS CORRESPONDIENTES AL RATING (rating_counter Y rating_total) DE UN VÍDEO CUALQUIERA. DIVIDE PARA OBTENER LA MEDIA DE RATINGS. ¿QUÉ MEDIA TIENE?. SELECCIONA EL IDENTIFICADOR DE VÍDEO Y tagged_date DE LA TABLA videos_by_tag CUANDO EL CAMPO tag SEA IGUAL A 'lol'.

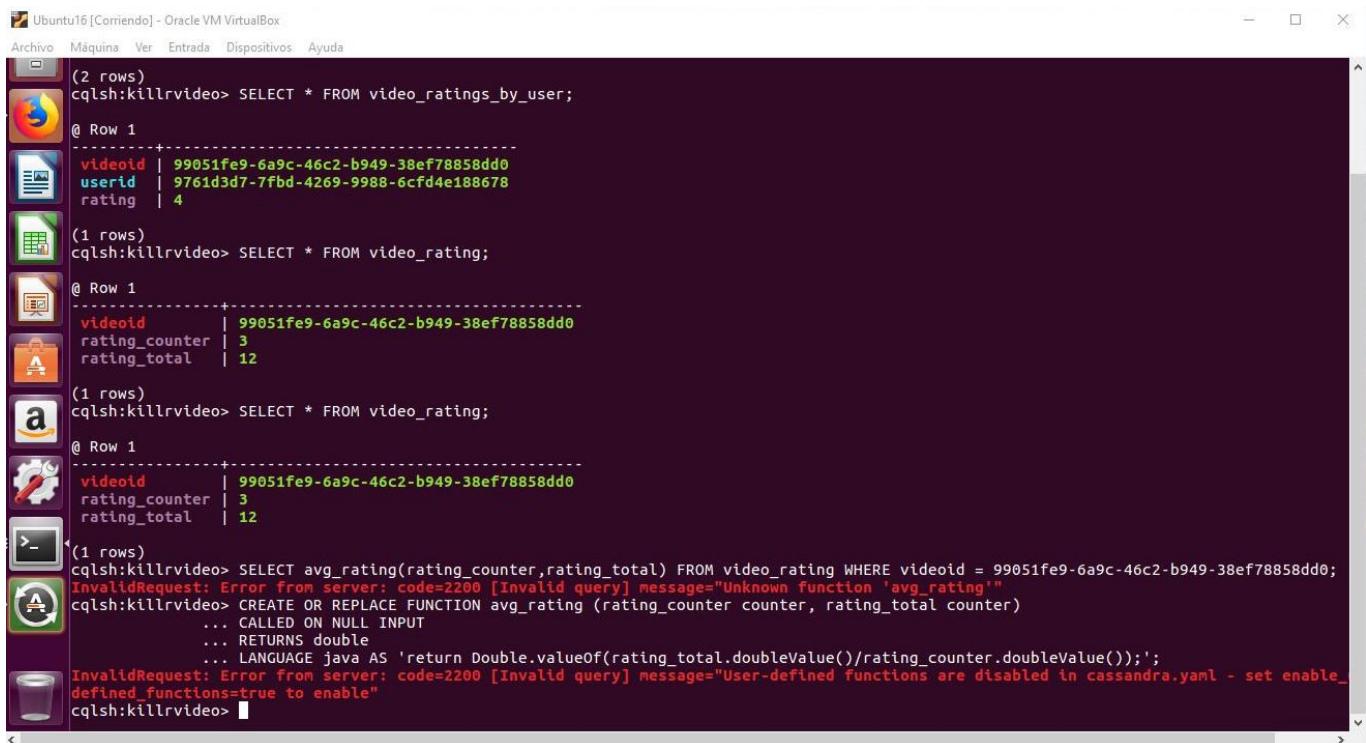
Para realizar esta consulta, de acuerdo al enlace con la información necesaria que se encuentra en la documentación aportada (killrvideo.pdf), es necesario definir primero una función de usuario llamada **avg_rating**:

```

// CQL 3.3 adds user defined functions
// This function takes the counters in the video rating table and averages them
// by dividing rating total by rating counter.
CREATE OR REPLACE FUNCTION avg_rating (rating_counter counter, rating_total counter)
CALLED ON NULL INPUT
RETURNS double
LANGUAGE java AS 'return Double.valueOf(rating_total.doubleValue()/rating_counter.doubleValue());';
    
```

Al intentar crear esta función en **Apache Cassandra**, me sale el siguiente mensaje de error:

```
InvalidRequest: Error from server: code=2200 [Invalid query]
message="User-defined functions are disabled in cassandra.yaml -
set enable_user_defined_functions=true to enable"
```



The screenshot shows a terminal window titled "Ubuntu16 [Corriendo] - Oracle VM VirtualBox". The terminal displays several Cassandra queries and their results:

- SELECT * FROM video_ratings_by_user; (2 rows)
- @ Row 1 (1 row)

videoid	99051fe9-6a9c-46c2-b949-38ef78858dd0
userid	9761d3d7-7fdb-4269-9988-6cf4e188678
rating	4
- SELECT * FROM video_rating; (1 rows)
- @ Row 1 (1 row)

videoid	99051fe9-6a9c-46c2-b949-38ef78858dd0
rating_counter	3
rating_total	12
- SELECT * FROM video_rating; (1 rows)
- @ Row 1 (1 row)

videoid	99051fe9-6a9c-46c2-b949-38ef78858dd0
rating_counter	3
rating_total	12
- SELECT avg_rating(rating_counter,rating_total) FROM video_rating WHERE videoid = 99051fe9-6a9c-46c2-b949-38ef78858dd0;
- InvalidRequest: Error from server: code=2200 [Invalid query] message="Unknown function 'avg_rating'"
- CREATE OR REPLACE FUNCTION avg_rating (rating_counter counter, rating_total counter)
 ... CALLED ON NULL INPUT
 ... RETURNS double
 ... LANGUAGE java AS 'return Double.valueOf(rating_total.doubleValue()/rating_counter.doubleValue());';
- InvalidRequest: Error from server: code=2200 [Invalid query] message="User-defined functions are disabled in cassandra.yaml - set enable_user_defined_functions=true to enable"

Por lo tanto, hay que poner a true ese parámetro (**enable_user_defined_functions**) en el fichero **cassandra.yaml** que se encuentra en la ruta **/etc/cassandra**.

Para que los cambios tengan efecto, se debe parar primero la instancia de **Apache Cassandra** y luego arrancarla de nuevo:

```
osboxes@osboxes:/etc/cassandra$ sudo service cassandra stop
osboxes@osboxes:/etc/cassandra$ sudo service cassandra start
osboxes@osboxes:/etc/cassandra$ cqlsh localhost
Connected to Test Cluster at localhost:9042.
[cqlsh 5.0.1 | Cassandra 3.11.6 | CQL spec 3.4.4 | Native
protocol v4]
Use HELP for help.
cqlsh> use killrvideo;
```

A continuación, ya podemos crear la función de usuario que calcula la media:



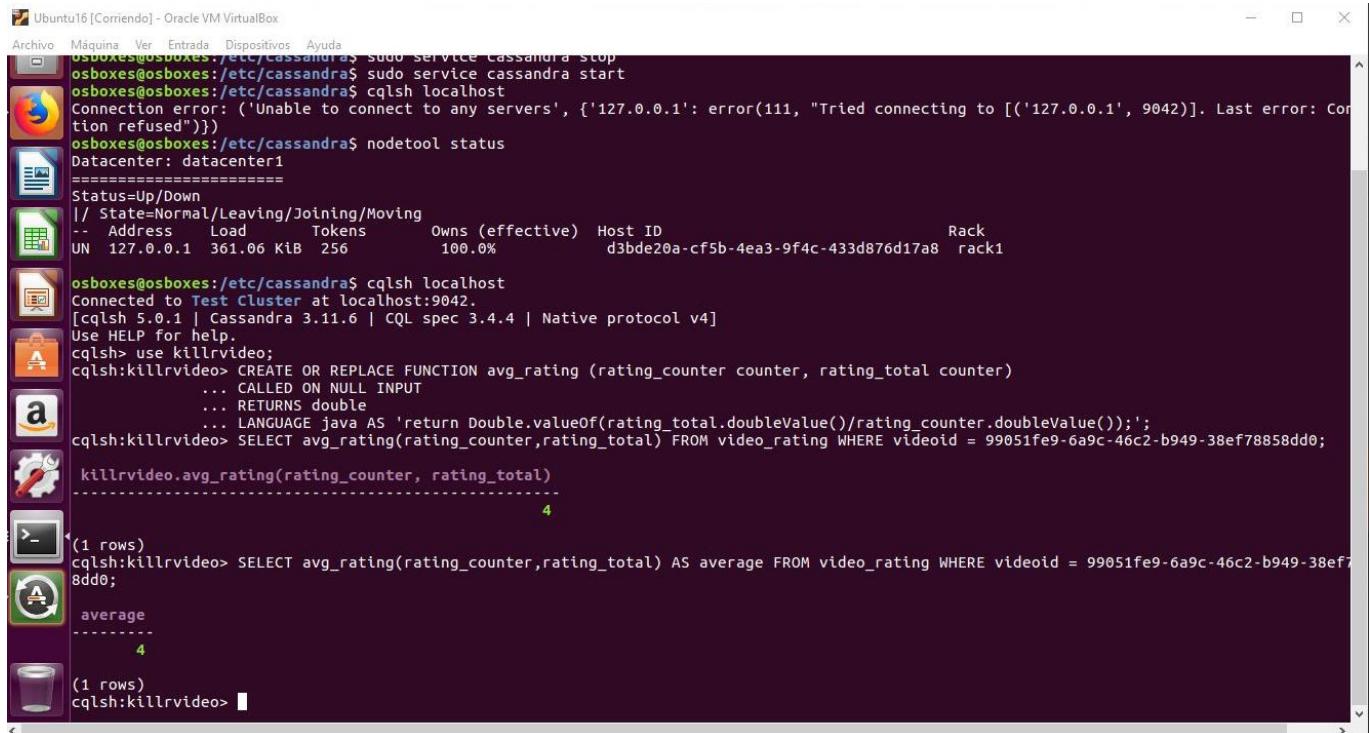
```
cqlsh:killrvideo> CREATE OR REPLACE FUNCTION avg_rating
(rating_counter counter, rating_total counter)
    ... CALLED ON NULL INPUT
    ... RETURNS double
    ... LANGUAGE java AS 'return
Double.valueOf(rating_total.doubleValue()/rating_counter.doubleV
alue());';
```

The screenshot shows a terminal window titled "Ubuntu16 [Corriendo] - Oracle VM VirtualBox". The terminal session starts with the command "cqlsh: use killrvideo;". It then attempts to create a function named "avg_rating" with parameters "rating_counter counter" and "rating_total counter". The function body consists of three lines: "... CALLED ON NULL INPUT", "... RETURNS double", and "... LANGUAGE java AS 'return Double.valueOf(rating_total.doubleValue()/rating_counter.doubleValue());';". An error message follows: "InvalidRequest: Error from server: code=2200 [Invalid query] message="User-defined functions are disabled in cassandra.yaml - set enable_use_defined_functions=true to enable"". The user then exits the session with "cqlsh:killrvideo> exit". They then run "sudo service cassandra stop" and "sudo service cassandra start". A connection error occurs: "Connection error: ('Unable to connect to any servers', {'127.0.0.1': error(111, "Tried connecting to [('127.0.0.1', 9042)]. Last error: Connection refused")})". Finally, they run "nodetool status" which shows a single node in the Datacenter: "datacenter1". The status is "Status=Up/Down |/ State=Normal/Leaving/Joining/Moving". The node details are: Address 127.0.0.1, Load 361.06 KiB, Tokens 256, Owns (effective) 100.0%, Host ID d3bde20a-cf5b-4ea3-9f4c-433d876d17a8, and Rack rack1. The session ends with "cqlsh:killrvideo>".

Una vez creada la función ya podemos realizar las consultas solicitadas:

```
cqlsh:killrvideo> SELECT avg_rating(rating_counter,rating_total)
FROM video_rating WHERE videoid = 99051fe9-6a9c-46c2-b949-
38ef78858dd0;
```

```
cqlsh:killrvideo> SELECT avg_rating(rating_counter,rating_total)
AS average FROM video_rating WHERE videoid = 99051fe9-6a9c-46c2-
b949-38ef78858dd0;
```

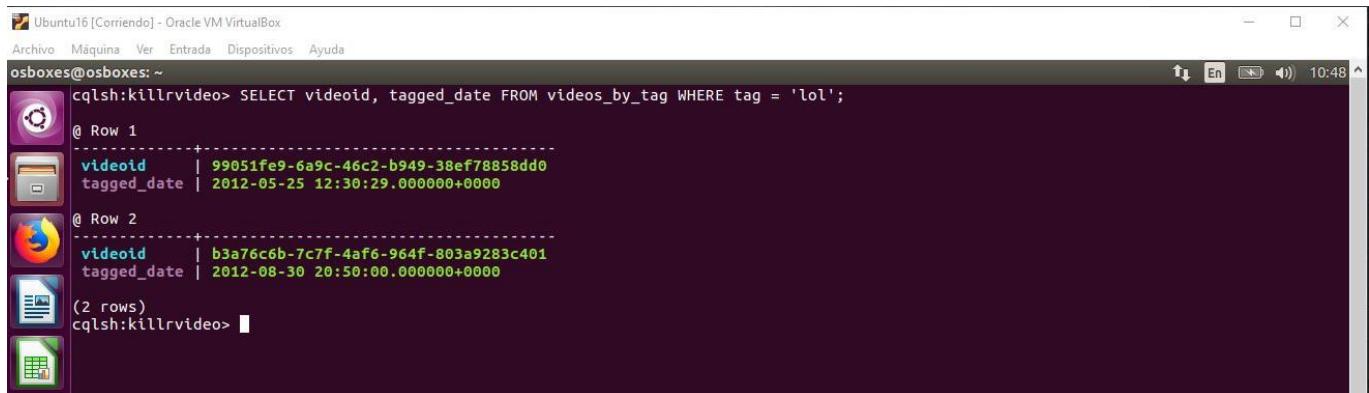


```

Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
osboxes@osboxes:~$ sudo service cassandra stop
osboxes@osboxes:~/etc/cassandra$ sudo service cassandra start
osboxes@osboxes:~/etc/cassandra$ cqlsh localhost
Connection error: ('Unable to connect to any servers', {'127.0.0.1': error(111, "Tried connecting to [('127.0.0.1', 9042)]. Last error: Connection refused")})
osboxes@osboxes:~/etc/cassandra$ nodetool status
Datacenter: datacenter1
=====
Status=Up/Down
!/ State=Normal/Leaving/Joining/Moving
-- Address     Load      Tokens     Owns (effective)  Host ID          Rack
UN 127.0.0.1   361.06 KiB  256        100.0%           d3bde20a-cf5b-4ea3-9f4c-433d876d17a8  rack1

osboxes@osboxes:~/etc/cassandra$ cqlsh localhost
Connected to Test Cluster at localhost:9042.
[cqlsh 5.0.1 | Cassandra 3.11.6 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
cqlsh> use killrvideo;
cqlsh:killrvideo> CREATE OR REPLACE FUNCTION avg_rating (rating_counter counter, rating_total counter)
    ... CALLED ON NULL INPUT
    ... RETURNS double
    ... LANGUAGE java AS 'return Double.valueOf(rating_total.doubleValue()/rating_counter.doubleValue());';
cqlsh:killrvideo> SELECT avg_rating(rating_counter,rating_total) FROM video_rating WHERE videoid = 99051fe9-6a9c-46c2-b949-38ef78858dd0;
cqlsh:killrvideo> killrvideo.avg_rating(rating_counter, rating_total)
=====
4
(1 rows)
cqlsh:killrvideo> SELECT avg_rating(rating_counter,rating_total) AS average FROM video_rating WHERE videoid = 99051fe9-6a9c-46c2-b949-38ef78858dd0;
average
-----
4
(1 rows)
cqlsh:killrvideo>
    
```

```
cqlsh:killrvideo> SELECT videoid, tagged_date FROM videos_by_tag
WHERE tag = 'lol';
```



```

Ubuntu16 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
osboxes@osboxes:~$
cqlsh:killrvideo> SELECT videoid, tagged_date FROM videos_by_tag WHERE tag = 'lol';
@ Row 1
+-----+
| videoid | 99051fe9-6a9c-46c2-b949-38ef78858dd0
| tagged_date | 2012-05-25 12:30:29.000000+0000
+-----+
@ Row 2
+-----+
| videoid | b3a76c6b-7c7f-4af6-964f-803a9283c401
| tagged_date | 2012-08-30 20:50:00.000000+0000
+-----+
(2 rows)
cqlsh:killrvideo>
    
```

2.8. ¿LA CONSULTA ANTERIOR SE PODRÍA REALIZAR SI NO HUBIÉRAMOS METIDO LA COLUMNA "tag" COMO ÍNDICE?. INDICA QUÉ OPCIONES TENDRÍAMOS SI NO HUBIÉRAMOS INTRODUCIDO ESE ÍNDICE.

Si no se hubiera establecido tag como PRIMARY KEY, habría dado un error como el siguiente:

```
cqlsh:killrvideo> SELECT * FROM videos WHERE tags = 'lol';
```

```
InvalidRequest: Error from server: code=2200 [Invalid query]
message="Collection column 'tags' (set<text>) cannot be
restricted by a '=' relation"
```

Es decir, al tratarse de un conjunto de elementos de texto, no se puede relacionar únicamente mediante una relación de igualdad.

Habría que haber realizado una consulta sobre todos los elementos de la tabla **videos** que contuvieran 'lol' en el conjunto **tags**.