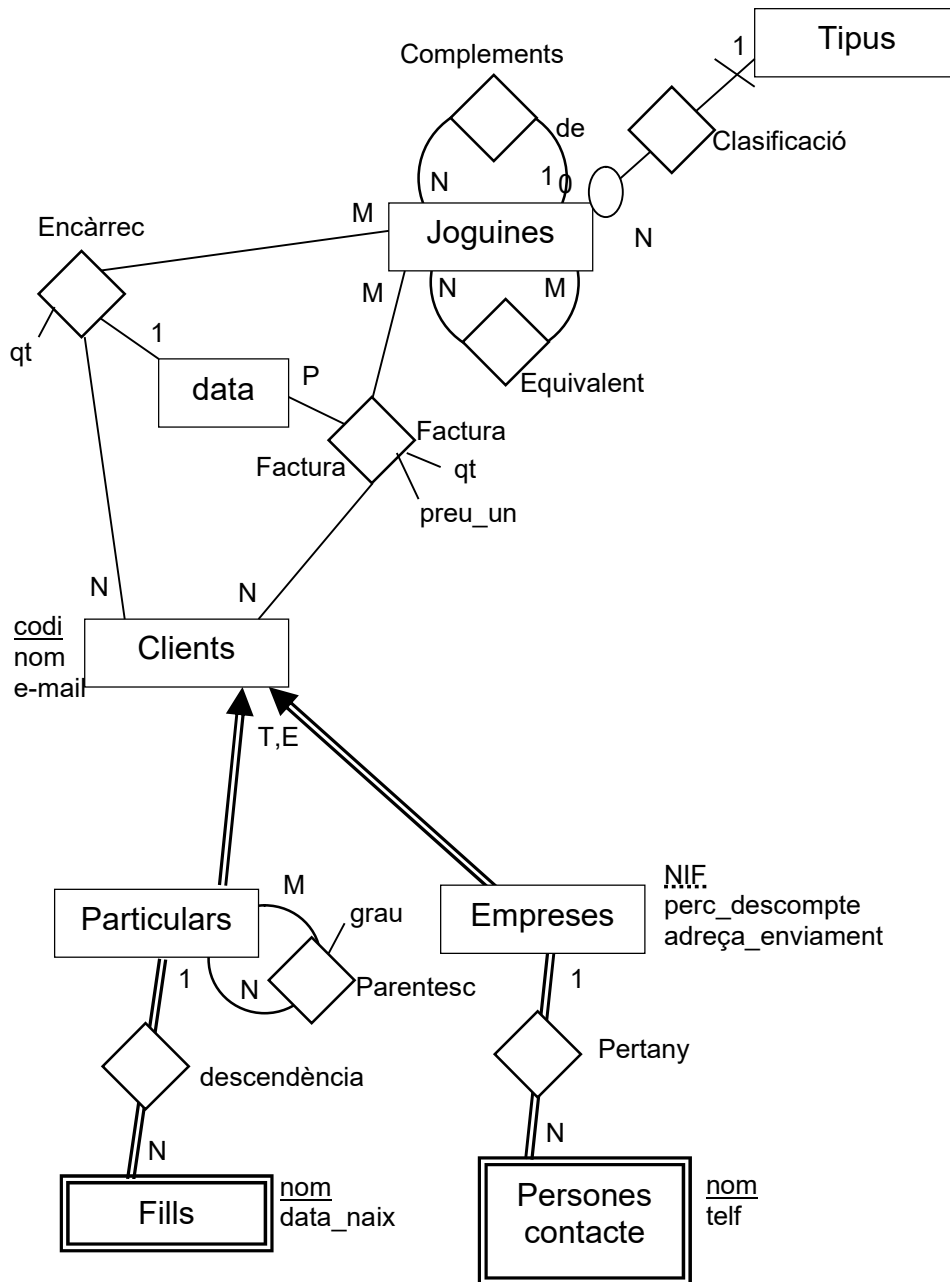


## PRÀCTICA 3<sup>a</sup> PART

DISSENY FÍSIC DE BASES DE DADES  
Data límit de lliurament: 06/6/2020

A partir de l'esquema EER i model relacional que hi ha a continuació, crear l'esquema físic en MySQL que implementi aquesta base de dades de manera que les dades mantinguin el màxim de coherència possible.

### ESQUEMA CONCEPTUAL



Notació:

—: clau primària

.....: clau alternativa

## **ESQUEMA LÒGIC**

### **TIPUS**(tipus)

**JOGUINES**(codi\_barres, nom, edat\_min, edat\_max, preu, tipus, complement\_de)

tipus és FK a Tipus

complement\_de és FK a Joguines

**EQUIVALENT**(joguina, joguina\_equivalent)

Joguina és FK a Joguines

Joguina\_equivalent és FK a joguines

NOTA: Aquesta interrelació és simétrica: si (A,B) aleshores (B,A); i també és transitiva: si (A,B) i (B,C) aleshores (A,C). Per aquest motiu convindria afegir aquestes regles:

I<sub>Joguines</sub> -> I<sub>Joguines</sub>: Si insereixo (A,B) cal inserir (B,A) si ho volem tenir repetit i si no prohibir-ho. També cal comprovar si existeixen tuples (A,C) i en aquest cas afegir (B,C) i deixar que s'insereixin possiblement altres tuples en executar-se les regles d'inserció.

E<sub>Joguines</sub> -> E<sub>Joguines</sub>: Si esborro (A,B) cal esborrar (B,A) si l'hem inserit abans, segons decisió disseny

M<sub>Joguines</sub> -> I<sub>Joguines</sub> i E<sub>Joguines</sub>: Si Modifico (A,B) per (A',B') cal esborrar (B,A) i inserir (B',A'), i deixar que s'insereixin i/o esborrin possiblement altres tuples en executar-se les regles d'inserció i/o esborrat.

**CLIENTS**(codi, nom, email)

I<sub>Clientss</sub> -> I<sub>Particularss</sub>: 0 I<sub>Empreses</sub>

**ENCARREC**(client, joguina, data, qt)

client és FK a Clients

joguina és FK a Joguines

**FACTURA**(client, data, joguina, qt, preu\_un)

client és FK a Clients

joguina és FK a Joguines

**PARTICULARS**(codi, telf., dni)

codi és FK a Clients

E<sub>Particularsss</sub> -> E<sub>Clients</sub>: Aquesta és una de les accions a fer pel fet de tenir una generalització total

**EMPRESSES**(codi, nif)

codi és FK a Clients

E<sub>Empreses</sub> -> E<sub>Clients</sub>: Aquesta és una de les accions a fer pel fet de tenir una generalització total

**FILLS**(nom, codi, data\_naix)

codi és FK a Particulars

**PERSONES\_CONTACTE**(nom, codi, telf)

Codi és FK a Particulars

**PARENTESC**(particular, parent, grau)

particular és FK a Particulars

parent és FK a Parents

NOTA: Aquesta interrelació és simétrica: si (A,B) aleshores (B,A); i també és transitiva: si (A,B) i (B,C) aleshores (A,C). Per aquest motiu convindria afegir aquestes regles:

I<sub>Parentesc</sub> -> I<sub>Parentesc</sub>: Si insereixo (A,B) cal inserir (B,A). També cal comprovar si existeixen tuples (A,C) i en aquest cas afegir (B,C) i deixar que s'insereixin possiblement altres tuples en executar-se les regles d'inserció.

E<sub>Parentesc</sub> -> E<sub>Parentesc</sub>: Si esborro (A,B) cal esborrar (B,A)

M<sub>Parentesc</sub> -> I<sub>Parentesc</sub> i E<sub>Parentesc</sub>: Si Modifico (A,B) per (A',B') cal esborrar (B,A) i inserir (B',A'), i deixar que s'insereixin i/o esborrin possiblement altres tuples en executar-se les regles d'inserció i/o esborrat.

## **ES DEMANA:**

1. Creeu un script d'SQL que creï l'estructura de la base de dades en MySQL ajustant-se al màxim als standards de SQL i fent prevaldre sempre la coherència de dades envers la rapidesa. A més de les restriccions expressades en el disseny EER, cal considerar també les següents:
  - A.- La quantitat en els encàrrecs ha de ser superior a 10 i inferior a 1000
  - B.- El grau en la relació parentesc estarà codificat amb una lletra que ha de ser P (=primer), S (=segon), T (=tercer), Q (=quart) o C (cinqué).
  - C.- A la relació equivalent al ser simètrica, es el mateix tenir la informació (A,B) que (B,A). Per tant es informació redundat, encara que pot ser interessant emmagatzemar-ho, Implementeu el que sigui necessari per evitar aquesta redundància)..
  - D.- (OPCIONAL:): Implementeu mitjançant triggers i/o stored procedures les restriccions de Total i Encavalcada de la generalització de Clients
- 2.- Creeu un segon script que inserti les dades que considereu necessàries per tal de poder provar les consultes dels apartats següents.

## **Resoldre les següents consultes:**

- A.- Obtenir les dades dels Clients Particulars que tenen algun fill major de 18 anys. Mostreu el resultat ordenat en ordre alfabètic segons el nom del Client.
- B- Obtenir els noms dels Clients als que no se els hi ha generat mai cap factura que superi els 10.000 euros.
- C- Obtenir quines joguines les han demanat mes de 10 Clients diferents que no siguin empreses. Concretament es demana el codi de barres de la joguina i el nom de la joguina, ordenat per nom de la joguina.
- D- Obtenir quins Clients han encarregat totes les joguines en que la seva edat mínima supera els 10 anys.
- E- Obtenir quins són els empleats particulars que han encarregat mes joguines que el client Empresa que te per NIF el B54543443F.

## **Realitzar les següents vistes:**

- F- Creeu una vista que obtingui el llistat de clients amb el codi de les joguines que ha encarregat en tota la seva vida. Tingueu en compte que si un empleat ha encarregat varies vegades la mateixa joguina, aquest binomi només ha d'apareixer una sola vegada.

## **Escriure una sentència (o un conjunt de sentències que s'executen com una de sola) per:**

- G- Modificar la taula factura per a introduir un cam que sigui "Pagat/no pagat" i posar com a no pagat totes aquelles factures que tenen com a data menys de 5 dies desde el dia d'avui.
- H- Suprimir tots els clients de la Base de dades que porten mes de cinc anys que no han encarregat cap joguina..

**IMPORTANT: Sempre que us sigui possible, seguïu els estandards de sintaxis vistos a classe (estandard de SQL).**