



Conceptos de Bases de Datos
Trabajo Práctico Obligatorio GRUPO 06 EQUIPO ALFA BUENA
MARAVILLA ONDA DINAMITA ESCUADRON LOBO

Dominio: Cine

Integrantes:

Joel David Jeckeln FAI-2830

Emiliano Lopez FAI-3357

Valentin Camusso FAI-3208

Julia Rossi FAI-2378

Lautaro Gonzalez FAI-2950

Ejercicio 1:

Dado el siguiente Modelo Relacional, construya el Modelo de Entidad Relación correspondiente marcando claramente Entidades, atributos, Relaciones, claves primarias, cardinalidades, entidades débiles, Generalización-Especialización

Primary Key: en negrita

Foreign Key: subrayado

PERSONA(tipoDoc, NroDocumento, apellido, nombre, fechaNacimiento, telefono, correoElectronico)

CLIENTE(tipoDoc, NroDocumento, situacionFiscal)

ENCARGADO(tipoDoc, NroDocumento, legajo, categoria)

ENTRADA(NumeroEntrada, tipoDocCliente, NroDocumentoCliente, NroAsiento, idSala, idFuncion, fechaEmision)

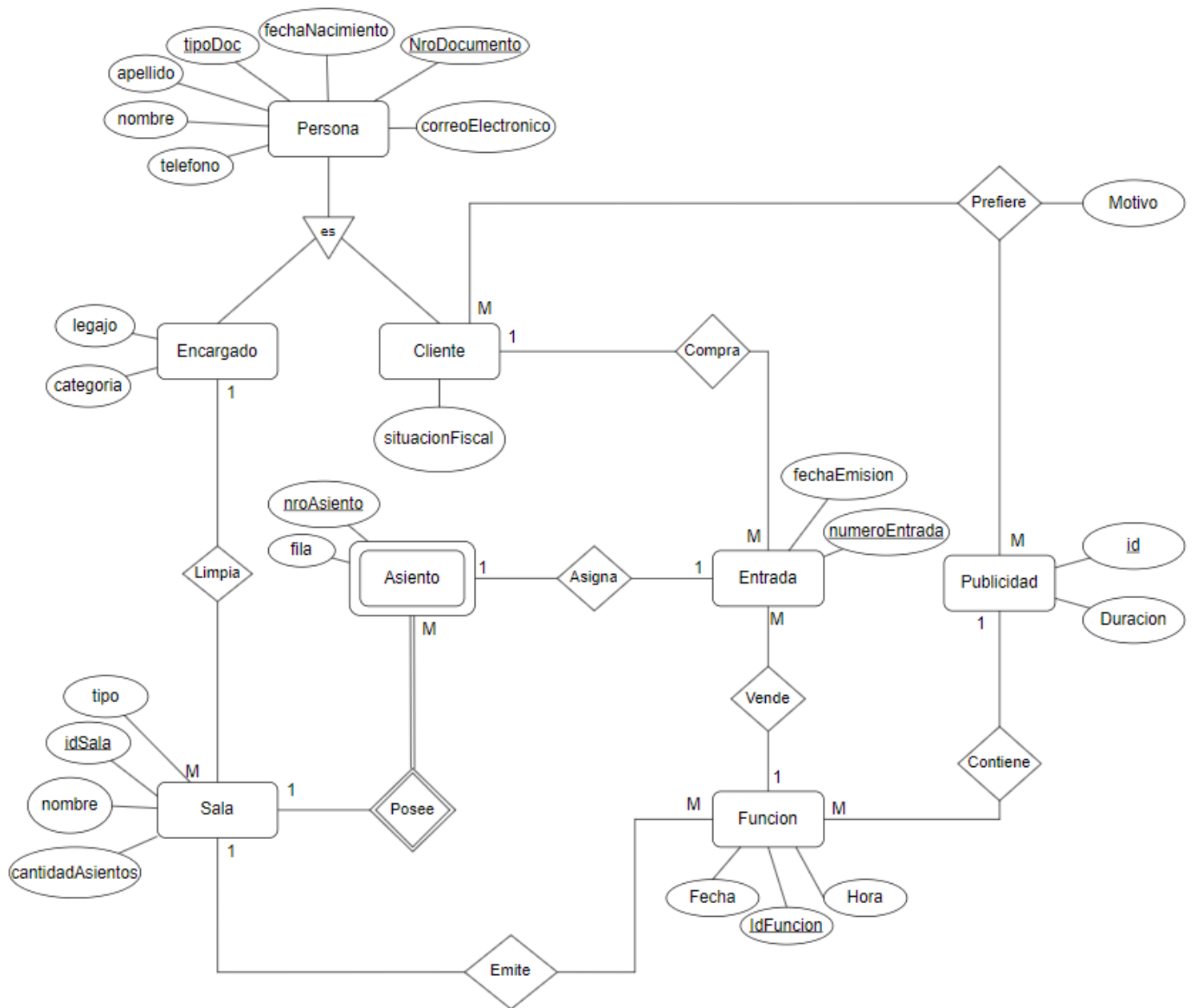
FUNCION(IdFuncion, Fecha, Hora, IdSala, idPublicidadInicio, idPublicidadFin)

PUBLICIDAD(id, Duracion)

SALA(idSala, nombre, cantidadAsientos, tipo, tipoDocEncargado, NroDocumentoEncargado)

ASIENTO(idSala, nroAsiento, fila)

PREFIERE(tipoDocCliente, NroDocumentoCliente, idPublicidad, Motivo)



Ejercicio 2: Realizar el DDL correspondiente al MR del ejercicio 1

```

CREATE TABLE persona (
    tipoDoc varchar(3),
    NroDocumento int(8),
    apellido varchar(20),
    nombre varchar(15),
    fechaNacimiento DATE,
    telefono int(15),
    correoElectronico varchar(30),
    PRIMARY KEY (tipoDoc, NroDocumento)
);

CREATE TABLE cliente (
    situacionFiscal varchar(10),
    tipoDoc varchar(3),
    NroDocumento int(8),

```

```

FOREIGN KEY (tipoDoc,NroDocumento) REFERENCES persona (tipoDoc,NroDocumento) ON
UPDATE CASCADE ON DELETE CASCADE,
PRIMARY KEY (tipoDoc, NroDocumento)
);

CREATE TABLE encargado(
    tipoDocEncargado varchar(3),
    NroDocumentoEncargado int(8),
    legajo varchar(15),
    categoria varchar(15),
    FOREIGN KEY (tipoDocEncargado, NroDocumentoEncargado) REFERENCES persona (tipoDoc,
    NroDocumento) ON UPDATE CASCADE ON DELETE CASCADE,
    PRIMARY KEY (tipoDocEncargado, NroDocumentoEncargado)
);

CREATE TABLE publicidad (
    id int(11) PRIMARY KEY,
    duracion TIME
);

CREATE TABLE sala(
    idSala int(5) PRIMARY KEY,
    nombre varchar(20),
    cantidadAsientos int(3),
    tipo varchar(15),
    tipoDocEncargado varchar(3),
    NroDocumentoEncargado int(8),
    FOREIGN KEY (tipoDocEncargado , NroDocumentoEncargado ) REFERENCES
    encargado (tipoDocEncargado , NroDocumentoEncargado ) ON UPDATE
    CASCADE ON DELETE CASCADE
);

CREATE TABLE funcion (
    IdFuncion int(5) PRIMARY KEY,
    Fecha DATE,
    Hora TIME,
    idSala int(5),
    idPublicidadInicio int(11),
    idPublicidadFin int(11),
    FOREIGN KEY (idSala) REFERENCES sala (idSala) ON UPDATE CASCADE ON DELETE
    CASCADE,
    FOREIGN KEY (idPublicidadInicio) REFERENCES publicidad (id) ON UPDATE CASCADE ON
    DELETE CASCADE,
    FOREIGN KEY (idPublicidadFin) REFERENCES publicidad (id) ON UPDATE CASCADE ON DELETE
    CASCADE
);

CREATE TABLE asiento(
    idSala int(5),
    nroAsiento int(3),

```

```

    fila int(2),
    FOREIGN KEY (idSala) REFERENCES sala (idSala) ON UPDATE CASCADE ON
    DELETE CASCADE,
    PRIMARY KEY (idSala, nroAsiento)
);

```

```

CREATE TABLE prefiere (
    tipoDocCliente varchar(3),
    NroDocumentoCliente int(8),
    idPublicidad int(11),
    motivo varchar(10),
    FOREIGN KEY (tipoDocCliente ,NroDocumentoCliente ) REFERENCES cliente
    (tipoDoc,NroDocumento) ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (idPublicidad) REFERENCES publicidad (id) ON UPDATE CASCADE ON DELETE
    CASCADE,
    PRIMARY KEY(tipoDocCliente,NroDocumentoCliente,idPublicidad)
);

```

```

CREATE TABLE entrada (
    NumeroEntrada int(10),
    tipoDocCliente varchar(3),
    NroDocumentoCliente int(8),
    nroAsiento int(3),
    idSala int(5),
    idFuncion int(5),
    fechaEmision DATE,
    FOREIGN KEY (tipoDocCliente, NroDocumentoCliente) REFERENCES cliente (tipoDoc,
    NroDocumento) ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (idSala, nroAsiento) REFERENCES asiento (idSala, nroAsiento) ON UPDATE
    CASCADE ON DELETE CASCADE,
    FOREIGN KEY (idFuncion) REFERENCES funcion (idFuncion) ON UPDATE CASCADE ON DELETE
    CASCADE,
    PRIMARY KEY (NumeroEntrada)
);

```

Ejercicio 3: Realizar las sentencias SQL para los siguientes ejercicios

- a) Agregar un Encargado a la base de datos. Tener en cuenta las restricciones de acuerdo a su DDL
- b) Actualizar la Hora de las funciones del viernes 24 de Junio de 2022: hora a actualizar 20:00
- c) Eliminar los Encargados que nunca han sido encargados de una sala

```

a)
INSERT INTO encargado (tipoDocEncargado,NroDocumentoEncargado,legajo,categoria) VALUES
('DNI',44041629,1,'Limpiador');

```

b)

UPDATE funcion SET Hora = '20:00:00' WHERE Fecha = '2022/06/24';

c) DELETE FROM encargado WHERE NroDocumentoEncargado NOT IN(SELECT NroDocumentoEncargado FROM sala);

Ejercicio 4: Realizar las siguientes consultas en SQL

a) Seleccionar NumeroEntrada, NroAsiento, IdSala, IdFuncion de aquellas entradas emitidas 2 días antes de la fecha de la función

```
SELECT NumeroEntrada, NroAsiento, entrada.idSala, entrada.idFuncion
FROM entrada
INNER JOIN funcion ON entrada.idFuncion = funcion.IdFuncion
WHERE DATEDIFF( funcion.Fecha, entrada.fechaEmision ) = 2
```

b) Seleccionar los asientos de las salas tipo '3D', que nunca han sido vendidos en Entradas. Listar idSala, NroAsiento, fila. Utilizar Not Exists

```
SELECT asiento.idSala, nroAsiento, fila
FROM asiento
INNER JOIN sala ON asiento.idSala = sala.idSala
WHERE sala.tipo = '3D'
AND NOT EXISTS (
    SELECT *
    FROM entrada
    WHERE entrada.idSala = asiento.idSala
)
```

c) Listar la cantidad de funciones por cada publicidad inicial en lo que va de este año 2022. Listar idPubliciad, Duración y cantidad

```
SELECT funcion.idPublicidadInicio, publicidad.duracion, COUNT( * ) AS cantidad
FROM funcion
JOIN publicidad ON funcion.idPublicidadInicio = publicidad.id
WHERE fecha
BETWEEN '2022-01-01'
AND '2022-12-31'
```

d) Listar la cantidad de entradas emitidas por fecha que correspondan a funciones del mes de Mayo de 2022 y además la publicidad final debe ser de 5 minutos. Listar fecha de emisión y cantidad. Utilice el operador IN para determinar la publicidad final.

```
SELECT entrada.fechaEmision, COUNT( * ) AS cantidad
FROM entrada
INNER JOIN funcion ON entrada.idFuncion = funcion.IdFuncion
WHERE YEAR( funcion.Fecha ) = '2022'
AND MONTH( funcion.Fecha ) = '05'
AND funcion.idPublicidadFin
```

```
IN (  
SELECT id  
FROM publicidad  
WHERE duracion = '00:05'  
)
```