

Feladatok – 2023.06.05

I. rész

Definiálja következő fogalmakat!

- Relációséma
- Hivatkozási integritási megszorítás
- Második normálforma
- Equijoin
- Tranzakció elkülönítése (ACID)

II. rész

1. Definiálja az Armstrong-axiómákat, és azok közül bizonyítsa be a reflexivitás szabályát!
2. Képezze le az alábbi ER diagramot a Neptun adatbázisról relációs modellé! Indokolja a lépéseit!

III. rész

1. Osztályozza a kapcsolattípusokat (fogalom és definíció)!
2. Ismertesse a háromséma-architectúra szintjeit!
3. Ismertesse a relációalgebra szelekció műveletét és annak tulajdonságait!
4. Ismertesse a relációs adatbázis-tervezés nem hivatalos irányelveit!
5. Sorolja fel egy tranzakció lehetséges állapotait!
6. Mit jelent a tranziencia és perzisztencia az OO adatbázisokban?
7. Sorolja fel a NoSQL adatmodelleket!

Megoldások – 2023.06.05

I. rész

Relációséma

Relációséma alatt az $R(A_1, A_2, \dots, A_n)$ jelölést értjük, ahol R a relációséma neve, A_1, A_2, \dots, A_n attribútumok. Minden A_i attribútum egy szerepkör neve, amelyet valamely D_i tartomány játszik. D_i -t az A_i attribútum tartományának nevezzük, és $\text{dom}(A_i)$ -vel is jelöljük, n a reláció foka.

Hivatkozási integritási megszorítás

Egy R_1 relációséma FK-val jelölt attribútumhalmaza külső (idegen) kulcsa R_1 -nek, amely hivatkozik az R_2 relációsémára, ha eleget tesz a következő feltételeknek:

- Az FK-beli attribútumoknak és az R_2 PK-val jelölt elsődleges kulcsattribútumainak páronként azonos a tartománya; ekkor azt mondjuk, hogy az FK attribútumok hivatkoznak az R_2 relációsémára.
- Bármely r_1 (R_1) aktuális állapotának egy t_1 rekordjában egy FK-beli érték vagy megjelenik egy r_2 (R_2) aktuális állapotának valamely t_2 rekordjában PK értékeként, vagy az értéke NULL. Az előbbi esetben $t_1[\text{FK}] = t_2[\text{PK}]$, ekkor azt mondjuk, hogy a t_1 rekord hivatkozik a t_2 rekordra.

Ha a két feltétel teljesül, egy hivatkozási integritási megszorítás áll fenn R_1 -ről R_2 -re vonatkozóan.

Második normálforma

Egy R relációséma második normálformában (2NF-ben) van, ha R minden másodlagos (leíró) attribútuma teljesen funkcionálisan függ R elsődleges kulcsától.

Equjoin

Azt az általános összekapcsolási műveletet, amelynek összekapcsolási feltételében csak az egyenlőségjel ($=$) szerepel összehasonlító műveleti jelként, egyenlőségen alapuló összekapcsolásnak vagy más szóval equjoin műveletének nevezzük.

Tranzakció elkülönítése (ACID)

Egy tranzakciónak látszólag más tranzakcióktól elkülönítve kell végrehajtnia. Ez azt jelenti, hogy a tranzakció végrehajtása nem állhat kölcsönhatásban semelyik másik konkurensen végrehajtott tranzakcióval sem.

II. rész

1. feladat

Armstrong-axióma definíciója

A reflexivitás, az augmentativitás és a tranzitivitás szabályait együtt Armstrong-Axiómának nevezzük. William Ward Armstrong 1974-ben bizonyította be, hogy a reflexivitás, az augmentativitás és a tranzitivitás szabálya együtt helyes és teljes.

Reflexivitás bizonyítása

Tegyük fel, hogy $X \supseteq Y$, és hogy léteznek t_1 és t_2 rekordok R valamely r relációjában úgy, hogy $t_1[X] = t_2[X]$. Ekkor $t_1[Y] = t_2[Y]$, mivel $X \supseteq Y$; ezért $X \rightarrow Y$ -nak teljesülnie kell r -ben.

2. feladat

III. rész

1. feladat

Kapcsolattípus definíciója

Egy vagy több egyedtípus közötti jól meghatározott viszony.

Kapcsolattípusok osztályozása

A kapcsolat foka: meghatározza, hogy hány egyedtípus vesz részt a kapcsolatban. Ez lehet bináris (másodfokú), ternáris (harmadfokú) vagy magasabb fokú.

A (bináris) kapcsolat számossága: meghatározza, hogy legfeljebb hány kapcsolat-előfordulásban vehet részt egy egyedelőfordulás. Ez lehet 1:1, 1:N, vagy M:N.

A (bináris) kapcsolat szorgossága meghatározza, hogy a kapcsolatban résztvevő egyedítpusok minden egyedének részt kell-e vennie legalább egy kapcsolat-előfordulásban. Ez lehet kötelező, félig kötelező vagy opcionális.

2. feladat

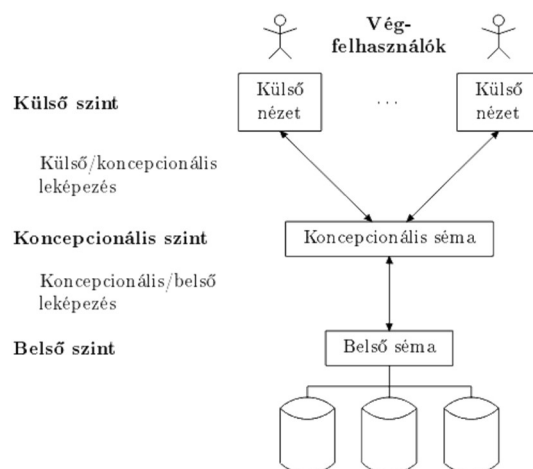
Háromséma-architektúra szintjei

A DBMS sémákat három szinten definiálja.

Külső sémák szinten a különböző felhasználói nézetek leírására. Rendszerint ugyanazt az adatmodellt használja mint a koncepcionális séma.

Koncepcionális séma koncepcionális szinten a teljes adatbázis szerkezetének és megszorításainak leírására a felhasználók közössége számára. Jellemzően koncepcionális vagy implementációs adatmodellt használ.

Belső séma belső szinten a szerkezet és az elérési utak (pl. indexek) fizikai tárolásának leírására. Jellemzően fizikai adatmodellt használ.



3. feladat

Relációalgebra

A matematikai halmazelméleten alapuló lekérdező nyelv. A lekérdezés egy kifejezés, amelyben az operátorok relációalgebrai műveletek, az operandusok pedig relációk. A lekérdezés eredménye szintén egy reláció.

Szelekció művelete

Általános alakja: $\sigma_{\langle \text{szelekciós feltétel} \rangle} (R)$, ahol R azt a relációt jelöli, amelyből a <szelekciós feltétel>-nek eleget tevő rekordokat válogatjuk ki. A <szelekciós feltétel> pedig, egy logikai kifejezés, amely logikai operátorokkal összekapcsolt részkifejezésekből épül föl. A részkifejezések alakja lehet:

- $\langle \text{attribútum} \rangle \langle \text{hasonlító operátor} \rangle \langle \text{konstans} \rangle$
- $\langle \text{attribútum} \rangle \langle \text{hasonlító operátor} \rangle \langle \text{attribútum} \rangle$,

ahol az <attribútum> az R egy attribútumának neve, a <hasonlító operátor> a $\{=, \neq, <, >, \leq, \geq\}$ operátorok egyike, a <konstans> pedig egy konstans érték az attribútum tartományból.

Egy általános szelekciós feltételben a részkifejezéseket az és, a vagy és a nem logikai operátorokkal kapcsolhatjuk össze.

Szelekció megvalósítása SQL-ben: `SELECT * FROM R WHERE szelekciós_feltétel;`

Szelekció tulajdonságai

A szelekció unáris művelet. Az eredményül kapott reláció foka és sémája megegyezik R fokával, illetve sémájával. A számossága pedig mindig kisebb vagy egyenlő R számosságánál, azaz bármely f feltétel esetén $|\sigma_f(R)| \leq |R|$.

Két egymásba ágyaztatott szelekciós művelet végrehajtási sorrendje felcserélhető. Minden többszörösen egymásba ágyaztatott szelekció átírható egyetlen szelekcióvá, amelynek a feltétele az eredeti feltételek konjukciója.

4. feladat

Relációs adatbázis-tervezés

Az attribútumok csoportosítás olyan módon, hogy „jó” relációsémákat alkossanak.

Relációs adatbázis-tervezés nem hivatalos irányelvei

1. A reláció attribútumának szemantikája

Egy reláció minden egyes rekordja egy egyed-előfordulást vagy kapcsolat-előfordulást reprezentáljon. (Az egyes relációkra és azok attribútumaira külön-külön vonatkozik).

- 📌 Különböző egyedek (DOLGOZÓ-k, OSZTÁLY-ok, PROJEKT-ek) attribútumai nem keverendők egyazon relációban.
- 📌 Más egyedekre való hivatkozás csak külső kulcsok használatával történjen.
- 📌 Az egyedekre és a kapcsolatokra vonatkozó attribútumokat a lehető legjobban el kell különíteni egymástól.

Összefoglalva: Olyan sémát kell tervezni, ami könnyen magyarázható relációról relációra. Az attribútumok szemantikájának könnyen értelmezhetőnek kell lennie.

Amikor az információt redundánsan tároljuk, az vagy tárhelyet pazarol, vagy karbantartási anomáliákat (beszúrási -, törlési-, módosítási anomália) okoz.

2. Redundáns információk a rekordokban és karbantartási anomáliák

Olyan sémát tervezzünk, amelyben nem jelennek meg karbantartási anomáliák. Ha mégis előfordulnak anomáliák, akkor jegyezzük fel azokat, hogy az alkalmazások számításba vehessék őket.

- 📌 Módosítási anomália: Ha megváltoztatjuk a projekt nevét, akkor minden rajta dolgozó ügyfélnél át kell írni az új projektnévre.
- 📌 Beszúrási anomália: Nem lehet új projektet beszúrní, ha nincs hozzá dolgozó rendelve, és nem lehet új dolgozót hozzárendelni egy nem létező projekthez.
- 📌 Törlési anomália: Ha törölünk egy projektet, akkor az összes hozzátartozó dolgozót is törölni kell.

3. NULL értékek a rekordban

A relációkat úgy kell megtervezni, hogy a rekordjaik a lehető legkevesebb NULL értéket tartsanak. Azok az attribútumok, amelyek gyakran vesznek fel NULL értéket, külön relációba tehetők (az elsődleges kulccsal). A NULL érték okai lehetnek:

- 📌 Az attribútum nem értelmezhető vagy érvénytelen.
- 📌 Az attribútóérték ismeretlen (de létezik).
- 📌 Az érték biztosan létezik, de nem elérhető.
- 📌 A relációs adatbázisok rossz tervezése bizonyos összekapcsolási műveletek esetén hibás eredményhez vezethet.
- 📌 A „veszteségmentes összekapcsolás” tulajdonsággal garantáljuk, hogy az összekapcsolási műveletek értelmes eredményt adnak.

4. Álrekord

A relációkat úgy kell megtervezni, hogy kielégítsék a veszteségmentes összekapcsolás feltételét. Egy tetszőleges, relációkon végrehajtott természetes összekapcsolás nem állíthat elő álrekordokat.

5. feladat

Tranzakció definíciója

Adatbázis folyamatok egy olyan logikai egysége, amely egy vagy több adatbázis-hozzáférési műveletet (olvasás – kinyerés, írás – beszúrási, frissítés, törlés) tartalmaz.

Tranzakció állapotai

- 📌 Aktív állapot
- 📌 Részlegesen véglegesített állapot
- 📌 véglegesített (commit) állapot
- 📌 hibás állapot
- 📌 megszakított állapot

6. feladat

Tranziencia OO adatbázisokban






Az objektumok az OO programozási nyelvekben tranziensek (átmenetiek), a futás befejeztével törlődnek.

Perzisztencia OO adatbázisokban

Az objektumok az O(R)DMS-ekben perzisztensek (állandóak), eltárolódnak, később kinyerhetőek és megoszthatóak más programokkal.

7. feladat

NoSQL adatmodellek

-  Kulcs-érték modell
-  Rendezett kulcs-érték modell
-  Oszlopcsalád modell
-  Dokumentum modell
-  Gráfmodell