PRACTICA 3

Nombre	**Aitor Larrinoa Rementeria**
BBDD	**Mysql**

SESION 6. EJERCICIO 9

ENUNCIADO

Repositorio a utilizar: airports. Se necesita almacenar en una tabla independiente de nombre osmairport_mx_cn los países, los nombres, códigos IATA, códigos ICAO y Operadores de los aeropuertos que tengan operador y cuyo país sea Mexico o Canada.

SOLUCION

```
use airports


create table osmairport_mx_cn as
select Country, Name, IATAcode, ICAOcode, Operator
from osmairport
where Operator is not null and
(upper(country) like '%MEXICO%' or upper(country) like '%CANADA%');


select * from osmairport_mx_cn
```

SESION 7. EJERCICIO 9

ENUNCIADO

Repositorio a utilizar videogames. Mostrar un resumen de videojuegos de Estrategia (metadata_genres) agrupados por creadores (metadata_publishers). Mostrar el número de videojuegos creados, asi como la suma de ventas (metrics_sales).

SOLUCION

```
use videogames


select count(*), sum(Metrics_Sales), Metadata_Publishers from videogame
where upper(Metadata_Genres) like '%Strategy%'
group by Metadata_Publishers;
```

SESION 8. PREGUNTA 11

ENUNCIADO

Repositorio a utilizar: moviesmall. Se quiere saber el número de películas en las que ha actuado cada actor. Mostrar el id del actor, el nombre, el apellido y el número de películas en las que ha actuado. Ordenarlo alfabéticamente por nombre y apellido.

SOLUCION

```
use moviesmall


select  count(*), act.id, act.first_name, act.last_name from actor act
inner join `role` r on (act.id = r.actor_id)
group by act.id, act.first_name, act.last_name
order by act.first_name, act.last_name;
```

SESION 11. EJERCICIO 6

ENUNCIADO

Un banco quiere gestionar sus movimientos mediante unos disparadores que actualicen el sado. El banco tiene una tabla de "saldo". Esta tabla tiene un id de la cuenta, un número de cuenta, y un saldo. **Además, la tabla "saldo" previamente ya tiene algún saldo registrado para alguna cuenta (os lo podéis inventar)**. El banco también tiene una tabla de "movimiento" que tiene un id de movimiento, el número de cuenta sobre la que se hace el movimiento, un concepto de movimiento y un importe de movimiento. Cada vez que se inserte un movimiento nuevo, el saldo (en la tabla "saldo") se tiene que actualizar.
Construir las tablas y el trigger para cumplir el requerimiento. Insertar un saldo inicial, y posteriormente un par de movimientos para asegurar el correcto funcionamiento del sistema.

RESPUESTA

```
create database banco


use banco


create table saldo (
id_cuenta int,
```

```
num_cuenta varchar(100),
saldo_cuenta int
)

insert into saldo values (1, "6370 0466 04 3693600089", 1000)

insert into saldo values (2, "7712 4326 10 1992210900", 560)

select * from saldo



create table movimiento (
id_movimiento int,
num_cuenta varchar(100),
concepto varchar (200),
importe double
)

create trigger movimiento_saldo after insert on movimiento
for each ROW
BEGIN
        update saldo
        set saldo_cuenta = saldo_cuenta + new.importe
        WHERE
        num_cuenta = new.num_cuenta;
END;

insert movimiento values (1, "6370 0466 04 3693600089", "ingreso", 100)

insert movimiento values (2, "7712 4326 10 1992210900", "gasto", -50)
```

```
select * from movimiento
```

SESION 13. EJERCICIO 6

ENUNCIADO

Ahora la concesionaria quiere prohibir la venta de los coches que se hayan fabricado antes del 2010. Rehacer la tabla para añadir este requerimiento e insertar un par de registros para asegurar que se cumpla.
(Usar como base la solución del Ejercicio 5 de esta sesión)

RESPUESTA

```
create database coches


use coches


CREATE TABLE venta (
id INTEGER AUTO_INCREMENT,
modelo VARCHAR(100) NOT NULL,
cliente VARCHAR(100),
fechaventa DATE,
matricula VARCHAR(10) UNIQUE,
marca VARCHAR(20) DEFAULT 'Ford',
fechafabricacion DATE CONSTRAINT antes2010 CHECK (fechafabricacion
>= '2010-01-01'),
precioorginal INTEGER,
preciosegundamano DOUBLE GENERATED ALWAYS AS ( precioorginal -
precioorginal*(YEAR(fechaventa) - YEAR(fechafabricacion))*(10/100)),
CONSTRAINT pkventa PRIMARY KEY (id)
);
```

```
-- lo siguiente nos dará error, pues introducimos fechafabricacion menor que 2010-01-01
insert into venta(modelo, fechafabricacion) values ('kuga', '2009-01-01');

-- lo siguiente se ejecutará bien, pues ya introducimos fechafabricacion mayor o igual que 2010-01-01
insert into venta(modelo, fechafabricacion) values ('mondeo', '2010-01-01');
```