

# Diseño de bases de datos

## PRA1: Instalación de sistemas y SQL práctico.

### ENUNCIADO

#### A. Creación de un “Virtual Host”.

En este primer ejercicio deberéis crear un host virtual de la práctica 1, que se llamará **prac1\_vh**.

**Se pide** adjuntar una captura de pantalla que muestre la dirección [www.prac1\\_vh.com](http://www.prac1_vh.com) y vuestro nombre en el contenido de la página de inicio del virtual host, tal y como se muestra a continuación:



## B. Creación de tablas con instrucciones del DDL

Una cadena de tiendas de videojuegos quiere tener información de sus tiendas, de los juegos que se venden en éstas, así como de las plataformas a las que pertenecen. El diseño de la base de datos es el siguiente:

### Modelo ER



### Modelo Relacional

TIENDA (**código**, **nombre**, **dirección**);

PLATAFORMA (**id**, **nombre**);

VIDEOJUEGO (**id**, **nombre**, precio, idPlataforma, idTienda);

Donde {idPlataforma} referencia a PLATAFORMA (id)

Donde {códigoTienda} referencia a TIENDA (código)

*NOTA: Las claves principales están subrayadas y los atributos obligatorios marcados en negrita.*

### Tablas informadas

#### TIENDA

<u>código</u>	nombre	dirección
12223A	Trafalgar	C/ Trafalgar, 5
52222222B	Pobla	Av/ La Pobla, 33
52333333C	Dolors	C/ Dels dolors, 177
456789V	Vila	C/ Vila, 76

## PLATAFORMA

id	nombre
1	Nintiendo
2	Sonyi
3	Segga
4	PC

## VIDEOJUEGO

id	nombre	precio	idPlataforma	idTienda
1	Zombies everywhere	2	1	12223A
2	Silent	4	2	12223A
3	Tetris	2	3	52222222B
4	Autos	1	3	52222222B
5	Spaces	6	4	52222222B
6	Evil Jump	3	4	52333333C
7	Silent	4	1	52333333C

Los tipos de datos que se utilizarán en la creación de los atributos de los campos de las tablas son los siguientes:

*Nombre y dirección: cadena variable de 150 caracteres.*

*Código: cadena variable de 10 caracteres.*

*Los ids de videojuegos y plataformas, así como el precio: entero.*

Dado el modelo conceptual, el modelo lógico y la información que se almacenará en el conjunto de tablas normalizadas que se muestra anteriormente **se pide**:

1. Escribir las instrucciones DDL necesarias para **crear todas las tablas** y ejecutarlas en MySQL en el entorno local.

```
create table TIENDA
(
  código varchar(10),
  nombre varchar(150),
  dirección varchar(150),
  primary key (código)
);

create table PLATAFORMA
(
  id integer,
  nombre varchar(150),
  primary key (id)
);

create table VIDEOJUEGO
(
  id integer,
  nombre varchar(150),
  precio integer,
  idPlataforma integer,
  códigoTienda varchar(10),
  primary key (id),
  foreign key (idPlataforma) references PLATAFORMA(id),
  foreign key (códigoTienda) references TIENDA(código)
);
```

2. Escribir, también, las instrucciones necesarias para **introducir todas las filas de datos** en cada una de las tablas resultantes en un entorno local.

```
insert into tienda values ("12223A", "Trafalgar", "C/Trafalgar, 5");
insert into tienda values ("52222222B", "Pobla", "Av/ La Pobla, 33");
insert into tienda values ("52333333C", "Dolors", "C/ Dels dolors, 177");
insert into tienda values ("456789V ", "Vila", "C/ Vila, 76");

insert into plataforma values (1,"Nintiendo");
insert into plataforma values (2,"Sonyi");
insert into plataforma values (3,"Segga");
insert into plataforma values (4,"PC");

insert into videojuego values (1, "Zombies everywhere", 2, 1, "12223A");
insert into videojuego values (2, "Silent", 4, 2, "12223A");
insert into videojuego values (3, "Tetris", 2, 3, "52222222B");
```

```
insert into videojuego values (4, "Autos", 2, 3, "52222222B");  
insert into videojuego values (5, "Spaces", 6, 4, "52333333B");  
insert into videojuego values (6, "Evil Jump", 3, 4, "52333333C");  
insert into videojuego values (7, "Silent", 4, 1, "52333333C");
```

3. Una vez creadas las tablas e insertadas las filas de datos en las tablas, realizar una exportación de base de datos a un archivo con extensión SQL.
4. Importar las tablas de la base de datos al DBMS remoto. Adjuntad una captura de pantalla que muestre vuestro MySQL remoto, con todas las tablas creadas y los datos insertados.

***Poned a partir de aquí las respuestas de los puntos 1, 2 y 4 de esta actividad***

## C. Uso del Lenguaje de Manipulación de Datos (DML) para gestionar la base de datos

A partir de la base de datos creada en el ejercicio anterior, **se pide** el conjunto de instrucciones que permitan hacer las gestiones que se detallan a continuación:

1.- Indicar las instrucciones necesarias para reflejar que el videojuego con id = 2 cambia de plataforma a PC (id = 4) y de precio; siendo el nuevo precio igual a 4. Asimismo, el videojuego con id = 3 pasa a venderse en la tienda con código=52333333C y su precio será igual a 6.

```
update videojuego
set precio = 4, idPlataforma=4
where id = 2;

update videojuego
set precio = 6, códigoTienda="52333333C"
where id = 3;
```

2.- Mostrar el código y el nombre de la tienda que vende más videojuegos. En el caso de que haya más de una tienda, mostrar sólo una tienda. Sólo hay que mostrar una de ellas.

```
select t.código, t.nombre
from tienda t inner join videojuego v on t.código = v.códigotienda
group by t.código
order by count(*) desc
limit 1;
```

3.- Mostrar todos los datos de las plataformas que tengan 2 o más videojuegos.

```
select p.id, p.nombre
from plataforma p inner join videojuego v on p.id = v.idPlataforma
group BY p.id, p.nombre
having count(*) >= 2;
```

4.- Mostrar el nombre, el precio y nombre de la plataforma de aquellos videojuegos cuyo precio número sea menor que la media de precios de todos los juegos de la plataforma con id = 3.

```
select v.nombre, v.precio, p.nombre
from plataforma p inner join videojuego v on p.id = v.idplataforma
where v.precio < (
    select avg(precio)
    from videojuego
    where idplataforma = 3)
```

5.- De todas las tiendas que tienen videojuegos a la venta, mostrar su código, nombre, así como el máximo de precio de todos los videojuegos de la plataforma con id = 4 que vende.

```
select t.código, t.nombre, max(v.precio)
from tienda t inner join videojuego v on t.código = v.códigotienda
where v.idplataforma = 4
group by t.código, t.nombre
```

6.- Añadir un nuevo atributo a la tabla PLATAFORMA llamado “numJuegos”, de tipo entero.

```
alter table PLATAFORMA
add numJuegos integer;
```

7.- Crear las instrucciones del DML necesarias para actualizar el campo “numJuegos” para todas las plataformas, en función de lo datos de la tabla VIDEOJUEGO.

```
update plataforma
set numJuegos = (
    select count(*)
    from videojuego
    where idPlataforma = 1
)
where id = 1;
```

```
update plataforma
set numJuegos = (
    select count(*)
    from videojuego
    where idPlataforma = 2
)
where id = 2;
```

```
update plataforma
set numJuegos = (
    select count(*)
    from videojuego
    where idPlataforma = 3
)
where id = 3;
```

```
update plataforma
set numJuegos = (
    select count(*)
    from videojuego
    where idPlataforma = 4
)
where id = 4;
```

O BIEN, Mysql también deja hacer esto, aunque no sea SQL estándar:

```
update plataforma
```

```
set numJuegos = (  
  select COUNT(idPlataforma)  
  from videojuego  
  where idPlataforma = plataforma.id  
)
```

8.- Crear las instrucciones del DML necesarias para eliminar aquellos juegos cuya plataforma sea Segga (id=3) y el precio sea menor que 4.

```
delete from videojuego  
where idPlataforma = 3 and precio < 4;
```

Para tener en cuenta:

- La base de datos se considera creada en local
- No se puede utilizar ningún dato en la instrucción SQL, que no esté en el enunciado de la pregunta
- No hace falta añadir ni eliminar ningún dato de la base de datos a menos que la propia pregunta lo pida.

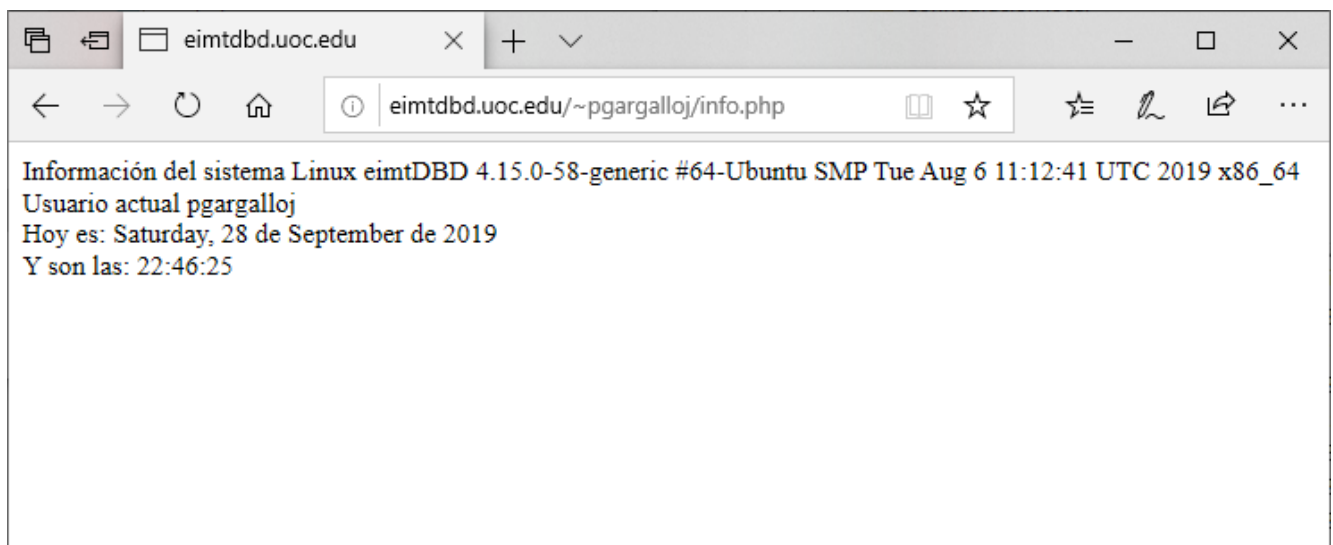


## D. Uso del espacio de almacenamiento remoto FTP

Tenéis que subir a vuestro espacio FTP en el servidor el fichero que se adjunta con la práctica (“info.php”).

Lo tenéis que hacer poniendo en la barra de direcciones de un navegador la siguiente dirección (modificada con los datos de vuestro usuario, claro):

<http://eimtdbd.uoc.edu/~pgargallo/info.php>



Como respuesta a este punto, sólo hace falta que pongáis la dirección URL que se pide en el apartado.

## Criterios de corrección

El criterio de corrección de esta práctica es el siguiente:

**Parte A (Virtual Host): 1 punto.** Se tendrá la captura de pantalla demostrando la realización del VirtualHost.

**Parte B (DDL): 3 puntos.** Se tendrá en cuenta la sintaxis de las instrucciones SQL, así como la correcta exportación e importación de la base de datos en el entorno remoto, indicada en la captura de pantalla a adjuntar.

**Parte C (DML): 4.5 puntos.** Se tendrá en cuenta la corrección sintáctica de las diferentes instrucciones SQL a implementar en cada apartado, así como la corrección de los resultados que generen. Una instrucción con errores de sintaxis se calificará automáticamente con 0 puntos.

**Parte D (Espacio de almacenamiento remoto): 1.5 puntos.** Se tendrá en cuenta que el fichero a subir esté exactamente en la URL que se pide en el enunciado, y que no haya sido modificado de ninguna forma.

## Formato de entrega

El formato del documento a entregar será obligatoriamente PDF para asegurar la correcta visualización de los diagramas. A pesar de que se facilita el enunciado en un formato editable, la solución final entregada ha de estar en formato PDF.

En el aula de teoría hay la información sobre las competencias y objetivos de esta PRA, así como las fechas importantes (fecha de inicio, de entrega, de publicación de notas y de solución) y los materiales complementarios necesarios para la realización de la PRA.