

Nombre

Profesor

Institución

Fecha:

Materia:

Cupo:

Nota:

1 MER

socio
id-socio (PK)
nombre
apellido
fecha-nacimiento

Disciplina
id-disciplina (PK)
nombre
nivel
categoria

Monitor
id-monitor (PK)
nombre
especialidad
años-experiencia

sala
id-sala (PK)
nombre
Capacidad
disponibilidad
ubicacion

Inscripción
id-socio (FK)
id-disciplina (FK)
fecha-inicio
horario

asignacion-monitor
id-monitor (FK) →
id-disciplina (FK)
fecha-asignacion
turno

reserva-sala
id-sala (FK)
id-disciplina (FK)
fecha-reserva
hora-inicio
hora-fin

Create database Club deportivo

Use club deportivo

Create Table Socio (

id_socio int Primary Key,
nombre varchar (50),
apellido varchar (50),
fecha_nacimiento Date,
);

Create Table Disciplinas

id_disciplina int Primary Key,
nombre varchar (50),
Categoría varchar (50),
nivel varchar (20)
);

Create Table Monitores

id_monitor int Primary Key,
nombre varchar (50),
especialidad varchar (100),
años_experiencia int
);

Create Table sala (

id_sala int Primary Key,
nombre varchar (50),
capacidad int,
ubicacion varchar (100)
);

Create)

id-cliente

id-disciplina

fecha_nacimiento

horario

);

Create)

id-monitor

id-disciplina

fecha_nacimiento

);

Create)

id-sala

id-monitores

);

Create)

id-cliente

id-disciplina

fecha_nacimiento

);

Create)

id-cliente

id-disciplina

fecha_nacimiento

);

Create)

id-cliente

id-disciplina

fecha_nacimiento

);

```
create table inscripcionc
id-cliente int foreign key cliente (id-cliente),
id-disciplina int foreign key disciplina (id-disciplina),
fecha-inicio Date,
horario Time
);
create table asignacion-monitorc
id-monitor int foreign key monitor (id-monitor),
id-disciplina int foreign key disciplina (id-disciplina),
fecha-asignacion Date,
turno varchar (20)
);
create table reserva-sala
id-sala int foreign key sala (id-sala),
id-disciplina int foreign key disciplina (id-disciplina),
hora-inicio Time,
hora-fin Time
);
```

```
insert into socio values
(1,'andres','garcia','1990-06-20','activo');
```

```
insert into disciplina values
(1,'spinning','cardio','intermedio');
```

```
insert into monitor values
```

```
(1,'Laura Perez','spinning',5);
```

```
insert into sala values
```

```
(1,'sala 1',30,'disponible','segundo piso'));
```

```
insert into inscripcion values
```

```
(1,1,'2024-01-10','18:00')).
```

Insert into asignacion-monitor values
(1,1,'2024-01-01','mañana'));

Insert into reserva-sala values
(1,1,'2024-06-25','8:00','10:00'));

Update socio set apellido = 'Perez' where id-socio = 1;

Update socio set nombre = 'santiago' where id-socio = 1;

Update socio set fecha-nacimiento = '2007-05-17' where id-socio = 1;

Update disciplina set nivel = 'difícil' where id-disciplina = 1;

Update disciplina set categoria = 'Fuerza' where id-disciplina = 1;

Update monitor set nombre = 'carlos' where id-monitor = 1;

Update monitor set especialidad = 'pesas' where id-monitor = 1;

Update monitor set años-experiencia = 10 where id-monitor = 1;

Select * from socio;

Select nombre from socio where id-socio = 1;

Select nombre from socio where id-socio != 1;

Select id-socio from Inscripcion where estatus = 'activo';

;

Select s.nombre , d.nombre from socios

Join inscripcion I on s.id-socio = I.id-socio

Join disciplina D on I.id-disciplina = D.id-disciplina;

Select nombre from socio where fecha-nacimiento < '1995-01-01';

Nombre _____
Profesor _____ Materia _____

Select * from disciplina;

Select nombre from disciplina where categoria = 'cardio';

Select nombre from disciplina where id-disciplina in (

Select id-disciplina from inscripcion group by id-disciplina
having count(*) > 1);

Select d.nombre, m.nombre from disciplina d
Join asignacion-monitor am on d.id-disciplina = am.id-disciplina
Join monitor m On am.id-monitor = m.id-monitor;

Select nombre from disciplina where categoria = 'intermedio';

Select * From monitors;

Select nombre from monitor where especialidad = 'spinning';

Select nombre from monitor where id-monitor in (

Select id-monitor from asignacion-monitor where achiote = 1);

Select m.nombre, d.nombre from monitor m
Join asignacion-monitor am on m.id-monitor = am.id-monitor
Join disciplina d on am.id-disciplina = d.id-disciplina;

Select nombre from monitor where años-experiencia >= 3;

Select * from sala;

Select nombre from sala where disponibilidad = 'disponible';

Select nombre from sala where capacidad > 20;

Select s.nombre, d.nombre from sala s

Join reserva_sala rs on s.id-sala = rs.id-sala

Join disciplina d on rs.id-disciplina = d.id-disciplina;

Select nombre from sala where ubicacion = 'segundo piso';

Select * from inscripcion

Select id-socio from inscripcion where estatus = 'activo';

Select id-disciplina from inscripcion where horario = '18:00';

Select s.nombre, d.nombre from inscripcion i

Join socio s on i.id-socio = s.id-socio

Join disciplina d on i.id-disciplina = d.id-disciplina;

Select count(*) from inscripcion where estatus = 'activo';

Select * from asignacion_monitor;

Select id-monitor from asignacion_monitor where id-monitor = 1;

Select id-disciplina from asignacion_monitor where turno = 'mañana';

Select m.nombre, d.nombre from asignacion_monitor am

Join monitor m on am.id-monitor = m.id-monitor

Join disciplina d on am.id-disciplina = d.id-disciplina;

Select count(*) from asignacion_monitor where activo = 'si';

select * from reserva_sala;
 select id_sala from reserva_sala where fecha_reserva = '2024-06-15';
 select id_disciplina from reserva_sala where horario_fino = '08:00';
 select sa.nombre, d.nombre from reserva_sala
 join sala sa on rs.id_sala = sa.id_sala
 join disciplina d on ss.id_disciplina = d.id_disciplina;
 select count(*) from reserva_sala where hora_fino > '10:00';

delete from socio where id_socio = 1;
 delete from inscripcion where id_socio = 1 and id_disciplina = 1;

Alter table cliente add telefono varchar(70);

diccionario de datos

nombre: socio

descripcion: almacena informacion del socio

clave	campo	tipo de dato	tamaño	descripcion
PK	socio id_socio	int		identificador de socio
	socio nombre	varchar	100	nombre del socio
	socio apellido	varchar	100	apellido del socio
	socio fecha_nacimiento	Date		fecha nacimiento del socio

nombre: disciplina

descripcion: almacena informacion de la disciplina

clave	campo	tipo de dato	tamaño	descripcion
PK	disciplina id_disciplina	int		identificador de disciplina
	disciplina nombre	varchar	100	nombre de la disciplina
	disciplina nivel	int		nivel de la disciplina
	disciplina categoria	varchar	100	categoria de la disciplina

nombre: monitor

descripcion: almacena informacion del monitor

clave campo	tipo de dato	largo	descripcion
PK monitor	int		identificador de monitor
id-monitor	varchar	100	nombre del monitor
monitor	varchar	100	especialidad del monitor
nombre	varchar	100	años de experiencia del monitor
monitpur	int		especialidad del monitor
especialidad	int		especialidad del monitor
monitor	varchar	100	nombre del monitor
expresionista	varchar	100	especialidad del monitor

nombre: sala

descripcion: almacena la informacion de la sala

clave campo	tipo de dato	largo	descripcion
PK sala	int		identificador de sala
idsala	varchar	100	nombre de la sala
sala	int		capacidad de la sala
nombre	varchar	100	ubicacion de sala
sala	int		capacidad de la sala
capacidad	varchar	100	ubicacion de sala
sala	varchar	100	ubicacion de sala
ubicacion	varchar	100	ubicacion de sala

nombre: inscripcion

descripcion: tabla pivote entre socio y disciplina

clave campo	tipo de dato	largo	descripcion
FK inscripcion	int		identificador de socio
id-socio	int		identificador de disciplina
FK inscripcion	int		fecha que inicia la inscripcion
id-disciplina	date		hora de inscripcion
inscripcion	time		
fechainicio			
horario			

nombre: asignacion-monitor

descripcion: tabla pivote entre disciplina y monitor

clave campo	tipo de dato	largo	descripcion
FK asignacion-monitor	int		identificador de monitor
id-monitor	int		identificador de disciplina
FK asignacion-monitor	int		fecha asignacion
id-disciplina	date		turno asignado
asignacion-monitor	varchar	100	
fecha-asignacion			
asignacion-monitor			
turno			

nombre: reserva-sala

descripción: tabla pivotee entre disciplina y sala

clave	campo	tipo de dato	tamaño	descripción
fk	reserva-sala id_sala	int		identificador de sala
fk	reserva-sala id_disciplina	int		identificador de disciplina
	reserva-sala fecha_reserva	Date		fecha de reserva
	reserva-sala hora_inicio	time		hora inicio de la reserva
	reserva-sala hora_fin	time		hora que acaba la reserva



Create database hospital

use hospital

Create Table paciente (

id-paciente int primary key,

nombre varchar (100),

apellido varchar (100),

fecha-nacimiento date,

estado varchar (20)

);

```
Create Table medico (
    id-medico int primary key,
    nombre varchar (100),
    especialidad varchar (100),
    años_experiencia int,
    estado varchar (20)
);
```

```
Create Table sala (
    id-sala int primary key,
    numero varchar (20),
    tipo varchar (30),
    capacidad int,
    disponible boolean
);
```

```
Create Table tratamiento (
    id-tratamiento int primary key,
    nombre varchar (100),
    descripción varchar (100),
    duracion_dias int,
    estado varchar (20)
);
```

```
Create Table consulta
```

```
id-paciente int foreign key reference paciente (id-paciente),
id-medico int foreign key reference medico (id-medico),
fecha_consulta date,
diagnóstico varchar (100),
estado varchar (100)
```

```
);
```

```
Create Table
id-Paciente 1
id-tratamiento
fecha_inicial
Fecha_final
resultado
);

Create Table
id-sala 1
id-paciente
fecha_inicial
fecha_final
estad0
);
```

```
Insert
(1, 'A', 'C', '2023-01-01', '2023-01-15', 'Disponible')
```

```
Insert
(1, 'A', 'C', '2023-01-01', '2023-01-15', 'Disponible')
```

```
Insert
(1, 'A', 'C', '2023-01-01', '2023-01-15', 'Disponible')
```

```
Insert
(1, 'A', 'C', '2023-01-01', '2023-01-15', 'Disponible')
```

Create table paciente

id-paciente int foreign key references paciente (id-paciente),
id-tratamiento int foreign key references tratamiento (id-tratamiento),
fecha-inicio Date,
Fecha-fin Date,
resultado varchar(100)

;

Create table asignacion-sala

id-sala int foreign key references sala (id-sala),
id-paciente int foreign key references paciente (id-paciente),
fecha-ingreso Date,

fecha-salida Date,

estado varchar(20);

;

Insert into paciente values

(1, 'ana', 'rojas', '1990-05-12', 'activo');

Insert into medico values

(1, 'Dr. Pedro', 'cardiologico', 15, 'Activo');

Insert into sala values

(1, 'U101', 'urgencias', 4, true);

Insert into tratamiento values

(1, 'fisioterapia', 'Terapia Postoperatoria', 10, 'activo');

Insert into consulta values

(1, 1, '2023-06-10', 'hipertension leve', 'alendriola');

Insert into aplicacion-tratamiento values

(1, 1, '2023-06-11', '2023-06-21', 'mejoria');

Insert into asignacion-salida values ('2023-06-10', '2023-06-15', 'Finalizado');
Update paciente set estado = 'egresado' where id-paciente = 1;
Update tratamiento set estado = 'inactivo' where id-tratamiento = 1;
Update consulta set estado = 'en seguimiento' where id-paciente = 1 AND id-medico = 1 AND fecha-consulta = '2023-06-10';
Update asignacion-salida set fecha-salida = '2023-06-14' where id-consulta = 1 and id-paciente = 1;
Select * from paciente;
Select nombre from paciente where estado = 'activo';
Select nombre from paciente where fechahabilitacion < '1990-01-01';
Select p.nombre, m.nombre from paciente p
Join consulta c on p.id-paciente = c.id-paciente
Join medico m on c.id-paciente = m.id-paciente;
Select count(*) from paciente where estado = 'egresado';
Select * from medico;
Select nombre from medico where especialidad = 'cardiologico';
Select nombre from medico where años-experiencia > 10;
Select m.nombre, p.nombre from medico m
Join consulta c on m.id-medico = c.id-medico
Join paciente p on c.id-paciente = p.id-paciente;
Select count(*) from medico where estado = 'Activo';

Select * from sala;

Select numero from sala where tipo = 'urgencias';

Select numero from sala where disponibilidad = true;

Select s.numero, p.nombre from salas
Join asignacionSalas a on s.id_sala = a.id_sala
Join paciente p on a.id_paciente = p.id_paciente;

Select count(*) from sala where capacidad >= 3;

Select * from tratamientos;

Select nombre from tratamiento where estado = 'activo';

Select nombre from tratamiento where descripcion like '%levar';

Select t.nombre, p.nombre from tratamiento t
Join aplicacion_tratamientos at on t.id_tratamiento = at.id_tratamiento
Join paciente p on at.id_paciente = p.id_paciente;

Select count(*) from tratamiento;

Select 1 from consulta;

Select id_medico from consulta where estadia = 'atendida';

Select fecha_consulta from consulta where diagnostico like '%levar';

Select c.fecha_consulta, p.nombre, m.nombre from consulta c
Join paciente p on c.id_paciente = p.id_paciente
Join medico m on c.id_medico = m.id_medico;

Select * from aplicacion_tratamiento;

Select id_tratamiento from aplicacion_tratamiento where resultado = 'negativo';

Select fecha_fin from aplicacion_tratamiento where fecha_inicio > '2023-06-10' ;

Select at.fecha_inicio, p.nombre, t.nombre from aplicacion_tratamiento at;

Join paciente p on at.id_paciente = p.id_paciente

Join tratamiento t on at.id_tratamiento = t.id_tratamiento;

Select count(*) from aplicacion_tratamientos;

Select * from asignacion_sala;

Select id_paciente from asignacion_sala where estado = 'finalizado';

Select fecha_salida from asignacion_sala where fechasingreso = '2023-06-10';

Select s.numero, a.nombre from asignacion_sala a,

Join sala s on a.id_sala = s.id_sala

Join paciente p on a.id_paciente = p.id_paciente;

Alter table paciente add correo varchar(100);

nombre: Paciente

descripcion: almacena informacion de paciente

clave	campo	tipo de dato	tamaño	descripcion
PK	Paciente	int		identificador de paciente
	id_paciente	varchar	100	nombre del paciente
	nombre	varchar	100	apellido del paciente
	apellido	date		fecha de nacimiento del paciente
	paciente	varchar	100	correo del paciente
	Fecha_nacimiento			
	paciente_estado			

nombre: medico

descripcion: almacena informacion de medico

clave	campo	tipo de dato	tamaño	descripcion
PK	medico id-medico	int		identificador de medico
	medico nombre	varchar	100	nombre del medico
	medico especialidad	varchar	100	especialidad del medico
	medico años-experiencia	int		años de experiencia del medico
	medico estado	varchar	100	estado del medico

nombre: sala

descripcion: almacena informacion de sala

clave	campo	tipo de dato	tamaño	descripcion
PK	sala id-sala	int		identificador de sala
	sala	varchar	100	numero de la sala
	numero	varchar	100	
	sala	varchar	100	tipo de sala
	tipo	int		
	sala	int		capacidad de sala
	capacidad	varchar	100	
	sala	varchar	100	disponibilidad de la sala
	disponible			

nombre: tratamiento

descripcion: almacena informacion de tratamiento

clave	campo	tipo de dato	tamaño	descripcion
PK	tratamiento id-tratamiento	int		identificador de tratamiento
	tratamiento	varchar	100	nombre del tratamiento
	nombre	varchar	100	descripcion del tratamiento
	tratamiento	date		duracion del tratamiento
	descripcion	varchar	100	estado del tratamiento
	tratamiento			
	duracion-dias			
	tratamiento			
	estado			

nombre: consulta

descripción: tabla pivote entre paciente y medico

clave	campo	tipo de dato	tamaño	descripción
FK	consulta	int		identificador de paciente
FK	id_Paciente	int		identificador de medico
	consultad			fecha de consulta
	id_medico			diagnostico de consulta
	Consulta	Date		
	Consulta	varchar	100	
	diagnostico	varchar	20	estado de consulta
	consultad			
	estadod			

nombre: aplicacion_tratamiento

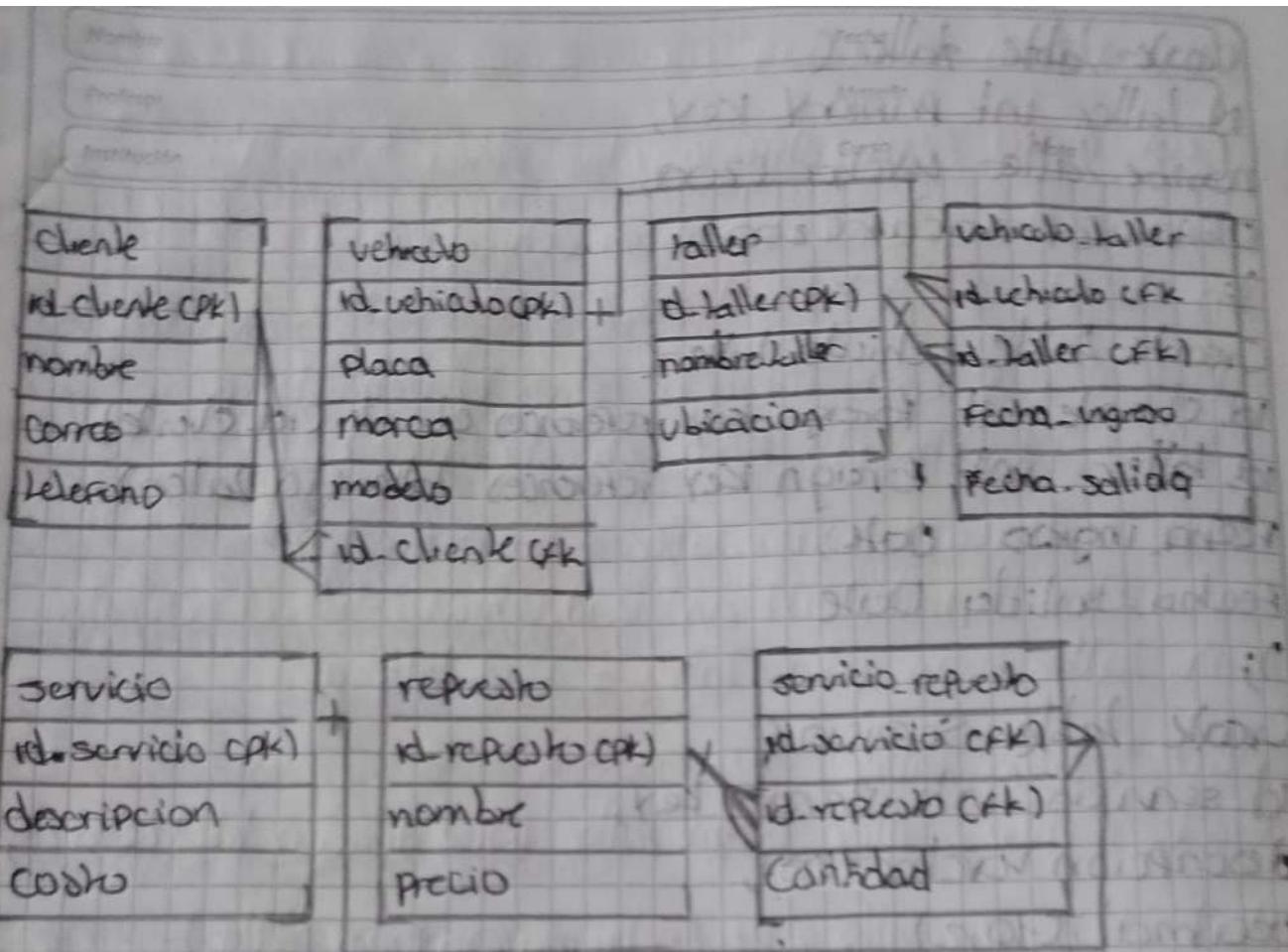
descripción: tabla pivote entre paciente y tratamiento

clave	campo	tipo de dato	tamaño	descripción
FK	aplicacion_tratamiento	int		identificador de paciente
FK	id_Paciente	int		identificador de tratamiento
FK	aplicacion_tratamiento	int		fecha inicio del tratamiento
	id_tratamiento			fecha fin del tratamiento
	aplicacion_tratamiento	Date		
	fecha_inicio	Date		
	aplicacion_tratamiento	varchar	100	estado del paciente
	fecha_fin			
	aplicacion_tratamiento			
	resultado	varchar		

nombre: asignacion_sala

descripción: tabla pivote entre sala y paciente

clave	campo	tipo de dato	tamaño	descripción
FK	asignacion_sala	int		identificador de sala
FK	id_sala			
FK	asignacion_sala	int		identificador de paciente
FK	id_Paciente			
	asignacion_sala	Date		fecha de ingreso
	fecha_ingreso			
	asignacion_sala	Date		fecha de salida
	fecha_salida			
	asignacion_sala	varchar	100	estado del paciente
	resultado			



Create database taller;

Use taller;

Create table Cliente C

id_cliente int primary key,
nombre varchar (100),
correo varchar (100),
telefono varchar (100)

;

Create table Vehiculo C

id_vehiculo int primary key
placa varchar (10),
marca varchar (20),
modelo varchar (50),
d_cliente int foreign key references Cliente (id_cliente)

Create table taller (

id_taller int primary key,
nombre_taller varchar(200),
ubicacion varchar(200),
);

Create table vehiculo_taller (

id_cliente int foreign key references cliente (id_cliente),
id_taller int foreign key references taller (id_taller),
fecha_ingreso Date,
fecha_salida Date
);

Create table servicio (

id_servicio int primary key,
descripcion varchar(100),
decimal (10,2)

);

Create table request (

id_request int primary key,
nombre varchar(100),
precio decimal (10,2)

);

Create table servicio_request (

id_servicio int foreign key references servicio (id_servicio),
id_request int foreign key references request (id_request),
cantidad int

);

insert into cliente values

(1, 'carlos perez', 'carlos@gmail.com', '300123456'),
(2, 'laura gomez', 'laura@gmail.com', '3001700');

insert into vehiculo values

(1, 'ABC123', 'toyota', 'corolla', 2),
(2, 'X123456', 'mazda', '3', 2);

insert into taller values

(1, 'taller Motor', 'cra 10 # 23-45'),
(2, 'auto fix', 'cra 50 # 15-30');

insert into vehiculo-taller values

(1, 1, '2025-06-01', '2025-06-05');

insert into servicio values

(1, 'cambio de aceite', 80000),
(2, 'alineacion y balanceo', 120000);

insert into repuesto values

(1, 'filto de aceite', 35000),

(2, 'llanta rin 15', 180000);

insert into servicio-repuesto values

(1, 1, 1);

select * from Cliente;

select nombre from Cliente where correo like '%@gmail.com';

select c.nombre, u.precio from Cliente c

join vehiculo v on c.id-cliente = v.id-cliente;

select nombre from Cliente where id-cliente in (

select id-cliente from Cliente group by id-cliente having count(*) >

1);

Select * from vehiculo;

Select placa from vehiculo where placa = 'mazda';

Select v.placa, c.nombre from vehiculo v;

Join cliente c on v.id-cliente = c.id-cliente;

Select placa from Vehiculo where id-vehiculo in (

select id-vehiculo from Vehiculo-Taller);

Select v.placa, t.nombre-taller from Vehiculo v;

Join Vehiculo-Taller vt on v.id-vehiculo = vt.id-vehiculo;

Join taller t on vt.id-taller = t.id-taller;

Select * from taller;

Select nombre-taller from taller where dirección like '%cra%';

Select t.nombre-taller, vt.Fecha-ingreso from taller t;

Join Vehiculo-Taller vt on t.id-taller = vt.id-taller;

Select nombre-taller from taller where id-taller in (

Select id-taller from Vehiculo-Taller

);

Select * from Vehiculo-Taller;

Select * from Vehiculo-Taller where Fecha-salida is null;

Select vt.Fecha-ingreso, v.placa from Vehiculo-Taller vt;

Join Vehiculo v on vt.id-vehiculo = v.id-vehiculo;

Select vt.Fecha-ingreso, c.nombre from Vehiculo-Taller vt;

Join Vehiculo v on vt.id-vehiculo = v.id-vehiculo;

Join cliente c on v.id-cliente = c.id-cliente

Nombre

Profesor

Institución

Select * from servicio;

Select description from servicio where costo > 10000;

Select s.descripcion, sr.cantidad from servicio s,

Join servicio-reuesto sr on s.id-servicio = sr.id-servicio;

Select id-servicio from servicio where id-servicio in

Select id-servicio from servicio-reuesto

);

Select * from reuesto;

Select nombre from reuesto where precio > 5000;

Select r.nombre from reuestor

Join servicio-reuesto sr on r.id-reuesto = sr.id-reuesto

Join servicios s on sr.id-servicio = s.id-servicio

Where s.descripcion like '%accidente%'

Select id-reuesto from reuesto

Where id-reuesto in (Select id-reuesto from servicio-reuesto),

Select * from servicio-reuesto;

Select * from servicio-reuesto where cantidad >= 1;

Select sr.cantidad, r.nombre from servicio-reuesto sr

Join reuesto r ON sr.id-reuesto = r.id-reuesto

Where sr.id-servicio = 1;

Select id-servicio from servicio-reuesto

Where id-reuesto = (Select id-reuesto from reuesto where nombre = 'kilómetro');

delete from cliente where id-cliente = 2;
 delete from vehiculo where id-vehiculo = 2;
 delete from taller where id-taller = 2;
 delete from servicio where id-servicio = 2;

Update cliente set telefono = '300000' where id-cliente = 1;
 Update vehiculo set modelo = 'Yaris' where id-vehiculo = 1;
 Update taller set direccion = 'AV Siempreviva' where id-taller = 1;
 Update servicio set costo = 90000 where id-servicio = 1;

nombre: cliente

descripcion: almacena informacion del cliente

Clave	Campo	Tipo de dato	Tamaño	descripcion
PK	cliente	int		identificador del cliente
	id-cliente	int		nombre del cliente
	cliente	varchar	100	correo del cliente
	correo	varchar	100	telefono del cliente
	cliente	varchar	100	telefono del cliente
	telefono	varchar	100	

nombre: vehiculo

descripcion: almacena informacion del vehiculo

Clave	Campo	Tipo de dato	Tamaño	descripcion
PK	vehiculo	int		identificador del vehiculo
	id-vehiculo	int		placa del vehiculo
	vehiculo	varchar	100	marca del vehiculo
	Placa	varchar	100	
	vehiculo	varchar	100	modelo del vehiculo
	marca	varchar	100	
	vehiculo	varchar	100	identificador de cliente
FK	vehiculo	int		
	id-cliente	int		

nombre: taller

descripción: almacena información de taller

clave	campo	tipo de dato	tamaño	descripción
PK	id_taller	int		identificador de taller
	id_taller	varchar	100	nombre del taller
	nombre_taller	varchar	100	ubicación del taller
	ubicacion			

nombre: vehiculo_taller

descripción: tabla pivote entre vehículo y taller

clave	campo	tipo de dato	tamaño	descripción
fk	vehiculo_taller	int		identificador de vehículo
	id_vehiculo			
fk	vehiculo_taller	int		identificador de taller
	id_taller			
	vehiculo_taller	date		fecha que ingresa el vehículo
	fecha_ingreso	date		fecha que sale el vehículo
	vehiculo_taller			
	fecha_salida			

nombre: servicio

descripción: almacena la información de servicio

clave	campo	tipo de dato	tamaño	descripción
PK	id_servicio	int		identificador de servicio
	descripcion	varchar		descripción del servicio
	costo	decimal	10,2	costo del servicio

nombre: repto

descripción: almacena la información de repto

clave	campo	tipo de dato	tamaño	descripción
PK	repto	int		identificador de repto
	id_repto			
	repto	varchar	100	nombre del repto
	nombre	decimal	10,2	precio del repto
	repto			
	precio			

Profesor

Curso

Nota

Institución

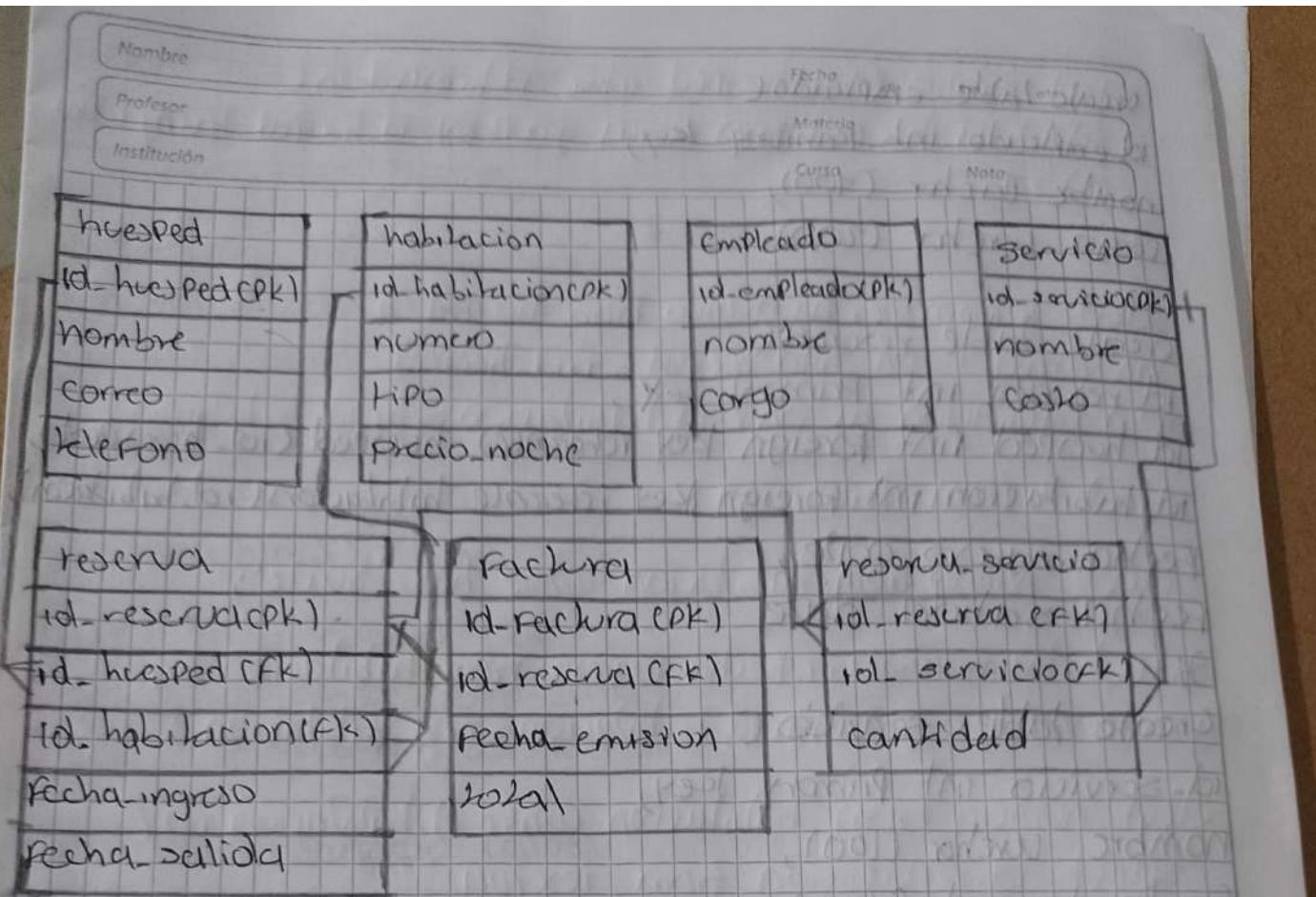
Alumno

Servicio

Nombre: servicio_reuesto

descripción: tabla que une entre servicio y repuesto

Clave	Campo	Tipo de dato	Tamaño	Descripción
FK	servicio_reuesto	int		identificador de servicio
FK	id_servicio	int		identificador de servicio
FK	servicio_reuesto	int		identificador de repuesto
	id_reuesto	int		cantidad de repuestos
	Conjunto			



Create database hotel;

USE hotel;

CREATE TABLE huesped (

id_huesped int Primary key,

nombre varchar (80),

correo varchar (100),

telefono varchar (20)

);

Create Table habitacion (

id_habitacion int Primary key,

numero varchar (20),

tipo varchar (100),

precio_noche decimal (10,2)

);

```
create table empleado (
    id_empleado int primary key,
    nombre varchar (100),
    cargo varchar (200)
);
```

```
create table reserva (
    id_reserva int primary key,
    id_huesped int foreign key references huesped (id_huesped),
    id_habitacion int foreign key references habitacion (id_habitacion),
    Fecha_ingreso date,
    Fecha_salida date
);
```

```
create table servicio (
    id_servicio int primary key,
    nombre varchar (100),
    costo decimal (10,2)
);
```

```
create table factura (
    id_factura int primary key,
    id_reserva int foreign key references reserva (id_reserva),
    fecha_emision date,
    total decimal (10,2)
);
```

```
create table reserva_servicio (
    id_reserva int foreign key references reserva (id_reserva),
    id_servicio int foreign key references servicio (id_servicio),
    cantidad int
);
```

```
insert into
    empleado
values
    (1, 'Juan'),  

    (2, 'Monica')
```

```
insert into
    reserva
values
    (1, '101'),  

    (2, '102')
```

```
insert into
    servicio
values
    (1, 'Carlos'),  

    (2, 'Laura')
```

```
Insert
    (1,1,1),
    (2,2,2)
```

```
Insert
    (1,0),
    (2,5)
```

```
Insert
    (1,1),
    (2,2)
```

```
Insert
    (1,1),
    (2,2)
```

```
(1,1)
```

insert into huSped values

(1, 'Juan Lopez', 'juan@gmail.com', 'solitario'),
(2, 'Maria Perez', 'Maria@gmail.com', 'suite')

insert into habitacion values

(1, '101', 'double', 20000),
(2, '102', 'suite', 30000)

insert into empleado values

(1, 'Carlos Mendez', 'receptionista'),
(2, 'Carmen Jones', 'limpieza')

insert into reserva values

(1, 1, 1, '2025-06-10', '2025-06-12'),
(2, 2, 2, '2025-06-11', '2025-06-15')

insert into servicio values

(1, 'desayuno buffet', 30000),
(2, 'SPA', 80000)

insert into factura values

(1, 1, '2025-06-12', 46000),
(2, 2, '2025-06-13', 51000)

insert into reserva-servicio values

(1, 1, 2),
(2, 2, 1)

Select * from huSped;

Select nombre from huSped where correo like '%@gmail.com';

Select h.nombre, r.fecha-ingreso from huSped h
Join reservar on h.id-huesped = r.id-huesped;

Nombre
Profesor

```
Select nombre from huésped where id_huesped in C
Select id_huesped from reservas group by id_huesped having count(*) > 1
);
3;

Select h.nombre, ha.numero from huésped h
Join reserva r on h.id_huesped = r.id_huesped
Join habitación ha on r.id_habitación = ha.id_habitación;
2;

Select * from habitación;
2;

Select numero from habitación where tipo = 'suite';
2;

Select ha.numero, r.Fecha_Ingreso from habitación ha
Join reserva r on ha.id_habitación = r.id_habitación;
2;

Select numero from habitación where id_habitación in (
Select id_habitación from reserva
);
2;

Select * from empleado;
2;

Select nombre from empleado where cargo = 'recepcionista';
2;

Select e.nombre from empleado e where e.id_empleado in (
Select id_empleado from empleado where cargo like '%limpieza'
);
2;

Select nombre from empleado where empleado is not null;
2;

Select nombre, cargo from empleado where cargo in ('recepcionista',
'limpieza');
2;
```

Nombre

Fecha

Profesor

Materia

Institución

Ciudad

Nº de
teléfono

Select * from reserva;

Select * from reserva where fecha-salida='2025-06-13';

Select r.id-reserva, h.nombre from reserva r

Join huiped d on r.id-huiped = d.id-huiped;

Select r.id-reserva, ha.numero from reserva r;

Select habitacion ha on r.id-habitacion=ha.id-habitacion;

Select * from servicio;

Select nombre from servicio where costo > 5000;

Select s.nombre, rs.Cantidad from servicio s

Join reserva-servicio rs ON s.id-servicio=rs.id-servicio;

Select id-servicio from servicio where id-servicio in (

Select id-servicio from reserva-servicio

);

Select * from Factura;

Select total from factura where total > 50000;

Select f.total, r.fecha-ingreso from Factura f

Join reserva r ON f.id-reserva = r.id-reserva;

Select id-factura from factura where id-reserva = 1;

Select f.total, h.nombre from Factura f

Join reserva r ON f.id-reserva = r.id-reserva

Join huiped h ON r.id-huiped = h.id-huiped;

Select * from reservaservicio;
 Select * from reservaservicio where cantidad >= 1;
 Select rs.cantidad, s.nombre from reservaservicio rs
 Join servicio s on rs.id_servicio = s.id_servicio;
 Select id_servicio from reservaservicio where id_reserva = 2;
 Select s.nombre from reservaservicio rs
 Join servicio s on rs.id_servicio = s.id_servicio
 Where rs.id_reserva = 1;
 Delete from huésped where id_huésped = 2;
 Delete from habitación where id_habitación = 2;
 Delete from empleado where id_empleado = 2;
 Delete from reserva where id_reserva = 2;
 Delete from servicio where id_servicio = 2;

Update huésped set teléfono = '3000000' where id_huésped = 1;
 Update habitación set precio_noche = 10000 where id_habitación = 1;
 Update empleado set cargo = 'supervisor' where id_empleado = 1;

nombre: huésped

Descripción: almacena información del huésped

clave	campo	tipo de dato	tamaño	descripción
PK	id_huesped	int		identificador de huésped
	nombre_huesped	varchar	100	nombre del huésped
	correo_huesped	varchar	100	correo del huésped
	telefono	varchar	100	teléfono del huésped

nombre:	habitación
descripción:	almacena
clave	campo
PK	habitación
	id_habitación
	habitación
	nombre
	habitación
	tipo
	habitación
	precio_noche

nombre:	empleado
descripción:	almacena
clave	campo
PK	empleado
	id_empleado
	empleado
	nombre
	empleado
	cargo

nombre:	servicio
descripción:	
clave	campo
PK	servicio
	id_servicio
	servicio
	nombre
	servicio

nombre:	
descripción:	
clave	
PK	
FK	

nombre: habitacion

descripcion: almacena informacion de la habitacion

clave	campo	tipo de dato	tamano	descripcion
PK	habitacion	int		identificador de habitacion
	id-habitacion			
	habitacion	varchar	100	numero de la habitacion
	nombre	varchar	100	lno. de habitacion
	habitacion	varchar	100	lno. de habitacion
	tipo	decimal	10,2	precio por noche
	habitacion			
	precio_noche			

nombre: empleado

descripcion: almacena informacion del empleado

clave	campo	tipo de dato	tamano	descripcion
PK	empleado	int		identificador de empleado
	id-empleado			
	empleado	varchar	100	nombre del empleado
	nombre	varchar	100	cargo del empleado
	empleado			
	cargo			

nombre: servicio

descripcion: almacena informacion de servicio

clave	campo	tipo de dato	tamano	descripcion
PK	servicio	int		identificador de servicio
	id-servicio			
	servicio	varchar	100	nombre del servicio
	nombre	decimal	10,2	costo del servicio
	servicio			
	costo			

nombre: reserva

descripcion: tabla pivote entre huiped + habitacion

clave	campo	tipo de dato	tamano	descripcion
PK	reserva	int		identificador de reserva
	id-reserva			
FK	reserva	int		identificador de huiped
	id-huiped			
FK	reserva	int		identificador de habitacion
	id-habitacion			
	reserva	date		fecha de ingreso
	fecha_ingreso			
	reserva	date		fecha de salida
	fecha_salida			

nombre: factura

descripción: Almacena la información de factura

Clave	Campo	Tipo de dato	Tamaño	Descripción
PK	factura	int		Identificador de factura
	Id_Factura	int		Identificador de reserva
FK	Factura	int		
	Id_reserva			Fecha que se hizo la factura
	Factura	date		
	Fecha_emision			Precio total de la reserva
	Factura			
	Total	decimal	10,2	

nombre: reserva_servicio

descripción: Tabla puente entre reserva y servicio

Clave	Campo	Tipo de dato	Tamaño	Descripción
FK	reserva_servicio	salvo		Identificador de reserva
	Id_reserva	int		
FK	reserva_servicio	int		Identificador de servicio
	Id_servicio			
	reserva_servicio	int		Cantidad de reserva
	cantidad			

Nombre Santiago Gordo Ponce

Profesor

Institución

Fecha

Materia

Carrera

Nota

5MER

estudiante

id-estudiante (PK)

nombre

apellido

profesor

id-profesor (PK)

apellido

email

curso

id-curso (PK)

descripción

id-departamento (FK)

id-salon (FK)

departamento

id-departamento (PK)

nombre

salon

id-salon (PK)

nombre

estudiante-curso

id-estudiante (FK)

id-curso (FK)

profesor-curso

id-profesor (FK)

id-curso (FK)

Relaciones

1) estudiantes ↔ cursos = es de muchos a muchos mediante la tabla pivote estudiante-curso

- Un estudiante puede inscribirse en muchos cursos.
- Un curso puede tener muchos estudiantes.

2) profesor ↔ curso = la relación es de muchos a muchos por medio de la tabla pivote profesor-curso

- Un profesor puede dar varios cursos.
- Un curso puede ser dado por varios profesores.

3) curso → departamento = la relación es de muchos a uno.

- Cada curso pertenece a un solo departamento.
- Pero un departamento puede tener muchos cursos.

- 4) curso → salones: es de muchos a uno
• Cada curso se da en un solo salón
• Un salón puede ser usado para muchos cursos

Sentencias

Create database Universidad

Use Universidad

Create Table Estudiante (

id_estudiante int primary key,

nombre varchar (30),

apellido varchar (30)

);

Create Table Profesor (

id_profesor int primary key

apellido varchar (30),

email varchar (30)

);

Create Table Departamento (

id_departamento int primary key,

nombre varchar (100)

);

Create Table Salón (

id_salon int primary key

nombre varchar (100)

);

Create Table Curso (

id_curso int primary key,

descripcion varchar (200),

id_departamento int foreign key references Departamento (id_departamento),

id_salon int foreign key references Salón (id_salon)

1;
Create h
1d-estud

Nombre Santiago Gordo Perez

Profesor

Institución

Fecha

Materia

Curso

Nota

1;
Create table estudiante-CURSO
id-estudiante int,

id-Curso int,

Primary key (id-estudiante, id-curso),

foreign key (id-estudiante) references estudiante (id-estudiante),

foreign key (id-curso) references curso (id-curso)

);

Create table Profesor-CURSO(

id-Profesor int,

id-Curso int,

Primary key (id-Profesor, id-curso),

foreign key (id-Profesor) references Profesor (id-Profesor),

foreign key (id-curso) references curso (id-curso)

);

Insert into departamento values (1, 'ingenieria');

Insert into departamento values (2, 'Ciencias Sociales');

Insert into salon values (101, 'aula A');

Insert into salon values (102, 'Laboratoria 1');

Insert into CURSO values (1001, 'programacion I', 1, 101);

Insert into CURSO values (1002, 'sociedad', 2, 102);

Insert into estudiante values (1, 'ana', 'Perez');

Insert into estudiante values (2, 'luis', 'Gomez');

Insert into Profesor values (10, 'Rodriguez', 'rodriguez@gmail.com');

Insert into Profesor values (11, 'Martinez', 'mrtinez@gmail.com');

Insert into profesor-curso values (10, 1001);
Insert into profesor-curso values (11, 1002);

Insert into estudiante-curso values (1, 1001);
Insert into estudiante-curso values (2, 1002);

Select * from estudiante;

Select * from curso
Where id-departamento = 1;

Select * from profesor
Where apellido Like 'M%';

Select * from salon
Where nombre Like '%aula%';

Select descripcion from curso
Where id-curso > 1000;

Select e.nombre, e.apellido, c.descripcion As curso
From estudiante e
JOIN estudiante-curso ec ON e.id-estudiante = ec.id-estudiante
JOIN curso c ON ec.id-curso = c.id-curso

Select c.descripcion, d.nombre As departamento
From curso c
Join departamento d on c.id-departamento = d.id-departamento;

Select p.apellido, c.descripcion As curso
From profesor p

Join profesor-curso pc ON p.id-profesor = pc.id-profesor
Join curso c ON pc.id-curso = c.id-curso;

Nombre Santiago Gordo Pérez

Profesor

Institución

5

Select C.descripcion AS curso, S.nombre AS salon
From curso C

Join salon S ON C.id_curso = S.id_salon;

Select e.nombre, C.descripcion AS curso, P.apellidos
Profesor

From estudiante e

Join estudiante-curso ec ON e.id_estudiante = ec.id_estudiante

Join curso C ON ec.id_curso = C.id_curso

Join profesor-curso PC ON C.id_curso = PC.id_curso

Join profesor P ON PC.id_profesor = P.id_profesor

Select nombre From salones

Where id_salon IN C

Select id_salon From curso

Where descripcion Like '%programacion%'

;

Select nombre From Departamento

Where id_departamento IN C

Select distinct id_departamento From cursos

;

Select descripcion From curso

Where id_curso IN C

Select id_curso

From estudiante-curso

group By id_curso

Having Count(*) > 1

;

Select apellido from profesor
Where id_profesor in

Select PC.id_Profesor
From Profesor_Curso PC

Join curso C on PC.id_curso = C.id_curso
Join Departamento D on C.id_departamento = D.id_departamento
Join departamento on C.id_departamento = D.id_departamento

Select nombre, apellido from estudiante

Where id_estudiante in (

Select id_estudiante from estudiante_curso

Where id_curso = 1001

);

Drop table estudiante_curso;

Alter table profesor

Drop column email;

Diccionario de datos

nombre de la tabla: estudiante

descripcion: Entidad que almacena la informacion del estudiante

clave	campo	tipo de dato	tamaño	descripcion
PK	estudiante.id_estudiante	int		identificador de estudiante
	estudiante.nombre	varchar	30	nombre del estudiante
	estudiante.apellido	varchar	30	apellido del estudiante

nombre de la tabla: Profesor

descripcion: almacena informacion del profesor

clave	campo	tipo de dato	tamaño	descripcion
PK	profesor.id_Profesor	int		identificador el profesor
	profesor.apellido	varchar	30	apellido del profesor
	profesor.email	varchar	100	correo del profesor

Nombre: Santiago
Profesor:
Institucion:

nombre de la tabla

descripcion: adm

Clave campo

PK id_curso

curso

FK curso_id

FK curso_id

nombre de la

descripcion:

Clave campo

PK departamento

nombre

nombre de la

descripcion:

Clave campo

PK salon

salon

nombre de

descripcion:

Clave campo

FK id

FK id

Nombre: Santiago Gómez Pérez

Profesor:

Materia:

Fecha:

Materia:

Cupo:

Nota:

nombre de la tabla: curso

Descripción: almacena la información de los cursos

Clave	Campo	Tipo de dato	Tamaño	Descripción
PK	id-curso	int		identificador del curso
	curso_descripción	varchar	100	descripción acerca del curso
FK	curso_id-departamento	int		identificador de departamento
FK	curso_id-salon	int		identificador de salón

nombre de la tabla: departamento

Descripción: almacena la información de los departamentos

Clave	Campo	Tipo de dato	Tamaño	Descripción
PK	departamento_id-departamento	int		identificador del departamento
	nombre	varchar	100	nombre del departamento

nombre de la tabla: salon

Descripción: almacena información de los salones

Clave	Campo	Tipo de dato	Tamaño	Descripción
PK	salon_id-salon	int		identificador del salón
	salon_nombre	varchar	100	nombre del salón

nombre de la tabla: estudiante-curso

Descripción: tabla pivote entre estudiante + curso

Clave	Campo	Tipo de dato	Tamaño	Descripción
FK	id-estudiante	int		identificador de estudiante
FK	id-curso	int		identificador de curso

From pedido_producto

Group By id_pedido;

Asunto o resultado de la consulta es el nombre As empleado, P.nombre

nombre de la tabla: Profesor_Curso

descripción: tabla pivote entre profesor y curso

clave	campo	tipo de dato	tamaño	descripción
FK	id_profesor	int		identificador de profesor
FK	id_curso	int		identificador de curso