

SOLUCIÓN

Ejercicio 1

GRUPO A

Supongamos que tenemos un videoclub,

Socio

<u>CodSocio</u>	numérico de 3 posisciones
nombreSocio	alfanumérico de 20
direcSocio	alfanumérico 30
teléfono:	alfanumérico de 9 posiciones

Pelicula

<u>codPelicula</u>	alfanumérico de 4 posiciones
titulo	alfanumérico de 30 posiciones
director	alfanumerico de 20 posiciones
artista	alfanumérico de 20 posiciones

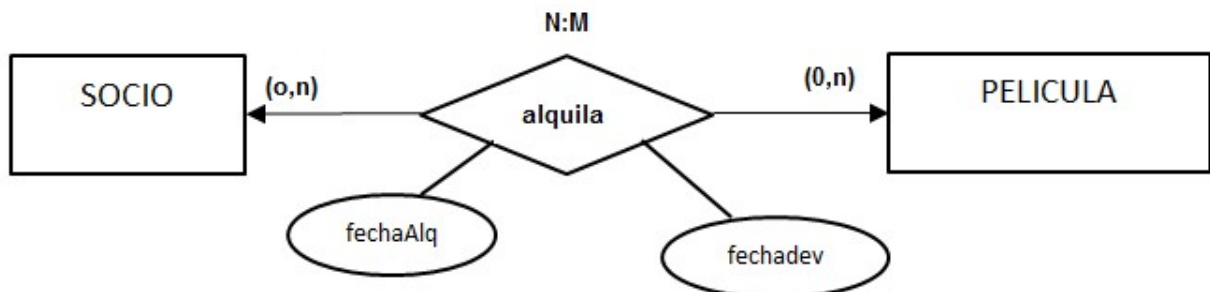
Alquiler

<u>CodSocio</u>	numerico de 3 posisciones
<u>codPelicula</u>	alfanumérico de 4 posiciones
feAlquiler	tipo fecha
feDevolucion	fecha

- 1.- Crear un usuario que se llame VIDEOCLUB con privilegios de administrador

```
CREATE USER VIDEOCLUB IDENTIFIED BY VIDEOCLUB;  
GRANT ALL PRIVILEGES TO VIDEOCLUB;
```

- 2.- Crear las tablas a partir del modelo en el usuario creado anteriormente.



SOLUCIÓN

```
CREATE TABLE socio
(
    CODSOCIO          NUMBER(3) NOT NULL,
    NOMBRESOCIO       VARCHAR2(20),
    DIRECSOCIO        VARCHAR2(30),
    TELEFONO           VARCHAR2(9),
    CONSTRAINT SOC_COD_PK1 PRIMARY KEY (CODSOCIO)
);
```

```
CREATE TABLE pelicula
(
    CODPELICULA        VARCHAR2(4) NOT NULL,
    TITULO              VARCHAR2(30),
    DIRECTOR            VARCHAR2(20),
    ARTISTA              VARCHAR2(20),
    CONSTRAINT PEL_COD_PK1 PRIMARY KEY (CODPELICULA)
);
```

```
CREATE TABLE alquiler
(
    CODSOCIO            NUMBER(3) NOT NULL,
    CODPELICULA          VARCHAR2(4) NOT NULL,
    FEALQUILER           DATE,
    FEDEVOLUCION          DATE,
    CONSTRAINT ALQ_COD_PK1 PRIMARY KEY (CODSOCIO,CODPELICULA),
    CONSTRAINT ALQ_COD_FK2 FOREIGN KEY (CODSOCIO) REFERENCES SOCIO (CODSOCIO),
    CONSTRAINT ALQ_COD_FK3 FOREIGN KEY (CODPELICULA) REFERENCES PELICULA (CODPELICULA)
);
```

COMMIT;

3.- Introducir los datos en las tablas e indicar el orden en que se deben de introducir

Usuario desde el que se realiza la acción: [VIDEOCLUB](#)

Instrucción

```
insert into SOCIO values(111,'Socio111','direccion socio111','111111111');
insert into SOCIO values(222,'Socio222','direccion socio222','222222222');
insert into SOCIO values(333,'Socio333','direccion socio333','333333333');
insert into SOCIO values(444,'Socio444','direccion socio444','444444444');
insert into SOCIO values(555,'Socio555','direccion socio555','555555555');
```

SOLUCIÓN

```
insert into PELICULA values('Pe11','Titulo 11','Director 11','Artista 11');
insert into PELICULA values('Pe22','Titulo 22','Director 22','Artista 22');
insert into PELICULA values('Pe33','Titulo 33','Director 33','Artista 11');
insert into PELICULA values('Pe44','Titulo 44','Director 44','Artista 22');
insert into PELICULA values('Pe55','Titulo 55','Director 55','Artista 33');
insert into PELICULA values('Pe66','Titulo 66','Director 66','Artista 44');
```

```
insert into ALQUILER values(111,'Pe11','01/01/17','08/01/17');
insert into ALQUILER values(222,'Pe22','07/01/17','16/01/17');
insert into ALQUILER values(333,'Pe11','16/01/17','25/01/17');
insert into ALQUILER values(444,'Pe33','07/01/17','16/01/17');
```

COMMIT;

- 4.- Crear el usuario EXAMEN1 solo con los privilegios básicos;

Usuario desde el que se realiza la acción: VIDEOCLUB

Instrucción SQL:

```
CREATE USER EXAMEN1 identified by examen1
Grant Connect, Resource to EXAMEN1
```

- 5.- Realizar las acciones oportunas para que el usuario EXAMEN1 pueda consultar las tablas anteriores

Usuario desde el que se realiza la acción: VIDEOCLUB

Instrucción SQL:

```
GRANT SELECT ON alquiler to examen1
GRANT SELECT ON pelicula to examen1
GRANT SELECT ON socio to examen1
```

Indicar si asignamos privilegios de sistema o de objeto:

- 6.- Comprobar las instrucciones para poder consultar dichas tablas el usuario EXAMEN1

Usuario desde el que se realiza la acción: EXAMEN1

Instrucción SQL:

```
SELECT * FROM VIDEOCLUB.SOCIO
SELECT * FROM VIDEOCLUB.PELICULA
SELECT * FROM VIDEOCLUB.ALQUILER
```

SOLUCIÓN

- 7.- Realizar las acciones necesarias para que el usuario EXAMEN1 pueda añadir en la tabla en la que se registran los alquileres

Usuario desde el que se realiza la acción: [VIDEOCLUB](#)

Instrucción SQL:

```
GRANT INSERT ON VIDEOCLUB.ALQUILER TO EXAMEN1
```

Acceder

- 8.- Comprobar que desde dicho usuario puede añadir un préstamo y consultar si se ha añadido el registro.

Usuario desde el que se realiza la acción: [EXAMEN1](#)

Instrucción SQL:

[ACCEDER A LA TABLA ALQUILERES DE VIDEOCLUB DESDE EL ESQUEMA EXAMEN1 Y PROBAR A INSERTAR UN NUEVO REGISTRO](#)

- 9.- Desde el usuario EXAMEN1 modificar en la tabla SOCIO el teléfono de uno de los socios. (Asegurarse de pulsar al finalizar la modificación el ENTER y después pulsar el botón de COMMIT para que la modificación sea efectiva y definitiva). ¿Qué sucede y por qué?

Usuario desde el que se realiza la acción: [EXAMEN1](#)

Instrucción SQL:

[NO SE PUEDE YA QUE NO TIENE PRIVILEGIOS PARA ACTUALIZAR.](#)

- 10.- Realizar los pasos necesarios para permitir que el usuario EXAMEN1 pueda modificar en la tabla se SOCIO solo los campos de dirección y teléfono

Usuario desde el que se realiza la acción: [VIDEOCLUB](#)

Instrucción SQL:

```
GRANT UPDATE (TELEFONO) ON VIDEOCLUB.SOCIO TO EXAMEN1
```

Indicar si asignamos privilegios de sistema o de objeto: