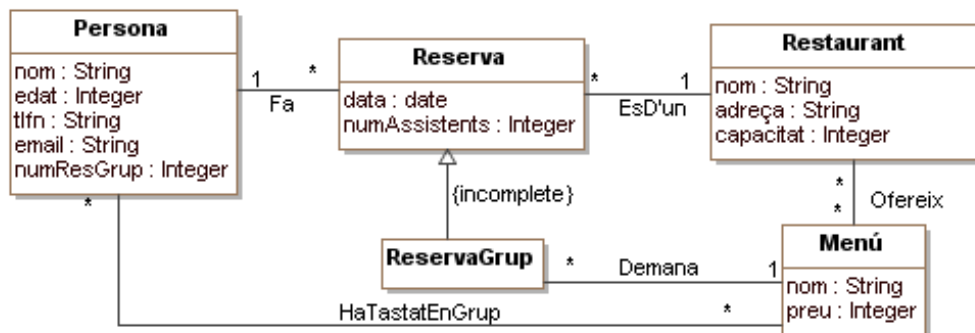


Exercici 4 de la llista – Adaptat a la primera classe de la Capa de Dades

Una cadena de restaurants que ofereix menús per sopars ens ha demanat que li dissenyem una part d'un sistema software per gestionar les seves reserves. L'esquema conceptual (de l'especificació) es mostra a continuació:



R.I. Textuals:

RI1 - Claus: (Persona, nom); (Restaurant, nom); (Menú, nom); (Reserva, Persona::nom+data);

RI2 - Es considera que no hi pot haver dos persones amb el mateix email.

RI3 - Els menús de les reserves de grup han de ser menús oferts pel restaurant on s'ha fet la reserva.

RI4 - La capacitat d'un restaurant ha de ser més gran que el sumatori dels assistents de les reserves previstes per aquell restaurant en una data determinada.

RI5 - El numAssistents d'una reserva de grup ha de ser més gran que 5.

RI6 - Els menús que ha tastat en grup una persona són els mateixos que ha reservat.

RI7 - El numResGrup d'una persona és igual al número de reserves de grup que ha fet la persona.

RI8 - En numResGrup ha de ser més gran o igual a 0. La resta d'atributs de tipus integer han de ser positius.

RI9 - Els únics atributs que apareixen en el diagrama de classes pels que es permet valors nuls és l'atribut edat, i per un dels dos atributs tlfn, email però mai per tots dos.

Altres restriccions no rellevants per l'exercici.

La capa de domini ofereix l'operació següent:

context novaReservaGrup(nomClient:String, nomRest:String, dr:date, nAss:Integer, nomMenú:String): Integer

pre client-existeix, data-ok, més-5assistents, reserva-no-existeix

exc restaurant-no-existeix: el restaurant amb *nomRest* no existeix.

exc menu-no-ofert: el restaurant *nomRest* no ofereix el menú *nomMenú*.

exc restaurant-sense-lloc: el restaurant *nomRest* no té disponibilitat pel *nAss* a la data *dr*.

post reserva-grup-creada: es crea una reserva de grup i es formen totes les associacions amb la persona, el restaurant i el menú.

post numResGrup-incrementat: s'incrementa el *numResGrup* de la persona *nomClient*.

post haTastat-creat: si la persona no ha tastat el menú, es crea una instància de *HaTastatEnGrup* entre la persona *nomClient* i el menú *nomMenú*.

post result= número de places disponibles del restaurant *nomRest* a la data *dr* (capacitat del restaurant – número de assistents (de les reserves) per la data *dr* - *nAss*).

Es demana:

- 1) Dissenyeu l'esquema de la base de dades relacional corresponent al diagrama de classes usant Class Table Inheritance per la jerarquia d'herència. Cal incloure taules i claus primàries i foranes.
- 2) Dissenyeu l'esquema de la base de dades relacional corresponent al diagrama de classes usant Concrete Table Inheritance per la jerarquia d'herència. Cal incloure taules i claus primàries i foranes.
- 3) Dissenyeu l'esquema de la base de dades relacional corresponent al diagrama de classes usant Single Table Inheritance per la jerarquia d'herència. Cal incloure taules i claus primàries i foranes.

- 4) Suposant el disseny de l'apartat 2 (Concrete Table Inheritance). Indiqueu cadascuna de les restriccions de l'esquema conceptual com la implementaríeu en una base de dades relacional. Les restriccions que podeu usar d'una base de dades relacional són:
- a. Taules
 - b. Columnnes
 - c. Claus primàries
 - d. Claus foranes
 - e. Restriccions de taula (Check, Not Null, Unique)
- 5) Suposem el disseny de taules de l'apartat 2 (Concrete Table Inheritance), i la implementació de restriccions que s'ha fet en l'apartat 4. Implementeu l'operació novaReservaGrup de l'enunciat com un procediment emmagatzemat.

AMPLIACIÓ

- A. Quines restriccions que no heu pogut implementar en l'apartat 4 es podrien implementar si es disposés de:
- a. Triggers per controlar restriccions d'integritat
 - b. Triggers per actualitzar atributs derivats/taules derivades.

- 1) Disseny de la base de dades amb Class Table Inheritance per la jerarquia de Reserves. Només s'ha tingut en compte la restricció textual RI1.

```
persones(nomP, edat, tlf, email, numResGrup)

restaurants(nomR, adreça, capacitat)

menus(nomM,preu)

ofertesMenus (nomR,nomM)
    {nomR} referencia Restaurant
    {nomM} referencia Menu

tastatsEnGrup(nomP,nomM)
    {nomP} referencia Persona
    {nomM} referencia Menu

reserves(nomP,data,numAssistents,nomR)
    {nomP} referencia Persona
    {nomR} referencia Restaurant

reservesGrup(nomP,data,nomM)
    {nomP,data} referencia Reserva
    {nomM} referencia Menu
```

- 2) Disseny de la base de dades amb Concrete Table Inheritance per la jerarquia de Reserves. Només s'ha tingut en compte la restricció textual RI1 i RI3.

```
persones(nomP, edat, tlf, email, numResGrup)

restaurants(nomR, adreça, capacitat)

menus(nomM,preu)

ofertesMenus (nomR,nomM)
    {nomR} referencia Restaurant
    {nomM} referencia Menu

tastatsEnGrup(nomP,nomM)
    {nomP} referencia Persona
    {nomM} referencia Menu

reservesIndividuals(nomP,data,numAssistents,nomR)
    {nomP} referencia Persona
    {nomR} referencia Restaurant

reservesGrup(nomP,data, numAssistents,nomR,nomM)
    {nomP} referencia Persona
    {nomR,nomM} referencia Ofereix
    /* la darrera restricció s'ha afegit */
    /* per controlar la RI3 */
```

- 3) Disseny de la base de dades amb Single Table Inheritance per la jerarquia de Reserves. Només s'ha tingut en compte la restricció textual RI1.

```
persones(nomP, edat, tlf, email, numResGrup)
```

```
restaurants(nomR, adreça, capacitat)
```

```
menus(nomM,preu)
```

```
ofertesMenus (nomR,nomM)  
  {nomR} referencia Restaurant  
  {nomM} referencia Menu
```

```
tastatsEnGrup(nomP,nomM)  
  {nomP} referencia Persona  
  {nomM} referencia Menu
```

```
reserves(nomP,data,tipus,numAssistents,nomR,nomM)  
  {nomP} referencia Persona  
  {nomR} referencia Restaurant  
  {nomM} referencia Menu  
/* El nomM tindrà valor nul en cas de reserva  
individual */
```

- 4) Suposant el disseny de l'apartat 2 (Concrete Table Inheritance). Indiqueu cadascuna de les restriccions de l'esquema conceptual com la implementaríeu en una base de dades relacional. Les restriccions que podeu usar d'una base de dades relacional són:
- Taules
 - Columnes
 - Claus primàries
 - Claus foranes
 - Restriccions de taula (Check, Not Null, Unique)

Esquema Conceptual	Base de Dades
Classe Persona	Taula persones
Classe Restaurant	Taula restaurants
Classe Menú	Taula menus
Classes/jerarquia Reserva, ReservaGrup	Taules reservesIndividuals, reservesGrup
Associació Fa (1..*)	Taula reservesIndividuals, reservesGrup <ul style="list-style-type: none"> {nomP} referencia Persona Com que és part de la PK ja és segur not null
Associació EsDun (1..*)	Taula reservesIndividuals, reservesGrup <ul style="list-style-type: none"> {nomR} referencia Restaurant nomR not null
Associació Demana (1..*)	Taula reservesGrup <ul style="list-style-type: none"> {nomM} referencia Menu nomM not null
Associació Ofereix (*..*)	Taula ofertesMenus <ul style="list-style-type: none"> PK {nomR, nomM} {nomR} referencia Restaurant {nomM} referencia Menu
Associació HaTastaEnGrup (*..*)	Taula tastatsEnGrup <ul style="list-style-type: none"> PK {nomP, nomM} {nomP} referencia Persona {nomM} referencia Menu
Jerarquia Reserva (incomplete, disjoint) – una reserva o és individual o és de grup	No es pot implementar
RI1 - Claus: (Persona, nom), Claus: (Restaurant, nom), Claus (Menú, nom), Claus (Reserva, Persona::nom+data)	Taula persones <ul style="list-style-type: none"> PK {nomP} Taula restaurants <ul style="list-style-type: none"> PK {nomR} Taula menus <ul style="list-style-type: none"> PK {nomM} Taules reservaIndividuals, reservesGrup <ul style="list-style-type: none"> PK {nomP, data}
RI2 - Es considera que no hi pot haver dos persones amb el mateix email.	Taula Persona <ul style="list-style-type: none"> Unique atribut email
RI3 - Els menús de les reserves de grup han de ser menús oferts pel restaurant on s'ha fet la reserva	Taula reservesGrup <ul style="list-style-type: none"> {nomR,nomM} referencia Ofereix Aquesta FK ens garanteix que quan es faci una reserva de grup el menú és ofert pel Restaurant.

RI4 - La capacitat d'un restaurant ha de ser més gran que el sumatori dels assistents de les reserves previstes per aquell restaurant en una data determinada.	No es pot implementar
RI 5 - El numAssistents d'una reserva de grup ha de ser més gran que 5	Taula reservesGrup <ul style="list-style-type: none"> • Check(numAssistents > 5)
RI 6 - Els menús que ha tastat en grup una persona són els mateixos que ha reservat.	No es pot implementar
RI7 - El numResGrup d'una persona és igual al número de reserves de grup que ha fet la persona	No es pot implementar
RI8 - En numResGrup ha de ser més gran o igual a 0. La resta d'atributs de tipus integer han de ser positius.	Taula persones <ul style="list-style-type: none"> • Check(numResGrup >= 0) Taula restaurants <ul style="list-style-type: none"> • Check(capacitat > 0) Taula menus <ul style="list-style-type: none"> • Check(preu > 0) Taula reservesIndividuals <ul style="list-style-type: none"> • Check(numAssistents > 0) Taula reservesGrup <ul style="list-style-type: none"> • Check(numAssistents > 5) /* veure RI5*/
RI 9 - Els únics atributs del diagrama de classes pels que es permet valors nuls és l'atribut edat	A continuació surten els atributs que no formen part de PKs, que ja són nuls per definició Taula Persona <ul style="list-style-type: none"> • numResGrup NOT NULL Taula Restaurant <ul style="list-style-type: none"> • adreca NOT NULL • capacitat NOT NULL Taula Menu <ul style="list-style-type: none"> • preu NOT NULL Taula ReservaIndividual <ul style="list-style-type: none"> • numAssistents NOT NULL Taula ReservaIndividual <ul style="list-style-type: none"> • numAssistents NOT NULL
RI9- I per un dels dos atributs tlfm, email però mai per tots dos	Check(tlf NOT NULL OR email NOT NULL)

```

DROP TABLE tastatsEnGrup;
DROP TABLE reservesGrup;
DROP TABLE reservesIndividuals;
DROP TABLE ofertesMenus;
DROP TABLE menus;
DROP TABLE restaurants;
DROP TABLE persones;

```

```

CREATE TABLE persones
(nomP VARCHAR(50),
edat INTEGER CHECK(edat>0),
tlfm CHAR(12),
email VARCHAR(50) UNIQUE,
numResGrup INTEGER CHECK(numResGrup>=0) NOT NULL,
PRIMARY KEY (nomP),
CHECK (tlfm IS NOT NULL OR email IS NOT NULL));

```

```

CREATE TABLE restaurants
(nomR VARCHAR(50),
adreca VARCHAR(200) NOT NULL,
capacitat INTEGER CHECK(capacitat>0) NOT NULL,
PRIMARY KEY (nomR));

CREATE TABLE menus
(nomM VARCHAR(50),
preu INTEGER CHECK(preu>0) NOT NULL,
PRIMARY KEY (nomM));

CREATE TABLE ofertesMenus
(nomR VARCHAR(50),
nomM VARCHAR(50),
PRIMARY KEY (nomR,nomM),
FOREIGN KEY (nomR) REFERENCES restaurants,
FOREIGN KEY (nomM) REFERENCES menus);

CREATE TABLE reservesIndividuals
(nomP VARCHAR(50),
data DATE,
nomR VARCHAR(50) NOT NULL,
numAssistents INTEGER CHECK(numAssistents>0) NOT NULL,
PRIMARY KEY (nomP,data),
FOREIGN KEY (nomP) REFERENCES persones,
FOREIGN KEY (nomR) REFERENCES restaurants);

CREATE TABLE reservesGrup
(nomP VARCHAR(50),
data DATE,
nomR VARCHAR(50) NOT NULL,
numAssistents INTEGER CHECK(numAssistents>5) NOT NULL,
nomM VARCHAR(50) NOT NULL,
PRIMARY KEY (nomP,data),
FOREIGN KEY (nomR,nomM) REFERENCES ofertesMenus);

CREATE TABLE tastatsEnGrup
(nomP VARCHAR(50),
nomM VARCHAR(50),
PRIMARY KEY (nomP,nomM),
FOREIGN KEY (nomP) REFERENCES persones,
FOREIGN KEY (nomM) REFERENCES menus);

```

- 1) Suposem el disseny de taules de l'apartat 2 (Concrete Table Inheritance), i la implementació de restriccions que s'ha fet en l'apartat 4. Implementeu l'operació novaReservaGrup de l'enunciat com un procediment emmagatzemat.

```
CREATE OR REPLACE FUNCTION novaReservaGrup(nomCli VARCHAR(50), nomRest VARCHAR(50),
                                           dr DATE, nAss INTEGER, nomMenu VARCHAR(50)) RETURNS
INTEGER AS $$
DECLARE
    cap integer;
    numAssResInd integer;
    numAssResGrup integer;

BEGIN
    -- pres garantides: clientExisteix, dataOK, mes5assistents, reservaNoEx
    -- post reserva-grup-creada - inserir reserva de grup
    INSERT INTO reservesGrup
    VALUES(nomCli,dr,nomRest,nAss,nomMenu);
    -- post numResGrup-incrementat - modificar el numero de reserves de grup del client
    UPDATE persones
    SET numResGrup = numResGrup + 1
    WHERE nomP = nomCli;
    -- post haTastatCreat - inserir una fila a tastatsEnGrup
    INSERT INTO tastatsEnGrup
    VALUES(nomCli,nomMenu);
    SELECT capacitat INTO cap
    FROM restaurants
    WHERE nomR = nomRest;
    SELECT SUM(numAssistents) INTO numAssResInd
    FROM reservesIndividuals
    WHERE nomR = nomRest AND data = dr;
    SELECT SUM(numAssistents) INTO numAssResGrup
    FROM reservesGrup
    WHERE nomR = nomRest AND data = dr;
    -- exc restaurant-sense-lloc
    IF (cap - numAssResInd - numAssResGrup < 0)
    THEN RAISE EXCEPTION 'El restaurant no té disponibilitat';
    END IF;
    -- post resultat
    RETURN cap - numAssResInd - numAssResGrup;
EXCEPTION
    -- exc restaurant-sense-lloc
    WHEN raise_exception THEN
        RAISE EXCEPTION '%',SQLERRM;
    -- exc restaurant-no-existeix, menu-no-ofert
    WHEN foreign_key_violation THEN
        RAISE EXCEPTION 'El restaurant no existeix o el menú no ofert pel restaurant';
    -- altres errors
    -- WHEN OTHERS THEN
    -- RAISE EXCEPTION 'Altres Errors';
END;
$$LANGUAGE plpgsql;

-- Dades per fer proves
INSERT INTO persones VALUES ('Carme',30,null,'carmepq',0);
INSERT INTO persones VALUES ('Josep',36,null,'josepty',0);

INSERT INTO restaurants VALUES ('Follia','Sant Just',100);
INSERT INTO restaurants VALUES ('Petit Pau','Barcelona',50);

INSERT INTO menus VALUES ('Tardor',50);
INSERT INTO menus VALUES ('Primavera',30);

INSERT INTO ofertesMenus VALUES ('Follia','Tardor');
```



```
INSERT INTO ofertesMenus VALUES ('Follia','Primavera');
INSERT INTO ofertesMenus VALUES ('Petit Pau','Tardor');

INSERT INTO reservesIndividuals VALUES ('Carme','2017-03-03','Follia',5);

SELECT * FROM novaReservaGrup('Carme','Follia','2017-03-03',6,'Primavera');
```

A. Suposarem el cas Concrete Table Inheritance (apartat 2). Amb Triggers. Restriccions noves.

Esquema Conceptual	Base de Dades
Jerarquia Reserva (incomplete, disjoint) – una reserva o és individual o és de grup	<p>Trigger reservesDisjunes</p> <ul style="list-style-type: none"> Es dispara quan hi ha una inserció de reservesIndividuals o d'una reservaGrup Fa saltar una excepció si inserim una reserva individual que ja és de grup, o inserim una reserva de grup que ja és individual
RI4 - La capacitat d'un restaurant ha de ser més gran que el sumatori dels assistents de les reserves previstes per aquell restaurant en una data determinada.	<p>Trigger capacitatSuficient</p> <ul style="list-style-type: none"> Es dispara quan inserim una reserva (individual o de grup) i quan modifiquem el numAssistents d'una reserva (individual o de grup) i també quan modifiquem la capacitat d'un Restaurant Saltarà una excepció si la capacitat d'un restaurant és inferior a la suma d'assistents en reserves per una mateixa data.
RI 6 - Els menús que ha tastat en grup una persona són els mateixos que ha reservat.	<p>Trigger mantenimentTastat</p> <ul style="list-style-type: none"> Es dispara quan hi ha una inserció de reservesGrup, també quan hi ha una modificació del nomM d'una reserva de grup, i quan hi ha un esborrat de reservesGrup, i quan es modifica el nom de la persona que ha fet la reserva.. Mantindrà a la taula haTastatEnGrup una fila d'una persona nomP i menu nomM, en cas que la persona tingui una o més reservesGrup que hagi demanat el nomM. <p>Trigger noActualitzarHaTastat</p> <ul style="list-style-type: none"> Es dispara quan hi ha una inserció/esborrats a la taula haTastatEnGrup Saltarà una excepció si s'intenta inserir/esborrar files en aquesta taula.
RI7 - El numResGrup d'una persona és igual al número de reserves de grup que ha fet la persona	<p>Trigger mantenimentNumResGrup</p> <ul style="list-style-type: none"> Es dispara quan hi ha una inserció d'una reservaGrup, també quan hi ha un esborrat d'una reservaGrup, i quan es modifica el nom de la persona que ha fet la reserva. Mantindrà actualitzat l'atribut numResGrup d'una persona, incrementant-el en 1 si hi ha una inserció d'una reserva de grup de la persona, i decrementant en 1 si s'esborra una reserva de grup de la persona.

Suposarem el cas Concrete Table Inheritance (apartat 2). Amb Triggers. Implementació amb Stored Procedures

```
CREATE OR REPLACE FUNCTION novaReservaGrup(nomCli VARCHAR(50), nomRest VARCHAR(50),
                                           dr DATE, nAss INTEGER, nomMenu VARCHAR(50)) RETURNS
INTEGER AS $$
DECLARE
    cap integer;
    numAssResInd integer;
    numAssResGrup integer;

BEGIN
    -- pres per clientExisteix, dataOK, mes5assistents, reservaNoEx
    INSERT INTO reservesGrup
    VALUES(nomCli,dr,nomRest,nAss,nomMenu);
    -- numResGrup-incrementat ho fa el trigger mantenimentNumResGrup
    -- haTastat-Creat ho fa el trigger mantenimentTastat
    SELECT capacitat INTO cap
    FROM restaurants
    WHERE nomR = nomRest;
    SELECT SUM(numAssistents) INTO numAssResInd
    FROM reservesIndividuals
    WHERE nomR = nomRest AND data = dr;
    SELECT SUM(numAssistents) INTO numAssResGrup
    FROM reservesGrup
    WHERE nomR = nomRest AND data = dr;
    RETURN cap - numAssResInd - numAssResGrup;
EXCEPTION
    --WHEN exception_trigger THEN
    --RAISE EXCEPTION 'restaurant sense lloc';
    WHEN foreign_key_violation THEN
    RAISE EXCEPTION 'el restaurant no existeix o el menu no es ofert pel restaurant';
    WHEN OTHERS THEN RAISE EXCEPTION 'Altres Errors';
END;
$$LANGUAGE plpgsql;
```