

Ejercicio 4 SQL - Relación 2

1º DAW - Bases de Datos

Autor: Mario Pérez

2018

Contenido

Enunciado.....	3
Modelo entidad-relacion y paso a tablas.....	4
Creación de la base de datos y las tablas.....	5
Ejercicios.....	9

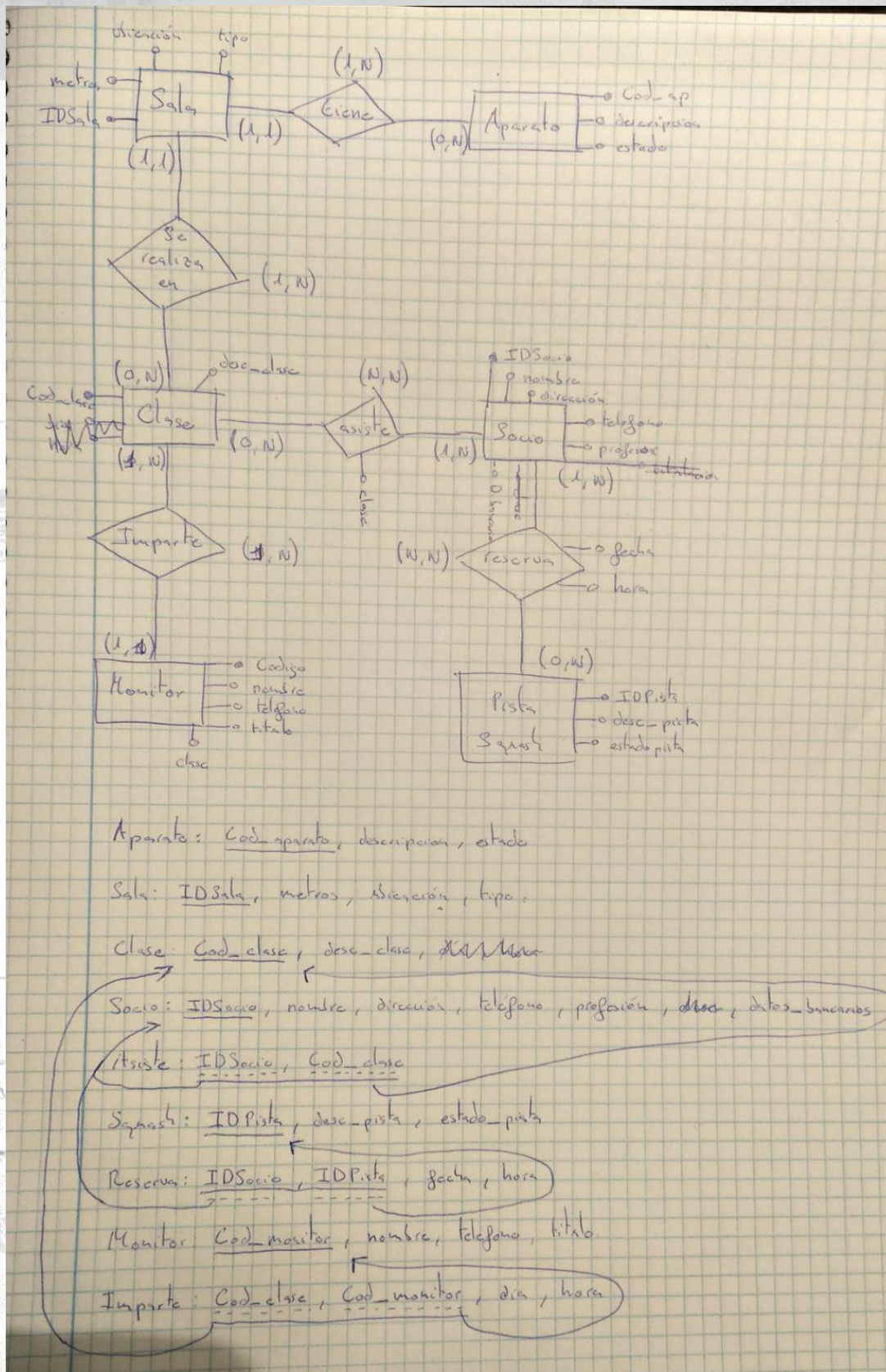
Ejercicio 4 SQL - Relación 2

Enunciado.

A.4. En el gimnasio “Siempre en forma” se quiere implantar una base de datos para llevar el control de los socios, recursos utilizados, etc. Las especificaciones que nos han dado son las siguientes:

- Existen varias salas, de las cuales se quiere guardar información, como los metros cuadrados que ocupa, ubicación y el tipo de sala (cardio, general, muscular). Cada sala se identifica por un número.
- Hay salas que tienen aparatos y salas que no. En las salas se pueden o no impartir clases.
- Cada aparato está asignado a una única sala, y de cada uno de ellos se quiere tener almacenado su código, descripción y estado de conservación.
- También se quiere mantener información relacionada con las clases que se imparten (descripción y día/hora en la que se imparten); cada clase se identifica por un código de clase. Cada clase tiene asignada una sala en la que se imparte y un monitor.
- De cada monitor se quiere conocer el DNI, nombre, teléfono, si tienen o no titulación y experiencia profesional, así como las clases que pueden impartir (preparación como monitor de aeróbic, *step*, *stretching*, etc.).
- De cada socio se quiere conocer el número de socio, nombre, dirección, teléfono, profesión y datos bancarios, así como las clases a las que asiste.
- El gimnasio dispone también de pistas de squash, de las que se quiere conocer el número de pista, ubicación y estado. Las pistas de squash pueden ser utilizadas por socios, y existe un servicio de reserva de pista (en una fecha y a una hora.)

Modelo entidad-relacion y paso a tablas



Creación de la base de datos y las tablas.

```
create database Ejercicio4
```

```
use Ejercicio4
```

```
create table Sala(
```

```
    IDSala int identity (100, 1) primary key,
```

```
    metros int,
```

```
    ubicacion varchar(40),
```

```
    tipo varchar(8) check (tipo='cardio' or tipo='general' or tipo='muscular')
```

```
)
```

```
insert into Sala values (50, 'area norte', 'cardio')
```

```
insert into Sala values (90, 'area sur', 'general')
```

```
insert into Sala values (60, 'area este', 'cardio')
```

```
insert into Sala values (30, 'area oeste', 'muscular')
```

```
insert into Sala values (50, 'area norte', 'muscular')
```

```
insert into Sala values (50, 'area sur', 'general')
```

```
create table Aparato(
```

```
    Cod_aparato int identity (100, 1) primary key,
```

```
    desc_aparato varchar (100),
```

```
    estado_aparato varchar(7) check (estado_aparato='bueno' or estado_aparato='malo'  
or estado_aparato='regular'),
```

```
    IDS int foreign key references Sala
```

```
)
```

```
insert into Aparato values('Prensa de piernas', 'bueno', 100)
```

```
insert into Aparato values('Dorsalera', 'malo', 101)
```

```
insert into Aparato values('Máquinas de femorales', 'regular', 105)
```



```
insert into Aparato values('Poleas cruzadas', 'regular', 100)
```

```
insert into Aparato values('Peck Deck', 'bueno', 104)
```

```
insert into Aparato values('Máquina de abductores', 'regular', 100)
```

```
create table Monitor(
```

```
    Cod_monitor int identity (10, 1) primary key,
```

```
    nombre varchar(40) not null,
```

```
    telefono int check (telefono>100000000),
```

```
    titulo varchar(2) check (titulo='Si' or titulo='No'),
```

```
    claseImpartida varchar(20)
```

```
)
```

```
insert into Monitor values ('Rosa Melpito', 685987642,'No', 'Danza')
```

```
insert into Monitor values ('Tomas Turbao', 654851356,'Si', 'Aerobic')
```

```
insert into Monitor values ('Aitor Menta', 615887023,'Si', 'Acrosport')
```

```
insert into Monitor values ('Juancho Talarga', 615887023,'No', 'Zumba')
```

```
create table Clase(
```

```
    Cod_clase int identity (50, 1) primary key,
```

```
    desc_clase varchar (40),
```

```
    IDS int foreign key references Sala,
```

```
    IDM int foreign key references Monitor,
```

```
    dia varchar(9),
```

```
    hora time,
```

```
)
```

```
insert into Clase values('Clase de acrobacias', 100, 10, 'Lunes', '12:00')
```

```
insert into Clase values('Clase de danza tradicional', 101, 12, 'Miercoles', '18:00')
```

```
insert into Clase values('Clase de gimnasia con musica', 102, 12, 'Viernes', '12:00')
```

Ejercicio 4 SQL - Relación 2

```
insert into Clase values('Clase de baile', 101, 13, 'Lunes', '18:00')
```

```
create table Socio(
```

```
    IDSocio int identity (1000, 1) primary key,
```

```
    nombre varchar(40) not null,
```

```
    direccion varchar(100),
```

```
    telefSocio int check (telefSocio>1000000000),
```

```
    profesion varchar(50),
```

```
    datos_bancarios int
```

```
)
```

```
insert into Socio values ('Mia Khalifa', 'C/Cervantes', 612358945, 'Actriz', null)
```

```
insert into Socio values ('Johnny Sins', 'C/Pericles', 694235681, 'Repartidor', null)
```

```
insert into Socio values ('Sasha Grey', 'C/Don Quijote', 666548235, 'Limpiadora', null)
```

```
insert into Socio values ('Torbe', 'C/Cristo del Repollo', 633475891, 'Traficante', null)
```

```
insert into Socio values ('Dunia Montenegro', 'C/Lucifer Salvador', 614726891, 'Profesora', null)
```

```
create table Asiste(
```

```
    CodigoClase int references Clase,
```

```
    CodigoSocio int references Socio,
```

```
    clase varchar(9),
```

```
    constraint PK_Asiste primary key(CodigoClase, CodigoSocio)
```

```
)
```

```
insert into Asiste values(51,1000,'Danza')
```

```
insert into Asiste values(53,1002,'Zumba')
```

```
insert into Asiste values(52,1004,'Aerobic')
```

```
insert into Asiste values(50,1001,'Acrosport')
```

```
-----  
create table Squash(  
    IDPista int identity (10, 1) primary key,  
    desc_pista varchar(100),  
    ubicacion varchar(40),  
    estado_squash varchar(7) check (estado_squash='bueno' or estado_squash='malo' or  
estado_squash='regular')  
)  
  
insert into Squash values('Pista de Squash numero 1', 'area norte', 'bueno')  
insert into Squash values('Pista de Squash numero 2', 'area sur', 'malo')  
insert into Squash values('Pista de Squash numero 3', 'area este', 'malo')  
insert into Squash values('Pista de Squash numero 4', 'area oeste', 'regular')
```

```
-----  
create table Reserva(  
    CodigoP int references Squash,  
    CodigoS int references Socio,  
    fecha date,  
    hora int,  
    constraint PK_Reserva primary key(CodigoP, CodigoS)  
)  
  
insert into Reserva values (10, 1004, '2018-02-24', 12)  
insert into Reserva values (10, 1000, '2018-02-25', 20)  
insert into Reserva values (10, 1003, '2018-02-24', 18)  
insert into Reserva values (13, 1002, '2018-02-27', 14)  
insert into Reserva values (12, 1004, '2018-02-28', 12)  
insert into Reserva values (13, 1001, '2018-02-24', 12)  
-----
```


Ejercicios.

--Mostrar el ID de las salas de cardio mayores o iguales a 50 metros cuadrados

```
select IDSala, metros  
  
from Sala  
  
where metros>=50  
  
and tipo='cardio'
```

--Mostrar los socios que no asisten a ninguna clase

```
select IDSocio, nombre from Socio  
  
where IDSocio not in (select CodigoSocio from Asiste)
```

--Mostrar los nombre, profesiones y telefonos de los socios que asisten a clase de Zumba

```
Select nombre, profesion, telefSocio from Socio, Asiste  
  
where IDSocio = CodigoSocio and clase = 'Zumba'
```

--Mostrar el nombre y el estado de los aparatos que están en la sala 100

```
select desc_aparato, estado_aparato from Aparato  
  
where IDS = 100
```

--Mostrar los socios que reservaron la pista 13 de Squash y cuando

```
select nombre, fecha from Socio, Reserva  
  
where CodigoP = 13 and IDSocio = CodigoS
```

--Mostrar los datos de los monitores que no tengan titulo o no impartan clase

```
select * from Monitor  
  
where titulo = 'no' or Cod_monitor not in (select idm from Clase)
```

Ejercicio 4 SQL - Relación 2

--Funcion que devuelva los datos de los aparatos que están en las salas del tipo que especifiquemos

```
create function fTipoAparatos (@nombre varchar(40))
```

```
returns table
```

```
as return select * from Aparato
```

```
where IDS in (select IDSala from sala
```

```
where tipo like @nombre)
```

```
select * from fTipoAparatos('muscular')
```