

ZONES BIOCONTENCIÓ

Grup de Laboratori: L02

Professor: Gabriel Plana

Membres: Pau Ferré - Florian Serb

Data d'entrega: 27/05/2022

Índex

| | |
|-------------------------------------|-----------|
| 1. Decisions de disseny..... | 3 |
| 2. Creates..... | 4 |
| 3. Inserts..... | 7 |
| 4. Selects..... | 11 |
| 5. Updates..... | 14 |

1. Decisions de disseny

Hi ha dos tipus de motors encarregats d'emmagatzemar, gestionar i recuperar informació en una taula. Els més utilitzats són MySQL i InnoDB. Nosaltres hem decidit utilitzar **InnoDB**, ja que ofereix una fiabilitat i consistència superior a MySQL, ja que InnoDB ha començat a reemplaçar MySQL.

Un dels exemples més clars pel qual hem utilitzat **InnoDB** és que soporta els **FOREIGN KEY**, mentre que MySQL és incapaç de gestionar-ho.

Un altre punt en compte a l'hora de realitzar la pràctica ha sigut que a l'hora de crear les taules, hem utilitzat **CONSTRAINT**. Amb l'ajuda de **CONSTRAINT**, si per qualsevol error algun dels paràmetres no funciona correctament, es pot eliminar aquest paràmetre sense haver d'eliminar tota la taula.

A l'hora de dur a terme una consulta, en molts d'aquests hem utilitzat **INNER JOIN** per a poder unir diferents taules que no tenen cap relació en comú.

Com a últim punt, a l'hora de realitzar l'últim **UPDATE**, nosaltres tenim tots els empleats qualificats assignats a la seva zona de biocontenció i al seu laboratori i si algun d'ells és cap, també està assignat a responsable. Hem pensat que si actualitzem la informació de la columna responsables de la taula zona de biocontenció, tots els seus integrants passaran a estar únicament assignats sense ser responsables de cap zona.

2. Creates

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:

- El potencial de les armes biològiques es codifica mitjançant un número que ha d'estar entre 1 i 10.
- El nivell de perillositat estarà codificat amb una lletra que ha de ser A (=alt), M (=mitjà) o B (=baix).
- L'atribut tractat_signat només pot ser 'S', 'N' o desconegut.

```
CREATE DATABASE BD1;
```

```
USE BD1;
```

```
CREATE TABLE paisos(  
    nom VARCHAR(15) NOT NULL,  
    pot_desenv CHAR(1) NOT NULL,  
    tractat_signat CHAR(1),  
    CONSTRAINT PK_paisos PRIMARY KEY (nom),  
    CONSTRAINT CHK_pot_desenv_paisos CHECK (pot_desenv='S' OR  
    pot_desenv='N'),  
    CONSTRAINT CHK_tractat_signat_paisos CHECK (tractat_signat='S'  
    OR tractat_signat='N')  
) engine=innodb;  
CREATE TABLE laboratoris(  
    codi INT(3) NOT NULL,  
    nom VARCHAR(15) NOT NULL,  
    pais VARCHAR(15) NOT NULL,  
    CONSTRAINT PK_laboratoris PRIMARY KEY(codi),  
    CONSTRAINT FK_laboratoris_paisos FOREIGN KEY (pais) REFERENCES  
    PAISOS(nom)  
) engine=innodb;
```

```
CREATE TABLE zones_biocontencio(  
    codi INT(3) NOT NULL,  
    codiLab INT(3) NOT NULL,  
    nivell CHAR(1) NOT NULL,  
    responsable INT,  
    CONSTRAINT PK_zones PRIMARY KEY(codi, codiLab),  
    CONSTRAINT FK_zones_laboratoris FOREIGN KEY (codiLab)  
    REFERENCES LABORATORIS(codi),
```

```
        CONSTRAINT CHK_nivell_zones CHECK (nivell='A' OR nivell='M' OR
        nivell='B')
    ) engine=innodb;

CREATE TABLE armes_biologiques(
    nom VARCHAR(15) NOT NULL,
    data DATE NOT NULL,
    potencial INT(2) NOT NULL,
    zona INT(3) NOT NULL,
    lab INT(3) NOT NULL,
    CONSTRAINT PK_armes PRIMARY KEY (nom),
    CONSTRAINT FK_armes_zones FOREIGN KEY (zona, lab) REFERENCES
    ZONES_BIOCONTENCIO(codi, codiLab),
    CONSTRAINT CHK_potencial_armes CHECK (potencial BETWEEN 1 AND
    10)
) engine=innodb;

CREATE TABLE empleats(
    num_pass INT NOT NULL,
    nom VARCHAR(15) NOT NULL,
    CONSTRAINT PK_empleats PRIMARY KEY (num_pass)
) engine=innodb;

CREATE TABLE ordinaris(
    num_pass INT NOT NULL,
    CONSTRAINT PK_ordinaris PRIMARY KEY (num_pass),
    CONSTRAINT FK_ordinaris_empleats FOREIGN KEY (num_pass)
    REFERENCES EMPLEATS(num_pass)
) engine=innodb;

CREATE TABLE qualificats(
    num_pass INT NOT NULL,
    titulacio VARCHAR(50) NOT NULL,
    zona_assignada INT(3) NOT NULL,
    lab INT(3) NOT NULL,
    CONSTRAINT PK_qualificats PRIMARY KEY (num_pass),
    CONSTRAINT FK_qualificats_empleats FOREIGN KEY (num_pass)
    REFERENCES EMPLEATS(num_pass),
    CONSTRAINT FK_qualificats_zones FOREIGN KEY (zona_assignada,
    lab) REFERENCES ZONES_BIOCONTENCIO(codi, codiLab)
) engine=innodb;

CREATE TABLE assignacions(
    data DATE NOT NULL,
```

```
empl_ord INT NOT NULL,  
zona INT(3) NOT NULL,  
lab INT(3) NOT NULL,  
data_fi DATE,  
CONSTRAINT PK_assignacions PRIMARY KEY(data, empl_ord),  
CONSTRAINT FK_assignacions_ordinaris FOREIGN KEY (empl_ord)  
REFERENCES ORDINARIS(num_pass),  
CONSTRAINT FK_assignacions_zones FOREIGN KEY (zona, lab)  
REFERENCES ZONES_BIOCONTENCIO(codi, codiLab)  
) engine=innodb;
```

```
ALTER TABLE zones_biocontencio  
ADD CONSTRAINT FK_responsables_zones  
FOREIGN KEY (responsable)  
REFERENCES qualificats(num_pass);
```

3. Inserts

Creeu un segon script que insereixi les dades que considereu necessàries per tal de poder provar les consultes dels apartats següents.

```
INSERT INTO països(nom, pot_desenv, tractat_signat)
VALUES('espanya', 'S', 'S'),
      ('finlandia', 'N', 'S'),
      ('alemanya', 'N', 'N'),
      ('italia', 'S', 'N'),
      ('romania', 'S', 'S'),
      ('andorra', 'S', 'S');
```

```
INSERT INTO laboratoris(codi, nom, país)
VALUES(100, 'BCN-magno', 'espanya'),
      (101, 'MUN-nenuco', 'alemanya'),
      (102, 'AND-vida', 'andorra'),
      (103, 'BCN-XXX', 'espanya'),
      (200, 'BUC-dacia', 'romania'),
      (201, 'HEL-johnson', 'finlandia'),
      (202, 'ROM-pisto', 'italia'),
      (203, 'TAM-cell', 'finlandia');
```

```
INSERT INTO zones_biocontencio(codi, codiLab, nivell)
VALUES(100, 102, 'B'),
      (101, 101, 'A'),
      (102, 203, 'A'),
      (103, 100, 'B'),
      (200, 202, 'A'),
      (201, 103, 'M'),
      (202, 200, 'M'),
      (203, 201, 'B');
```

```
INSERT INTO armes_biologiques(nom, data, potencial, zona, lab)
VALUES('Àntrax', '2000-12-09', 3, 100, 102),
      ('Tularemia', '2017-11-30', 6, 101, 101),
      ('Botulismo', '2009-01-10', 4, 102, 203),
      ('Viruela', '2014-10-04', 1, 103, 100),
      ('Virus del ébola', '2007-09-04', 9, 200, 202),
      ('Peste', '2016-07-15', 5, 201, 103),
      ('Bunyavirus', '2014-12-09', 4, 202, 200),
      ('Còlera', '2017-02-15', 2, 203, 201),
      ('COVID-19', '2021-01-30', 7, 100, 102),
      ('mathipo', '2022-05-21', 9, 201, 103);
```

```
INSERT INTO empleats(num_pass, nom)
VALUES(1, 'Pepe Ciruela'),
      (2, 'Juan Grano'),
      (3, 'Pedro Pica'),
      (4, 'Carlos Fernandez'),
      (5, 'Ivan Garcia'),
      (6, 'Jack Diamon'),
      (7, 'Gustabo Herrera'),
      (8, 'Ruben Jimenez'),
      (9, 'Florian Serb'),
      (10, 'Adria Artunyedo'),
      (11, 'Ricard Ferguson'),
      (12, 'Angela Patric'),
      (13, 'Paula Sancho'),
      (14, 'Jordi Gomez'),
      (15, 'Sergio Hernandez'),
      (16, 'David Ferrero'),
      (17, 'Xavier Romeu'),
      (18, 'Sofia Bernaltes'),
      (19, 'Carla Montanyes'),
      (20, 'Maria Lago'),
      (21, 'Marco Polo'),
      (22, 'Josep Trujillo'),
      (23, 'Santi Macdon'),
      (24, 'Victor Serpin'),
      (25, 'Cristian Hiust'),
      (26, 'Marc Gavilan'),
      (27, 'Miquel Masdeu'),
      (28, 'Ana Meseguer'),
      (29, 'Nahia Rudo'),
      (30, 'Jordi Guasch'),
      (31, 'Alberto Gonzalez'),
      (32, 'Ana Anguera'),
      (33, 'Arnau Rubiales'),
      (34, 'Pau Ferré'),
      (35, 'Franco Franci'),
      (36, 'Marina Dalmau'),
      (37, 'Andrea Navarro'),
      (38, 'Raquel Rudo'),
      (39, 'Didac Estrada'),
      (40, 'Pol Rodriguez');
```

```
INSERT INTO ordinariis(num_pass)
VALUES(4),
```




(5), (9), (10),
(14), (15), (19),
(20), (24), (25),
(29), (30), (34),
(35), (39), (40);

```
INSERT INTO qualificats(num_pass, titulacio, zona_assignada, lab)
VALUES(1, 'Llicenciatura en Químic Fàrmac biòleg', 102, 203),
      (2, 'Llicenciatura en Enginyeria Química', 102, 203),
      (3, 'Enginyeria química industrial', 203, 201),
      (6, 'Enginyeria química industrial', 202, 200),
      (7, 'Enginyeria química industrial', 202, 200),
      (8, 'Enginyeria química', 100, 102),
      (11, 'Enginyeria química', 201, 103),
      (12, 'Enginyeria química industrial', 201, 103),
      (13, 'Llicenciatura en Enginyeria Química', 200, 202),
      (16, 'Llicenciatura en Químic Fàrmac biòleg', 200, 202),
      (17, 'Llicenciatura en Enginyeria Química', 200, 202),
      (18, 'Enginyeria química', 200, 202),
      (21, 'Llicenciatura en Enginyeria Química', 200, 202),
      (22, 'Llicenciatura en Químic Fàrmac biòleg', 103, 100),
      (23, 'Enginyeria química industrial', 103, 100),
      (26, 'Llicenciatura en Enginyeria Química', 100, 102),
      (27, 'Llicenciatura en Químic Fàrmac biòleg', 100, 102),
      (28, 'Enginyeria química', 201, 103),
      (31, 'Llicenciatura en Enginyeria Química', 100, 102),
      (32, 'Llicenciatura en Químic Fàrmac biòleg', 101, 101),
      (33, 'Enginyeria química', 101, 101),
      (36, 'Enginyeria química industrial', 101, 101),
      (37, 'Llicenciatura en Químic Fàrmac biòleg', 101, 101),
      (38, 'Llicenciatura en Químic Fàrmac biòleg', 101, 101);
```

```
INSERT INTO assignacions(data, empl_ord, zona, lab, data_fi)
VALUES('2010-09-12', 4, 100, 102, '2014-01-26'),
      ('2008-06-29', 5, 100, 102, '2009-07-29'),
      ('2022-05-12', 9, 101, 101, NULL),
      ('2010-03-13', 10, 101, 101, '2019-10-10'),
      ('2009-06-15', 14, 102, 203, '2017-08-26'),
      ('2020-07-01', 15, 102, 203, NULL),
      ('2012-02-13', 19, 103, 100, '2020-12-13'),
      ('2000-12-09', 20, 103, 100, '2007-03-04'),
      ('2018-02-23', 24, 200, 202, NULL),
      ('2019-08-23', 25, 200, 202, NULL),
```



```
('2012-03-10', 29, 201, 103, NULL),  
( '2017-10-15', 30, 201, 103, '2018-03-09'),  
( '2018-11-25', 34, 202, 200, '2020-05-06'),  
( '2016-07-25', 35, 202, 200, NULL),  
( '2010-05-11', 39, 203, 201, '2015-02-27'),  
( '2006-10-30', 40, 203, 201, '2018-10-21'),  
( '2014-01-27', 4, 200, 202, '2022-02-21'),  
( '2009-07-30', 5, 103, 100, NULL),  
( '2019-10-11', 10, 100, 102, NULL),  
( '2017-08-27', '14', 101, 101, NULL),  
( '2020-12-14', 19, 102, 203, '2022-01-16'),  
( '2007-03-05', 20, 203, 201, NULL),  
( '2018-03-10', 30, 103, 100, NULL),  
( '2020-05-07', 34, 203, 201, '2022-03-27'),  
( '2015-02-28', 39, 103, 100, NULL),  
( '2018-10-22', 40, 201, 103, '2022-04-14'),  
( '2022-02-22', 4, 203, 201, NULL),  
( '2022-01-17', 19, 200, 202, NULL),  
( '2022-03-28', 34, 201, 103, NULL),  
( '2022-04-15', 40, 102, 203, NULL);
```

```
UPDATE zones_biocontencio SET responsable = 8 WHERE codi = 100;  
UPDATE zones_biocontencio SET responsable = 32 WHERE codi = 101;  
UPDATE zones_biocontencio SET responsable = 1 WHERE codi = 102;  
UPDATE zones_biocontencio SET responsable = 23 WHERE codi = 103;  
UPDATE zones_biocontencio SET responsable = 18 WHERE codi = 200;  
UPDATE zones_biocontencio SET responsable = 28 WHERE codi = 201;  
UPDATE zones_biocontencio SET responsable = 6 WHERE codi = 202;  
UPDATE zones_biocontencio SET responsable = 3 WHERE codi = 203;
```

4. Selects

Obtenir el codi i nom dels laboratoris que tenen alguna zona de biocontenció d'alt nivell de perillositat. Mostreu el resultat ordenat en ordre alfabètic segons el nom del laboratori.

```
SELECT DISTINCT laboratoris.codi, laboratoris.nom
FROM laboratoris, zones_biocontencio
WHERE zones_biocontencio.nivell = 'A'
ORDER BY nom ASC;
```

Obtenir els noms dels empleats ordinaris que no han estat mai assignats a cap zona de biocontenció de nivell de perillositat Alt.

```
SELECT DISTINCT empleats.nom
FROM empleats
INNER JOIN ordinaris
    ON ordinaris.num_pass = empleats.num_pass
INNER JOIN assignacions
    ON assignacions.empl_ord = ordinaris.num_pass
WHERE NOT EXISTS (
    SELECT ordinaris.num_pass
    FROM ordinaris
    INNER JOIN zones_biocontencio
        ON assignacions.zona = zones_biocontencio.codi
    WHERE zones_biocontencio.nivell = 'A'
);
```

Obtenir quines zones tenen més de 3 empleats qualificats. Concretament es demana el codi de la zona conjuntament amb el nom del laboratori, ordenat per laboratori i zona.

```
SELECT laboratoris.nom, zones_biocontencio.codi
FROM laboratoris
INNER JOIN zones_biocontencio
    ON laboratoris.codi = zones_biocontencio.codiLab
INNER JOIN qualificats
    ON zones_biocontencio.codi = qualificats.zona_assignada
GROUP BY qualificats.zona_assignada
HAVING COUNT(qualificats.zona_assignada) > 3
ORDER BY laboratoris.nom;
```

Obtenir quins empleats ordinaris han estat en totes les zones de biocontenció del laboratori que té per nom 'BCN-XXX'

```
SELECT DISTINCT ordinaris.num_pass
FROM ordinaris
INNER JOIN assignacions
    ON ordinaris.num_pass = assignacions.empl_ord
INNER JOIN laboratoris
    ON assignacions.lab = laboratoris.codi
INNER JOIN zones_biocontencio
    ON laboratoris.codi = zones_biocontencio.codiLab
WHERE EXISTS (
    SELECT zones_biocontencio.codi FROM zones_biocontencio
    WHERE laboratoris.nom = 'BCN-XXX'
);
```

Obtenir quins són els empleats ordinaris que han estat en més zones de biocontenció que l'empleat anomenat "Jordi Guasch".

```
SELECT empleats.nom, COUNT(assignacions.zona)
FROM EMPLEATS
INNER JOIN ordinaris
    ON empleats.num_pass = ordinaris.num_pass
INNER JOIN assignacions
    ON ordinaris.num_pass = assignacions.empl_ord
INNER JOIN zones_biocontencio
    ON assignacions.zona = zones_biocontencio.codi
GROUP BY empleats.nom
HAVING COUNT(assignacions.empl_ord) > (SELECT
COUNT(assignacions.empl_ord)
    FROM assignacions
    INNER JOIN ordinaris
        ON ordinaris.num_pass = assignacions.empl_ord
    INNER JOIN empleats
        ON empleats.num_pass = ordinaris.num_pass
    INNER JOIN zones_biocontencio
        ON assignacions.zona = zones_biocontencio.codi
    WHERE empleats.nom = 'Jordi Guasch');
```

Creeu una vista que obtingui el llistat d'empleats amb el codi de la zona que té assignada. Tingueu en compte que també volem que els responsables de zona apareguin com a empleats assignats, i que no volem que cap empleat aparegui més d'una vegada.

```
CREATE VIEW VW_zona_empleats
AS SELECT DISTINCT empleats.nom, zones_biocontencio.codi
   FROM empleats
   INNER JOIN ordinaris
       ON empleats.num_pass = ordinaris.num_pass
   INNER JOIN assignacions
       ON ordinaris.num_pass = assignacions.empl_ord
   INNER JOIN zones_biocontencio
       ON assignacions.zona = zones_biocontencio.codi
 WHERE assignacions.data_fi IS NULL
 UNION
 SELECT DISTINCT empleats.nom, zones_biocontencio.codi
   FROM empleats
   INNER JOIN qualificats
       ON empleats.num_pass = qualificats.num_pass
   INNER JOIN zones_biocontencio
       ON qualificats.num_pass = zones_biocontencio.responsable;

SELECT * FROM VW_zona_empleats;
```

5. Updates

Assignar el nivell de perillositat a 'A' (=alt) a aquelles zones de biocontenció que desenvolupin alguna arma de potencial superior a 5.

```
UPDATE zones_biocontencio  
INNER JOIN armes_biologiques  
ON armes_biologiques.zona = zones_biocontencio.codi  
SET nivell = 'A'  
WHERE armes_biologiques.potencial > 5;
```

Destituir a tots els responsables dels seus càrrecs, és a dir, passar-los de la relació de “cap” a la relació “assignats”. Potser et sigui necessari fer un petit canvi en el disseny per tal de poder aconseguir-ho. Si aquest és el vostre cas, escriviu també la sentència per tal de fer aquest canvi.

```
UPDATE zones_biocontencio  
SET zones_biocontencio.responsable = NULL;  
  
-- Motiu explicat a les decisions de disseny
```