

Exercicis SQL 1

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 27 February 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (30%)

Doneu una sentència SQL per obtenir els números i els noms dels departament situats a MADRID, que tenen empleats que guanyen més de 200000.

Pel joc de proves que trobareu al fitxer adjunt, la sortida ha de ser:

NUM_DPT	NOM_DPT
5	VENDES

[Fitxer adjunt](#)

Solució: `SELECT DISTINCT e.num_dpt, d.nom_dpt FROM departaments d, empleats e WHERE e.num_dpt = d.num_dpt AND d.ciutat_dpt = 'MADRID' AND e.sou >200000`

Exercicis SQL 1

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 27 February 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (30%)

- Pregunta 2 (30%)

Doneu una sentència SQL per obtenir el nom del departament on treballa i el nom del projecte on està assignat l'empleat número 2

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

Nom_dpt	Nom_proj
MARKETING	IBDVID

[Fitxer adjunt](#)

Solució:

```
select d.Nom_dpt, p.Nom_proj from departaments d, empleats e, projectes p WHERE e.Num_dpt = d.Num_dpt AND e.Num_empl =2 AND e.Num_proj = p.Num_proj;
```

Exercicis SQL 1

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 27 February 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (30%)

+ Pregunta 2 (30%)

- Pregunta 3 (40%)

Obtenir per cada departament situat a MADRID la mitjana dels sous dels seus empleats. Concretament, cal donar el número de departament, el nom de departament i la mitjana del sou.

Pel joc de proves que trobareu al fitxer adjunt, la sortida ha de ser:

NUM_DPT	NOM_DPT	SOU
5	VENDES	250000

[Fitxer adjunt](#)

Solució:

```
select d.num_dpt, d.nom_dpt, avg(e.sou) as SOU from empleats e , departaments d where d.num_dpt = e.num_dpt and d.ciutat_dpt = 'MADRID' group by d.num_dpt;
```

Exercicis SQL 2

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 6 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (20%)

Doneu una sentència SQL per obtenir el número i el nom dels departaments que no tenen cap empleat que visqui a MADRID.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

NUM_DPT	NOM_DPT
3	MARKETING

[Fitxer adjunt](#)

Solució:

```
select d.num_dpt, d.nom_dpt  
from departaments d  
where not exists ( select *  
                        from empleats e  
                        where e.num_dpt = d.num_dpt and e.ciutat_empl = 'MADRID')
```

Exercicis SQL 2

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 6 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ **Pregunta 1** (20%)

- **Pregunta 2** (30%)

Doneu una sentència SQL per obtenir les ciutats on hi viuen empleats però no hi ha cap departament.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

CIUTAT_EMPL
GIRONA

[Fitxer adjunt](#)

Solució:

```
select distinct e.ciutat_empl from empleats e where not exists (select d.ciutat_dpt from departaments d where e.ciutat_empl = d.ciutat_dpt);
```

Exercicis SQL 2

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 6 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ [Pregunta 1 \(20%\)](#)

+ [Pregunta 2 \(30%\)](#)

- [Pregunta 3 \(30%\)](#)

Doneu una sentència SQL per obtenir el número i nom dels departaments que tenen dos o més empleats que viuen a ciutats diferents.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

NUM_DPT	NOM_DPT
3	MARKETING

[Fitxer adjunt](#)

Solució:

```
select d.num_dpt, d.nom_dpt
from departaments d, empleats e
where d.num_dpt = e.num_dpt
group by d.num_dpt
having count(distinct e.ciutat_empl) >= 2
```

Exercicis SQL 2

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 6 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (20%)

+ Pregunta 2 (30%)

+ Pregunta 3 (30%)

- Pregunta 4 (20%)

Tenint en compte l'esquema de la BD que s'adjunta, proposeu una sentència de creació de les taules següents:

comandes(numComanda, instantComanda, client, encarregat, supervisor)

productesComprats(numComanda, producte, quantitat, preu)

La taula *comandes* conté les comandes fetes.

La taula *productesComprats* conté la informació dels productes comprats a les comandes de la taula *comandes*.

En la creació de les taules cal que tingueu en compte que:

- No hi poden haver dues comandes amb un mateix número de comanda.
 - Un client no pot fer dues comandes en una mateix instant.
 - L'encarregat és un empleat que ha d'existir necessàriament a la base de dades, i que té sempre tota comanda.
 - El supervisor és també un empleat de la base de dades i que s'assigna a algunes comandes en certes circumstàncies.
 - No hi pot haver dues vegades un mateix producte en una mateixa comanda. Ja que en cas de el client comprar més d'una unitat d'un producte en una mateixa comanda s'indica en l'atribut quantitat.
 - Un producte sempre s'ha comprat en una comanda que ha d'existir necessàriament a la base de dades.
 - La quantitat de producte comprat en una comanda no pot ser nul, i té com a valor per defecte 1.
 - Els atributs numComanda, instantComanda, quantitat i preu són de tipus *integer*.
 - Els atributs client, producte són *char(30)*, i *char(20)* respectivament.
 - L'atribut instantComanda no pot tenir valors nuls.
- Respecteu els noms i l'ordre en què apareixen les columnes (fins i tot dins la clau o claus que calgui definir). Tots els noms s'han de posar en majúscules/minúscules com surt a l'enunciat.

NOTA 8.5

CREATE TABLE comandes

(numComanda INTEGER,

instantComanda INTEGER NOT NULL UNIQUE,

client CHAR(30),

encarregat INTEGER NOT NULL,

supervisor INTEGER,

PRIMARY KEY (numComanda),

FOREIGN KEY (encarregat) REFERENCES EMPLEATS (NUM_EMPL),

FOREIGN KEY (supervisor) REFERENCES EMPLEATS (NUM_EMPL));

CREATE TABLE productesComprats

(numComanda INTEGER,

producte CHAR(20),
quantitat INTEGER DEFAULT 1 NOT NULL,
preu INTEGER,
PRIMARY KEY (numComanda, producte),
UNIQUE (numComanda, producte),
FOREIGN KEY (numComanda) REFERENCES comandes
(numComanda));

Exercicis SQL 3

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 13 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (30%)

Doneu una sentència SQL per obtenir el nom dels professors que o bé se sap el seu número de telèfon (valor diferent de null) i tenen un sou superior a 2500, o bé no se sap el seu número de telèfon (valor null) i no tenen cap assignació a un despatx amb superfície inferior a 20.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

NomProf

toni

[Fitxer adjunt](#)

Solució:

```
select p.nomprof
from professors p
where (p.telefon is not null and p.sou > 2500) or (p.telefon is null and not exists (
select *
from despatxos d, assignacions a
where a.numero = d.numero and a.dni = p.dni and a.modul = d.modul and d.superficie < 20))
```


Exercicis SQL 3

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 13 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (30%)

- Pregunta 2 (40%)

Donar una sentència SQL per obtenir els professors que tenen alguna assignació finalitzada (instantFi diferent de null) a un despatx amb superfície superior a 15 i que cobren un sou inferior o igual a la mitjana del sou de tots els professors. En el resultat de la consulta ha de sortir el dni del professor, el nom del professor, i el darrer instant en què el professor ha estat assignat a un despatx amb superfície superior a 15.

Pel joc de proves que trobareu al fitxer adjunt, la sortida ha de ser:

DNI NomProf Darrer_instant

111 toni 344

[Fitxer adjunt](#)

Solució:

```
select distinct p.dni, p.nomProf, max(a.instantFi) as Darrer_instant
  from professors p natural inner join assignacions a natural inner join despatxos d
 where a.instantFi is not null and d.superficie > 15 and p.sou <= (select avg(p2.sou)
 from professors p2)
 group by p.dni;
```

Exercicis SQL 3

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 13 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (30%)

+ Pregunta 2 (40%)

- Pregunta 3 (10%)

Suposem la base de dades que podeu trobar al fitxer adjunt.
Suposem que aquesta base de dades està en un estat on no hi ha cap fila.
Doneu una seqüència de sentències SQL d'actualització (INSERTs i/o UPDATEs) sobre la taula que assignacions que violi la integritat referencial de la clau forana que referencia la taula Despatxos.
Les sentències **NOMÉS** han de violar aquesta restricció.

[Fitxer adjunt](#)

Solució:

```
insert into professors values ('111','iosif','1212',100000);  
insert into despatxos values ('omega','1312',1312);  
insert into assignacions values ('111','omega','1213',4,null);
```

Exercicis SQL 3

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 13 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (30%)

+ Pregunta 2 (40%)

+ Pregunta 3 (10%)

- Pregunta 4 (20%)

Suposeu la base de dades que podeu trobar al fitxer adjunt.
Doneu una seqüència de sentències SQL d'actualització (INSERTs i/o UPDATEs) de tal manera que, un cop executades, el resultat de la consulta següent sigui el que s'indica. El nombre de files de cada taula ha de ser el més petit possible, i hi ha d'haver com a màxim un professor.

Per a la consulta:

```
Select count(*) as quant  
From assignacions ass  
Where ass.instantlnici>50  
Group by ass.instantlnici  
order by quant;
```

El resultat haurà de ser:

```
quant  
1  
2
```

[Fitxer adjunt](#)

Solució:

```
insert into professors values('111','toni','3111',100);  
  
insert into despatsos values('omega','119',16);  
insert into despatsos values('omega','120',16);  
  
insert into assignacions values('111','omega','119',109,344);  
insert into assignacions values('111','omega','119',345,null);  
insert into assignacions values('111','omega','120',109,null);
```

Exercicis àlgebra relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 20 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

Recordeu que en el corrector d'Àlgebra Relacional:

Els noms de les taules o relacions intermèdies o resultat han de tenir per nom un únic caràcter.

Teniu més informació al manual d'àlgebra del corrector [Manual d'Àlgebra Relacional al LearnSQL](#).

- Pregunta 1 (25%)

Doneu una seqüència d'operacions d'àlgebra relacional per obtenir el nom del departament on treballa i el nom del projecte on està assignat l'empleat número 2.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

Nom_dpt	Nom_proj
MARKETING	IBDVID

[Fitxer adjunt](#)

Solució:

```
A = empleats(num_empl = 2)
C = A*departaments
D = C*projectes
B = D[nom_dpt,nom_proj]
```

Exercicis àlgebra relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 20 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

Recordeu que en el corrector d'Àlgebra Relacional:

Els noms de les taules o relacions intermèdies o resultat han de tenir per nom un únic caràcter.

Teniu més informació al manual d'àlgebra del corrector [Manual d'Àlgebra Relacional al LearnSQL](#).

+ Pregunta 1 (25%)

- Pregunta 2 (25%)

Doneu una seqüència d'operacions d'àlgebra relacional per obtenir el número i nom dels departaments tals que tots els seus empleats viuen a MADRID. El resultat no ha d'incloure aquells departaments que no tenen cap empleat.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

Num_dpt	Nom_dpt
3	MARKETING

[Fitxer adjunt](#)

Solució:

```
A = empleats(CIUTAT_EMPL<>'MADRID')
B = departaments*A
C = B[num_dpt,nom_dpt,planta,edifici,ciutat_dpt]
E = departaments-C
F = E*empleats
R = F[num_dpt,nom_dpt]
```

Exercicis àlgebra relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 20 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

Recordeu que en el corrector d'Àlgebra Relacional:

Els noms de les taules o relacions intermèdies o resultat han de tenir per nom un únic caràcter.

Teniu més informació al manual d'àlgebra del corrector [Manual d'Àlgebra Relacional al LearnSQL](#).

+ [Pregunta 1 \(25%\)](#)

+ [Pregunta 2 \(25%\)](#)

- [Pregunta 3 \(25%\)](#)

Doneu una seqüència d'operacions de l'àlgebra relacional per obtenir el número i nom dels departaments que tenen dos o més empleats que viuen a ciutats diferents.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

Num_dpt	Nom_dpt
3	MARKETING

[Fitxer adjunt](#)

Solució:

```
A = empleats[num_dpt,ciutat_empl]
B = A{ciutat_empl -> ciutat, num_dpt -> num}
C = A[ciutat_empl <> ciutat, num_dpt = num]B
D = C*departaments
E = D[num_dpt,num_dpt]
```

Exercicis àlgebra relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 20 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

Recordeu que en el corrector d'Àlgebra Relacional:

Els noms de les taules o relacions intermèdies o resultat han de tenir per nom un únic caràcter.

Teniu més informació al manual d'àlgebra del corrector [Manual d'Àlgebra Relacional al LearnSQL](#).

+ Pregunta 1 (25%)

+ Pregunta 2 (25%)

+ Pregunta 3 (25%)

- Pregunta 4 (25%)

Donar una seqüència d'operacions d'àlgebra relacional per obtenir informació sobre els despatxos que només han estat ocupats per professors amb sou igual a 100000. Es vol obtenir el modul i el numero d'aquests despatxos.

Pel joc de proves que trobareu al fitxer adjunt, la sortida ha de ser:

Modul Numero
Omega 128

[Fitxer adjunt](#)

Solució:

```
A=despatxos[modul,numero]
B=professors(sou<=100000)
C=B*assignacions
D=C[modul,numero]
E=A-D
F=E*assignacions
R=F[dni,modul,numero,instantInici]
```

Correcció $R=F[\text{mòdul},\text{numero}]$

Exercicis de SQL i Àlgebra Relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 27 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (20%)

Donar una sentència SQL per obtenir per cada mòdul on hi hagi despatxos, la suma de les durades de les assignacions finalitzades (instantFi diferent de null) a despatxos del mòdul. El resultat ha d'estar ordenat ascendentment pel nom del mòdul.

Pel joc de proves que trobareu al fitxer adjunt, la sortida ha de ser:

MODUL	SUMAA
Omega	235

[Fitxer adjunt](#)

Solució:

```
select a.MODUL, SUM(a.instantfi- a.instantinici) AS SUMAA
from assignacions a
where a.instantfi is not null
group by a.modul
order by a.modul ASC;
```


Exercicis de SQL i Àlgebra Relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 27 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (20%)

- Pregunta 2 (30%)

Doneu una sentència SQL per obtenir els departaments tals que tots els empleats del departament estan assignats a un mateix projecte.

No es vol que surtin a la consulta els departaments que no tenen cap empleat.

Es vol el número, nom i ciutat de cada departament.

Cal resoldre l'exercici **sense fer servir funcions d'agregació**.

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

Num_dpt	Nom_dpt	Ciutat_dpt
1	DIRECCIO	BARCELONA

[Fitxer adjunt](#)

Solució:

```
select DISTINCT d.num_dpt, d.nom_dpt, d.ciutat_dpt
from departaments d, empleats e
where e.num_dpt = d.num_dpt AND NOT EXISTS(select *
      from empleats e2
      where e.num_dpt = e2.num_dpt AND e.num_proj <> e2.num_proj)
order by d.num_dpt ASC;
```

Exercicis de SQL i Àlgebra Relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 27 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (20%)

+ Pregunta 2 (30%)

- Pregunta 3 (20%)

Doneu una seqüència d'operacions en àlgebra relacional per obtenir el nom dels professors que o bé tenen un sou superior a 2500, o bé que cobren menys de 2500 i no tenen cap assignació a un despatx amb superfície inferior a 20..

Pel joc de proves que trobareu al fitxer adjunt, la sortida seria:

NomProf

toni

[Fitxer adjunt](#)

Solució:

```
A = PROFESSORS(sou > 2500)
B = PROFESSORS * DESPATXOS
C = B * ASSIGNACIONS
D = C(superficie < 20)
X = D[nomProf]
T = C[nomProf]
M = T - X
J = professors[nomProf]
H = J - X
Q = A[nomProf]
K = M_u_H
R = Q_u_K
```

Exercicis de SQL i Àlgebra Relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 27 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (20%)

+ Pregunta 2 (30%)

+ Pregunta 3 (20%)

- Pregunta 4 (15%)

Doneu una sentència d'inserció de files a la taula `cost_ciutat` que l'ompli a partir del contingut de la resta de taules de la base de dades. Tingueu en compte el següent:

Cal inserir una fila a la taula `cost_ciutat` per cada ciutat on hi ha un o més departaments, però no hi ha cap departament que tingui empleats.

Per tant, només s'han d'inserir les ciutats on cap dels departaments situats a la ciutat tinguin empleats.

El valor de l'atribut `cost` ha de ser 0.

Pel joc de proves públic del fitxer adjunt, un cop executada la sentència d'inserció, a la taula `cost_ciutat` hi haurà les tuples següents:

CIUTAT_DPT	COST
BARCELONA	0

[Fitxer adjunt](#)

Solució:

```
INSERT INTO cost_ciutat (SELECT DISTINCT d.ciutat_dpt, 0
FROM departaments d
WHERE d.ciutat_dpt NOT IN (SELECT d2.ciutat_dpt
FROM empleats e1, departaments d2
WHERE e1.num_dpt = d2.num_dpt));
```

Exercicis de SQL i Àlgebra Relacional

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 27 March 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (20%)

+ Pregunta 2 (30%)

+ Pregunta 3 (20%)

+ Pregunta 4 (15%)

- Pregunta 5 (15%)

Tenint en compte l'esquema de la BD que s'adjunta, proposeu una sentència de creació de la taula següent:

presentacioTFG(idEstudiant, titolTFG, dniDirector, dniPresident, dniVocal, instantPresentacio, nota)

Hi ha una fila de la taula per cada treball final de grau (TFG) que estigui pendent de ser presentat o que ja s'hagi presentat.

En la creació de la taula cal que tingueu en compte que:

- No hi pot haver dos TFG d'un mateix estudiant.
- Tot TFG ha de tenir un títol.
- No hi pot haver dos TFG amb el mateix títol i el mateix director.
- El director, el president i el vocal han de ser professors que existeixin a la base de dades, i tot TFG té sempre director, president i vocal.
- El director del TFG no pot estar en el tribunal del TFG (no pot ser ni president, ni vocal).
- El president i el vocal no poden ser el mateix professor.
- L'identificador de l'estudiant i el títol del TFG són chars de 100 caràcters.
- L'instant de presentació ha de ser un enter diferent de nul.
- La nota ha de ser un enter entre 0 i 10.
- La nota té valor nul fins que s'ha fet la presentació del TFG.

Respecteu els noms i l'ordre en què apareixen les columnes (fins i tot dins la clau o claus que calgui definir). Tots els noms s'han de posar en majúscules/minúscules com surt a l'enunciat.

create table presentacioTFG

(idEstudiant CHAR(100),

unique(titolTFG,dniDirector),

titolTFG CHAR(100) not null,

dniDirector char(50) not null,

dniPresident char(50) not null,

dniVocal char(50) not null,

instantPresentacio integer not null,

nota integer check (nota >= 0 and nota <= 10) default null,

primary key (idEstudiant),

foreign key (dniDirector) references professors,

foreign key (dniPresident) references professors,

```

foreign key (dniVocal) references professors,

check (dniDirector <> dniPresident and dniDirector <>
dniVocal),

check (dniPresident <> dniVocal)

);

```

Pràctica de procediments

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 24 April 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (100%)

Donat un interval de DNIs, programar un procediment emmagatzemat "llistat_treb(dniIni,dniFi)" per obtenir la informació de cadascun dels treballadors amb un DNI d'aquest interval.

Per cada treballador de l'interval cal obtenir:

- Les seves dades personals: dni, nom, sou_base i plus
- En cas que el treballador tingui 5 o més lloguers actius, al llistat hi ha de sortir una fila per cadascun dels cotxes que té llogats.
- En qualsevol altre cas, al llistat hi ha de sortir una única fila amb les dades del treballador, i nul a la matrícula.

Tingueu en compte que:

- Es vol que retorneu els treballadors ordenats per dni i matrícula de forma ascendent.
- El tipus de les dades que s'han de retornar han de ser els mateixos que hi ha a la taula on estan definits els atributs corresponents.

El procediment ha d'informar dels errors a través d'excepcions. Les situacions d'error que heu d'identificar són les tipificades a la taula missatgesExcepcions, que podeu trobar definida i amb els inserts corresponents al fitxer adjunt. En el vostre procediment heu d'incloure, on s'identifiquin aquestes situacions, les sentències:

```
SELECT texte INTO missatge FROM missatgesExcepcions WHERE num=___; ( 1 o 2, depenent de l'error)
RAISE EXCEPTION '%',missatge;
```

On la variable missatge ha de ser una variable definida al vostre procediment.

Per joc de proves que trobareu al fitxer adjunt i la crida següent,

```
SELECT * FROM llistat_treb('11111111','33333333');
```

el resultat ha de ser:

DNI	Nom	Sou	Plus	Matrícula
22222222	Joan	1700	150	1111111111
22222222	Joan	1700	150	2222222222
22222222	Joan	1700	150	3333333333
22222222	Joan	1700	150	4444444444
22222222	Joan	1700	150	5555555555

CREATE TYPE treballador as (

dni varchar(8),

nom varchar(30),

sou_base real,

sou_plus real,

matricula varchar(30));

```
CREATE FUNCTION llistat_treb(dniIni char(8), dniFi char(8)) returns setof  
treballador AS $$
```

```
DECLARE
```

```
    miss varchar(50);
```

```
    tre treballador;
```

```
BEGIN
```

```
    for tre IN SELECT *
```

```
        FROM treballadors
```

```
        WHERE (dni >= dniIni and dniFI >= dni)
```

```
        ORDER BY dni ASC
```

```
LOOP
```

```
    IF ((select count(*)
```

```
        from lloguers_actius
```

```
        where tre.dni =dni) > 4)
```

```
    THEN
```

```
        for tre.matricula in select matricula
```

```
            from lloguers_actius
```

```
            WHERE tre.dni =dni
```

```
            ORDER BY matricula ASC
```

```
LOOP
```

```
    return next tre;
```

END loop;

ELSE

tre.matricula := null;

return next tre;

END IF;

END LOOP;

IF NOT found THEN

SELECT texte

INTO miss

FROM missatgesExcepcions

WHERE num=1;

RAISE EXCEPTION '%',miss;

END IF;

EXCEPTION

WHEN raise_exception then RAISE EXCEPTION '%',sqlerm;

WHEN others THEN

SELECT texte

INTO miss

FROM missatgesExcepcions

```

WHERE num=2;

RAISE EXCEPTION '%',miss;

return;

END;

```

```

$$LANGUAGE plpgsql;

```

Pràctica de disparadors

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Tuesday, 3 May 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (34%)

Implementar mitjançant disparadors la restricció d'integritat següent:
No es pot esborrar l'empleat 123 ni modificar el seu número d'empleat.

Cal informar dels errors a través d'excepcions tenint en compte les situacions tipificades a la taula missatgesExcepcions, que podeu trobar definida (amb els inserts corresponents) al fitxer adjunt. Concretament en el vostre procediment heu d'incloure, quan calgui, les sentències:
 SELECT texte INTO missatge FROM missatgesExcepcions WHERE num=___; (el número que sigui, depenent de l'error)
 RAISE EXCEPTION '%',missatge;
 La variable missatge ha de ser una variable definida al vostre procediment, i del mateix tipus que l'atribut corresponent de l'esquema de la base de dades.

Per joc de proves que trobareu al fitxer adjunt i la instrucció:
 DELETE FROM empleats WHERE nempl=123;
 La sortida ha de ser:

No es pot esborrar l'empleat 123 ni modificar el seu número d'empleat

```

CREATE OR REPLACE function no_tocar123() RETURNS TRIGGER AS $$

```

```

DECLARE

```

```

    missatge varchar(100);

```

```

BEGIN

```

```

    if (old.nempl = 123) then

```

```
SELECT texte INTO missatge FROM missatgesExcepcions WHERE  
num=1;
```

```
RAISE EXCEPTION '%',missatge;
```

```
else
```

```
if(tg_op = 'DELETE') then return old;
```

```
elseif (tg_op = 'UPDATE') then return new;
```

```
end if;
```

```
end if;
```

```
EXCEPTION
```

```
WHEN raise_exception THEN
```

```
RAISE EXCEPTION '%', SQLERRM;
```

```
end;
```

```
$$ language plpgsql;
```

```
create trigger mod123 before update of nempl on empleats
```

```
for each row execute procedure no_tocar123();
```

```
create trigger esb123 before delete on empleats
```

```
for each row execute procedure no_tocar123();
```


Pràctica de disparadors

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Tuesday, 3 May 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (34%)

Implementar mitjançant disparadors la restricció d'integritat següent:
No es pot esborrar l'empleat 123 ni modificar el seu número d'empleat.

Cal informar dels errors a través d'excepcions tenint en compte les situacions tipificades a la taula missatgesExcepcions, que podeu trobar definida (amb els inserts corresponents) al fitxer adjunt. Concretament en el vostre procediment heu d'incloure, quan calgui, les sentències:

SELECT texte INTO missatge FROM missatgesExcepcions WHERE num=___; (el número que sigui, depenent de l'error)

RAISE EXCEPTION '%',missatge;

La variable missatge ha de ser una variable definida al vostre procediment, i del mateix tipus que l'atribut corresponent de l'esquema de la base de dades.

Pel joc de proves que trobareu al fitxer adjunt i la instrucció:

DELETE FROM empleats WHERE nempl=123;

La sortida ha de ser:

No es pot esborrar l'empleat 123 ni modificar el seu número d'empleat

CREATE OR REPLACE function no_tocar123() RETURNS TRIGGER AS \$\$

DECLARE

missatge varchar(100);

BEGIN

if (old.nempl = 123) then

SELECT texte INTO missatge FROM missatgesExcepcions WHERE
num=1;

RAISE EXCEPTION '%',missatge;

else

```
        if(tg_op = 'DELETE') then return old;
        elseif (tg_op = 'UPDATE') then return new;
        end if;
    end if;
```

```
EXCEPTION
```

```
    WHEN raise_exception THEN
```

```
        RAISE EXCEPTION '%', SQLERRM;
```

```
end;
```

```
$$ language plpgsql;
```

```
create trigger mod123 before update of nempl on empleats
for each row execute procedure no_tocar123();
```

```
create trigger esb123 before delete on empleats
for each row execute procedure no_tocar123();
```

Pràctica de disparadors

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Tuesday, 3 May 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (34%)

- Pregunta 2 (33%)

Implementar mitjançant disparadors la restricció d'integritat següent:

No es poden esborrar empleats el dijous

Tiguen en compte que:

- Les restriccions d'integritat definides a la BD (primary key, foreign key,...) **es violen amb menys freqüència que la restricció comprovada per aquests disparadors.**

- El dia de la setmana serà el que indiqui la única fila que hi ha d'haver sempre insertada a la taula "dia". Com podreu veure en el joc de proves que trobareu al fitxer adjunt, el dia de la setmana és el 'dijous'. Per fer altres proves podeu modificar la fila de la taula amb el nom d'un altre dia de la setmana. IMPORTANT: Tant en el programa com en la base de dades poseu el nom del dia de la setmana en MINÚSCULES.

Cal informar dels errors a través d'excepcions tenint en compte les situacions tipificades a la taula missatgesExcepcions, que podeu trobar definida (amb els inserts corresponents) al fitxer adjunt. Concretament en el vostre procediment heu d'incloure, quan calgui, les sentències:

SELECT texte INTO missatge FROM missatgesExcepcions WHERE num=___;(el número que sigui, depenent de l'error)

RAISE EXCEPTION '%',missatge;

La variable missatge ha de ser una variable definida al vostre procediment, i del mateix tipus que l'atribut corresponent de l'esquema de la base de dades.

Pel joc de proves que trobareu al fitxer adjunt i la instrucció:

DELETE FROM empleats WHERE salari<=1000

la sortida ha de ser:

No es poden esborrar empleats el dijous

create or replace function esborrar() returns trigger as \$\$

declare

missatge varchar(100);

dj varchar(10);

begin

select dia into dj from dia;

if (dj = 'dijous')

then

SELECT texte INTO missatge

```
FROM missatgesExcepcions
```

```
WHERE num= 1; --(el número que sigui, depenent de l'error)
```

```
RAISE EXCEPTION '%',missatge;
```

```
else return old;
```

```
end if;
```

```
exception
```

```
when raise_exception then
```

```
        RAISE EXCEPTION '%', SQLERRM;
```

```
end
```

```
$$ language plpgsql;
```

```
create trigger dij before delete on empleats
```

```
for each statement execute procedure esborrar();
```

Pràctica de disparadors

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Tuesday, 3 May 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (34%)

+ Pregunta 2 (33%)

- Pregunta 3 (33%)

En aquest exercici es tracta de mantenir de manera automàtica, mitjançant triggers, l'atribut derivat *import* de la taula *comandes*.

En concret, l'import d'una comanda és igual a la suma dels resultats de multiplicar per cada línia de comanda, la quantitat del producte de la línia pel preu del producte .

Només heu de considerar les operacions de tipus INSERT sobre la taula línies de comandes.

Pel joc de proves que trobareu al fitxer adjunt, i la sentència: **INSERT INTO líniesComandes VALUES (110, 'p111', 2);** La sentència s'executarà sense cap problema, i l'estat de la taula de comandes després de la seva execució ha de ser:

numcomanda	instantfeta	instantservida	numtelf	import
110	1091	1101	null	30

create or replace function mantenir() returns trigger as \$\$

DECLARE

missatge varchar(100);

BEGIN

UPDATE comandes

SET "import" = "import"+new.quantitat*(SELECT p.preu

FROM productes p

WHERE new.idProducte =

p.idproducte)

WHERE numcomanda = new.numcomanda;

RETURN NULL;

END;

\$\$language plpgsql;

CREATE TRIGGER auto AFTER INSERT ON líniescomandes FOR EACH ROW
EXECUTE PROCEDURE mantenir();

Pràctica de procediments emmagatzemats i disparadors

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 8 May 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (50%)

En aquest exercici es tracta definir els disparadors necessaris sobre empleats2 (veure definició de la base de dades al fitxer adjunt) per mantenir la restricció següent:

Els valors de l'atribut ciutat1 de la taula empleats1 han d'estar inclosos en els valors de ciutat2 de la taula empleats2

Per mantenir la restricció, la idea és que:

En lloc de treure un missatge d'error en cas que s'intenti executar una sentència sobre empleats2 que pugui violar la restricció,

cal executar operacions compensatòries per assegurar el compliment de l'assertió. En concret aquestes operacions compensatòries ÚNICAMENT podran ser operacions DELETE.

Pel joc de proves que trobareu al fitxer adjunt, i la sentència:

DELETE FROM empleats2 WHERE nemp2=1;

La sentència s'executarà sense cap problema, i l'estat de la base de dades just després ha de ser:

Taula empleats1

nemp1	nom1	ciutat1
1	joan	bcn
2	maria	mad

Taula empleats2

nemp2	nom2	ciutat2
2	pere	mad
3	enric	bcn

CREATE FUNCTION delete_empleat() RETURNS TRIGGER AS \$\$

BEGIN

if (not exists (select *

from empleats2

where ciutat2 = old.ciutat2))

then

delete from empleats1

where ciutat1 = old.ciutat2;

end if;

return null;

END;

\$\$ language plpgsql;

create trigger ciutats after delete on empleats2

for each row execute procedure delete_empleat();

create trigger ciutat_update after update of ciutat2 on empleats2

for each row execute procedure delete_empleat();

Pràctica de procediments emmagatzemats i disparadors

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 8 May 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

+ Pregunta 1 (50%)

- Pregunta 2 (50%)

Disposem de la base de dades del fitxer adjunt que gestiona clubs esportius i socis d'aquests clubs. Cal implementar un procediment emmagatzemat "assignar_individual(nomSoci,nomClub)".

El procediment ha de:

- Enregistrar l'assignació del soci *nomSoci* al club *nomClub*, inserint la fila corresponent a la taula *Socisclubs*.
- Si el club *nomClub* passa a tenir més de 5 socis, inserir el club a la taula *Clubs_amb_mes_de_5_socis*.
- El procediment no retorna cap resultat.

Les situacions d'error que cal identificar són les tipificades a la taula *missatgesExcepcions*.

Quan s'identifiqui una d'aquestes situacions cal generar una excepció:

```
SELECT texte INTO missatge FROM missatgesExcepcions WHERE num=___; ( 1 .. 5, depenent de l'error)
RAISE EXCEPTION '%',missatge; (missatge ha de ser una variable definida al vostre procediment)
```

Suposem el joc de proves que trobareu al fitxer adjunt i la sentència

select * from assignar_individual('anna','escacs');

La sentència s'executarà sense cap problema, i l'estat de la base de dades just després ha de ser:

Taula Socisclubs

anna	escacs
joanna	petanca
josefa	petanca
pere	petanca

Taula clubs_amb_mes_de_5_socis

sense cap fila

```
CREATE OR REPLACE FUNCTION assignar_individual(nomSoci
char(10),nomClub char(10))
```

```
RETURNS VOID AS $$
```

```
DECLARE
```

```
    missatge varchar(50);
```

```
    num_omens integer;
```

```
    num_mulheres integer;
```

```
BEGIN
```



```
insert into socisclubs values(nomSoci, nomClub);
```

```
num_omens = (select count(*)
```

```
from socis s natural inner join socisclubs sc
```

```
where s.sexe = 'M' and sc.nclub = nomClub);
```

```
num_mulheres = (select count(*)
```

```
from socis s natural inner join socisclubs sc
```

```
where s.sexe = 'F' and sc.nclub = nomClub);
```

```
if ((select count(*)
```

```
from socisclubs s
```

```
where s.nclub = nomClub) > 5 and not exists (select *
```

```
from
```

```
clubs_amb_mes_de_5_socis
```

```
where nclub
```

```
= nomClub))
```

```
then
```

```
insert into clubs_amb_mes_de_5_socis values(nomClub);
```

```
END IF;
```

```
if ((select count(*)
```

```
from socisclubs s
```

```
        where s.nclub = nomClub) > 10)

    then

        SELECT texte INTO missatge FROM missatgesExcepcions
WHERE num=1;

        RAISE EXCEPTION '%',missatge;

    END IF;
```

```
    if (num_omens > num_mulheres)

        then

            SELECT texte INTO missatge FROM missatgesExcepcions
WHERE num=2;

            RAISE EXCEPTION '%',missatge;

        END IF;
```

```
EXCEPTION
```

```
when raise_exception then

    RAISE EXCEPTION '%',missatge;
```

```
    when unique_violation THEN

        SELECT texte INTO missatge FROM missatgesExcepcions
WHERE num=3;

        RAISE EXCEPTION '%',missatge;
```

```
    when foreign_key_violation THEN
```

```
SELECT texte INTO missatge FROM missatgesExcepcions
WHERE num=4;

RAISE EXCEPTION '%',missatge;

when not_null_violation THEN

SELECT texte INTO missatge FROM missatgesExcepcions
WHERE num=4;

RAISE EXCEPTION '%',missatge;

when others THEN

SELECT texte INTO missatge FROM missatgesExcepcions
WHERE num=5;

RAISE EXCEPTION '%',missatge;

END;

$$ LANGUAGE plpgsql;
```

Pràctica JDBC 1

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 15 May 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (100%)

AQUEST EXERCICI NO ES CORREGIRÀ DE MANERA AUTOMÀTICA

PASSOS A SEGUIR

- **Apartat 1**

- Obriu Guia de Programació en JDBC. Repasseu la informació que se us ofereix en aquesta guia.
- Executeu el programa Eclipse
- Importeu el projecte Eclipse que podeu trobar al zip adjunt
- Descarregueu el driver JDBC de la pàgina del curs a una carpeta de l'ordinador.
- Copieu el driver de JDBC a la carpeta "libraries" del projecte Eclipse (es pot fer simplement arrossegant des d'una carpeta on el tingueu).
- Repasseu el contingut de les carpetes i fitxers que es poden trobar dins del projecte Eclipse.
- Prepareu la base de dades des de DBeaver Executeu les sentències SQL de creació de taules (fitxer crea.txt) i càrrega de files a les taules (fitxer carrega.txt).
- Editeu codi del programa *gestioProfes.java* (carpeta "src")
 - Poseu el nom de la vostra base de dades (*LaVostraBD*)
 - Poseu l'esquema on estan les taules (*ElVostreEsquema*).
 - Poseu el vostre username de connexió a la base de dades (*ElVostreUsername*, *ElVostrePassword*).
 - Comproveu que el programa no té errors

- **Apartat 2**

- **Execució 1**

- Abans d'executar el programa, des del DBeaver feu ***select * from Professors***
- Mireu el codi del programa *gestioProfes* per veure què fa sobre la taula *Professors*.
- Compileu el programa.
- Executeu el programa.
- Des del DBeaver torneu a fer ***select * from Professors***

- Quin ha estat l'efecte de l'execució del programa sobre la taula Professors?
Perquè?
- **Execució 2**
 - Editeu el programa. Substituïu la sentència "rollback" per una sentència "commit".
 - Compileu el programa.
 - Executeu el programa.
 - Des del DBeaver feu torneu a fer ***select * from Professors***
 - Quin ha estat l'efecte de l'execució del programa sobre la taula Professors?
Perquè?
- **Execució 3**
 - Executeu una altra vegada el programa.
 - Quina excepció es produeix?
 - Quin ha estat l'efecte de l'execució del programa sobre la taula Professors?
Perquè?
 - Com podrieu fer (sense afegir accessos a la base de dades des del programa) que quan es dongui ***aquesta excepció*** en lloc del missatge obtingut surti "El professor ja existeix"?
 - Editeu el programa i afegiu la implementació d'aquesta excepció.
- **Execució 4**
 - Esborreu la fila que el programa insereix, des del DBeaver.
 - Editeu el programa per tal d'implementar el bloc IMPLEMENTAR
 - En aquest bloc cal implemenar en jdbc:
 - Una consulta per obtenir el dni i el nom dels professors que tenen el telèfon amb un número inferior al número de la variable buscaTelf
 - En cas que no hi hagi cap professor que tingui un telèfon amb número inferior al indicat a la variable, treure un missatge "NO TROBAT"
 - Cal mostrar amb System.out.println el resultat de la consulta.

- Executeu una altra vegada el programa
- Indiqueu quin és el resultat del select.
- Doneu el codi de la part del programa des del bloc IMPLEMENTAR fins al final.

- **Escriviu la resposta a les preguntes anteriors en el formulari del qüestionari i premeu "Envia".**

Ex1:

-No ha tingut cap efecte. Perque ha fet un rollback.

Ex2:

-L'efecte ha estat afegir un professor a la base de dades. Degut al commit.

Ex3:

-Error unique_violation. Perque es repeteix la clau primària.

```
if (se.getSQLState().equals("23505"))
```

```
system.out.println("El professor ja existeix");
```

Ex4:

-111 ruth

222 ona

333 anna

```
String buscaTelf="3334";
```

```
ResultSet r = s.executeQuery("select dni, nomprof " + "from professors " + "where  
telefon < " + buscaTelf"";");
```

```
boolean found = false;
```

```
while (r.next())
```

```
{
```

```
System.out.println(r.getString("dni") + " " + r.getString("nomprof"));
```

```
found = true;
```

```
}
```

```
if (!found) System.out.println("NO TROBAT");
```

Pràctica JDBC 2

[Amagar instruccions](#)

Aquest qüestionari remot es va tancar el Sunday, 22 May 2022, 23:55

El mode lectura està habilitat. Quan el tanqui el qüestionari es podran visualitzar les respostes

No existeix temps extra per entregar el qüestionari remot

La nota definitiva de cada qüestió és la nota de l'última resposta

Els reintents no penalitzen

- Pregunta 1 (100%)

AQUEST EXERCICI NO ES CORREGIRÀ DE MANERA AUTOMÀTICA

PASSOS A SEGUIR

- Importeu el projecte Eclipse que podeu trobar al zip adjunt
- Prepareu el projecte per a ser executat (driver, dades de connexió dins del programa,...)
- Prepareu la base de dades des de DBeaver (fitxers crea.txt, carrega.txt).

- **Apartat 1**

- Editeu el programa gestioProfes per tal d'implementar el bloc IMPLEMENTAR CONSULTA
- En aquest bloc cal implemenar en jdbc:
 - Una consulta per obtenir el dni i el nom dels professors que tenen els telèfons que hi ha al array telfsProf.
 - En cas que hi hagi un telèfon que no sigui de cap professor caldrà que surti el número de telèfon i el text "NO TROBAT"
 - Cal mostrar amb System.out.println el resultat de la consulta.
- Executeu el programa
- Indiqueu quin és el resultat del select.

- **Apartat 2**

- Editeu el programa gestioProfes per tal d'implementar el bloc IMPLEMENTAR CANVI BD
- En aquest bloc cal implemenar en jdbc:
 - Per cada despatx del mòdul 'omega' que no té cap assignació amb instant fi null, incrementar la superfície del despatx en 3 metres quadrats.
 - Cal mostrar amb System.out.println la quantitat de files modificades.
 - En cas que la superfície d'algun dels despatxos passi a ser més gran o igual a 25, no s'ha de modificar cap despatx, i cal mostrar un missatge "Algun despatx passaria a tenir superfície superior o igual a 25".
- Indiqueu quina/es sentències SQL us ha/n fet falta per implementar el canvi, quin és el resultat de l'execució

del programa, i com ho heu fet per identificar si es produeix l'excepció.

- **Escriviu la resposta a les preguntes anteriors, i el codi del programa que heu editat per a la implementació dels apartats 1 i 2, en el formulari del qüestionari i premeu "Envia".**

1- Apartat 1:

111	ruth
222	ona
333	anna
4444	NO TROBAT

Commit i desconnexio realitzats correctament.

- Codi:

```
String[] telfsProf = {"3111", "3222", "3333", "4444"};
```

```
PreparedStatement ps = c.prepareStatement("select dni, nomprof from  
professors where telefon = ?;");
```

```
for (int i = 0; i < 4; i++) {
```

```
    ps.setString(1, telfsProf[i]);
```

```
    ResultSet r = ps.executeQuery();
```

```
    if (r.next()) {
```

```
        System.out.println(r.getString(1) + ' ' + r.getString(2));
```

```
        while (r.next()) {
```

```
            System.out.println(r.getString(1) + ' ' + r.getString(2));
```

```

        }
    }
    else {
        System.out.println(telfsProf[i] + " NO TROBAT");
    }
}

```

2- Apartat 2:

Primer s'ha hagut de fer un update per tal de modificar la taula despatxos i comprovar que no hi hagi una assignació al despatx amb instantFi igual a null. Si cap dels despatxos que compleixen les condicions del where passa a tenir superfície de 25 o més, els despatxos incrementen la seva superfície en 3 metres, sinó es viola una condició del check i es fa rollback imprimint un missatge d'excepció.

- Codi:

```

SQLException se;

Statement st = c.createStatement();

int count = st.executeUpdate("Update despatxos d set superficie = superficie
+ 3 where modul = 'omega' and not exists(Select * from assignacions a where
a.modul = 'omega' and a.numero = d.numero and a.instantfi is null);");

System.out.println(count + " Rows modificades");

if (se.getSQLState().equalsIgnoreCase("23514")) {

```

```
        System.out.println("Algun despatx passaria a tenir superfície superior o  
igual a 25");  
    }
```