

Activitat 3.2 - Transformació inicial d'una Entitat-Relació en una BD

Realitzar les tasques per adquirir els coneixements per aprendre a crear taules d'una base de dades usant el llenguatge SQL. Podeu consultar la documentació DDL des de *l'apartat 1 al 4*.

Enunciat

Realitza el model conceptual (model Entitat-Relació), el disseny lògic (model relacional) i el disseny físic (codi SQL) per aquesta tasca.

Un cop superat cada pas de la transformació de les tasques, consulteu les solucions i partiu d'aquesta per tal de realitzar el següent pas de la transformació. No ho feu a partir de les vostres solucions mentre no coincideixin amb la proposta.

Temps estimat: 3 hores.

Sense Avaluació: solució a classe.

Tasca 1. Productes informàtics

A partir de la base de dades de la [Tasca 2. BD M02 UF2](#) (de l'Activitat 2.1. Base de dades). Volem crear una gestió per guardar els productes informàtics d'una botiga i certes dades dels seus fabricants. En concret hem de tenir en compte:

- Crearem una taula amb els productes on guardarem un codi numèric únic, el nom, la descripció, la categoria i el preu.

```
CREATE TABLE productes(  
    codi          INTEGER          UNIQUE  
    nom          VARCHAR(25)  
    descripcio   TEXT  
    categoria    VARCHAR(25)  
    preu         FLOAT);
```

- També crearem una taula on guardarem les característiques dels fabricants i de cadascun posarem un codi de fabricant únic, el nom, l'adreça web i un telèfon de contacte.

```
CREATE TABLE fabricants(  
    codi          INTEGER    UNIQUE  
    nom          VARCHAR(25)  
    adreca       VARCHAR(25)  
    telefon      INTEGER);
```

- Les dues taules han d'estar vinculades; és a dir, de cada producte volem saber de quin és el fabricant.

```
FOREIGN KEY (codi_fabricant) REFERENCES productes(codi_producte)
```

- Els camps en forma de text tindran una longitud màxima de 20 caràcters (excepte que algun veieu que podeu reduir la mida).

```
CREATE TABLE productes(  
    codi_producte INTEGER          UNIQUE,  
    nom           VARCHAR(20),  
    descripcio    VARCHAR(20),  
    categoria     VARCHAR(10),  
    preu          FLOAT  
);
```

```
CREATE TABLE fabricants(  
    codi_fabricant INTEGER    UNIQUE,  
    nom           VARCHAR(20),  
    adreca       VARCHAR(20),  
    telefon      INTEGER,  
    FOREIGN KEY (codi_fabricant) REFERENCES productes(codi_producte)  
);
```

Tasca 2. Institut

A partir de la base de dades de la [Tasca 2. BD M02 UF2](#) (de l'Activitat 2.1. Base de dades). Un institut vol emmagatzemar les assignatures, els professors que les imparteixen i els alumnes que estudien cada assignatura.

- De les assignatures volem guardar el codi de la assignatura (3 lletres), el nom de l'assignatura, el cicle a la que pertany i el curs al que pertany.

```
CREATE TABLE assignatura(  
    codi_assignatura    CHAR(3),  
    nom                 VARCHAR(20),  
    cicle               VARCHAR(20),  
    curs                INTEGER  
);
```

- De cada professor volem emmagatzemar un codi numèric de professor, el seu nom, els cognoms, la data de naixement, l'adreça i el seu telèfon.

```
CREATE TABLE professor(  
    codi_professor      INTEGER  
    nom                 VARCHAR(20),  
    cognoms             VARCHAR(20),  
    data_naixement      DATE,  
    adreca              VARCHAR(20),  
    telefon             INTEGER  
);
```

- De cada alumne guardarem un codi numèric únic, el seu nom, els cognoms, la data de naixement, l'adreça i el seu telèfon. També volem guardar quan va fer la seva primera matrícula.

```
CREATE TABLE alumne(  
    codi_alumne        INTEGER    UNIQUE,  
    nom                VARCHAR(20),  
    cognoms             VARCHAR(20),  
    data_naixement     DATE,  
    adreca              VARCHAR(20),  
    telefon             INTEGER,  
    data_matricula     DATE  
);
```

- Els camps textuais tindran una longitud màxima de 20 caràcters.

Tasca 3. Botiga DVD

Volem crear una **nova base de dades** per gestionar una botiga de préstec de DVDs. L'objectiu de la base de dades és que gestioni els usuaris i les pel·lícules dvd que tenim en stock.

Dels *usuaris* volem emmagatzemar el seu dni, el nom i cognoms, l'adreça i ciutat on viuen, el sexe, el telèfon de casa, el mòbil, el correu electrònic i data de naixement.

De les *pel·lícules dvd* emmagatzemarem el títol, el director, els actors, la companyia productora, l'any en que va ser rodada, un text llarg on podrem posar l'argument, la durada en minuts (no pot tenir decimals), la data en que va ser donada d'alta a la nostra base de dades, la nacionalitat i un camp que ens deixi indicar si està donada de baixa (perquè s'ha perdut, no la lloga ningú, ...). A més té un identificador per identificar-la.

Existeixen diferents *categories* (ciència-ficció, històrica, comèdia, ...), i cada pel·lícula serà d'una d'aquestes categories (només d'una). En qualsevol moment ens pot interessar afegir una nova categoria de les que s'han posat d'exemple.

Cada *pel·licula dvd* té diverses còpies, cadascuna amb un codi que comença per l'1 per a cada *pel·licula dvd*. Els usuaris poden *llogar* les còpies. Volem saber l'usuari que la té llogada i des de quina data. En el moment que la retorni afegirem la data del retorn.

La modelització de la base de dades ens afegeix una sèrie de requisits:

1. Els identificadors de les taules són enters autonumèrics.
2. La longitud dels camps de text no serà superior a 50.
3. La categoria és un valor únic.
4. Per la pel·lícula:
 - a. El nom és un valor únic.
 - b. L'any de filmació ha de ser més gran que 1900.
 - c. La durada va entre 1 fins 300 minuts.
 - d. La data d'alta és a partir de l'1/1/2001 i per defecte agafa la data d'avui.
 - e. Baixa pot ser 'S' o 'N', per defecte 'N'.
 - f. Són requerits tots els atributs menys l'argument.
5. Pels usuaris:
 - a. Per defecte la població és 'Terrassa'.
 - b. El dni és un valor únic.
 - c. El mòbil és un valor únic.
 - d. El correu electrònic és un valor únic.
 - e. La data de naixement ha de ser més gran que l'1/1/1900.
 - f. El dni ha de complir el format 'XXXXXXXXXL'.
 - g. Sexe pot ser 'H' o 'D'.
 - h. Són requerits el dni, nom, cognoms, adreça, població, sexe i la data de naixement.
6. Quan es lloga un DVD:
 - a. La data del lloguer ha de ser la data d'avui per defecte.
 - b. Són requerits tots els atributs menys la data de retorn.

CREATE TABLE usuari(

id_usuari	INTEGER	AUTO_INCREMENT	PRIMARY KEY,
dni	CHAR(9)	NOT NULL,	
nom	VARCHAR(50)	NOT NULL,	
cognoms	VARCHAR(50)	NOT NULL,	
adrec	VARCHAR(50)	NOT NULL,	
poblacio	VARCHAR(20)	DEFAULT("Terrassa")	NOT NULL,
sexe	ENUM('H','D')	NOT NULL,	
telefon	INTEGER,		
mobil	INTEGER,		
correu	VARCHAR(50),		
data_naixement	DATE	NOT NULL,	
CONSTRAINT	uk_usuari_dni_mobil_correu UNIQUE (dni, mobil, correu)		

);

CREATE TABLE pelicula(

id_pelicula	INTEGER	AUTO_INCREMENT	PRIMARY KEY,
nom	VARCHAR(50)	NOT NULL,	
director	VARCHAR(50)	NOT NULL,	
actors	VARCHAR(50)	NOT NULL,	
companyia	VARCHAR(50)	NOT NULL,	
data_film	DATE	NOT NULL,	
argument	TEXT,		
durada	TIME	NOT NULL,	
data_alta	DATETIME	NOT NULL,	
nacionalitat	VARCHAR(50)	NOT NULL,	
baixa	ENUM('S','N')	DEFAULT('N')	NOT NULL,
categoria	VARCHAR(50)	NOT NULL,	
CONSTRAINT	uk_pelicula_nom_categoria UNIQUE (nom, categoria)		

);

```
CREATE TABLE lloguer(  
    codi_pelicula    INTEGER        NOT NULL,  
    usuari           VARCHAR(50)    NOT NULL,  
    data_lloguer     DATE           NOT NULL,  
    data_retorn      DATE  
);
```

Solucions

[Models E-R](#)

[Models Relacionals](#)

Recursos

- [Moodle M2](#)
- [DDL \(MariaDB documentation\)](#)