

## Scripts al final

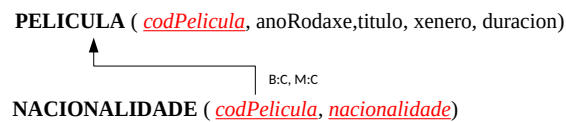
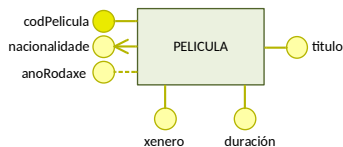
## CREACIÓN DE TÁBOAS

1. Crear a base de datos *practicass1*.

Crear nesa base de datos a táboa *fabricante*. A estrutura da táboa é a seguinte:

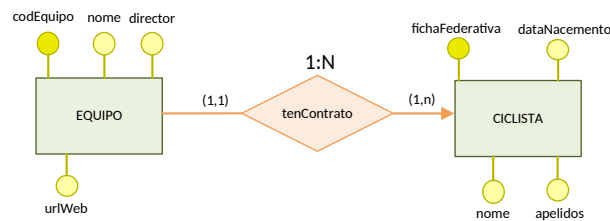
Nome columna	Tipo	Null	Clave	Observacións
idFabricante	char(5)	N	Primaria	Código que identifica a cada fabricante
nome	varchar(80)	N		

2. Crear na base de datos *practicass1* as táboas correspondentes á interpretación do seguinte diagrama E/R e ao seu grafo relacional:



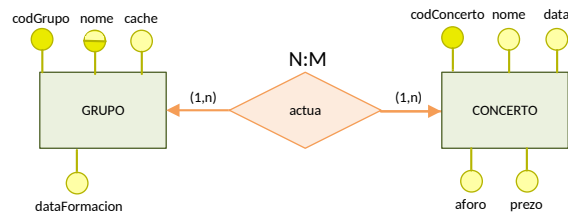
Terase en conta que:

- O atributo *nacionalidade* é un atributo multivaluado da entidade *película*.
  - O atributo *codPelícula* será un valor numérico que se irá incrementando, de forma automática, nunha unidade cada vez que insira unha nova fila na táboa *película*.
  - O atributo *título* ocupa como máximo 100 caracteres, e non admitirá valores nulos.
  - O atributo *duración* gardarase como unha cantidade numérica positiva, con dous díxitos enteiros e dous decimais.
3. Crear na base de datos *practicass1*, as táboas que correspondan ao seguinte diagrama E/R, interpretando a relación de tipo 1:N.



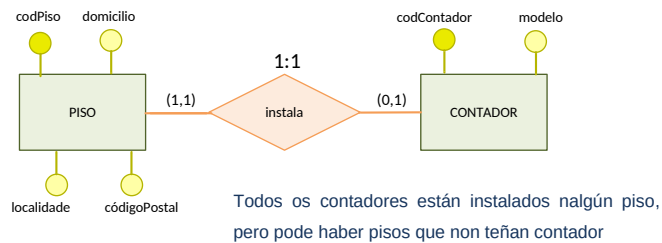
Terase en conta que o código do equipo está composto por unha combinación de letras e números cun tamaño fixo de 9 caracteres.

4. Crear na base de datos *practicass1*, as táboas que correspondan ao seguinte diagrama E/R, interpretando a relación de tipo N:M.



Terase en conta que:

- O *aforo* representa o número máximo de espectadores que poden asistir ao concerto e como máximo será 100000 persoas.
  - O *prezo* das entradas ven dado en euros con dous decimais e o prezo máximo é 200 euros.
5. Crear na base de datos *practicass1*, as táboas que correspondan ao seguinte diagrama E/R, interpretando a relación de tipo 1:1.



Terase en conta que só existen tres modelos de contadores que se identifican como 'aeg235', 'samsung20', e 'aeg55'.

## SCRIPTS

```
CREATE DATABASE IF NOT EXISTS practicas1;
```

```
USE practicas1;
```

```
CREATE TABLE fabricante (
    idFabricante CHAR(5)
        NOT NULL
    PRIMARY KEY
    COMMENT 'Código que identifica a cada fabricante',
    nome VARCHAR(80) NOT NULL
```

);

```
CREATE TABLE PELICULA (  
    codPelicula INTEGER UNSIGNED  
        PRIMARY KEY  
            AUTO_INCREMENT,  
    anoRodaxe INT NULL,  
    titulo VARCHAR(100)  
        NOT NULL,  
    xenero ENUM('DRAMA','COMEDIA','TERROR')  
        NOT NULL,  
    duracion DECIMAL(4,2)  
        NOT NULL  
        CONSTRAINT duracionMaiorQueCero  
            CHECK (duracion >= 0)  
);
```

```
CREATE TABLE NACIONALIDADE (  
    codPelicula INTEGER UNSIGNED  
        NOT NULL,  
    nacionalidade VARCHAR(100) NOT NULL,  
    CONSTRAINT fk_NACIONALIDADE  
        FOREIGN KEY (codPelicula) REFERENCES PELICULA (codPelicula)  
            ON DELETE CASCADE  
            ON UPDATE CASCADE,  
    CONSTRAINT pk_NACIONALIDADE  
        PRIMARY KEY (codPelicula, nacionalidade)  
);
```

```
CREATE TABLE EQUIPO (  
    codEquipo CHAR(9)  
        COMMENT "eSTO VA DE TAL FORMA" NOT NULL
```

```
PRIMARY KEY,  
nome VARCHAR(50)  
NOT NULL,  
director VARCHAR(50)  
NOT NULL,  
urlWEB VARCHAR(100)  
NOT NULL  
);
```

```
CREATE TABLE CICLISTA (  
    fichaFederativa INTEGER NOT NULL PRIMARY KEY,  
    nome VARCHAR(50) NOT NULL,  
    apellidos VARCHAR(60) NOT NULL,  
    codEquipo CHAR(9) NOT NULL,  
    dataNacimiento DATE NOT NULL,  
    CONSTRAINT fk_CICLISTA  
        FOREIGN KEY (codEquipo) REFERENCES EQUIPO (codEquipo)  
        ON DELETE RESTRICT  
        ON UPDATE RESTRICT  
);
```

```
CREATE TABLE GRUPO (  
    codGrupo INTEGER UNSIGNED NOT NULL PRIMARY KEY COMMENT "Clave principal",  
    nome VARCHAR(60) NOT NULL UNIQUE COMMENT "Nombre del grupo",  
    cache DECIMAL(7,2) NOT NULL COMMENT "Caché do grupo",  
    dataFormacion DATE NOT NULL COMMENT "data de formación do grupo");
```

```
CREATE TABLE ACTUA (  
    codGrupo INTEGER UNSIGNED,  
    codConcerto INTEGER UNSIGNED,  
    CONSTRAINT ACTUA_FK_GRUPO -- NOMENCLATURA HABITUAL  
        FOREIGN KEY (codGrupo)
```

```
REFERENCES GRUPO (codGrupo)
ON DELETE RESTRICT
ON UPDATE RESTRICT,
    CONSTRAINT ACTUA_FK_CONCERTO
        FOREIGN KEY (codConcerto)
REFERENCES CONCERTO (codConcerto)
ON UPDATE RESTRICT
ON DELETE RESTRICT,
    CONSTRAINT ACTUA_PK -- esta es la nomenclatura habitual
        PRIMARY KEY (codGrupo, codConcerto)
);
```

```
CREATE TABLE CONCERTO (
    codConcerto INTEGER UNSIGNED,
    nome VARCHAR(100) NOT NULL,
    data DATETIME NOT NULL COMMENT "Podría ser un timestamp también",
    aforo INTEGER(6) UNSIGNED NOT NULL,
    prezzo DECIMAL(5,2) NOT NULL,
    CONSTRAINT pk_CONCERTO
        PRIMARY KEY (codConcerto),
    CONSTRAINT prezomin_e_max
        CHECK (prezzo > 0 AND prezzo <= 200),
    CONSTRAINT aforoMaximo
        CHECK (aforo < 100000)
);
```

```
CREATE TABLE PISO (
    codPiso INTEGER UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    domicilio VARCHAR (100) NOT NULL,
    localidade VARCHAR(100)NOT NULL,
    codigoPostal CHAR(5) NOT NULL -- El cp es CHAR,no números, porque no se hacen operaciones, ni se
restan, ni se suman, ni nada
```

);

CREATE TABLE CONTADOR (

    codContador INTEGER UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

    modelo ENUM("aeg235", "samsung20", "aeg55") NOT NULL,

    codPiso INTEGER NOT NULL,

    CONSTRAINT CONTADOR\_FK\_PISO

        FOREIGN KEY (codPiso)

        REFERENCES PISO (codPiso)

        ON UPDATE RESTRICT

        ON DELETE RESTRICT

);