

Ktorý príkaz SQL sa používa na vrátenie iba rôznych hodnôt?

DISTINCT

Uveďte údajový typ, ktorý vám umožní zadávať finančnú hodnotu v Eurách od -99,99 EUR do 99,99 EUR (po zaokrúhlení na dve desatinné miesta), ale napr. 100 EUR už nie. Bez použitia CHECK.

Uveďte údajový typ, ktorý vám umožní zadávať finančnú hodnotu v Eurách od -99,99 EUR do 99,99 EUR (po zaokrúhlení na dve desatinné miesta), ale napr. 100 EUR už nie. Bez použitia CHECK.

Odpoveď zapíšte bez medzier!

Odpoved:

V dopyte `SELECT * FROM customers WHERE email OPERATOR '%@gmail.com'` chýba operátor (nahradený slovom OPERATOR). **LIKE**

Zapíšte údajový typ, ktorý vám umožní zadávať číslo s počtom všetkých číslic 5 a počtom číslic desatinnej časti 3. **numeric(5,3)**

Ktorá funkcia vráti tretie až šieste písmeno z textu (String), vo vzorovom príklade atribútu „title“?

Otázka 2
Ešte nezodpovedané
Max. hodnotenie 3,00
Oznáčť otázku

Ktorá funkcia vráti tretie až šieste písmeno z textu (String), vo vzorovom príklade atribútu „title“?

- a. `substr(title, 2,5)` ❌
- b. `substr(title, 3,3)`
- c. `substring(title, 3,6)` ❌
- d. `substr(title, 3,6)` ❌

[Zrušiť moju voľbu](#)

Ktoré kľúčové slová sa používajú v súvislosti s transakciami v PostgreSQL?:

Ktoré kľúčové slová sa používajú v súvislosti s transakciami v PostgreSQL?:

Označte jednu alebo viac odpovedí:

- a. ROLLBACK
- b. EXCEPT
- c. GRANT
- d. SAVEPOINT
- e. BEGIN
- f. COMMIT

Pomocou ktorého z uvedených SQL vyberiete záznamy z tabuľky „car“ kde „name“ sa nachádza abecedne medzi (a vrátane) „Audi a „Ferrari“?

Pomocou ktorého z uvedených SQL vyberiete záznamy z tabuľky „car“ kde „name“ sa nachádza abecedne medzi (a vrátane) „Audi“ a „Ferrari“?

- a. SELECT * FROM car WHERE name IN ('Audi ', 'Ferrari ');
- b. SELECT * FROM car WHERE name > 'Audi ' AND name < 'Ferrari ';
- c. SELECT * FROM car WHERE name BETWEEN 'Audi ' AND 'Ferrari ';
- d. SELECT * FROM car WHERE name <> 'Audi ' AND name <> 'Ferrari ';

[Zrušiť moju voľbu](#)

Vyberte z nasledujúcich funkcií tie, ktoré sú agregačné:

Otázka **8**
Ešte nezodpovedané
Max. hodnotenie 3,00
[Označiť otázku](#)

Vyberte z nasledujúcich funkcií tie, ktoré sú agregačné:

Označte jednu alebo viac odpovedí:

- a. ROUND
- b. AVG
- c. COUNT
- d. TO_CHAR

Vyberte vlastnosti primárneho kľúča:

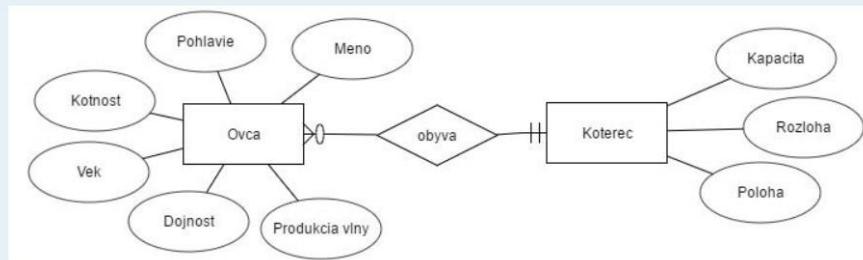
Vyberte vlastnosti primárneho kľúča:

Označte jednu alebo viac odpovedí:

- a. odkazuje sa na záznamy v inej tabuľke
- b. jednoznačne identifikuje záznamy v tabuľke
- c. môže mať hodnotu NULL
- d. má unikátnu hodnotu

Koľko tabuľiek vznikne prepisom nasledovného ER diagramu do relačného modelu?

Koľko tabuľiek vznikne prepisom nasledovného ER diagramu do relačného modelu?

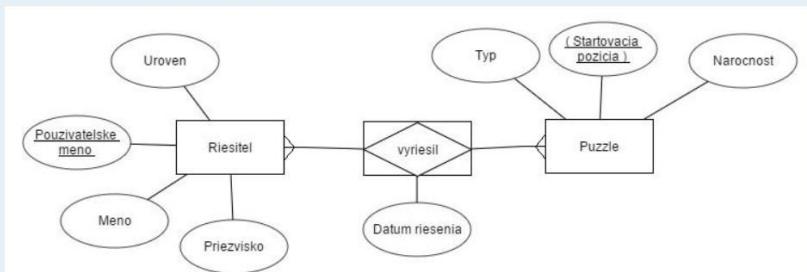


Označte jednu odpoved:

- a. 1
- b. 4
- c. 3
- d. 2

Máme ERD opisujúce databázu šachových hlavolamov ako je ukázané na nasledujúcim obrázku. Koľko stĺpcov bude mať v relačnom modele tabuľka pre asociatívnu entity "vyriesil"?

Máme ERD opisujúce databázu šachových hlavolamov ako je ukázané na nasledujúcim obrázku. Koľko stĺpcov bude mať v relačnom modele tabuľka pre asociatívnu entity "vyriesil"?



Označte jednu alebo viac odpovedí:

- a. 4
- b. 3
- c. 5
- d. 1
- e. 2

Vyberte pravdivé tvrdenia o agregačných funkciách:

Vyberte pravdivé tvrdenia o agregačných funkciách:

Označte jednu alebo viac odpovedí:

- a. Môžu sa použiť iba v selecte, ktorý má klauzulu GROUP BY, bez nej ich nie je možné použiť
- b. Agregujú viaceré hodnoty (napr. hodnoty v stĺpci) do jednej hodnoty, napr. vypočítajú priemernú hodnotu zo zoznamu čísel
- c. Umožňujú vnárať poddopyty aj vo WHERE klauzule
- d. Hodnota NULL je agregačnými funkciemi ignorovaná

Ktoré z nasledujúcich tvrdení o restrikčnom prístupe k zabezpečeniu referenčnej integrity sú pravdivé?

Ktoré z nasledujúcich tvrdení o restrikčnom prístupe k zabezpečeniu referenčnej integrity sú pravdivé?

Označte jednu alebo viac odpovedí:

- a. Ak je aktualizovaná referencia na nejaký záznam, automaticky sa aktualizuje aj primárny klúč, na ktorý sa referencia odkazovala
- b. Ak je zmazaný záznam, na ktorý existuje referencia, referencia je automaticky nastavená na NULL
- c. Ak je aktualizovaný primárny klúč záznamu, na ktorý existuje referencia, databáza vyhodí výnimku
- d. Ak je zmazaný záznam, na ktorý existuje referencia, je automaticky zmazaný aj referujúci záznam
- e. Ak je mazaný záznam, na ktorý existuje referencia, databáza vyhodí výnimku

Vyberte správnu alternatívu k podmienke: cena NOT BETWEEN 2000 AND 300 **MUSÍ BYŤ OR**

Vyberte správnu alternatívu k podmienke: cena NOT BETWEEN 1000 AND 4000

Vyberte správnu alternatívu k podmienke:

cena NOT BETWEEN 1000 AND 4000

- a. cena ≥ 1000 AND cena ≤ 4000
- b. cena = 1000 AND cena = 4000
- c. cena < 1000 OR cena > 4000
- d. cena ≤ 1000 OR cena ≥ 4000

[Zrušiť moju voľbu](#)

Vyberte, čo bude platiť po vykonaní nasledujúceho príkazu za predpokladu, že tabuľka penazenka(okamih,financie) existuje, obsahuje 100 riadkov a jej riadky nie sú referované žiadnymi cudzími klúčmi: TRUNCATE TABLE penazenka WHERE financie ≥ 100 ;

Vyberte, čo bude platiť po vykonaní nasledujúceho príkazu za predpokladu, že tabuľka penazenka(okamih,financie) existuje, obsahuje 100 riadkov a jej riadky nie sú referované žiadnymi cudzími klúčmi:

TRUNCATE TABLE penazenka WHERE financie ≥ 100 ;

- a. obsah tabuľky sa nezmiení
- b. odstráňa sa všetky riadky z tabuľky, pre ktoré platí uvedená podmienka ❌
- c. odstráňa sa všetky riadky z tabuľky ❌
- d. odstráňa sa všetky riadky z tabuľky, pre ktoré neplatí uvedená podmienka ❌

[Zrušiť moju voľbu](#)

Napíšte príkaz, ktorým obmedzíte počet riadkov, ktorý vráti dotaz na 5. Zariadťte taktiež aby bola prvá hodnota vyniechaná, teda vrátia sa riadky 2-6. **LIMIT 5 OFFSET 1**

Aký reťazec dostanete aplikáciou funkcie replace('matematika','ka','te')?

matematite

Ktorým príkazom vieme zrušiť transakciu, aby sa neuložili v databáze zmeny?

Ktorým príkazom vieme zrušiť transakciu, aby sa neuložili v databáze zmeny?

- a. CANCEL
- b. UNDO
- c. ROLLBACK
- d. REVERSE
- e. ROLLUP

Zrušiť moju voľbu

Vyberte tvrdenia, ktoré opisujú význam nasledujúceho dopytu: SELECT priezvisko FROM zamestnanec INTERSECT SELECT priezvisko FROM zákazník;

Vyberte tvrdenia, ktoré opisujú význam nasledujúceho dopytu:

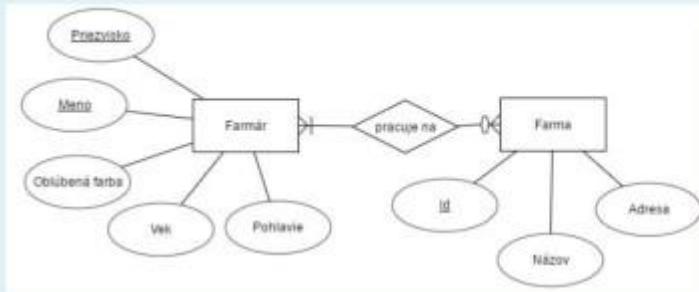
```
SELECT priezvisko FROM zamestnanec  
INTERSECT  
SELECT priezvisko FROM zákazník;
```

- a. Výstupom selectu sú všetky priezviská, ktoré sa nachádzajú v tabuľke zamestnanec okrem tých, ktoré sú v tabuľke zákazník
- b. Ani jedno z tvrdení nie je správne
- c. Výstupom selectu je zjednotená množina priezvisk z tabuľiek zamestnanec a zákazník
- d. Výstupom selectu je prienik priezvisk, nachádzajúcich sa v tabuľkách zákazník a zamestnanec

Zrušiť moju voľbu

Majme ERD pre farmu ako je ukázaný na nasledujúcom obrázku. Vyberte pravdivé tvrdenia vzhl'adom na vzťah "pracuje na".

Majme ERD pre farmu ako je ukázany na nasledujúcom obrázku. Vyberte pravdivé tvrdenia vzhľadom na vzťah "pracuje na".

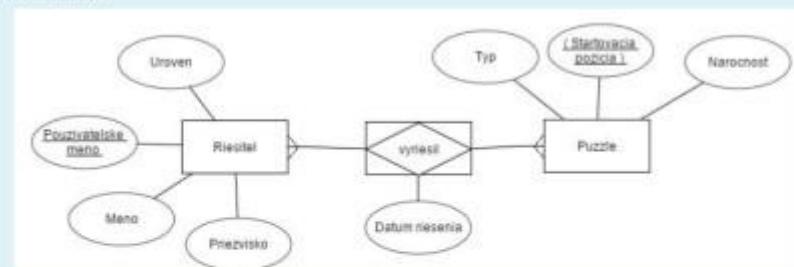


Označte jednu alebo viac odpovedí:

- a. ten istý farmár mohol pracovať na viacerých farmách, ale nemôže naraz pracovať na viacerých farmách v tom istom čase
- b. na farme môže pracovať viaceri farmári, ale každý farmár môže v tom istom časovom okamžiku pracovať len na jednej farme
- c. v relačnom modele bude tento vzťah vyjadrený cudzím kľúčom na strane farmára
- d. jeden farmár môže v tom istom čase pracovať naraz na viacerých farmách
- e. na jednej farme môže súčasne pracovať viaceri farmári, pričom každý z nich môže naraz pracovať na viacerých farmách

Máme ERD opisujúce databázu šachových hlavolamov ako je ukázané na nasledujúcom obrázku. Prečo je atribút "Používateľské meno" entitnej množiny "Riešiteľ" podčiarknutý?

Máme ERD opisujúce databázu šachových hlavolamov ako je ukázané na nasledujúcom obrázku. Prečo je atribút "Používateľské meno" entitnej množiny "Riešiteľ" podčiarknutý?



Označte jednu alebo viac odpovedí:

- a. lebo používateľské meno je povinný atribút - každý riešiteľ ho musí mať definované
- b. lebo hodnota používateľského mena je odvodnená (derivovaná) z mena a priezviska riešiteľa
- c. lebo používateľské meno riešiteľa musí byť unikátné (jedinečné)
- d. lebo používateľské meno je aktualizovateľné - je ho možné meniť

Vyberte pravdivé tvrdenia o referenčnej integrite:

Vyberte pravdivé tvrdenia o referenčnej integrite:

Označte jednu alebo viac odpovedí:

- a. nulitný prístup riešenia referenčnej integrity namiesto zmazania entity nastaví hodnoty všetkých jej atribútov na null, ale jej primárny kľúč nechá, čím zabezpečí, že cudzie kľúče sa budú mať na čo odkazovať
- b. kaskádny prístup k riešeniu referenčnej integrity kaskáduje zmenu (zmienky) na všetky záznamy odkazujúce sa na práve mazaný (menený) záznam
- c. reštrikčný prístup k riešeniu referenčnej integrity zabraňuje zmazať záznam v databáze, ak sa naň odkazuje nejaký cudzí kľúč
- d. restrikčný prístup riešenia referenčnej integrity zabráni zmazať záznamy, ktoré sa odkazujú na iné tabuľky

Vyberte pravdivé tvrdenia o zjednotení ako množinovej operácii v SQL:

Otázka 8
Odpoveď bola uložená
Max. hodnotenie 3,00
[Označiť otázku](#)

Vyberte pravdivé tvrdenia o zjednotení ako množinovej operácii v SQL:

Označte jednu alebo viac odpovedí:

- a. UNION ALL zjednotí všetky záznamy a pritom odstráni duplikáty
- b. v rámci jedného dopytu sa môže UNION použiť iba raz
- c. UNION je výkonovery rýchlejší ako UNION ALL
- d. ORDER BY sa môže pri použití UNION použiť iba v rámci druhého SELECT-u
- e. UNION ALL zjednotí záznamy tak, že neodstráni duplikáty

Vyberte, čo musí platiť, aby bola databáza v 1. normálnej forme:

Vyberte, čo musí platiť, aby bola databáza v 1. normálnej forme:

Označte jednu alebo viac odpovedí:

- a. v žiadnom stĺpci sa nesmú zbytočne opakovať rovnaké hodnoty
- b. databáza nemôže obsahovať zložené stĺpce
- c. všetky primárne kľúče musia byť číselné umelé kľúče
- d. hodnoty všetkých stĺpcov v databáze musia byť atomické
- e. databáza nemôže obsahovať viachodnotové stĺpce

Vyberte pravdivé tvrdenia o relačnom modeli:

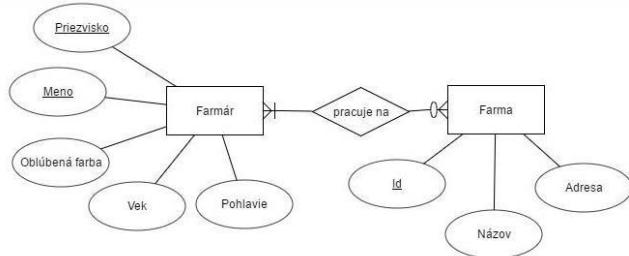
Vyberte pravdivé tvrdenia o relačnom modeli:

Označte jednu alebo viac odpovedí:

- a. Každý stĺpec obsahuje hodnoty toho istého atribútu
- b. Na poradí stĺpcov v tabuľke záleží
- c. Každá tabuľka má jednoznačný názov
- d. Pre daný záznam musí byť zadaná hodnota každého stĺpca
- e. Nie všetky stĺpce musia mať svoj názov
- f. Na poradí záznamov nezáleží
- g. Všetky hodnoty v danom zázname sú jednoznačne a plne závislé na primárnom kľúči záznamu

Majme ERD pre farmu ako je ukázaný na nasledujúcom obrázku. Vyberte pravdivé tvrdenia vzhľadom na vzťah "pracuje na".

Majme ERD pre farmu ako je ukázaný na nasledujúcom obrázku. Vyberte pravdivé tvrdenia vzťahom na vzťah "pracuje na".



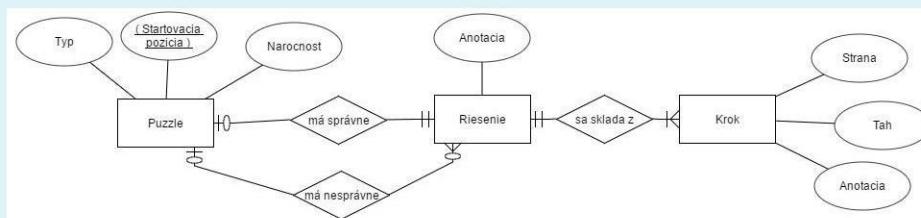
Označte jednu alebo viac odpovedí:

- a. na farme môže pracovať viaceri farmári, ale každý farmár môže v tom istom časovom okamžiku pracovať len na jednej farni
- b. na jednej farni môže súčasne pracovať viaceri farmári, pričom každý z nich môže naraz pracovať na viacerých farmach
- c. ten isty farmár mohol pracovať na viacerich farmach, ale nemôže naraz pracovať na viacerich farmach v tom istom čase
- d. v relačnom modeli bude tento vzťah vyjadrený cudzím klúčom na strane farmára
- e. jeden farmár môže v tom istom čase pracovať naraz na viacerich farmach

Nasledujúci jednoduchý ERD opisuje databázu šachových hlavolamov (puzzle).

Ktorá/é z nasledujúcich možností platí pre prevod vzťahu “má správne“ do relačného modelu?

Nasledujúci jednoduchý ERD opisuje databázu šachových hlavolamov (puzzle). Ktorá/é z nasledujúcich možností platí pre prevod vzťahu “má správne“ do relačného modelu?



Označte jednu alebo viac odpovedí:

- a. vzťah "má správne" môže byť v relačnom modeli vyjadrený pridaním cudzieho klúča do tabuľky "Puzzle", odkazujúceho sa na tabuľku "Riesenie"
- b. vzťah "má správne" bude v relačnom modeli vyjadrený prepájacou medzitabuľkou "maSpravne" s dvoma stĺpcami, ktoré budú cudzími klúčmi prepájacimi tabuľky "Riesenie" a "Puzzle"
- c. vzťah "má správne" môže byť v relačnom modeli vyjadrený pridaním cudzieho klúča do tabuľky "Riesenie", odkazujúceho sa na tabuľku "Puzzle"
- d. vzťah "má správne" bude v relačnom modeli vyjadrený prepájacou medzitabuľkou "maRiesenie" s dvoma stĺpcami, ktoré budú cudzími klúčmi prepájacimi tabuľky "Riesenie" a "Puzzle", pričom táto medzitabuľka bude zároveň vyjadrovať aj vzťah "má nesprávne" z ERD

Vyberte správnu alternatívu k podmienke: cena NOT IN (2000,3000)

Vyberte správnu alternatívku k podmienke:

cena NOT IN (2000,3000)

- a. cena = 2000 AND cena = 3000
- b. cena <> 2000 OR cena <> 3000
- c. cena = 2000 OR cena = 3000
- d. cena <> 2000 AND cena <> 3000

Zrušiť moju volbu

Vyberte, čo musí platiť, aby bola databáza v 1. normálnej forme:

Vyberte, čo musí platiť, aby bola databáza v 1. normálnej forme:

Označte jednu alebo viac odpovedí:

- a. všetky primárne kľúče musia byť číselné umelé kľúče
- b. hodnoty všetkých stĺpcov v databáze musia byť atomické
- c. databáza nemôže obsahovať viachodnotové stĺpce
- d. databáza nemôže obsahovať zložené stĺpce

Uveďte údajový typ, ktorý vám umožní zadávať textové reťazce pevnej (fixnej) dĺžky rozsahu 20 znakov.

Uveďte údajový typ, ktorý vám umožní zadávať textové reťazce pevnej (fixnej) dĺžky rozsahu 20 znakov.

Odpoveď zapísť bez medzier!

Odpoveď:

Ktoré z nasledujúcich jazykov patria do SQL štandardu?

Ktoré z nasledujúcich jazykov patria do SQL štandardu?

Označte jednu alebo viac odpovedí:

- a. Data Declaration Language - jazyk na deklaráciu štruktúry údajov
- b. Data Manipulation Language - jazyk na manipuláciu s údajmi
- c. Data Updating Language - jazyk na aktualizáciu údajov
- d. Data Definition Language - jazyk na definíciu štruktúry databázy
- e. Data Control Language - jazyk na riadenie prístupu k údajom

Vyberte, ktoré tvrdenia opisujú definíciu 3. normálnej formy:

Vyberte, ktoré tvrdenia opisujú definíciu 3. normálnej formy:

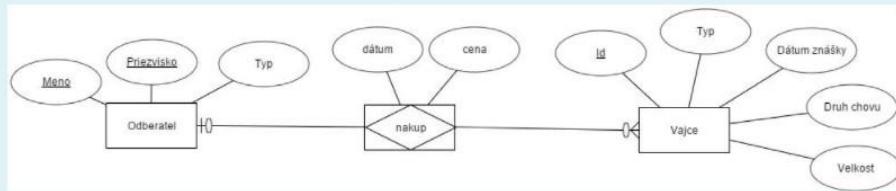
Označte jednu alebo viac odpovedí:

- a. primárny kľúč nesmie byť zároveň cudzím kľúčom
- b. neklúčové atribúty sú navzájom nezávislé
- c. neklúčové atribúty sú na primárnom kľúči závislé priamo a nie tranzitívne
- d. v neklúčových atribútoch sa neopakujú rovnaké hodnoty
- e. primárny kľúč nesmie byť zložený z viacerých atribútov

Na nasledujúcom obrázku máme ER model pre predaj vajec. Ktoré z nasledujúcich tvrdení sú pre daný diagram správne?

Otázka 5
Ešte nezodpovedané
Max. hodnotenie 3,00
 Označiť otázkú

Na nasledujúcim obrázku máme ER model pre predaj vajec. Ktoré z nasledujúcich tvrdení sú pre daný diagram správne?



Označte jednu alebo viac odpovedí:

- a. keďže stĺpec "typ" je aj v tabuľke "Odberatel" aj v tabuľke "Vajce", jeden z nich musí byť cudzí klúč - pravdepodobne ten v tabuľke odberateľ sa odkazuje na typ vajec, ktorý obľubuje
- b. cudzí klúč na tabuľku "Odberatel" bude stĺpec, v ktorom bude kombinácia mena a priezviska obderateľa
- c. cudzí klúč na tabuľku "Vajce" bude stĺpec, ktorý sa bude odkazovať na stĺpec "id" z tabuľky "Vajce"
- d. cudzí klúč na tabuľku "Odberatel" sa bude skladať z dvojice stĺpcov obsahujúcich hodnoty mena a priezviska obderateľa
- e. správna kardinalita vzťahu reprezentovaného asociatívou entitou "nakup" môže byť reprezentovaná aj cudzím klúčom v tabuľke "Vajce", ktorý bude odkazovať na tabuľku "Odberatel", avšak tabuľka "Vajce" potom musí mať aj stĺpce predstavujúce cenu a dátum nákupu vajca

Ktoré z nasledujúcich tvrdení o nulitnom prístupe k zabezpečeniu referenčnej integrity sú pravdivé?

Ktoré z nasledujúcich tvrdení o nulitnom prístupe k zabezpečeniu referenčnej integrity sú pravdivé?

Označte jednu alebo viac odpovedí:

- a. Ak je zmazaný záznam, na ktorý existuje referencia, je automaticky zmazaný aj referujúci záznam
- b. Ak je aktualizovaný záznam, na ktorý existuje referencia, databáza vyhodí výnimku
- c. Ak je zmazaný záznam, na ktorý existuje referencia, referencia sa nastaví na NULL
- d. Ak je aktualizovaný primárny klúč záznamu, na ktorý existuje referencia, referencia sa nastaví na NULL
- e. Ak je aktualizovaná referencia na nejaký záznam, automaticky sa aktualizuje aj primárny klúč, na ktorý sa referencia odkazovala

Vyberte pravdivé tvrdenia o INNER JOIN:

Vyberte pravdivé tvrdenia o INNER JOIN:

Označte jednu alebo viac odpovedí:

- a. Je to karteziánsky súčin, kde každý záznam z jednej tabuľky je spojený s každým záznamom z druhej tabuľky
- b. Výsledkom je kombinácia záznamov z jednej tabuľky so záznamami druhej tabuľky, pre ktoré platí podmienka v ON kľauzu
- c. Vyžaduje sa, aby spájané tabuľky mali rovnakú relačnú schému, t.j. aby mali rovnaký počet stĺpcov a aby každý prislúchajúci pári stĺpcov mal rovnaký (resp. kompatibilný) údajový typ
- d. Použitím kľauzuly USING namiesto ON vieme určiť stĺpce, ktorých hodnoty musia byť rovnaké v oboch tabuľkách rovnaké, aby sa kombinácia záznamov vyskytla vo výsledku

Vyberte pravdivé tvrdenia o LEFT OUTER JOIN:

Vyberte pravdivé tvrdenia o LEFT OUTER JOIN:

Označte jednu alebo viac odpovedí:

- a. Neodporúča sa používať, vhodnou náhradou je INNER JOIN s USING, kde sa explicitne vymenujú stĺpce s rovnakým názvom, ktoré sa majú použiť pri spájaní
- b. Žiadny OUTER JOIN neumožňuje použiť USING kľauzu, pri OUTER JOINoch je povolená iba ON kľauza
- c. Vo výsledku sa objaví každý záznam z ľavej tabuľky aspoň raz
- d. Výsledkom je kombinácia záznamov z jednej tabuľky so záznamami druhej tabuľky, pre ktoré platí podmienka v ON kľauzule, a aj záznamy z ľavej tabuľky, ktoré nemajú pári splňajúci podmienku, doplnené o NULL hodnoty

Vyberte pravdivé tvrdenia o RIGHT OUTER JOIN:

Vyberte pravdivé tvrdenia o RIGHT OUTER JOIN:

Označte jednu alebo viac odpovedí:

- a. Žiadny OUTER JOIN neumožňuje použiť USING klauzulu, pri OUTER JOINoch je povolená iba ON klauzula
- b. Neodporúča sa používať; vhodnou náhradou je INNER JOIN s USING, kde sa explicitne vymenujú stĺpce s rovnakým názvom, ktoré sa majú použiť pri spájaní
- c. Výsledkom je kombinácia záznamov z jednej tabuľky so záznamami druhej tabuľky, pre ktoré platí podmienka v ON klauzule, a vo výsledku sú aj záznamy z pravej tabuľky, ktoré nemajú pár spôsobujúci podmienku, doplnené o NULL hodnoty
- d. Je nahraditeľný LEFT OUTER JOINom, ak prehodíme ľavú tabuľku za pravú a naopak

Vyberte pravdivé tvrdenia o CROSS JOIN:

Vyberte pravdivé tvrdenia o CROSS JOIN:

Označte jednu alebo viac odpovedí:

- a. Je to karteziánsky súčin, kde každý záznam z jednej tabuľky je spojený s každým záznamom z druhej tabuľky
- b. Skrátený zápis pre CROSS JOIN je čiarka ,
- c. Vyžaduje si, aby spájané tabuľky mali rovnakú relačnú schému, t.j. aby mali rovnaký počet stĺpcov a aby každý pricíslúchajúci pári stĺpcov mal rovnaký (resp. kompatibilný) údajový typ
- d. Výsledkom je kombinácia záznamov z jednej tabuľky so záznamami druhej tabuľky, pre ktoré platí podmienka v ON klauzule

SQL dopyt, ktorý vráti všetky oblasti, ich názov, populáciu a id štátu ktorému patria(), ktoré majú viac ako 100.000 (stotisíc) obyvateľov

Otázka 11

Odpoveď bola uložená

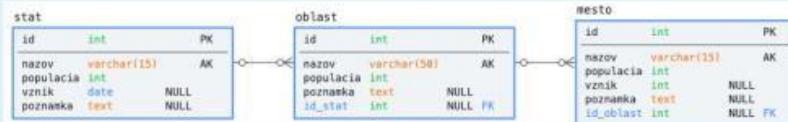
Max. hodnotenie 3,00

Označiť otázku

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti všetky oblasti, ich názov, populáciu a id štátu ktorému patria (len tieto 3 stĺpce v takomto poradí), ktoré majú viac ako 100.000 (stotisíc) obyvateľov a zoradte ich podľa populácie od najmenšej.

Vo vzorovej DB 9 oblastí.



```
select nazov, populacia, id_stat from oblast
where populacia > 100000
order by populacia
```

SQL dopyt, ktorý vráti všetky oblasti a počet ich obyvateľov žijúcich mimo miest v danej oblasti.

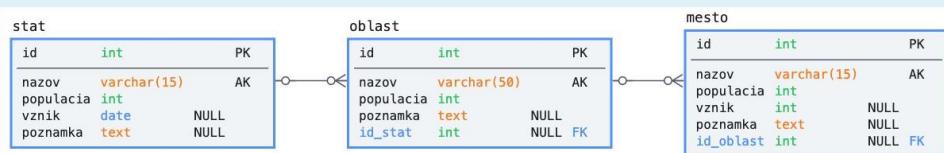
Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti všetky oblasti a počet ich obyvateľov žijúcich mimo miest v danej oblasti (podľa obsahu vzorovej DB).

• Výsledný select bude obsahovať 2 stĺpce – názov oblasti a jeho populáciu žijúcu mimo miest v uvedenom poradí; druhý stĺpec pomenujte na_dedine.

• Výsledok zoraďte zosunute podľa vypočítaného počtu obyvateľov, s NULL hodnotami na konci.

Vo vzorovej DB výsledok dopytu vráti 9 riadkov, iba pre oblasť Mesto Kyjev vyjde hodnota v stĺpci na_dedine na 0.



```
SELECT o.nazov, o.populacia - (SELECT SUM(m.populacia) FROM mesto m WHERE m.id = o.id_oblast) as na_dedine
FROM oblast o
ORDER BY na_dedine ASC
```

SQL dopyt, ktorý vráti názov a čas vzniku pre presne dva podľa populácie najväčšie štáty, ale vyberajte len z tých štátov ktoré nemajú poznámku.

Otázka 12

Odpoveď bola uložená.

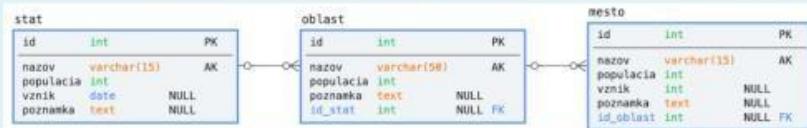
Max.

hodnotenie 3,00

Označiť otázkú

Vytvorte SQL dopyt, ktorý vráti názov a čas vzniku pre presne dva podľa populácie najväčšie štáty, ale vyberajte len z tých štátov ktoré nemajú poznámku.

Tieto dva štáty môžu zostať zoradené od populáciou najväčšieho po populáciou najmenší štát.



```
select nazov, vznik from stat
where poznamka is null
order by populacia desc
limit 2
```

SQL dopyt, ktorý vráti počet miest v jednotných oblastiach. Výsledný select bude obsahovať 2 stĺpce – počet miest a názov príslušnej oblasti

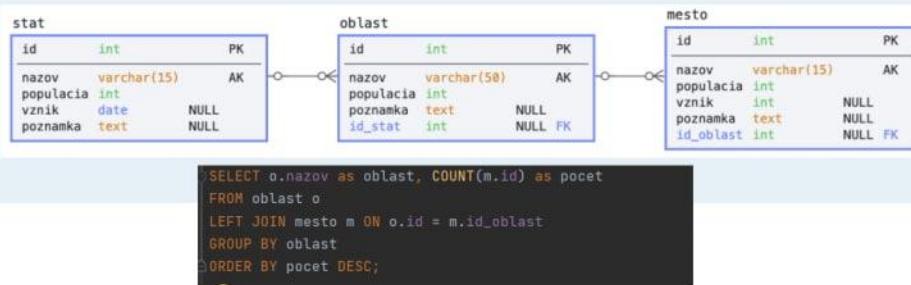
Napište SQL dopyt, ktorý vráti počet miest v jednotných oblastiach.

Výsledný select bude obsahovať 2 stĺpce – názov oblasti a počet miest.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napište SQL dopyt, ktorý vráti počet miest v jednotných oblastiach.
- Výsledný select bude obsahovať 2 stĺpce – názov oblasti a počet miest.
- Pomenujte ich ako oblast a pocet, v uvedenom poradí, bez diakritiky.
- Výsledok zoradte zostupne.

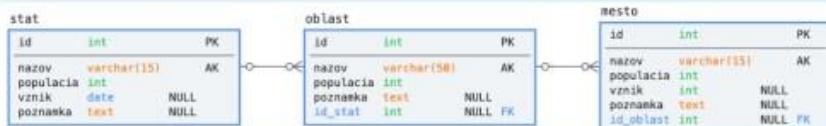
Vo vzorovej DB výsledok dopytu vráti 9 riadkov.



Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napište SQL dopyt, ktorý vráti počet miest v jednotných oblastiach.
- Výsledný select bude obsahovať 2 stĺpce – počet miest a názov príslušnej oblasti.
- Pomenujte ich ako pocet a oblast, v uvedenom poradí, bez diakritiky.
- Výsledok zoradte zostupne.

Vo vzorovej DB výsledok dopytu vráti 5 riadkov.

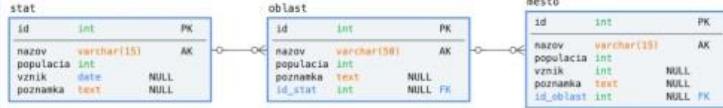


```
select count(m.id) as pocet, oblast.nazov as oblast from oblast
join mesto m on oblast.id = m.id_oblast
group by oblast.nazov
order by pocet desc
```

SQL dopyt, ktorý vráti názov štátu a priemernú populáciu miest štátu, ktorého priemerný počet obyvateľov v mestách je najvyšší

- Napište SQL dopyt, ktorý vráti názov štátu a priemernú populáciu miest štátu, ktorého priemerný počet obyvateľov v mestách je najvyšší.
- Výsledný select bude obsahovať 2 stĺpce – názov štátu a jeho priemernú populáciu miest v uvedenom poradí.
- Pomenujte ich ako stat a priemer, v uvedenom poradí, bez diakritiky.

Vo vzorovej DB výsledok dopytu vráti 1 riadok.



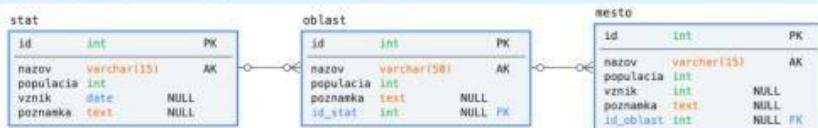
```

select stat.nazov as stat, avg(m.populacia) as priemer from stat
join oblast o on stat.id = o.id_stat
join mesto m on o.id = m.id_oblast
group by stat.nazov
order by priemer desc limit 1;
    
```

SQL dopyt, ktorý vráti všetky mestá a ich počet obyvateľov (len tieto 2 stĺpce), z oblasti v ktorej súčet obyvateľov jej všetkých miest aj najnižší.

- Napište SQL dopyt, ktorý vráti všetky mestá a ich počet obyvateľov (len tieto 2 stĺpce), z oblasti v ktorej súčet obyvateľov jej všetkých miest je najnižší.
- Výsledok zoradte zostupne.
- Výsledný select bude obsahovať 2 stĺpce – názov mesta a jeho populáciu v uvedenom poradí, bez aliasov.

Vo vzorovej DB výsledok dopytu vráti 3 riadky.



```

select nazov, populacia from mesto
where id_oblast = ( select oblast.id from oblast
join mesto m on oblast.id = m.id_oblast
group by oblast.id
order by sum(m.populacia) limit 1);
    
```

SQL dopyt, ktorý vráti počet oblastí v jednotlivých štátach. Výsledný select bude obsahovať 2 stĺpce – počet oblastí a názov príslušného štátu.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti počet oblastí v jednotlivých štátoch.
- Výsledný select bude obsahovať 2 stĺpce – počet oblastí a názov príslušného štátu.
- Pomenujte ich ako pocet a stat, v uvedenom poradí, bez diakritiky.
- Výsledok zoradťte zosunute podľa populácie štátu.

Vo vzorovej DB výsledok dopytu vráti 5 riadkov

```

stat
+-----+
| id   | int    | PK |
| nazov | varchar(15) | AK |
| populacia | int | |
| vznik | date | NULL |
| poznamka | text | NULL |
+-----+



oblast
+-----+
| id   | int    | PK | |
| nazov | varchar(50) | AK |
| populacia | int | |
| poznamka | text | NULL |
| id_stat | int | NULL | FK |
+-----+



mesto
+-----+
| id   | int    | PK | |
| nazov | varchar(15) | AK |
| populacia | int | |
| vznik | int | NULL |
| poznamka | text | NULL |
| id_oblast | int | NULL | FK |
+-----+

```

SQL query:

```

SELECT COUNT(o.id) AS pocet, s.nazov AS stat
FROM stat s
LEFT JOIN oblast o ON s.id = o.id_stat
GROUP BY s.nazov, s.populacia
ORDER BY s.populacia DESC

```

SQL dopyt, ktorý vráti všetky oblasti, ich názov a populáciu (len tieto 2 stĺpce v takomto poradí), ktoré majú menej ako 1 milión obyvateľov a zoradťte ich podľa populácie od najmenšieho.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti všetky oblasti, ich názov a populáciu (len tieto 2 stĺpce v takomto poradí), ktoré majú menej ako 1 milión obyvateľov a zoradťte ich podľa populácie od najmenšieho.

Vo vzorovej DB sú 3 takéto oblasti.

```

stat
+-----+
| id   | int    | PK |
| nazov | varchar(15) | AK |
| populacia | int | |
| vznik | date | NULL |
| poznamka | text | NULL |
+-----+



oblast
+-----+
| id   | int    | PK | |
| nazov | varchar(50) | AK |
| populacia | int | |
| poznamka | text | NULL |
| id_stat | int | NULL | FK |
+-----+



mesto
+-----+
| id   | int    | PK | |
| nazov | varchar(15) | AK |
| populacia | int | |
| vznik | int | NULL |
| poznamka | text | NULL |
| id_oblast | int | NULL | FK |
+-----+

```

SQL query:

```

SELECT nazov, populacia
FROM oblast
WHERE populacia < 1000000
ORDER BY populacia ASC

```

SQL dopyt, ktorý vráti všetky štáty, ich názov a populáciu v miliónoch (len tieto 2 stĺpce v takomto poradí, pri populácii uveďte aj desatinnú časť v celej dĺžke). **float8 miesto float4**

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti všetky štáty, ich názov a populáciu v miliónoch (len tieto 2 stĺpce v takomto poradí, pri populácii uvedťe aj desatinu časť v celej dĺžke). Výsledky zoradte podľa dĺžky názvu štátu začínajúc najkratším názvom, a v prípade rovnakej dĺžky podľa abecedy vzostupne.

Vo vzorovej DB by bol druhý riadok „Polsko 38.4336“ a Slovensko by bolo posledné v poradí.



```
SELECT s.nazov, float(s.populacia)/1000000 as populacia  
FROM stat s  
ORDER BY length(s.nazov), s.nazov
```

Co opisuje interna (fyzicka) schema databazy? B

NAVIGÁCIA V TESTE

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15

Ukončiť pokus ...

Čas zostávajúci do ukončenia testu

0:57:37

Otázka 1

Ešte nezodpovedané
Max. hodnotenie 2,00
 Označiť otázku

Čo opisuje interná (fyzická) schéma databázy?

Označte jednu odpoved:

- a. opisuje vlastnosti fyzických zariadení, na ktorých je databáza implementovaná (napr. potrebnú veľkosť RAM pamäte, a pod.)
- b. fyzické uloženie údajov, formát ich uloženia, a pod.
- c. opisuje konceptuálny model používaný databázou, prostredníctvom ktorého komunikuje klientský program s databázou (napr. relačný model v prípade relačných databáz), ktorý je v rámci programu interný (používateľovi programu nie sú údaje prezentované v rovnakom formáte)
- d. sú to konkrétné dátové súbory fyzicky uložené na disku, ktoré obsahujú samotné údaje

Ďalší

Nasledujuci jednoduchy ERD opisuje databazu sachovych hlavolamov (puzzle). Ako prevedieme do relacneho modelu vztah "sa sklada z" medzi Riesenim a Krokm? C

**NAVIGÁCIA V
TESTE**

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15

[Ukončiť pokus ...](#)

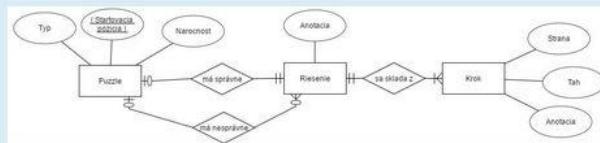
Čas zostávajúci do
ukončenia testu
0:57:32

Otázka 2

Ešte
nezodpovedané
Max. hodnotenie
2,00

[Označiť otázku](#)

Nasledujúci jednoduchý ERD opisuje databázu šachových hlavolamov (puzzle). Ako prevedieme do relačného modelu vztah "sa sklada z" medzi Riesenim a Krokm?



Označte jednu alebo viac odpovedí:

- a. do tabuľky "Riesenie" pridáme cudzí kľúč odkazujúci sa na tabuľku "Krok"
- b. vytvoríme prepájacie tabuľku "skladaSaZ", ktorá bude obsahovať dva cudzie kľúče odkazujúce sa na "Krok" a "Riesenie"
- c. do tabuľky "Krok" pridáme cudzí kľúč odkazujúci sa na tabuľku "Riesenie"
- d. do tabuľky "Krok" pridáme cudzí kľúč odkazujúci sa na tabuľku "Puzzle"
- e. vytvoríme prepájacie tabuľku "skladaSaZ", ktorá bude obsahovať dva cudzie kľúče odkazujúce sa na "Riesenie" a "Krok"

Vyberte, ktore tvrdenia opisuju definiciu 3. Normalnej formy A B C

**NAVIGÁCIA V
TESTE**

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15

[Ukončiť pokus ...](#)

Čas zostávajúci do
ukončenia testu
0:57:29

Otázka 3

Ešte
nezodpovedané
Max. hodnotenie
2,00

[Označiť otázku](#)

Vyberte, ktoré tvrdenia opisujú definíciu 3. normálnej formy:

Označte jednu alebo viac odpovedí:

- a. nekľúčové atribúty sú navzájom nezávislé
- b. primárny kľúč nesmie byť zložený z viacerých atribútov
- c. v nekľúčových atribútoch sa neopakujú rovnaké hodnoty
- d. primárny kľúč nesmie byť zároveň cudzím kľúčom
- e. nekľúčové atribúty sú na primárnom kľúči závislé priamo a nie tranzitívne

[Ďalší](#)

V jazyku SQL je platným zapisom logických podmienok výraz:

OR AND

NAVIGÁCIA V TESTE

1 2 3 4 5
6 7 8 9 10
11 12 13 14 15

Ukončiť pokus ...
Čas zostávajúci do ukončenia testu
0:57:23

Otázka 4
Ešte nezodpovedané Max. hodnotenie 2,00
Označiť otázku

V jazyku SQL je platným zápisom logických podmienok výraz:
Označte jednu alebo viac odpovedí:

- expression OR expression
- expression || expression
- expression AND expression
- expression && expression
- expression ^^ expression

Vyberte pravdivé tvrdenia o riadkových triggeroch **VYKONAJU SA RAZ, FOR EACH ROW**

NAVIGÁCIA V TESTE

1 2 3 4 5
6 7 8 9 10
11 12 13 14 15

Ukončiť pokus ...
Čas zostávajúci do ukončenia testu
0:55:52

Otázka 9
Ešte nezodpovedané Max. hodnotenie 2,00
Označiť otázku

Vyberte pravdivé tvrdenia o riadkových triggeroch:
Označte jednu alebo viac odpovedí:

- a. vykonajú sa raz pre každý riadok ovplyvnený udalosťou
- b. nemôžu reagovať ako náhrada udalosti (teda nemôžu byť v kombinácii s INSTEAD OF)
- c. musia mať definovanú podmienku v klazule WHEN
- d. sú to triggery označené klauzulou FOR EACH ROW

Majme nasledovny dopyt v relacnej algebra:

Ktorý z nasledovnych SQL dopytov vrati rovnaky vysledok pri akychkolvek udajoch v databaze?

D

NAVIGÁCIA V TESTE

1 2 3 4 5
6 7 8 9 10
11 12 13 14 15

[Ukončiť pokus ...](#)

Čas zostávajúci do ukončenia testu
0:55:58

Otázka 10
Ešte nezodpovedané
Max. hodnotenie 2,00

Označiť otázku

Majme nasledovný dopyt v relačnej algebre:

$\rho_{meno}(\pi_{composer}(track))$

Ktorý z nasledujúcich SQL dopytov vráti rovnaký výsledok pri akýchkoľvek údajoch v databáze?

Označte jednu odpoveď:

- a. `SELECT meno
FROM track
WHERE name='composer';`
- b. `SELECT composer AS meno
FROM track;`
- c. `SELECT DISTINCT composer AS meno
FROM track;`
- d. `SELECT DISTINCT composer
FROM track
WHERE composer='meno';`

[Ďalší](#)

Majme dopyt:

Ktorý z nasledujúcich indexov databazovy system nevie vyuzit pre zefektivnenie jeho vykonania

[Majme dopyt:](#)

```
SELECT name  
FROM track  
WHERE (unitPrice > 0.5 AND unitPrice <= 1.0)  
      OR composer = 'U2';
```

Ktorý z nasledujúcich indexov databázový systém **nevie** využiť pre zefektívnenie jeho vykonávania?

Označte jednu odpoveď:

- a. index na báze hešovacej tabuľky nad stĺpcom unitPrice ✓
- b. index na báze vyváženého stromu nad stĺpcom unitPrice
- c. index na báze vyváženého stromu nad stĺpcom composer
- d. index na báze hešovacej tabuľky nad stĺpcom composer

[Clear my choice](#)

[Skontrolovať](#)

Správna odpoved

Hodnotenie pre tento pokus: 2,00/2,00.

Vyberte pravdivé tvrdenia o NATURAL JOINe

Vyberte pravdivé tvrdenia o NATURAL JOINe:

Označte jednu alebo viac odpovedí:

- a. Výsledkom je kombinácia záznamov z jednej tabuľky so záznamami druhej tabuľky, pre ktoré sú hodnoty v rovnako pomenovaných stĺpcoch rovnaké ✓
- b. Je to karteziánsky súčin, kde každý záznam z jednej tabuľky je spojený s každým záznamom z druhej tabuľky
- c. Vyžaduje si, aby spájané tabuľky mali rovnakú relačnú schému, t.j. aby mali rovnaký počet stĺpcov a aby každý pricislúchajúci pári stĺpcov mal rovnaký (resp. kompatibilný) údajový typ
- d. Neodporúča sa používať, vhodnou náhradou je INNER JOIN s USING, kde sa explicitne vymenujú stĺpce s rovnakým názvom, ktoré sa majú použiť pri spájaní ✓

Skontrolovať

Správna odpoveď

Hodnotenie pre tento pokus: 2,00/2,00.

Ktore z nasledujúcich indexov databazový system vie využiť pre zefektívnenie jeho vykonania?

Majme dopyt:

```
SELECT 'Zamestnanec: ' || e.firstname || e.lastname, 'Zákazník: ' || c.firstname || c.lastname
FROM customer c JOIN employee e ON c.supportRepId = e.employeeId
WHERE e.hiredate < '1.1.2010';
```

Ktoré z nasledujúcich indexov databázový systém **vie** využiť pre zefektívnenie jeho vykonávania?

Označte jednu alebo viac odpovedí:

- a. zložený index nad stĺpcami firstname a lastname tabuľky customer
- b. index nad stĺpcom birthdate tabuľky employee
- c. index nad stĺpcom employeeId tabuľky employee
- d. index nad stĺpcom supportRepId tabuľky customer

Skontrolovať

Vyberte pravdivé tvrdenia o entitnej integrite udajov v rámci relacneho modelu

Vyber pravdivé tvrdenia o entitnej integrite údajov v rámci relačného modelu:

Označte jednu alebo viac odpovedí:

- a. primárny kľúč tabuľky musí byť vždy uvedený (nesmie mať hodnotu NULL) ✓
- b. akékoľvek neuvedenie povinného atribútu entity porušuje entitnú integritu databázy
- c. použitie viachodnotového atribútu (napr. titul s možnosťou uvedenia viacerých titulov v rámci jednej entity) porušuje entitnú integritu
- d. entitná integrita zabezpečuje jednoznačnú identifikateľnosť každej entity v databáze ✓
- e. rôzne entity v tabuľke sú vždy rozlíšiteľné prinajmenšom primárnym kľúčom ✓

Skontrolovať

Správna odpoveď
Hodnotenie pre tento pokus: 2,00/2,00.

Napiste operator urceny pre zlucenie retazcov ||

Čo opisuje konceptuálna schéma databázy?

Čo opisuje konceptuálna schéma databázy?

Označte jednu odpoved:

- a. je to sada dopytov, ktoré vyjadrujú, čo daný program chce z databázy získať, prostredníctvom nej vieme povedať, aké práva pre prístup do databázy daný program potrebuje
- b. Je to konkrétny diagram, ktorý je pri návrhu databázy použitý (napr. entitno-relačný diagram)
- c. opisuje logickú štruktúru údajov a ich vzájomných väzieb a používa sa na komunikáciu medzi databázou a klientskym programom (napr. relačná schéma v relačných databázach) ✓
- d. opisuje, ako sú údaje uložené na disku v údajových súboroch, ako sú vyjadrené ich vzájomné väzby a aké údajové typy sú použité

[Clear my choice](#)

Skontrolovať

Napíšte kľúčové slovo, pomocou ktorého je možné vrátiť DB do stavu pred začiatkom transakcie.

ROLLBACK

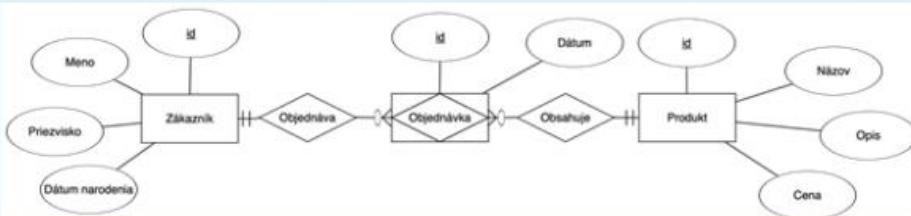
Napíšte kľúčové slovo, pomocou ktorého je možné vrátiť stav DB do stavu pred začiatkom transakcie.

Odpoveď zapíšte bez medzier!

Odpoved:

Majme ERD pre obchod ako je naznačené na nasledujúcim obrázku.

Majme ERD pre obchod ako je naznačené na nasledujúcim obrázku. Vyberte pravdivé tvrdenia vzhladom na diagram.



Označte jednu alebo viac odpovedí:

- a. Zákazník si mohol v rámci jednej objednávky objednať viacero produktov
- b. Jeden produkt si mohlo objednať viacero zákazníkov
- c. Jeden produkt sa môže v tom istom čase nachádzať v rôznych objednávkach
- d. V relačnom modeli bude vzťah „Objednáva“ vyjadrený cudzím kľúčom na strane Zákazníka
- e. Dve alebo viac objednávok môžu mať presne rovnaký dátum

O dopyte uvedenom vyššie vieme že existujú v databáze zmrzliny ktoré ešte nepatria k žiadnemu stánku, ku každému stánku patrí ale minimálne jedna zmrzlina.

```
SELECT * FROM zmrzinovy_stanok s TAJOMSTVO zmrzlina z ON s.StanokID = z.ZmrzlinalID;
```

O dopyte uvedenom vyššie vieme že existujú v databáze zmrzliny ktoré ešte nepatria k žiadnemu stánku, ku každému stánku patrí ale minimálne jedna zmrzlina.

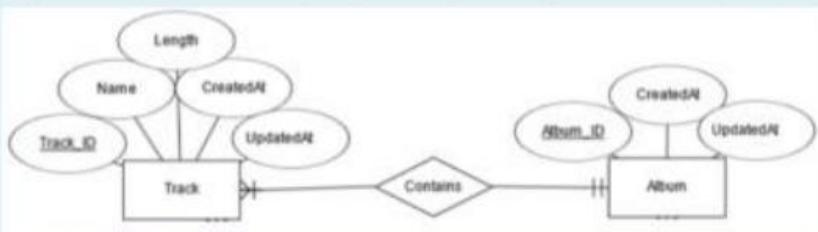
Naším cieľom je použiť JOIN, ktorý nám vráti záznamy v ktorých budú údaje o zmrzline v prvom stĺpici a údaje o stánku, ku ktorému zmrzlina patrí, v druhom. Chceme však aby nám dopytol aj zmrzliny ktoré nepatria ku žiadnemu stánku.

Akým druhom JOIN-u by mali byť tabuľky spojené (na mieste kde je teraz v dopyte slovo TAJOMSTVO) aby sme dosiahli uvedený cieľ? Vyberte jednu alebo viac správnych odpovedí.

- a. NATURAL JOIN
- b. INNER JOIN
- c. RIGHT OUTER JOIN
- d. LEFT OUTER JOIN
- e. EQUAL JOIN

Majme ERD pre albumové skladby ako je ukázaný na nasledujúcom obrázku.

Majme ERD pre albumové skladby ako je ukázaný na nasledujúcom obrázku. Vyberte pravdivé tvrdenia vzhľadom na vztah "contains".

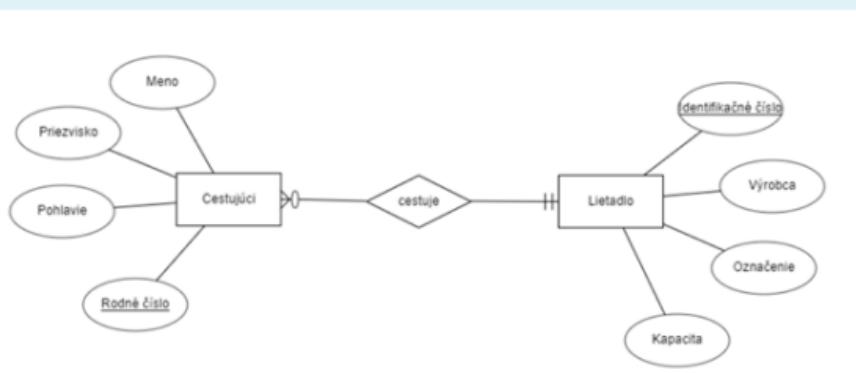


Označte jednu alebo viac odpovedí:

- a. tá istá skladba sa mohla vyskytovať na viacerých albumoch, ale nemôže sa vyskytovať na viacerých albumoch v tom istom čase
- b. v relačnom modeli bude tento vztah vyjadrený cudzím klúčom na strane skladby
- c. v relačnom modeli bude tento vztah vyjadrený cudzím klúčom na strane albumu
- d. na albe me môže byť súčasne viacero skladieb, pričom každá z týchto skladieb sa môže nachádzať práve na jednom albe

Majme ERD pre farmu ako je ukazany na nasledujucom obrazku. Vyberte pravdivie tvrdenia vzhľadom na vztah "pracuje na".

Majme ERD pre farmu ako je ukázaný na nasledujúcom obrázku. Vyberte pravdivé tvrdenia vzhľadom na vztah "pracuje na".

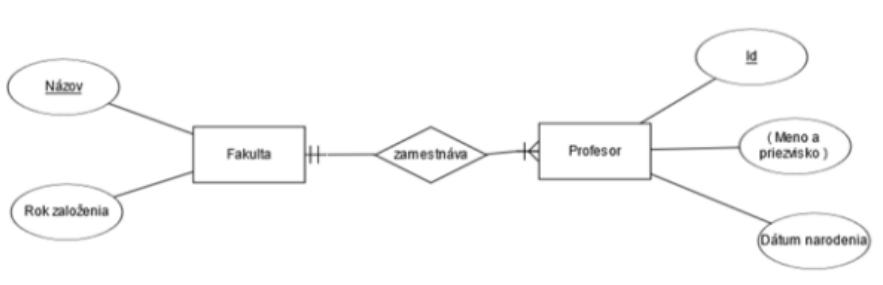


Označte jednu alebo viac odpovedí:

- a. jeden cestujúci môže letieť práve jediným lietadlom
- b. v jednom lietadle musí cestovať minimálne jeden cestujúci
- c. v relačnom modeli bude tento vztah vyjadrený cudzím klúčom na strane lietadla
- d. v relačnom modeli bude tento vztah vyjadrený cudzím klúčom na strane cestujúceho
- e. jeden cestujúci nemusí letieť žiadnym lietadlom

Majme ERD pre fakultu ako je ukázaný na nasledujúcom obrázku. Vyberte pravdivé tvrdenia vzhľadom na vzťah "zamestnáva".

Majme ERD pre fakultu ako je ukázaný na nasledujúcom obrázku. Vyberte pravdivé tvrdenia vzhľadom na vzťah "zamestnáva".

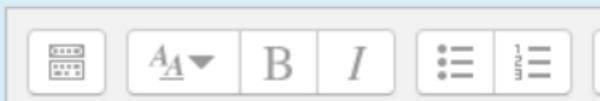


Označte jednu alebo viac odpovedí:

- a. ERD reprezentuje databázu ktorá je v prvej normálnej forme.
- b. Na fakulte musí byť vždy aspoň jeden profesor.
- c. Profesor nemôže pracovať na viacerých fakultách súčasne.
- d. Všetky atribúty sú atomické.
- e. V relačnom modeli bude tento vzťah vyjadrený cudzím kľúčom na strane profesora.

Na čo slúžia Indexy v SQL?

Na čo slúžia Indexy v SQL?



Na urýchlenie vyhľadávania v tabuľkách.

Popíšte 2 časti databázového systému.

Popíšte 2 časti databázového systému.



Databáza (Báza dát) - komplex dát, medzi ktorými existujú vztahy

Systém riadenia bázy dát - systém umožňujúci používateľovi pracovať s dátami

Napiste dopyt ktorý vrati dvojice oblast (prvy stlpec) – mesto patriace danej oblasti (druhy stlpec), ale len pre mesta ktore vznikli medzi rokmi 1200 a 1300 (vratane) a tiež pre oblasti bez mesta (vo vysledku pre ne bude v pravom stĺpci namiesto nazvu mesta NULL).

Vysledky zoradte abecedne podla nazvu oblasti.

Vo vzoroveq DB vysledok dopytu vrati 8 riadkov

Otázka 14

Este
nezodpovedané
Max.
hodnotenie 5,00
Poznámka
Označiť
otázkú

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište dopyt ktorý vráti dvojice oblast' (prvý stlpec) – mesto patriace danej oblasti (druhý stlpec), ale len pre mestá ktoré vznikli medzi rokmi 1200 a 1300 (vrátane) a tiež pre oblasti bez mesta (vo výsledku pre ne bude v pravom stĺpci namiesto názvu mesta NULL).

Výsledky zoradte abecedne podla názvu oblasti.

Vo vzorovej DB výsledok dopytu vráti 8 riadkov.



```
SELECT o.nazov,m.nazov FROM oblast LEFT JOIN mesto m on o.id = m.id_oblast WHERE m.vznik BETWEEN 1200 AND 1300 OR m.id IS NULL ORDER BY o.nazov;
```

Napiste SQL dopyt, ktorý vrati pre kazdy stat pocet miest v jednotlivych statoch

Vysledny select bude obsahovať 2 stpce – nazov statu a miest prislusneho statu

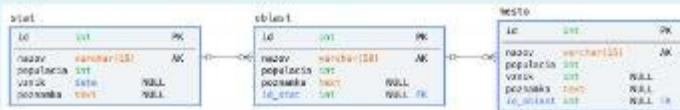
Pomenujte ich ako stat a pocet, v uvedenom poradi, bez diakrity

Vysledok zoskupte podla statu

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti pre každý štát počet miest v jednotlivých štátoch.
- Výsledný select bude obsahovať 2 stĺpce – názov štátu a miest príslušného štátu.
- Pomenujte ich ako stat a pocet, v uvedenom poradí, bez diakritiky.
- Výsledok zoskupte podľa štátu.

Vo vzorovej DB výsledok dopytu vráti 2 riadky.



```
select stat.nazov as stat, count(m.id) as pocet from
stat left outer join oblast o on stat.id = o.id_stat inner
join mesto m on o.id = m.id_ostat group by
stat.nazov;
```

Vo vzorovej DB vysledok dopytu vrati 10 riadkov, pricom rozdiel medzi suctom populacie v oblastiach pre jednotlive staty splnajuce podmienky by mal byť 4347150

--Vo vzorovej DB výsledok dopytu vráti **10 riadkov**, pričom rozdiel medzi súčtom populácie v oblastiach pre jednotlivé štaty spĺňajúce podmienky by mal byť **4 347 150**.

```
SELECT s.nazov, SUM(o.populacia)
FROM stats
JOIN oblast o on s.id = o.id_stat
WHERE s.poznamka IS NOT NULL
GROUP BY s.nazov
UNION
SELECT m.nazov, MAX(m2.populacia)
FROM mesto m
JOIN mesto m2 ON m.id_oblast = m2.id_oblast
WHERE m.nazov NOT LIKE 'K%'
GROUP BY m.nazov
ORDER BY nazov;
```

Napiste sql dopyt, ktorý vrati pocet miest v jednotnych oblastiach.

Vysledny select bude obsahovať 2 stlpce – pocet miest a nazov prislusnej oblasti.

Pomenujte ich ako pocet a oblast, v uvedenom poradi, bez diakritiky.

Vysledok zoradte zostupne

Vo vzoroveq DB bysledok dopytu vrati 5 riadkov

--15.

--Napište SQL dopyt, ktorý vráti počet miest v jednotných oblastiach.

--Výsledný select bude obsahovať 2 stĺpce - počet miest a názov príslušnej oblasti.

--Pomenujte ich ako pocet a oblast, v uvedenom poradí, bez diakritiky.

--Výsledok zoradte zostupne

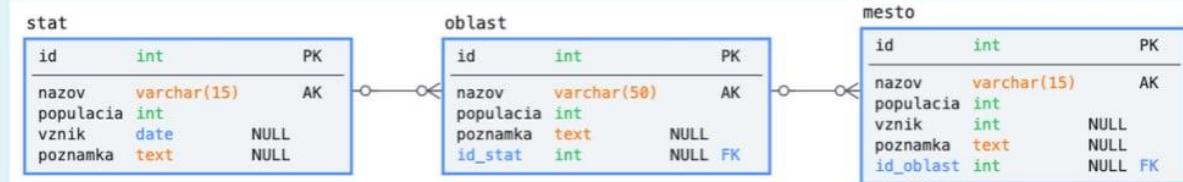
--Vo vzorovej DB výsledok dopytu vráti 5 riadkov.

```
SELECT COUNT(m.id) AS pocet, o.nazov AS oblast
FROM oblast o
JOIN mesto m ON o.id = m.id_oblast
GROUP BY oblast
ORDER BY pocet DESC;
```

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napište SQL dopyt, ktorý vráti počet oblastí v jednotlivých štátach.
- Výsledný select bude obsahovať 2 stĺpce – počet oblastí a názov príslušného štátu.
- Pomenujte ich ako pocet a stat, v uvedenom poradí, bez diakritiky.
- Výsledok zoradte zostupne podľa populácie štátu.

Vo vzorovej DB výsledok dopytu vráti 5 riadkov



SELECT COUNT(o.id) AS pocet, s.nazov AS stat

FROM stat s

LEFT JOIN oblast o ON s.id = o.id_stat

GROUP BY s.nazov, s.populacia

ORDER BY s.populacia DESC

Napiste SQL dopyt, ktorý vrati zoznam oblasti v ktorých sa NENACHADZA mesto s písmenom "v" ani "V" v nazve.

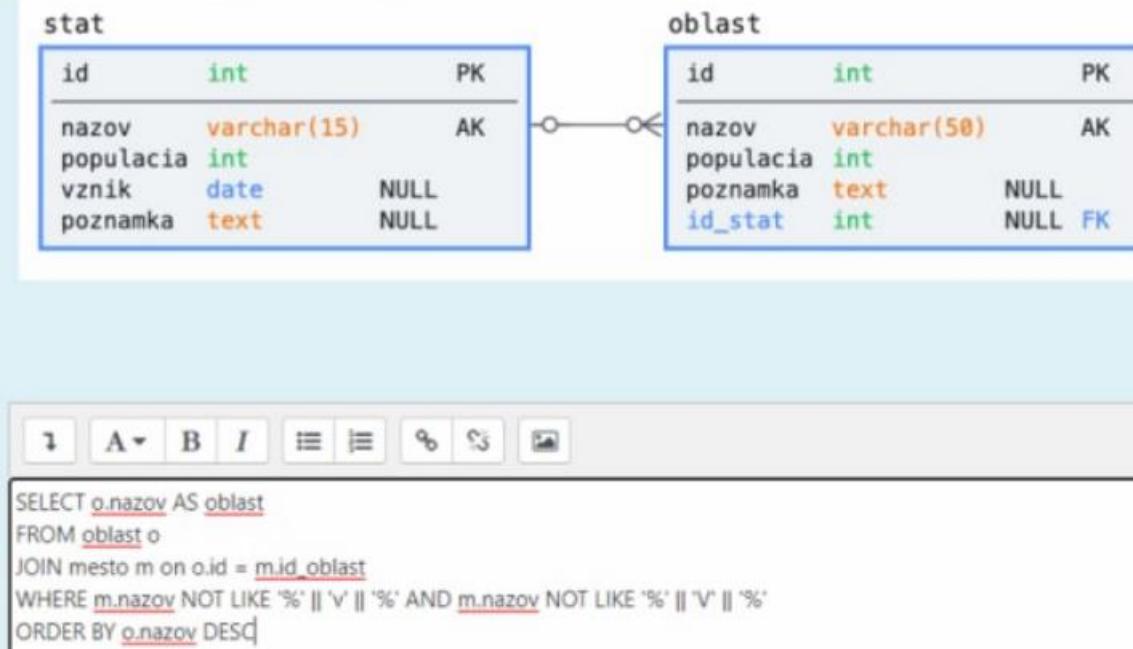
Vysledok zoradte vzostupne podla nazvu oblasti

Vysledny select nech obsahuje 1 stlpec pomenujte ho ako "oblast" (bez uvodzoviek a bez diakritiky).

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti zoznam oblastí v ktorých sa NENACHÁDZA mesto s písmenom "v" ani "V" v názve.
- Výsledok zoradte vzostupne podľa názvu oblasti.
- Výsledný select nech obsahuje 1 stĺpec pomenujte ho ako "oblast" (bez úvodzoviek a bez diakritiky).

Vo vzorovej DB výsledok dopytu vráti 5 riadkov.



Pracujeme s databazou, ktorej relacny model je zobrazeny na nasledujucej scheme.

Napiste SQL dopyt, ktory vrati vsetky mesta a ich pocet obyvateľov (len tieto 2 stlpce), z oblasti v ktorej suet obyvateľov jej vsetkych miest je najnizsi.

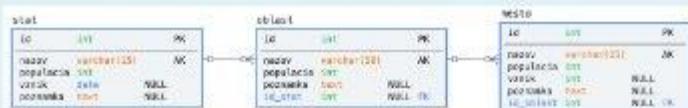
Vysledok zoradte zostupne

Vysledny select bude obsahovať 2 stlpce – nzaov mesta a jeho populaciu v uvedenim poradi, bez aliasov. Vo vzorovej DB vysledok dopytu vrati 3 riadky.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti všetky mestá a ich počet obyvateľov (len tieto 2 stĺpce), z oblasti v ktorej súčet obyvateľov jej všetkých miest je najnižší.
- Výsledok zoradte zostupne.
- Výsledný select bude obsahovať 2 stĺpce – názov mesta a jeho populáciu v uvedenom poradí, bez aliasov.

Vo vzorovej DB výsledok dopytu vráti 3 riadky.



```
select nazov, populacia from mesto where id_ostan  
in(select oblast.id from oblast inner join mesto m on  
oblast.id = m.id_ostan group by oblast.id order by  
count(m.id) desc limit 1);
```

Ktore maju menej ako 1 milion.



A ▾

B

I

1
2
3

```
select nazov, populacia from oblast where populacia < 1000000;
```

Napiste SQL dopyt ktory vrati vsetky oblasti a pocet ich obyvateľov zjucih mimo miest v danej oblasti (podla obsahu vzorovej DB)

Vysledny select bude obsahovať 2 stlpce – nazov oblasti a jeho populaciu zjucu mimo miest v uvedenom poradi, druhý stlpec pomenujte na_dedine.

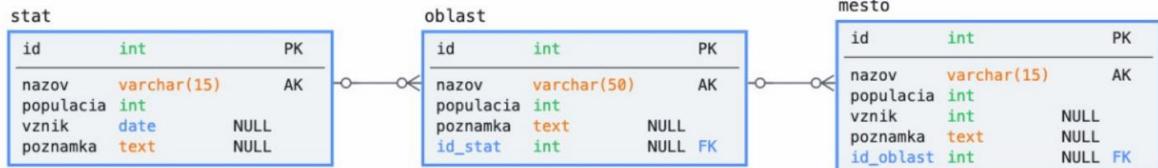
Vo vzorovej DB vysledok dopytu vrati 9 riadok, iba pre oblast Mesto Kyjev vyjde hodnota v stlpci na_dedine na 0

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti všetky oblasti a počet ich obyvateľov žijúcich mimo miest v danej oblasti (podľa obsahu vzorovej DB).

- Výsledný select bude obsahovať 2 stĺpce – názov oblasti a jeho populáciu žijúcu mimo miest v uvedenom poradí, druhý stĺpec pomenujte na_dedine.
- Výsledok zoradte zosunute podľa vypočítaného počtu obyvateľov, s NULL hodnotami na konci.

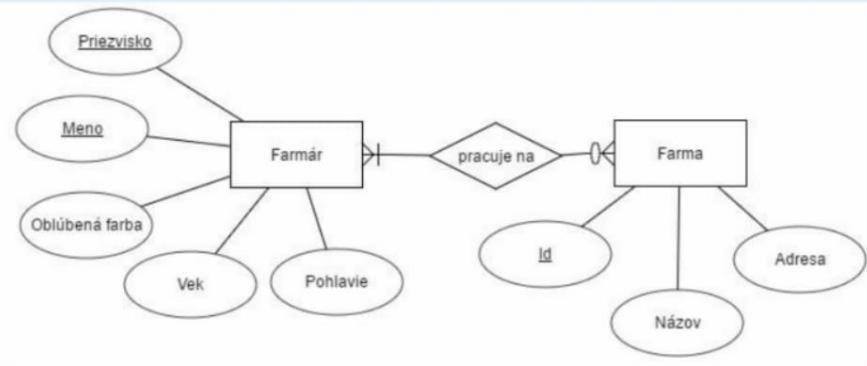
Vo vzorovej DB výsledok dopytu vráti 9 riadkov, iba pre oblast Mesto Kyjev vyjde hodnota v stĺpici na_dedine na 0.



```
select o.nazov, o.populacia - sum(m.populacia) as na_dedine from oblast o left outer join mesto m on o.id = m.id_oblast group by o.nazov, o.populacia order by o.populacia - sum(m.populacia) desc NULLS last;
```

Majme ERD pre farmu aké je ukazany na nasledujucom obrazku. Vyberte pravdivé tvrdenia vzhľadom na vztah "pracuje na".

Majme ERD pre farmu ako je ukázaný na nasledujúcom obrázku. Ako bude v relačnom modele vyjadrený vzťah "pracuje na"?



Označte jednu alebo viac odpovedí:

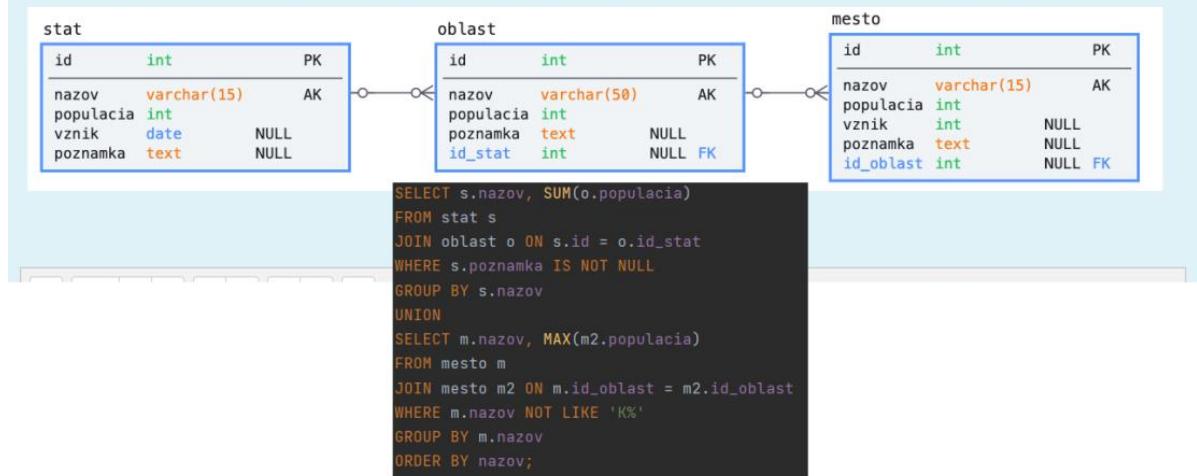
- a. prepájacou medzitabuľkou, ktorá bude mať dva stĺpce - cudzí klúč na "Farmar" a cudzí klúč na "Farma"
- b. cudzím klúčom na strane tabuľky "Farmar", ktorý sa bude odkazovať na tabuľku "Farma"
- c. cudzím klúčom na strane tabuľky "Farma", ktorý sa bude odkazovať na tabuľku "Farmar"
- d. v tabuľke "Farmar" bude stĺpec "id", ktorý bude mať hodnoty podľa stĺpca "id" tabuľky "Farma"

Vytvorte dopyt ktorý vráti 2 stĺpce. V prvom budú názvy takých štátov ktoré majú poznámku a okrem názvov takýchto štátov tu budú aj názvy všetkých miest ktoré nezačínajú písmenom K.

Vytvorte dopyt ktorý vráti 2 stĺpce.

- V prvom budú názvy takých štátov ktoré majú poznámku a okrem názvov takýchto štátov tu budú aj názvy všetkých miest ktoré nezačínajú písmenom K.
- V druhom stĺpci bude pre štáty splňajúce spomenuté podmienky určená suma populácie všetkých im priradených oblastí.
- Pre mestá bude v druhom stĺpci populácia najväčšieho mesta v oblasti ku ktorej patria.
- Výsledok zoradte abecedne podľa názvov v prvom stĺpco.

Vo vzorovej databáze výsledok dopytu vráti 10 riadkov, pričom rozdiel medzi súčtom populácie v oblastiach pre jednotlivé štáty splňajúce podmienky by mal byť 4 347 150

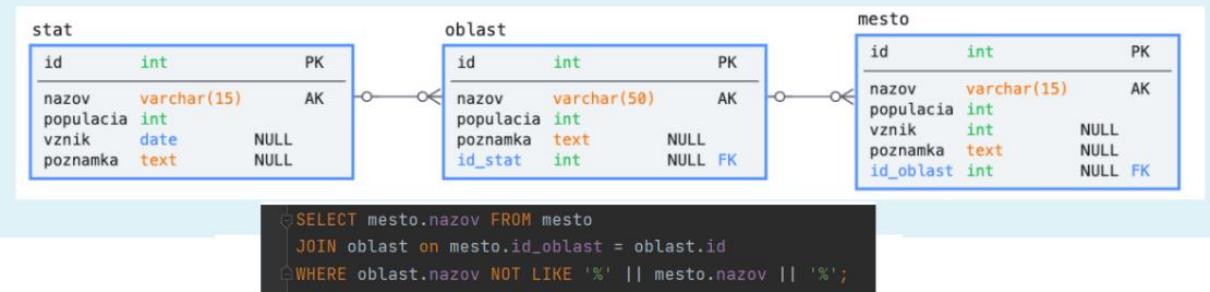


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte SQL dopyt, ktorý vráti názvy všetkých miest, ktorých názov sa nenachádza v plnej dĺžke v názve oblasti, v ktorej sa dané mestá nachádzajú.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti názvy všetkých miest, ktorých názov sa nenachádza v plnej dĺžke v názve oblasti, v ktorej sa dané mestá nachádzajú.

Vo vzorovej DB výsledok dopytu vráti 8 riadkov, nájde aj Košice lebo oblasť sa volá Košický kraj (písmeno e v názve kraja chýba)

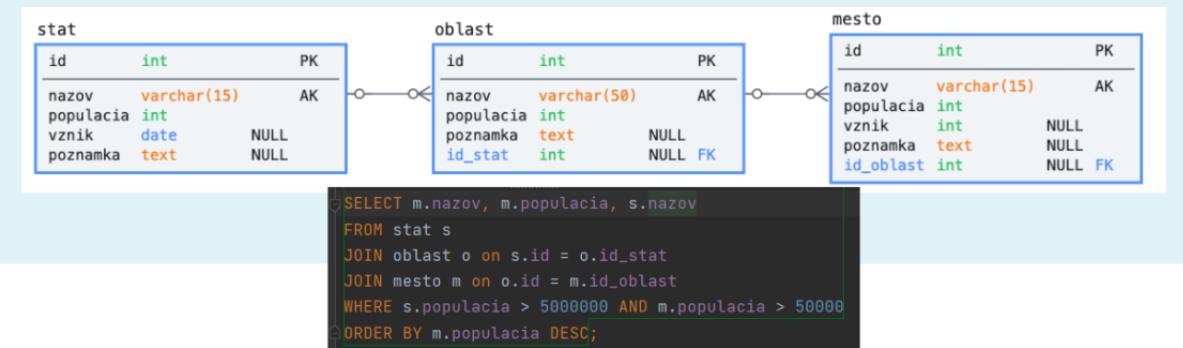


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte SQL dopyt, ktorý vráti názvy miest, ich populáciu a názov príslušného štátu, pričom platí podmienka, že štát musí mať viac ako 5 000 000 obyvateľov a mesto musí mať viac ako 50 000 obyvateľov.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti názvy miest, ich populáciu a názov príslušného štátu, pričom platí podmienka, že štát musí mať viac ako 5 000 000 obyvateľov a mesto musí mať viac ako 50 000 obyvateľov.
- Výsledok zoradte zopustne podľa populácie mesta.
- Výsledný select bude obsahovať 3 stĺpce – názov mesta a jeho populáciu a názov štátu v uvedenom poradí, bez aliasov.

Vo vzorovej DB výsledok dopytu vráti 6 riadkov.

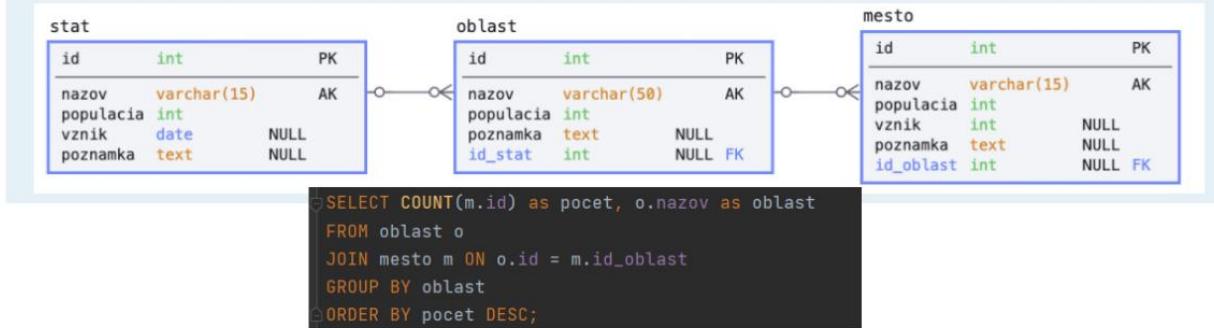


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte SQL dopyt, ktorý vráti počet miest v jednotných oblastiach. Výsledný select bude obsahovať 2 stĺpce – počet miest a názov príslušnej oblasti.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napište SQL dopyt, ktorý vráti počet miest v jednotných oblastiach.
- Výsledný select bude obsahovať 2 stĺpce – počet miest a názov príslušnej oblasti.
- Pomenujte ich ako pocet a oblast, v uvedenom poradí, bez diakritiky.
- Výsledok zoradte zostupne.

Vo vzorovej DB výsledok dopytu vráti 5 riadkov.

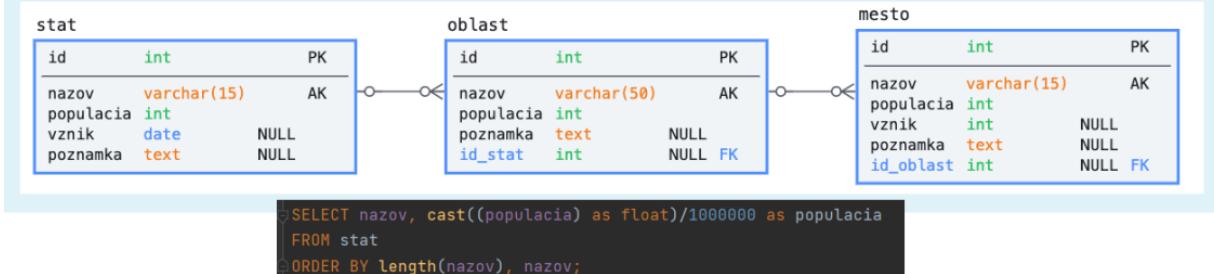


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napište SQL dopyt, ktorý vráti všetky štáty, ich názov a populáciu v miliónoch (len tieto 2 stĺpce v takomto poradí, pri populácii uveďte aj desatininnú časť v celej dĺžke).

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti všetky štáty, ich názov a populáciu v miliónoch (len tieto 2 stĺpce v takomto poradí, pri populácii uveďte aj desatininnú časť v celej dĺžke). Výsledky zoradte podľa dĺžky názvu štátu začínajúc najkratším názvom, a v prípade rovnakej dĺžky podľa abecedy vzostupne.

Vo vzorovej DB by bol druhý riadok „Polsko 38.4336“ a Slovensko by bolo posledné v poradí.

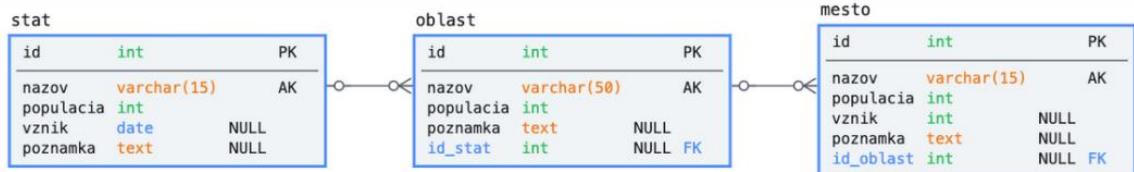


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napište SQL dopyt, ktorý vráti názvy štátov, ich populáciu a dátum vzniku pre štáty, ktoré vznikli v rokoch 1918 alebo 1991 a ich populácia je väčšia ako 10000000.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti názvy štátov, ich populáciu a dátum vzniku pre štáty, ktoré vznikli v rokoch 1918 alebo 1991 a ich populácia je väčšia ako 10000000.
- Výsledok zoradte od najmladšieho štátu po najstarší.
- Výsledný select nech obsahuje 3 stĺpce: názov štátu, populáciu a dátum vzniku, v uvedenom poradí, bez aliasov.

Vo vzorovej DB sú 2 takéto štáty.



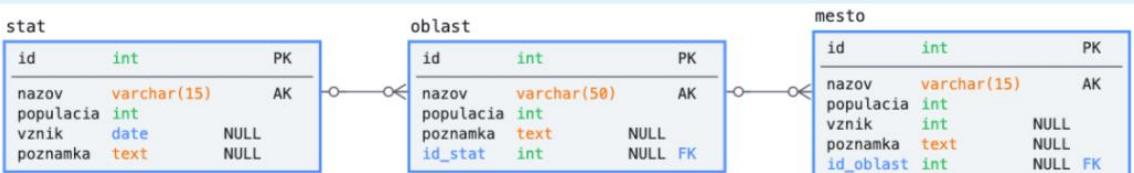
```
SELECT nazov, populacia, vznik
FROM stat
WHERE (date_part('year', vznik) = 1918 OR date_part('year', vznik) = 1991) AND populacia > 10000000
ORDER BY vznik DESC;
```

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte SQL dopyt, ktorý vráti všetky mestá, ich názov, populáciu a kedy vznikli (len tieto 3 stĺpce v takomto poradí), ktoré majú viac ako 100 tisíc obyvateľov a zoradte ich podľa vzniku od najmladšieho.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napíšte SQL dopyt, ktorý vráti všetky mestá, ich názov, populáciu a kedy vznikli (len tieto 3 stĺpce v takomto poradí), ktoré majú viac ako 100 tisíc obyvateľov a zoradte ich podľa vzniku od najmladšieho.

Vo vzorovej DB sú 3 takéto mestá.



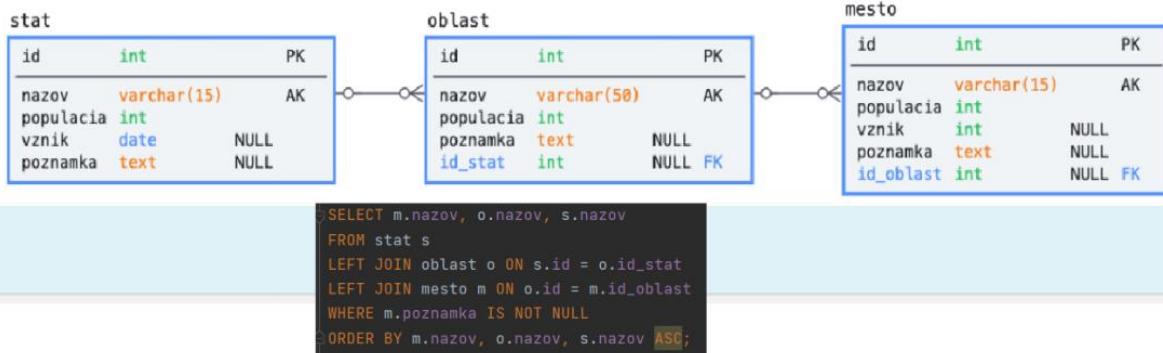
```
SELECT nazov, populacia, vznik
FROM mesto
WHERE populacia > 100000
ORDER BY vznik DESC;
```

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte SQL dopyt, ktorý vráti názvy všetkých miest, oblasti a štátu do ktorého patria pre mestá, ktoré majú uvedenú poznámku.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti názvy všetkých miest, oblasti a štátu do ktorého patria pre mestá, ktoré majú uvedenú poznámku.
- Výsledok zoradťte vzostupne podľa názvu mesta, oblasti a štátu.
- Výsledný select nech obsahuje 3 stĺpce: názov mesta, názov oblasti a názov štátu, v uvedenom poradí, bez aliasov.

Vo vzorovej DB výsledok dopytu vráti 3 riadok.



Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte SQL dopyt, ktorý vráti názov štátu a priemernú populáciu miest štátu, ktorého priemerný počet obyvateľov v mestách je najvyšší.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti názov štátu a priemernú populáciu miest štátu, ktorého priemerný počet obyvateľov v mestách je najvyšší.
- Výsledný select bude obsahovať 2 stĺpce – názov štátu a jeho priemernú populáciu miest v uvedenom poradí.
- Pomenujte ich ako stat a priemer, v uvedenom poradí, bez diakritiky.

Vo vzorovej DB výsledok dopytu vráti 1 riadok.

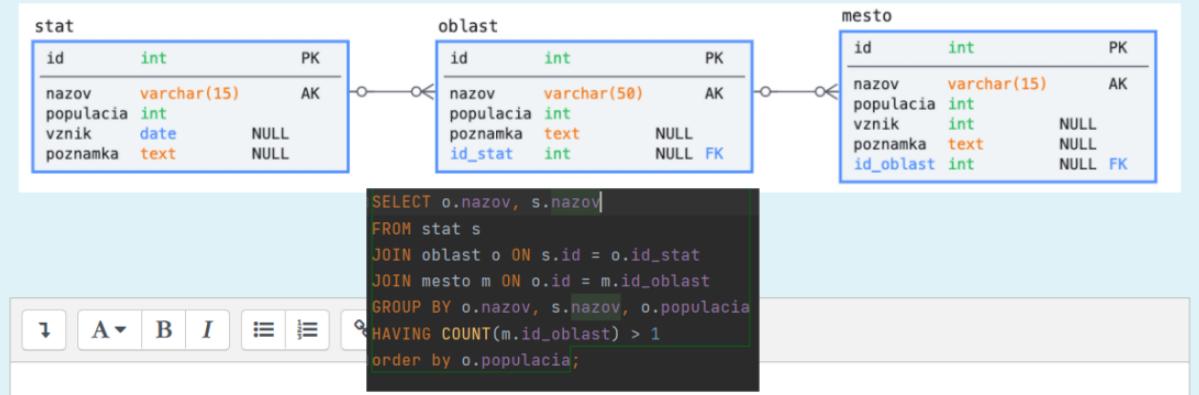


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte SQL dopyt, ktorý vráti všetky oblasti, ich názov a názov štátu ktorému patria(len tieto 2 stĺpce v takomto poradí), ktoré majú vo vzorovej DB viac ako 1 mesto a zoradťe ich podľa populácie od najmenšej.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti všetky oblasti, ich názov a názov štátu ktorému patria (len tieto 2 stĺpce v takomto poradí), ktoré majú vo vzorovej DB viac ako 1 mesto a zoradte ich podľa populácie od najmenšej.
- Výsledný SELECT bude obsahovať 2 stĺpce – názov oblasti a názov štátu v uvedenom poradí, bez aliasov.

Vo vzorovej DB výsledok dopytu vráti 4 riadkov.



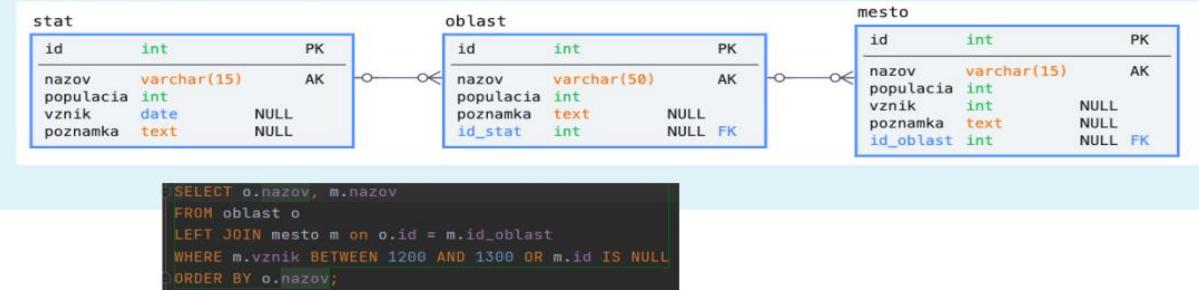
Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte dopyt ktorý vráti dvojice oblasť (prvý stĺpec) – mesto patriace danej oblasti (druhý stĺpec), ale len pre mestá ktoré vznikli medzi rokmi 1200 a 1300 (vrátane) a tiež pre oblasti bez mesta (vo výsledku pre ne bude v pravom stĺpci namiesto názvu mesta NULL).

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napíšte dopyt ktorý vráti dvojice oblasť (prvý stĺpec) – mesto patriace danej oblasti (druhý stĺpec), ale len pre mestá ktoré vznikli medzi rokmi 1200 a 1300 (vrátane) a tiež pre oblasti bez mesta (vo výsledku pre ne bude v pravom stĺpci namiesto názvu mesta NULL).

Výsledky zoradte abecedne podľa názvu oblasti.

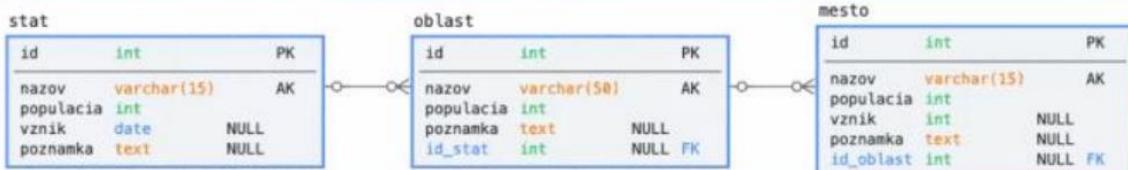
Vo vzorovej DB výsledok dopytu vráti 8 riadkov.



Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napíšte SQL dopyt, ktorý vráti maximum, minimum a priemer populácie všetkých miest.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti maximum, minimum a priemer populácie všetkých miest.



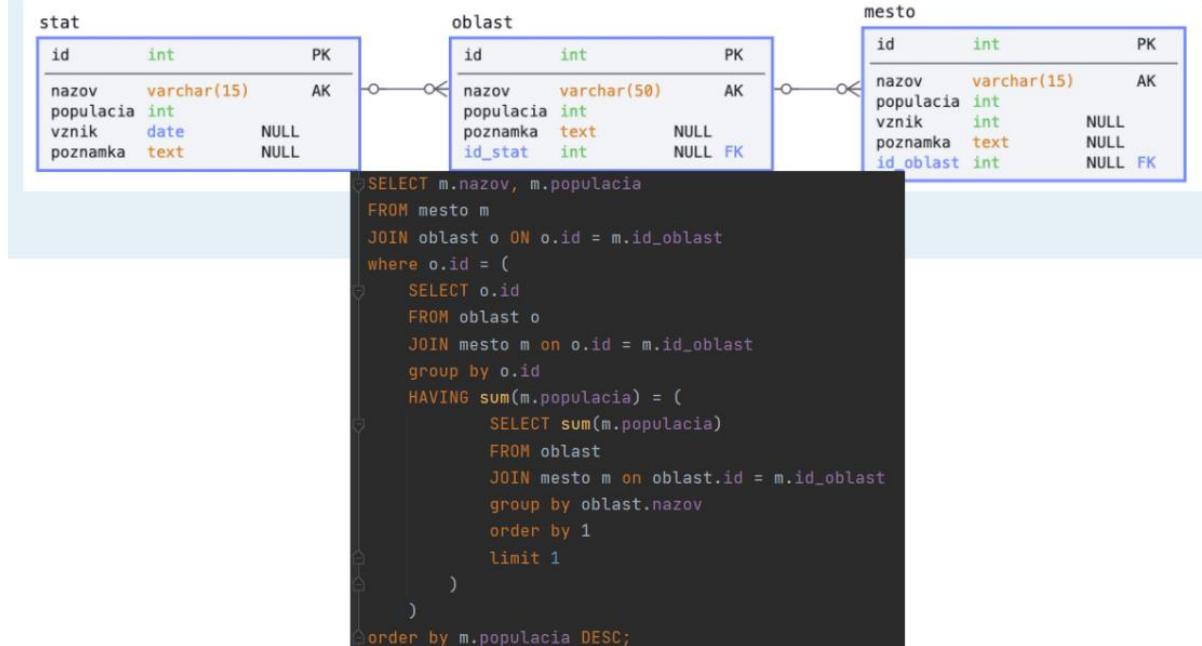
```
SELECT MAX(populacia), MIN(populacia), AVG(populacia)
FROM mesto;
```

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napište SQL dopyt, ktorý vráti všetky mestá a ich počet obyvateľov (len tieto 2 stĺpce), z oblasti v ktorej súčet obyvateľov jej všetkých miest je najnižší.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napište SQL dopyt, ktorý vráti všetky mestá a ich počet obyvateľov (len tieto 2 stĺpce), z oblasti v ktorej súčet obyvateľov jej všetkých miest je najnižší.
- Výsledok zoradte zopustne.
- Výsledný select bude obsahovať 2 stĺpce – názov mesta a jeho populáciu v uvedenom poradí, bez aliasov.

Vo vzorovej DB výsledok dopytu vráti 3 riadky.

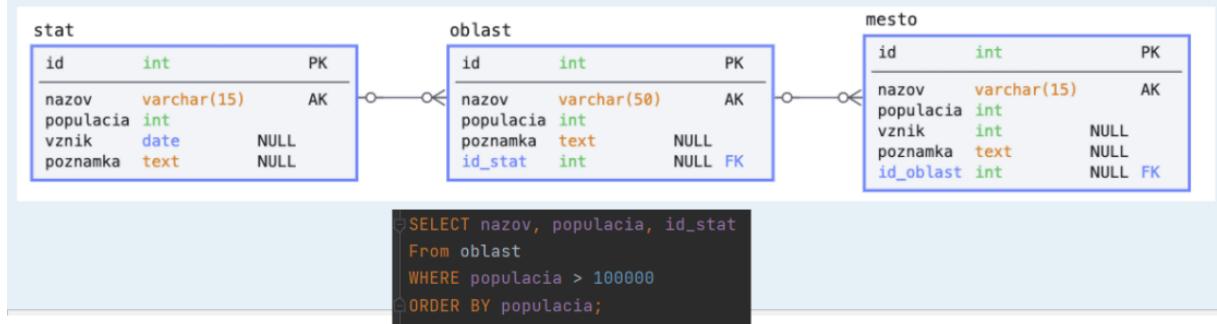


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napište SQL dopyt, ktorý vráti všetky oblasti, ich názov, populáciu a id štátu ktorému patria (len tieto 3 stĺpce v takomto poradí), ktoré majú viac ako 100.000 (stotisíc) obyvateľov a zoradte ich podľa populácie od najmenšej.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Napište SQL dopyt, ktorý vráti všetky oblasti, ich názov, populáciu a id štátu ktorému patria (len tieto 3 stĺpce v takomto poradí), ktoré majú viac ako 100.000 (stotisíc) obyvateľov a zoradťte ich podľa populácie od najmenšej.

Vo vzorovej DB 9 oblastí.

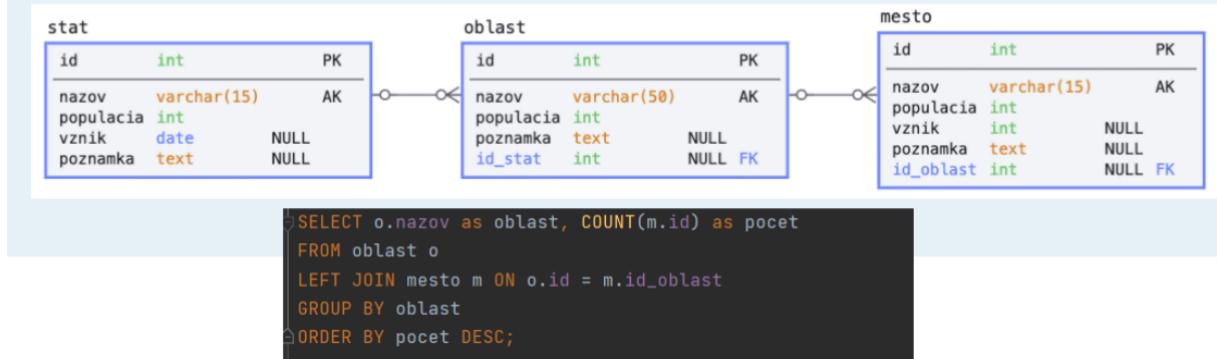


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napište SQL dopyt, ktorý vráti počet miest v jednotných oblastiach. Výsledný select bude obsahovať 2 stĺpce – názov oblasti a počet miest

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napište SQL dopyt, ktorý vráti počet miest v jednotných oblastiach.
- Výsledný select bude obsahovať 2 stĺpce – názov oblasti a počet miest.
- Pomenujte ich ako oblast a pocet, v uvedenom poradí, bez diakritiky.
- Výsledok zoradťte zostupne.

Vo vzorovej DB výsledok dopytu vráti 9 riadkov.

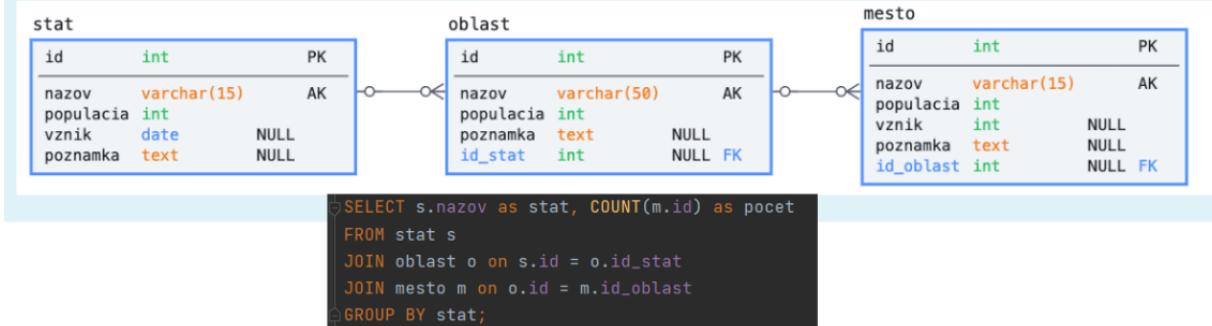


Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Napište SQL dopyt, ktorý vráti pre každý štát počet miest v jednotlivých štátoch.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

- Napíšte SQL dopyt, ktorý vráti pre každý štát počet miest v jednotlivých štátoch.
- Výsledný select bude obsahovať 2 stĺpce – názov štátu a miest príslušného štátu.
- Pomenujte ich ako stat a pocet, uvedenom poradí, bez diakritiky.
- Výsledok zoskupte podľa štátu.

Vo vzorovej DB výsledok dopytu vráti 2 riadky.



Vytvorte SQL dopyt, ktorý vráti názov a čas vzniku pre presne dva podľa populácie najväčšie štáty, ale vyberajte len z tých štátov ktoré nemajú poznámku.

Select nazov, vznik

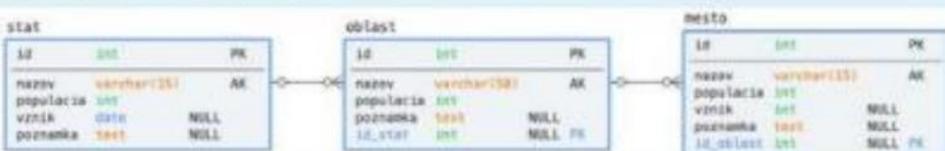
FROM stat

where stat.poznamka is null

order by populacia DESC limit 2;

Vytvorte SQL dopyt, ktorý vráti názov a čas vzniku pre presne dva podľa populácie najväčšie štáty, ale vyberajte len z tých štátov ktoré nemajú poznámku.

Tieto dva štáty môžu zostať zoradené od populáciou najväčšieho po populáciou najmenší štát.



Zobrazte z tabuľky OE.orders roky (order_date) v kt. bolo viac ako 10 objednávok.

```
Select order_date, count(*)  
from orders  
group by order_date  
having count(*)>10
```

Zobrazte z tabuľky OE.orders roky (order_date) v kt. bolo viac ako 10 objednávok.

Zobrazte meno (cust_first_name) a priezvisko (cust_last_name) zákazníkov (OE.customers), ktorých účet manažuje (account_mgr_id) zamestnanec (HR.employees) s menom Gerald Cambrault. Zákazníkov zoradte abecedne najprv podľa priezviska, potom podľa mena.

Odpoved:

```
select cust_first_name, cust_last_name  
from oe.customers  
where account_mgr_id = (select employee_id  
                         from hr.employees  
                         where first_name = 'Gerald' and last_name='Cambrault')  
ORDER BY cust_last_name,cust_first_name;
```

Zobrazte zoskupený identifikátor jazyka (language_id) a priemernú dĺžku názvu produktu (translated_name) z tabuľky OE.product_descriptions pre jazyky, pri ktorých priemerná dĺžka názvu produktu je aspoň 17 znakov.

Odpoved:

```
SELECT language_id, AVG(length(translated_name))  
FROM OE.product_descriptions  
WHERE length(translated_name)>= 17;
```

Vyberte, co musí platiť, aby bola databaza v 2. normalnej forme.

vyberte, čo musí platiť, aby bola databáza v 2. normálnej forme:

Označte jednu alebo viac odpovedí:

- a. v žiadnom stĺpci sa nesmú opakovať rovnaké hodnoty, okrem stĺpca, ktorý je primárnym klúčom
- b. atribút, ktorý nepatrí do primárneho klúča, nesmie závisieť iba na nejakej časti primárneho klúča
- c. databáza musí byť v 1. NF
- d. každý neklúčový atribút funkčne závisí iba na celom primárnom klúči
- e. neklúčové atribúty nesmú závisieť na iných neklúčových atribútoch

Pre kazdu z evidovaných deviatich oblasti zistí ich celkovu populaciu a v dalsom riadku vusku populacie mimo evidovaných miest danej oblasti, teda dostaneme 18 riadkov

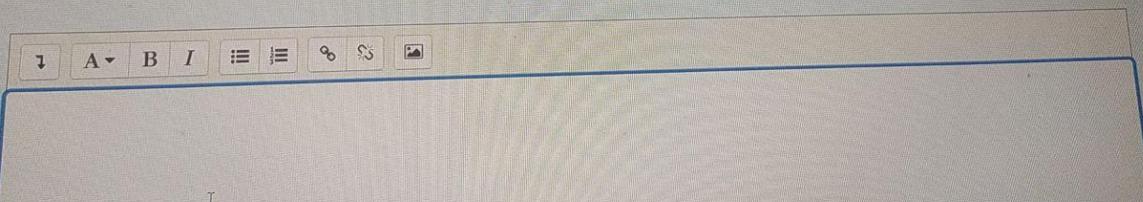
6
vedané
e 7,00

Zvyšný čas 0:43:26

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme. Vytvorte dotaz, ktorý nad vzorovou databázou uskutočňuje nasledujúcu štatistiku.

Pre každú z evidovaných deviatich oblastí zistí ich celkovú populáciu a v ďalšom riadku výšku populácie mimo evidovaných miest danej oblasti, teda dostaneme 18 riadkov.

- V prvom stĺpci má byť názov oblasti, v druhom výška populácie.
- Výpis formátujte tak, aby sa v ňom riadky zobrazili zoradené v abecednom poradí podľa názvu oblasti.
- Pre každú dvojicu pod sebou zobrazených riadkov k rovnakej oblasti musí platiť, že v prvom riadku je celková populácia danej oblasti a v druhom zase výška populácie mimo evidovaných miest danej oblasti.
- Oblast' s názvom Mesto Kyjev bude mať v 1. riadku populáciu 2963199, v 2. riadku 0, lebo podľa našej databázy žijú všetci v evidovanom meste.
- Ak oblasť nemá evidované mesto, tak sa čísla celkovej populácie a populácie mimo miest zhodujú, a dostaneme vo výsledku dva rovnaké riadky pre danú oblasť, napr. pre Hlavné mesto Praha bude populácia v obidvoch riadkoch 1324277. Eliminácia takejto duplicity je chyba v riešení úlohy!
- Populácia je číslo, t.j. výskyt hodnoty NULL vo výstupu je rovnako chyba!



```
select nazov, populacia from
oblast
union all
select
o.nazov, COALESCE(o.populacia-
sum(m.populacia), o.populacia)
from oblast o
left join mesto m on o.id =
m.id_ostat
group by o.nazov, o.populacia
order by nazov;
```

Vytvorte dotaz nad vzorovou databazou, ktorý zoskupí nazvy oblastí do jedného retazca podla ich umiestnenia v spoločnej krajinie v prvom stlpci a krajinu v druhom stlpci.

Pracujeme s databázou, ktorej relačný model je zobrazený na nasledujúcej schéme.

Zväžený čas 0:33:39

Vytvorte dotaz nad vzorovou databázou, ktorý zoskupí nazvy oblastí do jedného reťazca podľa ich umiestnenia v spoločnej krajinie v prvom stĺpco a krajinu v druhom stĺpci.
Stĺpce premenujte na "Oblasti zo spoločnej krajiny" a "Stat".
Správny dotaz vráti 5 riadkov, medzi ktorými bude aj: Kosicky kraj, Presovsky kraj | Slovensko.
POZOR: Pri spájaní nezabudnite na medzeru za čiarkou!

```
select
string_agg(oblast.nazov::char
varying, ', ') , stat.nazov
from oblast
full join stat on oblast.id_stat
= stat.id
group by stat.nazov;
```

ON DELETE NO ACTION

Označte, ako sa nazýva uvedený spôsob dodržania integrity v databáze:

ON DELETE NO ACTION

- a. kaskádny
- b. pasívny
- c. reštrikčný
- d. nulitný