

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

07/01/2018

Segunda práctica

Arquitectura Orientada a Servicios

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

Adrián Partida y Joan Manyà

ÍNDICE

INTRODUCCIÓN	2
Modelo Entidad/Relación	5
MODELO DE OPERACIONES.....	6

INTRODUCCIÓN

Somos una empresa de creación de software, encargada de gestionar los problemas que nos proponen nuestros clientes, en este caso la empresa de videojuegos GameBernat quiere gestionar un webservice, mediante el cual pueda registrar usuarios, videojuegos y generar inscripciones entre ambos.

Para ello crearemos una base de datos con todos los usuarios y otra con todos los videojuegos, y las enlazaremos entre si mediante los métodos explicados a continuación.

Realizaremos un servicio web capaz de trabajar contra una base de datos para almacenar todos aquellos datos que le proporcionemos a través de un modelo de datos JSON.

A su vez gracias a JPA usaremos la relación ManyToOne con tal de poder trabajar con objetos de datos en memoria.

Nombre	Descripción	Operación/URL	Parámetros de entrada	Salida OK
Alta persona	Dar de alta una persona introduciendo los parámetros de entrada que se muestran en el siguiente campo.	POST http://localhost:8080/users	int DNI, String Nombre, String Apellidos, Date Fecha Nacimiento, String Email	ID+ HTTP OK+Json de datos
Modificar persona	Modifica una sola persona, introduciendo su ID	PUT http://localhost:8080/users/1	int DNI, campo a modificar	ID+ HTTP OK+Json de datos
Mostrar persona	Llamando a la URL nos muestra todas las personas que tiene	GET http://localhost:8080/users		ID+ HTTP OK+Json de datos
Eliminar persona	Introduciendo la ID eliminamos a una sola persona.	DELETE http://localhost:8080/users/1	int ID	ID+ HTTP OK+Json de datos
Alta Inscripción	Introducimos la ID de persona, ID Videojuego y la fecha de inscripción.	POST http://localhost:8080/inscripciones/2/1	int DNI, int Codigo_videojuegos, date fecha inscripción	ID+ HTTP OK+Json de datos
Mostrar inscripción	Llamando a la URL nos muestra todas las	GET http://localhost:8080/inscripciones		ID+ HTTP OK+Json de datos

	inscripciones que tiene			
Eliminar Inscripción	Para Eliminar la inscripción nos pide la Id de inscripción.	DELETE 'http://localhost:8080/inscripciones/3'	Int ID Inscripción	ID+ HTTP OK+Json de datos
Alta Videojuego	Dar de alta un videojuego introduciendo los parámetros de entrada que se muestran en el siguiente campo.	POST http://localhost:8080/games	int Identificador, String Nombre, String Compañía, String Genero, Date Fecha Lanzamiento	ID+ HTTP OK+Json de datos
Modificar Videojuego	Modifica un solo videojuego, introduciendo su ID	PUT 'http://localhost:8080/games/2'	int ID, campo a modificar	ID+ HTTP OK+Json de datos
Eliminar videojuego	Introduciendo el DNI eliminamos a una sola persona	DELETE 'http://localhost:8080/games/2'	int ID Videojuego	ID+ HTTP OK+Json de datos
Mostrar Videojuego	Llamando a la URL nos muestra todos los videojuegos que tiene	GET/http://localhost:8080/games		ID+ HTTP OK+Json de datos

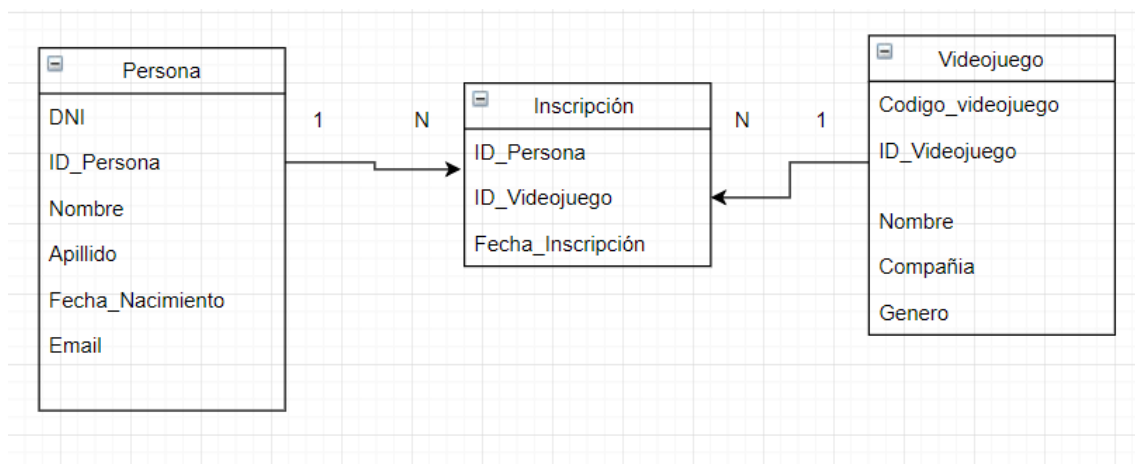
Salida KO En todos los casos HTTP 404 / 500

Modelo Entidad/Relación

One to many es un tipo de cardinalidad que se refiere a la relación entre dos entidades. Se utiliza a través de JPA para poder trabajar con los objetos en memoria y así ahorrar programación y mejorar la integridad.

Utilizamos esta relación ya que una persona puede estar relacionada con muchos videojuegos y un videojuego puede estar relacionado con muchas personas.

De la relación entre videojuegos y personas sale Inscripción creando un nuevo campo llamado fecha de inscripción.



MODELO DE OPERACIONES

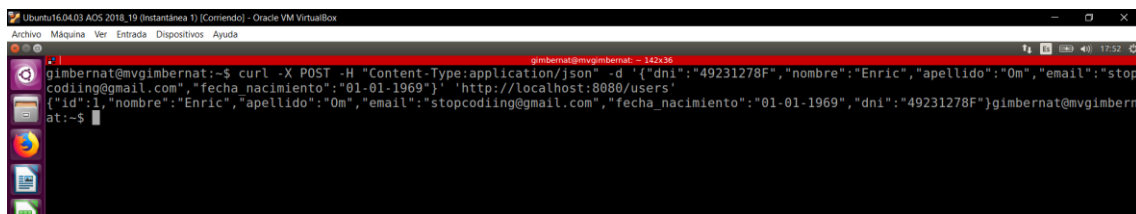
Alta persona

Mediante el método POST con la operación curl hacemos la introducción de una persona:

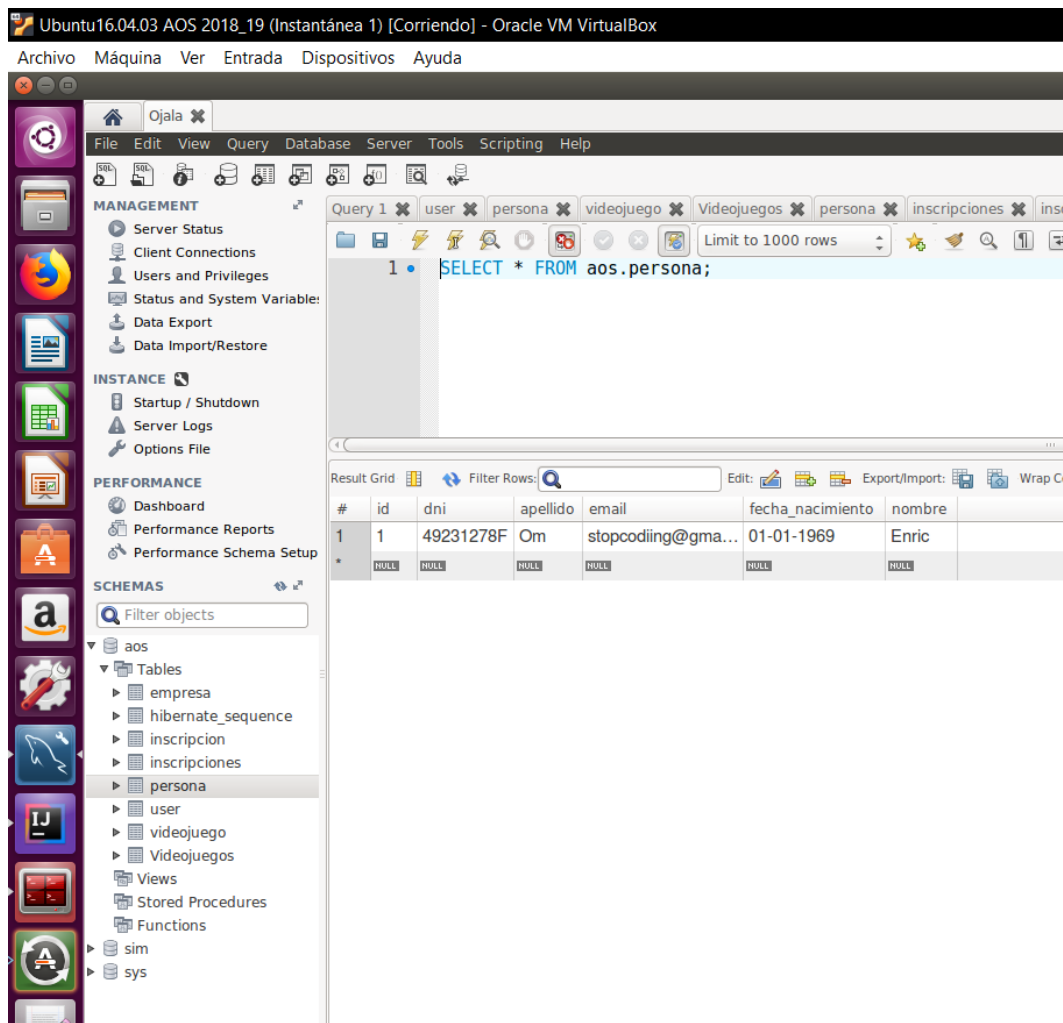
DNI+Nombre+Apellido+Email+Fecha_Nacimiento

Llamamos a la url <http://localhost:8080/users>

```
curl -X POST -H "Content-Type:application/json" -d '{"dni":"49231278F","nombre":"Enric","apellido":"Om","email":"stopcodiing@gmail.com","fecha_nacimiento":"01-01-1969"}' http://localhost:8080/users'
```



El Curl se ha realizado correctamente ya que una vez introducida todos los datos en su correcto orden nos devuelve la información.



Verificamos en la base de datos que la persona se ha creado correctamente y se le ha asignado un ID, en este caso es 1.

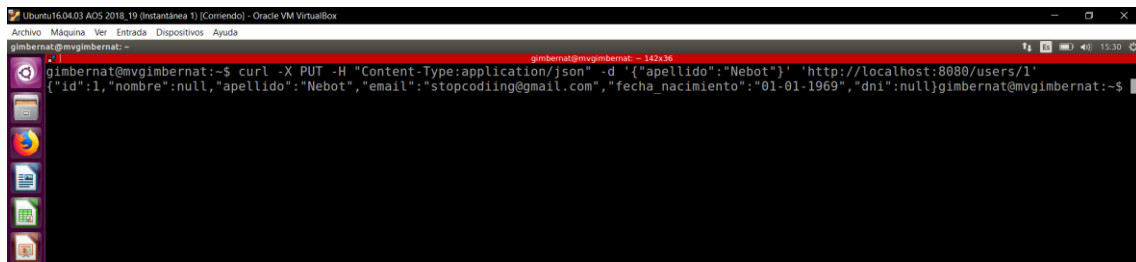
Modificar persona:

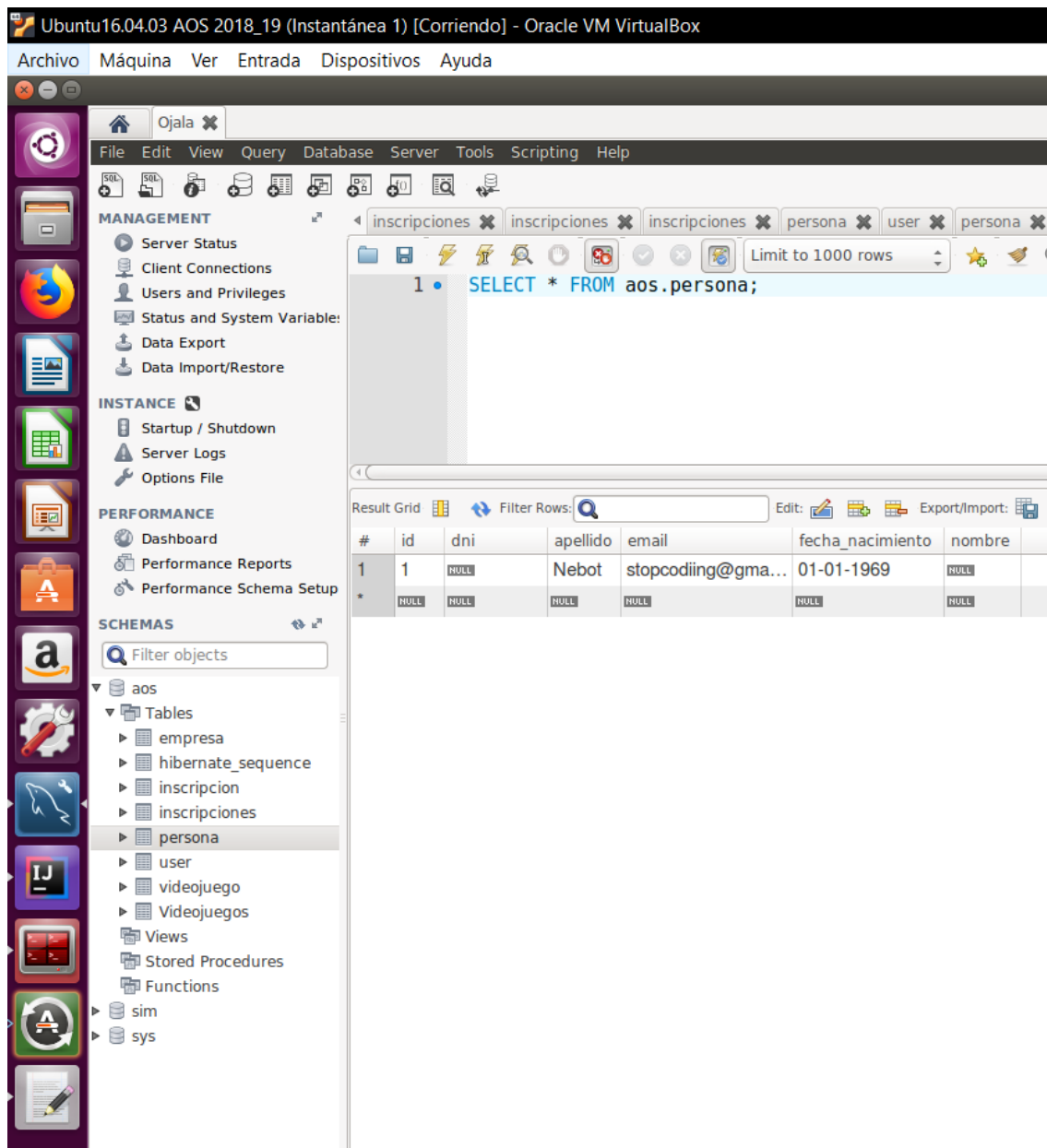
La operación de modificar persona nos permitirá cambiar datos puntuales de la clase Persona. Para ello se pedirá el ID del usuario y a continuación el dato a cambiar.

En la pantalla nos mostrará todos los campos incluido el campo modificado.

```
curl -X PUT -H "Content-Type:application/json" -d '{"apellido":"Nebot"}' 'http://localhost:8080/users/1'
```

Cambiamos el apellido con el put





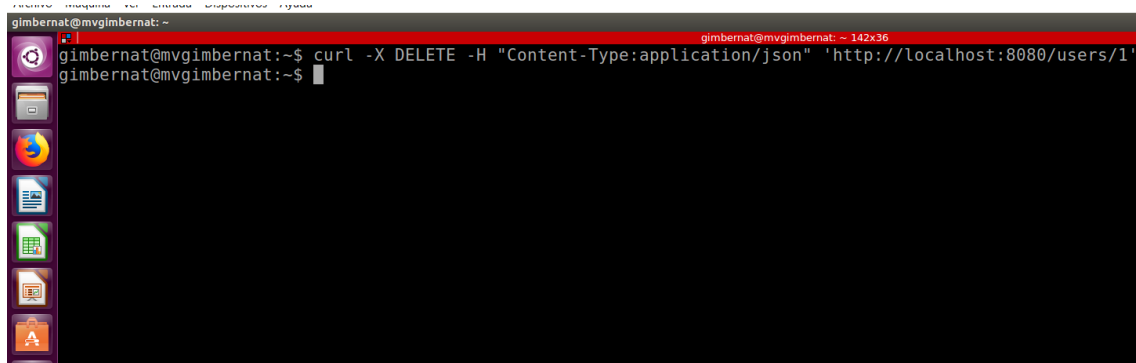
Verificamos en la base de datos que el campo apellido ha sido modificado correctamente.

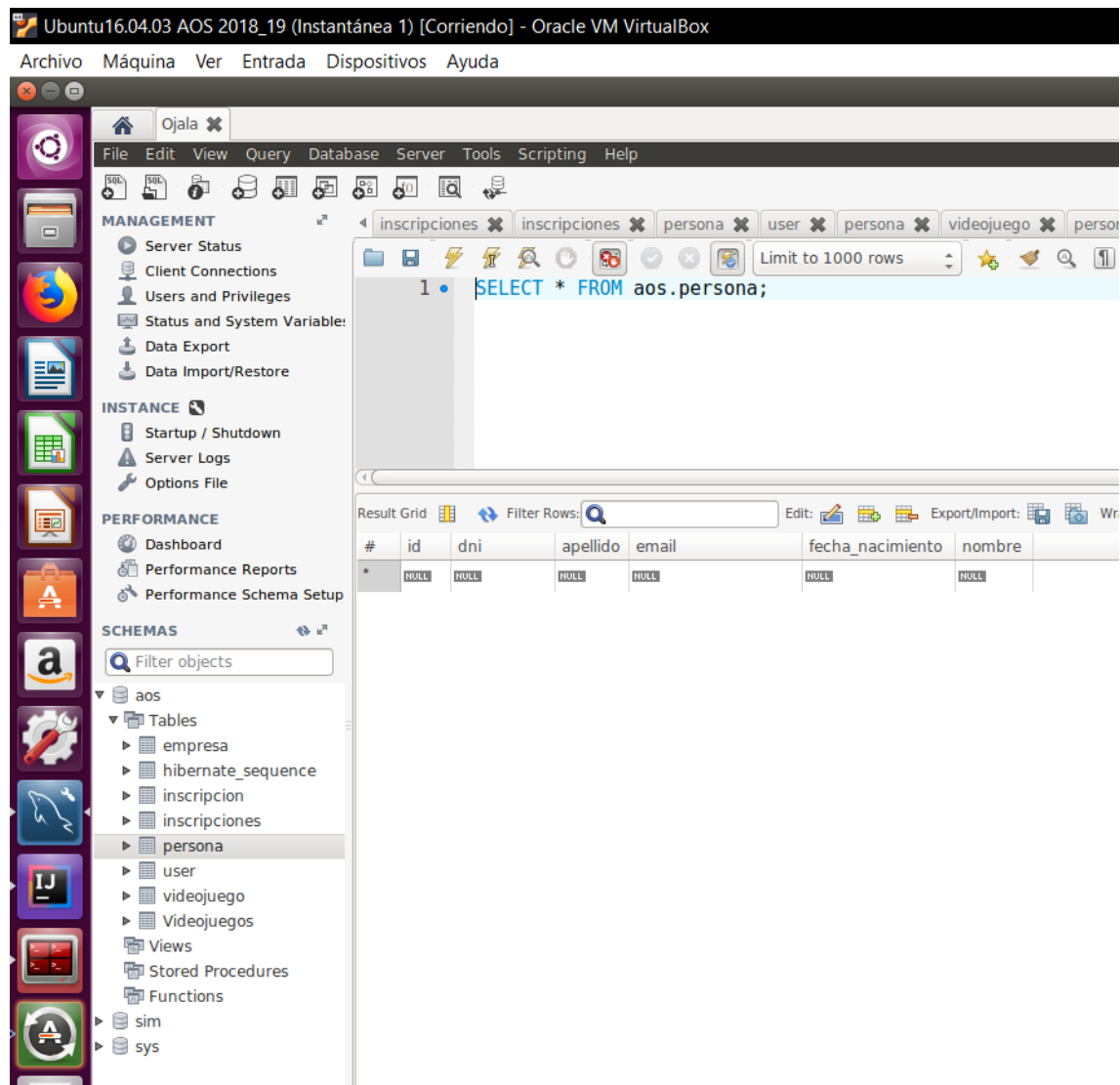
Eliminar persona:

La operación eliminar persona se encargará de quitar de la base de datos aquellos usuarios que no quieran seguir registrados.

Para ello indicaremos el ID de la persona que queramos eliminar. Acto seguido este registro será eliminado y se actualizará en nuestra base de datos.

```
curl -X DELETE -H "Content-Type:application/json" 'http://localhost:8080/users/1'
```



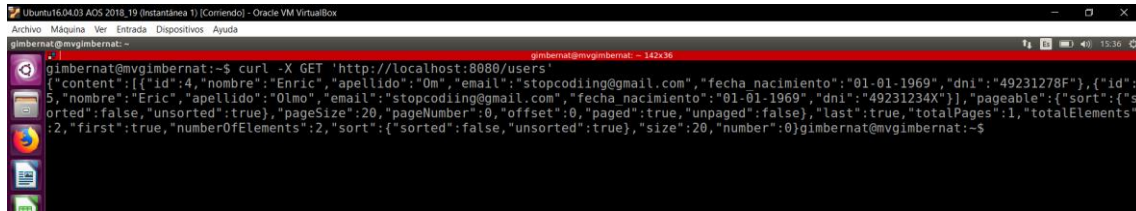


Verificamos en la base de datos que la Persona ha sido eliminada correctamente.

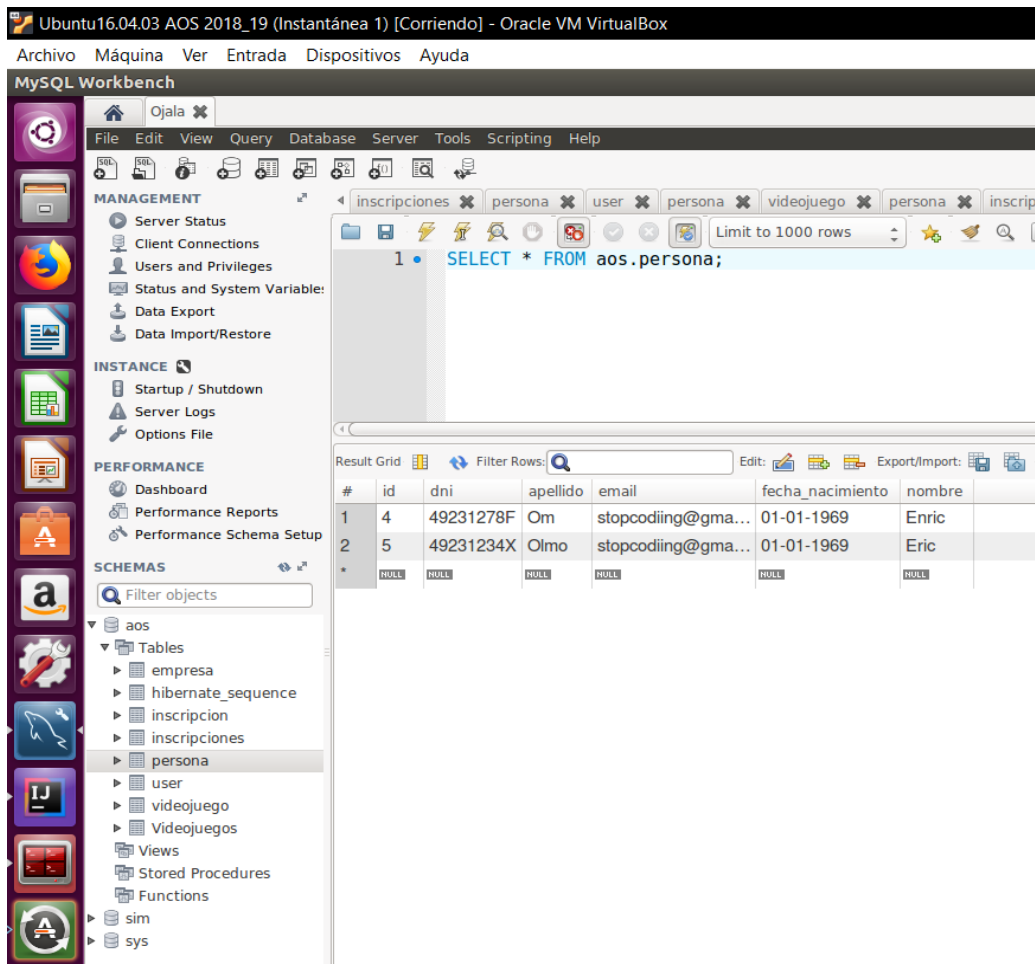
Mostrar Persona

Con la operación de mostrar Persona nos muestra todas las filas de la base de datos de Persona.

```
curl -X GET 'http://localhost:8080/users'
```



```
gimbernat@mvngimbernat:~$ curl -X GET 'http://localhost:8080/users'
{"content":[{"id":4,"nombre":"Enric","apellido":"Om","email":"stopcoding@gmail.com","fecha_nacimiento":"01-01-1969","dni":"49231278F"}, {"id":5,"nombre":"Eric","apellido":"Olmo","email":"stopcoding@gmail.com","fecha_nacimiento":"01-01-1969","dni":"49231234X"}], "pageable":{"sort":{"sorted":false,"unsorted":true},"pageSize":20,"pageNumber":0,"offset":0,"paged":true,"unpaged":false},"last":true,"totalPages":1,"totalElements":2,"first":true,"numberOfElements":2,"sort":{"sorted":false,"unsorted":true},"size":20,"number":0}gimbernat@mvngimbernat:~$
```



En la tabla persona nos muestra todas sus filas, pudiendo observar las personas creadas.

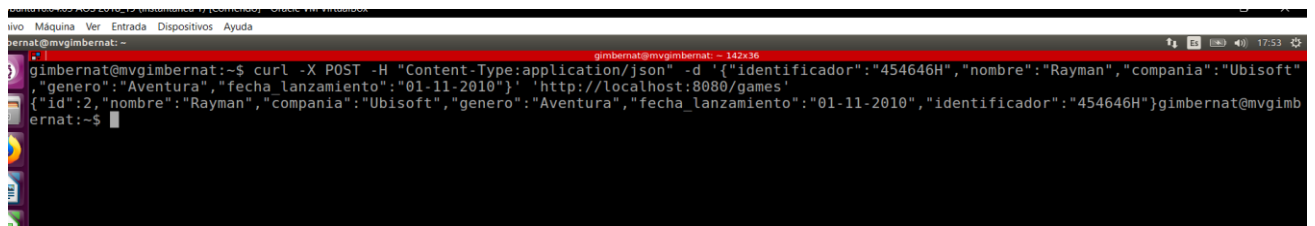
Alta Videojuegos

Mediante el método POST con la operación curl hacemos la introducción de un videojuego:

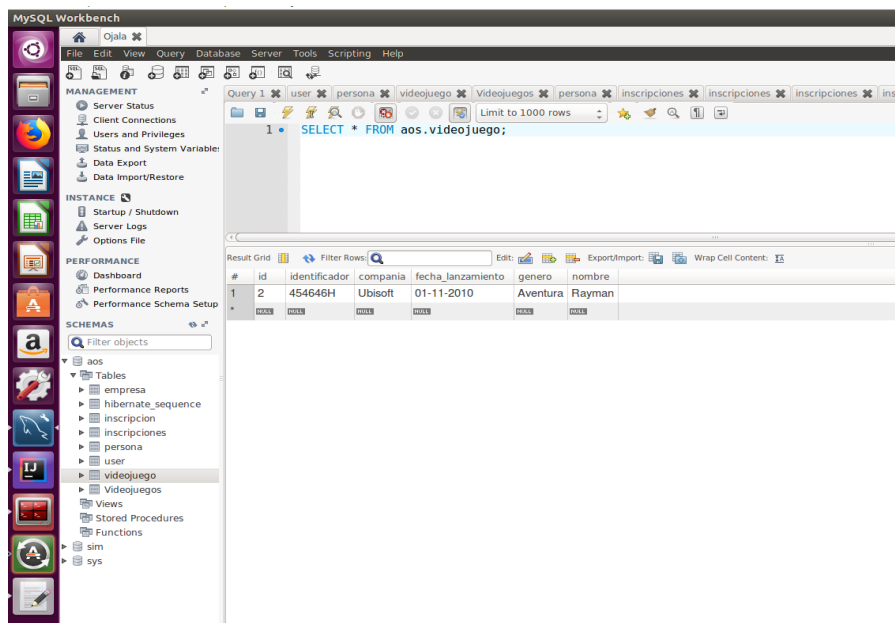
Identificador+ Nombre+ Compañía+ Genero+ Fecha Lanzamiento

```
curl -X POST -H "Content-Type:application/json" -d '{"identificador":"454646H","nombre":"Rayman","compania":"Ubisoft","genero":"Aventura","fecha_lanzamiento":"01-11-2010"}' 'http://localhost:8080/games'
```

Llamamos a la url “ <http://localhost:8080/games> ”



El Curl se ha realizado correctamente ya que una vez introducida todos los datos en su correcto orden nos devuelve la información.

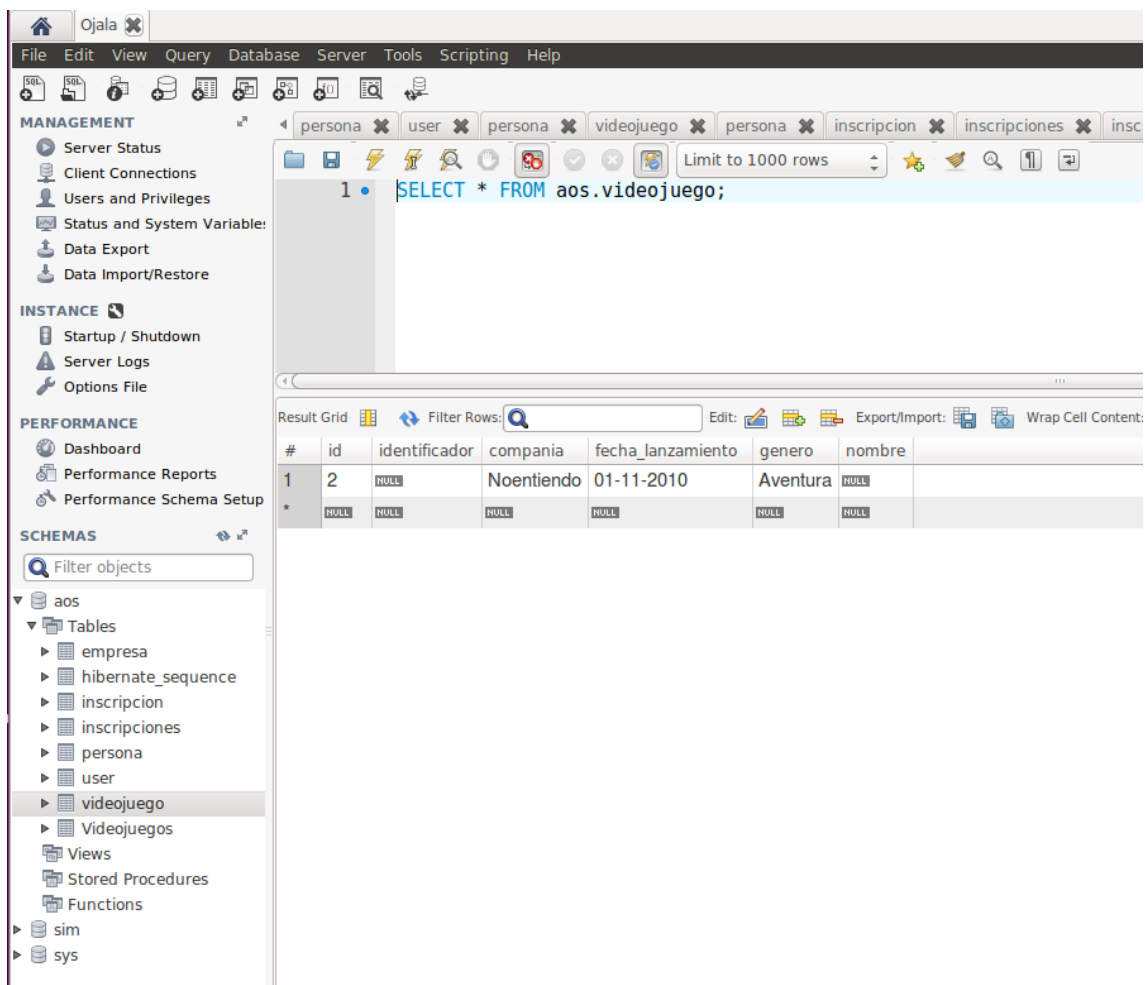
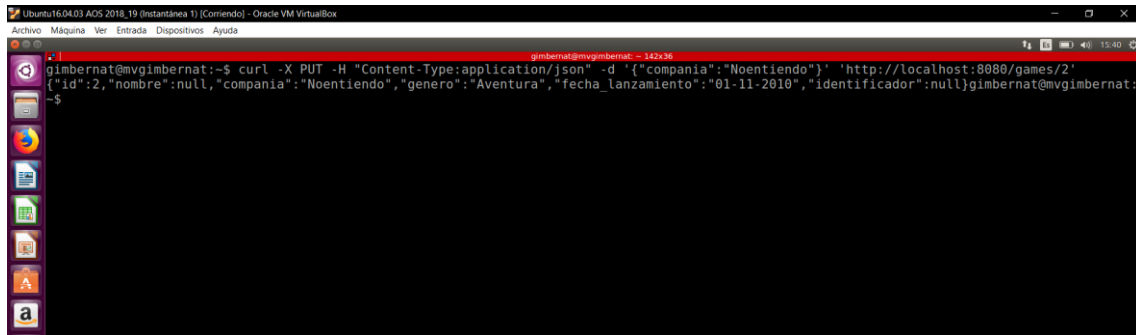


Verificamos en la base de datos la creación del videojuego y el número de identificación que se le ha asignado, en este caso 2.

Modificar videojuego:

En este apartado nos permite modificar los campos de la tabla videojuego, para ello insertaremos la siguiente comanda junto con los campos a modificar, en este caso la compañía.

```
curl -X PUT -H "Content-Type:application/json" -d '{"compañía":"Noentiendo"}' 'http://localhost:8080/games/2'
```

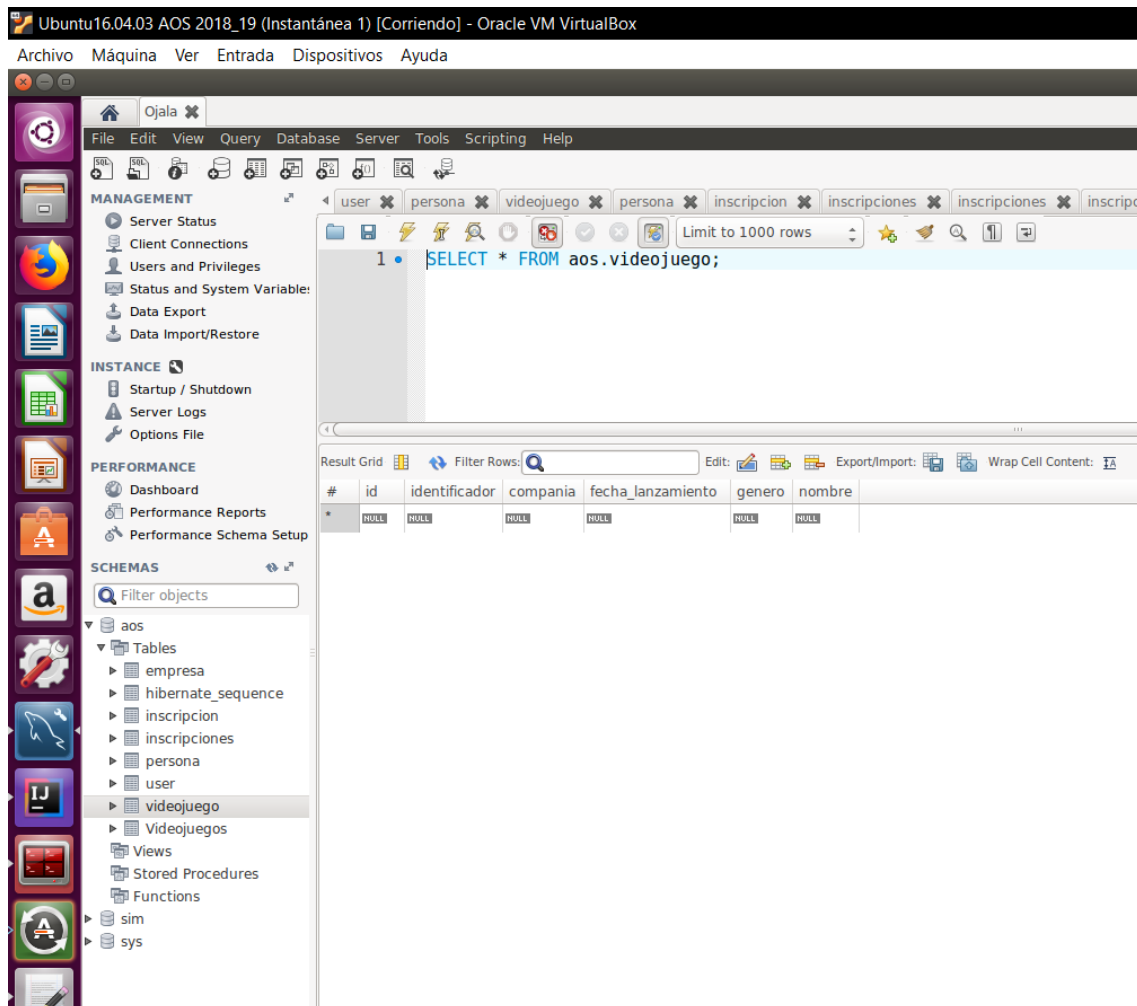
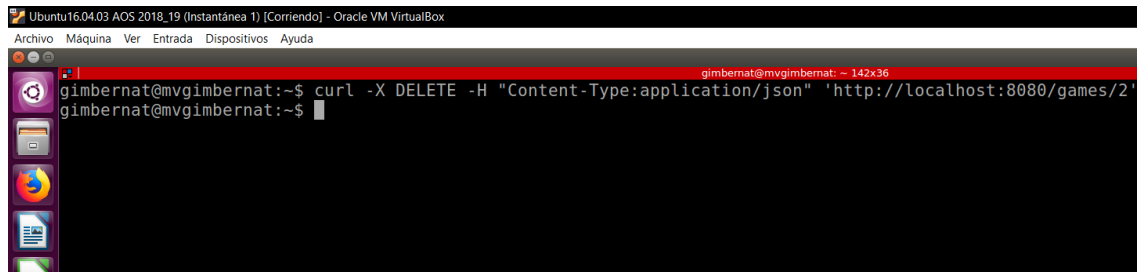


Comprobando en la base de datos que la modificación del campo de compañía se ha realizado correctamente.

Eliminar videojuego:

En este apartado queremos eliminar un videojuego de nuestra base de datos, para ello aplicaremos la siguiente comanda:

```
curl -X DELETE -H "Content-Type:application/json" 'http://localhost:8080/games/2'
```



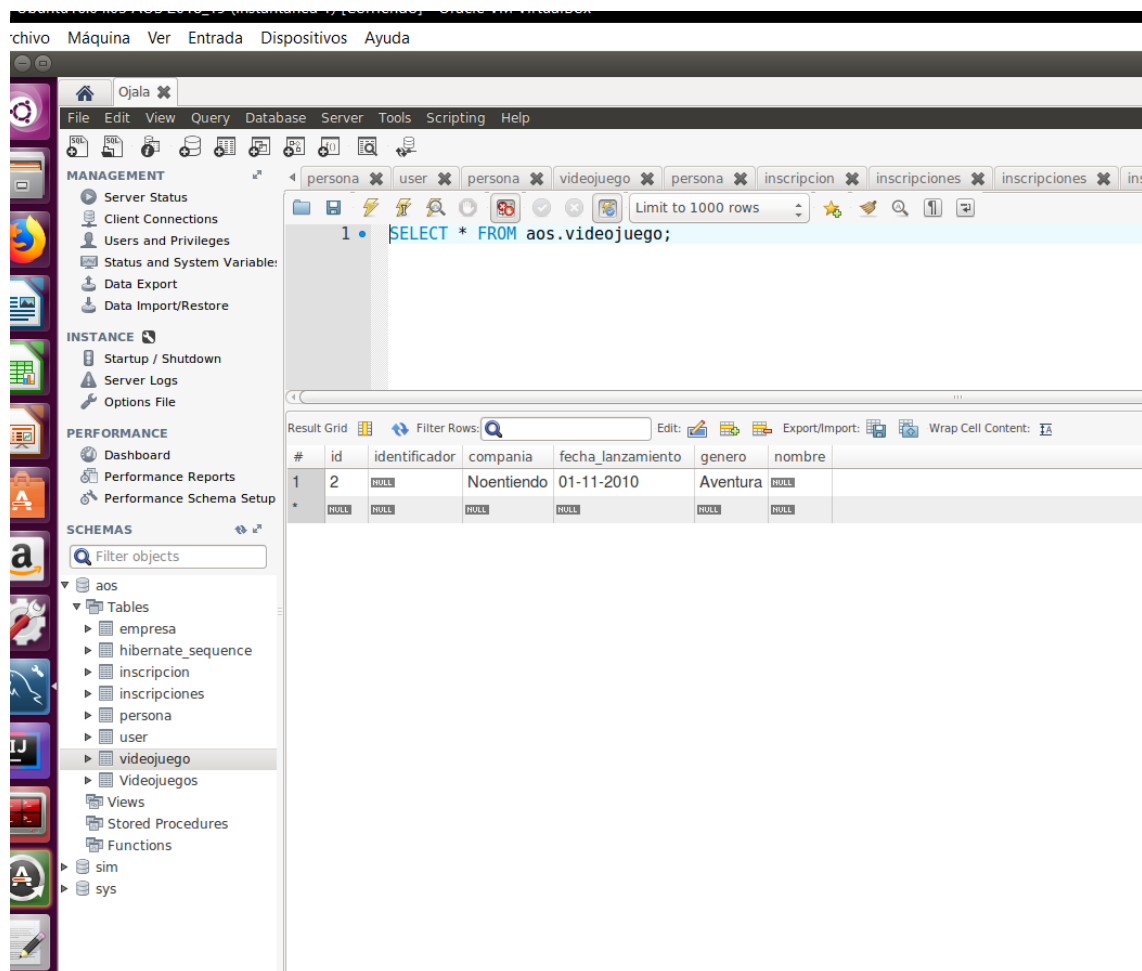
Observando en la base de datos que el videojuego ha sido eliminado.

Mostrar videojuego

Para mostrar todas las filas de la tabla de videojuego introducimos la siguiente comanda:
curl -X GET 'http://localhost:8080/games'



```
gimbernat@mvngimbernat:~$ curl -X GET 'http://localhost:8080/games'
{"content":[{"id":2,"nombre":null,"compania":"Noentiendo","genero":"Aventura","fecha_lanzamiento":"01-11-2010","identificador":null}], "pageable":{"sort":{"sorted":false,"unsorted":true},"pageSize":20,"pageNumber":0,"offset":0,"paged":true,"unpaged":false},"last":true,"totalPages":1,"totalElements":1,"first":true,"numberOfElements":1,"sort":{"sorted":false,"unsorted":true},"size":20,"number":0}gimbernat@mvngimbernat:~$
```



The screenshot shows the DBeaver database client interface. The query editor contains the SQL statement: `SELECT * FROM aos.videojuego;`. The results are displayed in a table with the following data:

#	id	identificador	compania	fecha_lanzamiento	genero	nombre
1	2	NULL	Noentiendo	01-11-2010	Aventura	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL

Nos muestra todo el listado de videojuegos que tenemos en la tabla.

Alta Inscripción

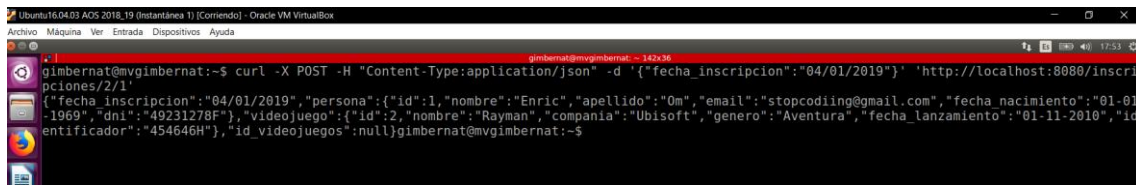
Alta de inscripción es una operación donde pedimos por pantalla el ID de la Persona que quiere realizar la inscripción y el videojuego al que quiere inscribirse.

Acto seguido se generará un nuevo registro en la tabla de inscripciones, mediante una relación entre las tablas Persona y Videojuego.

Mediante el método POST con la operación curl hacemos la introducción de una inscripción con:

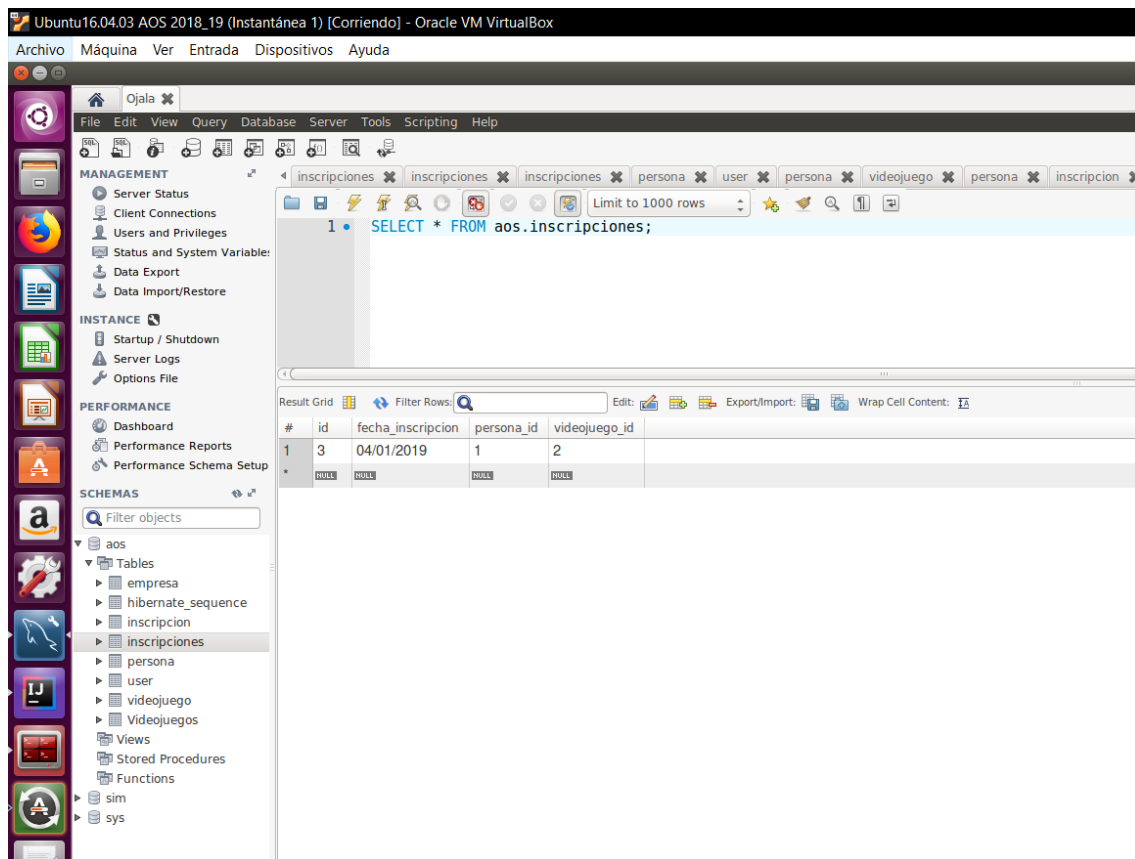
Identificador persona+ Identificador videojuego

Llamamos a la curl <http://localhost:8080/inscripciones/2/1> siendo este 2 el número identificador único del videojuego y el 1 el número identificador único de la persona. De esta manera cuando realizamos el alta de la inscripción lo hacemos en conjunto con los objetos persona y videojuego que les pertenece. En la captura podemos ver como se ha reconocido dichos objetos y en la creación salen sus datos por pantalla.



```
gimbernat@mvngimbernat:~$ curl -X POST -H "Content-Type:application/json" -d '{"fecha_inscripcion":"04/01/2019"}' 'http://localhost:8080/inscripciones/2/1'
{"fecha_inscripcion":"04/01/2019","persona":{"id":1,"nombre":"Enric","apellido":"Om","email":"stopcoding@gmail.com","fecha_nacimiento":"01-01-1969","dni":"49231278F"},"videojuego":{"id":2,"nombre":"Rayman","compania":"Ubisoft","genero":"Aventura","fecha_lanzamiento":"01-11-2010","id_identificador":"454646H"},"id_videojuegos":null}gimbernat@mvngimbernat:~$
```

Aquí podemos ver como se ha creado la inscripción correctamente.



Verificamos en la base de datos que en la tabla inscripción se ha generado una nueva fila con la relación entre persona y videojuego.

Eliminar Inscripción

Por último, con la operación eliminar inscripción podremos borrar de nuestra base de datos una inscripción realizada previamente, introduciendo el ID de la inscripción, como se muestra en la siguiente comanda:

```
curl -X DELETE -H "Content-Type:application/json" 'http://localhost:8080/inscripciones/3'
```

