

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

05.567 12 01 19 EX

Enganxeu en aquest espai una etiqueta identificativa
amb el vostre codi personal
Examen

Aquest enunciat correspon també a les assignatures següents:

- 11.503 - Bases de dades

Fitxa tècnica de l'examen

- Comprova que el codi i el nom de l'assignatura corresponen a l'assignatura matriculada.
- Només has d'enganxar una etiqueta d'estudiant a l'espai corresponent d'aquest full.
- No es poden adjuntar fulls addicionals, ni realitzar l'examen en llapis o retolador gruixut.
- Temps total: **2 hores** Valor de cada pregunta: **indicat a cada exercici**
- En cas que els estudiants puguin consultar algun material durant l'examen, quins són?
no es pot fer servir material En cas de poder fer servir calculadora, de quin tipus? **CAP**
- Si hi hagi preguntes tipus test: Descompten les respostes errònies? **NO** Quant?
- Indicacions específiques per a la realització d'aquest examen:

Enunciats

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

EXERCICI 1 (3 punts)

Volem disposar d'una base de dades per enregistrar informació sobre pel·lícules, actors i directors. A continuació, es descriu cadascuna de les taules.

PERSON (PERSONA)

La taula *PERSON* conté les dades generals sobre les persones que apareixen a la BD. En concret, per a cada persona s'enregistra un nombre identificador (*id*) que és clau primària, el nom (*name*), la data de naixement (*birth*), la data de defunció (*death*), el gènere (*gender*), la nacionalitat (*nationality*) i l'adreça (*address*). Només els atributs data de defunció (*death*) i adreça (*address*) poden prendre valors nuls.

DIRECTOR (DIRECTOR)

La taula conté informació sobre els directors. En concret, per cada director, es guarda l'identificador (*id*) que és clau primària i clau forana de *PERSON*. L'any de debut (*debut_year*) i el nombre de premis que ha rebut durant la seva carrera (*num_awards*). Només l'atribut any de debut (*debut_year*) pot ser nul.

ACTOR (ACTOR)

Guarda informació sobre els actors. Per cada actor es guarda l'identificador (*id*) que és clau primària i clau forana de *PERSON* i l'any de debut (*debut_year*), que pot ser nul.

MOVIE (PEL·LÍCULA)

Informació sobre les pel·lícules. Per a cada pel·lícula s'enregistra el seu identificador (*id*) que és clau primària, el títol (*title*), el gènere (*genre*), l'any d'estrena (*year*), el país de producció (*country*), la duració en minuts (*duration*), el pressupost en euros (*budget*), el ingressos generats en euros (*incomes*), una referència a la pel·lícula anterior si la pel·lícula pertany a una saga (*id_movie_prev*) i l'identificador del director (*id_director*) que és clau forana a *DIRECTOR*.

Només l'atribut que fa referència a la pel·lícula anterior pot prendre valors nuls. L'atribut *duration* ha de ser positiu estricte i els atributs *incomes* i *budget* han de ser iguals o majors a zero.

L'atribut gènere només pot prendre els valors {'Action', 'Adventure', 'Comedy', 'Crime', 'Drama', 'Historical', 'Horror', 'Musical', 'Science-fiction', 'War', 'Western'}.

PARTICIPATION (PARTICIPACIÓ)

Informació sobre les participacions dels actors a les pel·lícules. S'emmagatzemen el valors {*id_actor*, *id_movie*} que són clau primària, on {*id_actor*} és clau forana de *ACTOR* i {*id_movie*} és

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

clau forana de *MOVIE*. El nom del personatge interpretat (*character*) i el rol (*role*) que només pot prendre els valors {'Leading', 'Supporting', 'Narrator', 'Cameo', 'Unclassified'}.

Cap atribut pot prendre valors nuls. Si s'esborra una pel·lícula de la BD, caldrà esborrar totes les participacions d'actors a la pel·lícula.

(1 punt) ES DEMANA, utilitzant expressions d'àlgebra relacional:

1) Obtenir els actors que han participat en més d'una pel·lícula a l'any 2017. Concretament volem l'identificador, el nom, i l'any de debut d'aquests actors.

$R1 := \text{MOVIE}(\text{year}=2017)$

$R2 := R1[\text{id}]$

$R3 := R2[\text{id}=\text{id_movie}]\text{PARTICIPATION}$

$R4 := R3[\text{id_actor}, \text{id_movie}]$

$R5(\text{id_actor1}, \text{id_movie1}) := R4$

$R6 := R5[\text{id_actor1}=\text{id_actor}, \text{id_movie1} \neq \text{id_movie}]R4$

$R7 := R6[\text{id_actor}=\text{id}]\text{ACTOR}$

$R8 := R7 * \text{PERSON}$

$R9 := R8[\text{id}, \text{name}, \text{debut_year}]$

A $R4$ tenim les participacions d'actors (*id_actor*) a pel·lícules (*id_movie*) fetes a l'any 2017. Per una altra banda, a $R5$ reanomenem atributs per tal de poder fer les operacions necessàries per generar $R6$. $R6$ conté els actors (*id_actor*) que com a mínim han participat a dues pel·lícules (*id_movie*) diferents. A $R7$ recuperem les dades d'aquests actors, i a $R8$ les dades personals dels actors. Per últim, a $R9$ tenim els atributs demanats com a resultat de la consulta.

(1 punt) ES DEMANA, utilitzant expressions d'SQL:

2) Trobar els directors que han guanyat més premis. En concret, volem saber l'identificador, nom i any de naixement d'aquests directors, el nombre de premis que han guanyat i el nombre total de pel·lícules que han dirigit. El resultat es vol ordenat de forma ascendent pel nombre total de pel·lícules que han dirigit.

```
SELECT d.id, p.name, p.birth, d.num_awards, COUNT(*) AS total_movies
FROM DIRECTOR d JOIN PERSON p ON d.id=p.id
      JOIN MOVIE m ON m.id_director=d.id
WHERE d.num_awards= (SELECT MAX(d1.num_awards) FROM DIRECTOR d1)
GROUP BY d.id, p.name, p.birth, d.num_awards
ORDER BY total_movies;
```

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

(1 punt) **ES DEMANA**, donada la seqüència següent d'expressions d'àlgebra relacional:

$R1 := \text{ACTOR}[\text{id}]$

$R2 := \text{DIRECTOR}[\text{id}]$

$R3 := R1 \cap R2$

$R4 := R3 * \text{PERSON}$

$R5 := R4(\text{nationality} = \text{'italian'} \wedge \text{death is null})$

$R6 := R5[\text{id}, \text{name}, \text{birth}]$

3.1) **(0.5 punts)** Explicar en llenguatge natural què fa la seqüència d'operacions prèvies (en definitiva, es tracta de donar l'enunciat de la consulta que es planteja).

Obtenir les persones que són, alhora, actors i directors, però només si són de nacionalitat italiana i estan vius (en aquest cas l'atribut *death* té valor nul). Concretament volem l'identificador, el nom i la data de naixement d'aquestes persones.

3.2) **(0.5 punts)** Proposar una consulta equivalent en llenguatge SQL.

```
SELECT p.id, p.name, p.birth
FROM PERSON p JOIN DIRECTOR d ON p.id=d.id
      JOIN ACTOR a ON a.id=d.id
WHERE p.nationality='italian' AND death IS NULL;
```

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

EXERCICI 2 (2.5 punts)

Sobre la base de dades de pel·lícules del primer exercici, hem afegit la columna ingressos totals del director (*total_incomes*) a la taula *DIRECTOR*. Aquesta columna indica el total d'ingressos aconseguits per les pel·lícules a les que ha estat director.

- Volem mantenir el càlcul del total d'ingressos (*total_incomes*) del director actualitzat. En concret, volem que aquesta columna sempre reflecteixi la suma dels ingressos de les pel·lícules on ha estat director. Podem suposar que els usuaris o programes mai no actualitzaran directament la columna *total_incomes* de la taula *DIRECTOR*, i que en el moment d'inserir un nou director, el valor de la columna *total_incomes* serà zero.

Digueu quins elements d'SQL faríeu servir per implementar en PostgreSQL les regles anteriors.

En cas que trieu disparadors, cal indicar l'operació (*INSERT*, *DELETE*, *UPDATE*) que activaria l'execució de cada disparador i les taules implicades. En cas d'operacions d'*UPDATE* cal especificar les columnes rellevants.

Adicionalment, indiqueu el tipus de disparador (*FOR EACH ROW/FOR EACH STATEMENT* i si ha de ser *BEFORE/AFTER*), així com una breu descripció de les comprovacions que caldria realitzar.

Per a la regla de negoci cal definir disparadors sobre les taules i esdeveniments següents:

	<i>INSERT</i>	<i>UPDATE</i>	<i>DELETE</i>
<i>MOVIE</i>	<p>S'afegirà una nova pel·lícula (el valor de la columna <i>total_incomes</i> s'incrementa amb el valor <i>incomes</i> al director relacionat).</p> <p><i>FOR EACH ROW</i> (cal accedir a valors de la fila a inserir mitjançant <i>NEW</i>).</p> <p><i>AFTER</i> així garantim que no s'ha violat cap regla de negoci.</p>	<p>En cas d'<i>UPDATE</i> hem de tenir en compte la modificació de les columnes següents:</p> <p><i>id_director</i>: Si canviem aquesta columna, hem de tornar a calcular el valor de la columna <i>total_incomes</i> per a l'antic director i per al nou.</p> <p><i>incomes</i>: Si canviem aquesta columna, hem de tornar a calcular el valor de la columna <i>total_incomes</i> restant l'antic ingrès i sumant el nou.</p> <p><i>FOR EACH ROW</i> ja que les sentències d'<i>UPDATE</i> poden actuar sobre un conjunt de files (accedirem als valors de la fila mitjançant <i>NEW</i> i <i>OLD</i>).</p> <p><i>AFTER</i> així garantim que no</p>	<p>En cas d'esborrar una fila de la taula, s'haurà de tornar a calcular el valor de la columna <i>total_incomes</i> del director relacionat.</p> <p><i>FOR EACH ROW</i> (cal accedir a valors de la fila a esborrar mitjançant <i>OLD</i> per restar el valor de <i>incomes</i> de la pel·lícula a <i>total_incomes</i> del director relacionat).</p> <p><i>AFTER</i> així garantim que no s'ha violat cap regla de negoci.</p>

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

	s'ha violat cap regla de negoci.	
--	----------------------------------	--

EXERCICI 3 (2.5 punts)

Sigui un SGBD **sense cap mecanisme de control de concurrència**, i suposem que es produeix l'horari que mostrem a continuació (on R=lectura, RU=lectura amb intenció d'actualització, W=escriptura; les accions s'han numerat per facilitar la seva referència):

Acc#	T1	T2	T3
10			RU(D)
20			W(D)
30	RU(C)		
40		R(A)	
50			RU(C)
60		RU(B)	
70		W(B)	
80			W(C)
90			R(B)
100			RU(D)
110			W(D)
120	W(C)		
130		RU(B)	
140	R(A)		
150		W(B)	
160		COMMIT	
170			COMMIT
180	COMMIT		

1) (1 punt) Indiqueu la interferència o interferències que s'hi produeixen: tipus, grànuls i transaccions implicades, així com el nivell mínim d'aïllament d'SQL amb el que haurien de treballar les transaccions per tal d'evitar simultàniament totes les interferències trobades.

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

Acc#	T1	T2	T3
10			RU(D)
20			W(D)
30	RU(C)		
40		R(A)	
50			RU(C)
60		RU(B)	
70		W(B)	
80			W(C)
90			R(B)
100			RU(D)
110			W(D)
120	W(C)		
130		RU(B)	
140	R(A)		
150		W(B)	
160		COMMIT	
170			COMMIT
180	COMMIT		

Lectura no confirmada entre T2 i T3 pel grànul B.

Actualització perduda entre T1 i T3 pel grànul C.

El nivell d'aïllament mínim d'SQL per evitar la interferència de lectura no confirmada és READ COMMITTED, i per evitar l'actualització perduda és READ UNCOMMITTED.

Per tant, per evitar totes dues interferències simultàniament ens hem de quedar amb el nivell més restrictiu, és a dir, READ COMMITTED.

2) (0.5 punts) L'horari proposat, és recuperable? Quins horaris serials donarien els mateixos resultats que l'horari proposat?

L'horari proposat és recuperable, ja que cap transacció que llegeixi o escriptui un grànul escrit per una altra transacció confirma abans que aquesta.

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

No hi ha cap horari serial equivalent ja que l'horari proposat no és serialable degut a que hi ha dues interferències.

3) **(1 punt)** Apliqueu (indicant com quedaria l'horari) el protocol de reserves en dues fases (PR2F) estricte a l'horari anterior. Quins horaris serials donarien els mateixos resultats que l'horari resultant?

Acc#	T1	T2	T3
10			LOCK(D,X)
20			RU(D)
30			W(D)
40	LOCK(C,X)		
50	RU(C)		
60		LOCK(A,S)	
70		R(A)	
80			LOCK(C,X)
90		LOCK(B,X)	
100		RU(B)	
110		W(B)	
120	W(C)		
130		RU(B)	
140	LOCK(A,S)		
150	R(A)		
160		W(B)	
170		COMMIT UNLOCK(A) UNLOCK(B)	
180	COMMIT UNLOCK(C) UNLOCK(A)		
190			RU(C)
200			W(C)
210			LOCK(B,S)

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

220			R(B)
230			RU(D)
240			W(D)
250			COMMIT UNLOCK(C) UNLOCK(D) UNLOCK(B)

T3 bloqueja la seva execució fins poder aconseguir la reserva sobre el grànul C. L'estona que cada transacció està bloquejada es marca en gris a la solució.

L'horari que obtenim no té cap interferència i és equivalent a l'horari T2;T1;T3.

L'horari també és recuperable. El protocol de reserves en dues fases estricte garanteix horaris serialitzables i recuperables.

EXERCICI 4 (2 punts)

1) (1 punt) Explica què fan les següents línies de codi usat per resoldre el tercer lliurament de la pràctica. **Extensió màxima mitja pàgina.**

```
DBAccessor dbaccessor = new DBAccessor();
dbaccessor.init();
Connection conn = dbaccessor.getConnection();
if (conn != null) {
    Statement cstmt = null;
    ResultSet resultSet = null;
    ...
    cstmt = conn.createStatement();
    resultSet = cstmt.executeQuery("SELECT * from REPORT_YEAR");
}
```

Es realitza la connexió a la base de dades i es declara un `resultSet` que recull el resultat de la consulta que s'executa, obtenint les files de la taula `REPORT_YEAR` (possiblement múltiples files).

2) (1 punt) Què és una base de dades distribuïda? En el context d'aquest tipus de base de dades, què s'entén per fragmentació de les dades? Quins tipus de fragmentació existeixen? **Extensió màxima mitja pàgina.**

Una base de dades distribuïda és un tipus de base de dades que permet emmagatzemar i recuperar dades que es troben repartides en ubicacions diferents i interconnectades mitjançant alguna xarxa de comunicació de manera que l'usuari pot treballar contra aquesta base de dades com si es tractés d'una única base de dades, és a dir, com si fos una base de dades centralitzada com les que hem estudiat a l'assignatura.

Examen 2018/19-1

Assignatura	Codi	Data	Hora inici
Ús de bases de dades	05.567	12/01/2019	15:30

L'esquema de la base de dades (les diferents taules que conformen la base de dades) es poden fragmentar en taules més petites, que es troben físicament emmagatzemades en els diferents nodes o punts d'emmagatzematge que conformen la base de dades distribuïda. Es possible fer 3 tipus de fragmentació: horitzontal (consisteix a fragmentar una taula en grups de files aplicant algun criteri, per exemple, atenent al valor que pren alguna columna), vertical (consisteix a fragmentar una taula en diverses taules que incorporen diferents conjunts de columnes de la taula original) i mixta (resulta de combinar fragmentació vertical i horitzontal).