# UPIT

Količina opreme koju koristi pojedina medicinska sestra.

Medicinska sestra Ivana Ivić odlučila se požaliti ravnatelju bolnice kako joj fali opreme koju mora koristiti za kvalitetno obavljanje posla. Kako bi riješio problem medicinske sestre Ivane, ali i možebitne probleme ostalih medicinskih sestara u bolnici, ravnatelj bolnice je za početak zatražio popis svih medicinskih sestara i količine opreme koja im je dostupna za rad.

TRAŽENO RJEŠENJE: **medicinske\_sestre** id, **medicinske\_sestre** ime\_i\_prezime, kolicina\_opreme

KOD ZA UPIT:

CREATE VIEW sestra\_soba AS

SELECT medicinske\_sestre.id, CONCAT(medicinske\_sestre.ime, ' ', medicinske\_sestre.prezime) AS ime\_i\_prezime, soba.id AS id\_sobe

    FROM medicinske\_sestre, soba

    WHERE medicinske\_sestre.id\_odjel=soba.id\_odjel;

CREATE VIEW kol\_po\_sobi AS

SELECT sestra\_soba.id, sestra\_soba.ime\_i\_prezime, COALESCE(kolicina.kol\_u\_sobi, 0) AS kolicina

    FROM sestra\_soba

LEFT JOIN

    (SELECT \*, SUM(kolicina) AS kol\_u\_sobi

        FROM stanje\_opreme

        GROUP BY id\_soba) AS kolicina ON sestra\_soba.id\_sobe=kolicina.id\_soba;

SELECT kol\_po\_sobi.id, kol\_po\_sobi.ime\_i\_prezime, SUM(kolicina) AS kolicina\_opreme

    FROM kol\_po\_sobi

    GROUP BY id;

OPIS UPITA:

S obzirom da je oprema raspoređena po sobama moramo medicinske sestre povezati sa sobama u kojima obavljaju rad kako bi mogli prebrojati opremu koja je dostupna pojedinoj medicinskoj sestri.

* Unutar prvog upita povezala sam tablicu **medicinske\_sestre** i tablicu **soba** pomoću kartezijevog produkta. Kartezijev produkt sam dobila na način da u FROM dijelu navedem obje tablice.
* U WHERE dio upita dodala sam uvjet povezivanja po jednakom **id\_odjel** atributu kako bi ispravna imena medicinskih sestra povezala sa ispravnim brojevima soba u kojima obavljaju rad.
* U SELECT dio navela sam atribute koji će mi koristiti za daljnje rješavanje, a to su **id** atribut iz tablice **medicinske\_sestre**, **ime\_i\_prezime** te **id\_sobe** iz tablice **soba**.
* Atribut **ime\_i\_prezime** sam spojila iz dva atributa naredbom CONCAT.
* Od prvog upita kreirala sam pogled pod nazivom **sestra\_soba** koristeći naredbu CREATE VIEW.
* Nakon što sam kreirala pogled koji nam govori u kojoj sobi radi koja medicinska sestra, pronašla sam količinu opreme koja se nalazi u svakoj od soba na način da sam tablicu **stanje\_opreme** grupirala prema atributu **id\_soba** naredbom GROUP BY.
* Unutar SELECT dijela drugog upita, osim cijele tablice **stanje\_opreme**, dodala sam i atribut **kol\_u\_sobi**.
* Atribut **kol\_u\_sobi** označava količinu opreme koja se nalazi u pojedinoj sobi, a s obzirom da sam već grupirala cijelu tablicu prema atributu **id\_soba**, lako sam izračunala i količinu koristeći naredbu SUM.
* Kako bi povezala medicinske sestre sa količinom opreme, drugi sam upit pomoću naredbe LEFT JOIN pridodala pogledu **sestra\_soba**. Pri povezivanju tablice preimenovala sam ju u **kolicina**, a povezivanje sam obavila prema uvjetu da su atributi **broj\_sobe** iz pogleda **sestra\_soba** i **id\_soba** iz tablice **kolicina** jednaki.
* U SELECT dio upita navela sam samo potrebne atribute, a to su: **id** i **ime\_i\_prezime** iz pogleda **sestra\_soba** i **kolicina**.
* Atribut **kolicina** rezultat je sumiranja količine opreme iz pridodane tablice **kolicina**, a naredbom COALESCE postigla sam upisivanje 0 gdje se nalazi NULL vrijednost unutar atributa.
* U konačnici sam pomoću naredbe CREATE VIEW kreirala pogled pod nazivom **kol\_po\_sobi** u kojem se nalaze sve medicinske sestre i količina opreme koju imaju na raspolaganju.
* Kako bi vidjeli točan popis medicinskih sestara i točnu količinu opreme koju imaju na raspolaganju ponaosob (kako se imena i prezimena ne bi ponavljala i ne bi se prepustilo nama da sami zbrajamo opremu po imenima), kreirala sam i treći upit.
* Unutar trećeg upita sam pogled **kol\_po\_sobi** grupirala prema atributu **id** koji se odnosi na **id** medicinskih sestara i u SELECT dijelu sam, osim atributa koji su potrebni za rješenje, konačno sumirala opremu pomoću funkcije SUM preko atributa **kolicina** i naredbom AS rezultat preimenovala u **kolicina\_opreme**.

REZULTAT:

|  |  |  |
| --- | --- | --- |
| id | ime\_i\_prezime | kolicina\_opreme |
| 500 | Ivana Ivić | 2 |
| 501 | Miliana Milić | 12 |
| 502 | Ivana Marić | 12 |
| 503 | Marko Marulić | 2 |
| 504 | Žarka Stanić | 12 |

# UPIT

Popis SOS kontakata svih pacijenata sa dijagnozom 'Infarctus myocardii acutus'.

Doktor Noris Grubor sa odjela Kardiologije zamolio je medicinsku sestru da obavijesti sve SOS kontakte pacijenata sa dijagnozom Infarctus myocardii acutus da im je stanje kritično zbog proživljenog stresa uzrokovanog potresom. Kako bi to učinila, medicinska sestra zatražila je popis svih SOS kontakata pacijenata s tom dijagnozom.

TRAŽENO RJEŠENJE: **pacijent** ime\_i\_prezime, **sos\_kontakt** sos\_ime\_i\_prezime , **sos\_kontakt** broj\_telefona

KOD ZA UPIT:

CREATE VIEW sa\_dijagnozom AS

SELECT prijem.id\_pacijent

    FROM prijem, dijagnoza

    WHERE prijem.id\_dijagnoza=dijagnoza.id AND dijagnoza.naziv='Infarctus myocardii acutus';

SELECT CONCAT(pacijent.ime, ' ', pacijent.prezime) AS ime\_i\_prezime,

CONCAT(sos\_pod.ime, ' ', sos\_pod.prezime) AS sos\_ime\_i\_prezime, sos\_pod.broj\_telefona

    FROM pacijent

RIGHT JOIN sa\_dijagnozom AS trazena\_dijag ON pacijent.id=trazena\_dijag.id\_pacijent

LEFT JOIN sos\_kontakt AS sos\_pod ON pacijent.id\_sos\_kontakt=sos\_pod.id;

OPIS UPITA:

Kako bi uspješno odgovorili na ovaj upit prvo moramo pronaći kojim je pacijentima dijagnosticiran 'Infarctus myocardii acutus', što ćemo učiniti preko tablice **prijem** unutar koje su nam pobrojane dijagnoze svakog pacijenta. Zatim povezujemo tablice **pacijent** (kako bi dobili potrebne podatke o pacijentu) i **sos\_kontakt** (kako bi dobili potrebne podatke o SOS kontaktu pacijenta).

* Unutar prvog upita u FROM dijelu napravila sam kartezijev produkt tablica **prijem** i **dijagnoza**.
* U WHERE dio upita navela sam dva uvjeta spajanja. Prvi uvjet spajanja odnosi se na atribut **id** dijagnoze kako bi unutar tablice **prijem** dobili ispravna imena postavljenih dijagnoza, a drugi uvjet spajanja odnosi se na konkretan **naziv** dijagnoze koju tražimo u rezultatu.
* U SELECT dijelu upita navela sam samo atribut **id\_pacijent** jer mi je on dovoljan za daljnji rad na upitu.
* Pomoću naredbe CREATE VIEW sam kreirala pogled imena **sa\_dijagnozom** od prvog upita.
* Kako bi na temelju **id\_pacijent** atributa iz pogleda **sa\_dijagnozom** dobila potrebne podatke vezane uz pacijente, pogled **sa\_dijagnozom** sam pod imenom **trazena\_dijag**, pomoću RIGHT JOIN naredbe povezala sa tablicom **pacijent**. Kao uvjet povezivanja navela sam jednakost atributa **id** iz tablice **pacijent** i atributa **id\_pacijent** iz tablice **trazena\_dijag**.
* Jedini podaci koji nam sad fale su podaci vezani uz SOS kontakt podjedinog pacijenta. Kako bi i taj dio upita uspješno riješila uključila sam tablicu **sos\_kontakt** pomoću naredbe LEFT JOIN pod nazivom **sos\_pod**, sa uvijetom da su atributi **id\_sos\_kontakt** iz tablice **pacijent** i **id** iz tablice **sos\_pod** jednaki.
* U SELECT dio upita navela sam atribute **ime\_i\_prezime** (odnosi se na ime i prezime pacijenta), **sos\_ime\_i\_prezime** (odnosi se na ime i prezime SOS kontakta), te **broj\_telefona** (odnosi se na broj telefona SOS kontakta).
* Atribute **ime\_i\_prezime** i **sos\_ime\_i\_prezime** sam spojila pomoću naredbe CONCAT od odvojenih atributa ime i prezime iz tablica **pacijent** i **sos\_pod**.

REZULTAT:

|  |  |  |
| --- | --- | --- |
| ime\_i\_prezime | sos\_ime\_i\_prezime | broj\_telefona |
| Sebastijan Milošević | Aden Kotolaš | 0997871285 |
| Lucas Perić | Vojmil Novaković | 0990409583 |

# UPIT

Doktor s najviše pacijenata.

Ravnatelj bolnice odlučio je dodijeliti nagradu doktoru koji je zadužen za najviše pacijenata.

TRAŽENO RJEŠENJE: **doktor** id, **doktor** ime\_i\_prezime, **doktor** id\_odjel, broj\_pacijenata

KOD ZA UPIT:

CREATE VIEW svi\_brojevi\_pac AS

SELECT doktor.id, CONCAT(doktor.ime, ' ', doktor.prezime) AS ime\_i\_prezime, doktor.id\_odjel,

COALESCE(broj\_pacijenata, 0) AS broj\_pacijenata

    FROM doktor

LEFT JOIN

    (SELECT id\_doktor, COUNT(id\_pacijent) AS broj\_pacijenata

        FROM prijem

        GROUP BY id\_doktor) AS izracun ON doktor.id=izracun.id\_doktor;

SELECT \*

    FROM svi\_brojevi\_pac

    ORDER BY broj\_pacijenata DESC

    LIMIT 1;

OPIS UPITA:

Za rješavanje ovoga upita ključna je tablica **prijem** unutar koje je evidentirana dodjela doktora pacijentu prilikom prijema pacijenta u bolnicu.

* Prvi korak koji sam poduzela u svrhu rješavanja ovoga upita je prebrojavanje pacijenata unutar tablice **prijem** za svakog doktora ponaosob.
* Naredbom GROUP BY grupirala sam prema atributu **id\_doktor** podatke iz tablice **prijem**.
* Kako bi mogla prebrojati pacijente upotrijebila sam naredbu COUNT na atributu **id\_pacijent** unutar SELECT dijela upita i naredbom AS imenovala dobiveni atrinut kao **broj\_pacijenata**.
* Također sam u SELECT dio upita navela atribut **id\_doktor** koji mi je potreban za povezivanje traženih podataka o doktoru.
* Kako bi dobila podatke o doktoru, naredbom LEFT JOIN dodala sam dobiveni rezultat upita unutar tablice **doktor** pod nazivom **izracun**. Kao uvijet spajanja iskoristila sam jednakost atributa **id** iz tablice **doktor** i atributa **id\_doktor** iz tablice **izracun**.
* U SELECT dio upita navela sam iz tablice **doktor** slijedeće atribute: **id**, **ime\_i\_prezime** i **id\_odjel**.
* Atribut **ime\_i\_prezime** dobila sam spajanjem atributa **ime** i atributa **prezime** iz tablice **doktor** koristeći naredbu CONCAT.
* Kreirala sam pogled s imenom **svi\_brojevi\_pac** pomoću naredbe CREATE VIEW.
* Na kraju sam kreirala upit u kojem sam u SELECT dijelu navela sve atribute pogleda **svi\_brojevi\_pac** koju sam navela u FROM dijelu upita.
* Naredbom ORDER BY sam sortirala podatke silazno prema atributu **broj\_pacijenata**.
* Naredbom LIMIT sa vrijednošću 1 sam u konačnici dobila doktora s najviše pacijenata.

RJEŠENJE:

|  |  |  |  |
| --- | --- | --- | --- |
| id | ime\_i\_prezime | id\_odjel | broj\_pacijenata |
| 200 | Krešimira Paspalj | 102 | 9 |

# UPIT

Popis svih lijekova kojima je datum isteka prije 2023. godine, a proizvođač im je Bayer.

Potrebno je napisati narudžbu lijekova za proizvođača Bayer.

TRAŽENO RJEŠENJE: **lijek** id, **lijek** vrsta, **lijek** proizvodac, **lijek** naziv

KOD ZA UPIT:

CREATE VIEW prije\_zadane AS

SELECT \*

    FROM stanje\_lijekova

    WHERE YEAR (rok\_valjanosti) < 2023

    ORDER BY rok\_valjanosti ASC;

SELECT lijek\_pod.id, lijek\_pod.vrsta, lijek\_pod.proizvodac, lijek\_pod.naziv

    FROM prije\_zadane

LEFT JOIN lijek AS lijek\_pod ON prije\_zadane.id\_lijek=lijek\_pod.id

HAVING proizvodac='Bayer';

OPIS UPITA:

Datum isteka roka svakog lijeka vidljiv je unutar tablice **stanje\_lijekova**, a ostali podaci o lijeku koji su nam potrebni da bi izvršili narudžbu proizvođaču nalaze se unutar tablice **lijek**.

* Unutar prvog upita pomoću naredbe YEAR izvršene na atributu **rok\_valjanosti** u WHERE dijelu navela sam kako godina isteka roka mora biti manja od 2023. godine.
* Tablica navedena u FROM dijelu upita, na kojoj sam izvršila ovaj uvijet je **stanje\_lijekova**, a pomoću ORDER BY naredbe datume sam poredala silazno.
* Pomoću naredbe CREATE VIEW kreirala sam pogled imena **prije\_zadane**.
* Kako bi dobila podatke vezane uz lijekove koji su potrebni da bi se izvršila narudžba, povezala sam tablicu **lijek** pomoću LEFT JOIN naredbe pod imenom **lijek\_pod** sa pogledom **prije\_zadane**. Uvijet koji sam postavila za povezivanje je jednakost **id\_lijek** atributa iz pogleda **prije\_zanadne** sa **id** atributom iz tablice **lijek\_pod**.
* Naredbom HAVING sam zadala uvijet da se prikazuju samo oni lijekovi koji imaju pod atributom **proizvodac** naziv traženog proizvođača.
* U SELECT dijelu navela sam samo atribute potrebne za izvršavanje narudžbe, a to su **id**, **vrsta**, **proizvodac** i **naziv** iz tablice **lijek\_pod**.

RJEŠENJE:

|  |  |  |  |
| --- | --- | --- | --- |
| id | vrsta | proizvodac | naziv |
| 800 | Anelgetik | Bayer | Aspirin |
| 812 | Antibiotik | Bayer | Meropenem |

# UPIT

Popis pacijenata sa brojem posjeta (odvojeno posjete SOS kontakata i onih koji nisu SOS kontakti).

Na otpusnom listu pacijenta bolnica svakom pacijentu daje podatak koliko je ukupno imao SOS kontakat posjeta i ostalih posjeta. Zbog potresa bolnica se seli u zamjensku zgradu dok oštećena zgrada ne bude ponovno u funkciji. Ravnatelj bolnice naredio je da se svakom pacijentu prije preseljenja izda dokument sličan otpusnom listu koji između ostalih sadrži i te podatke.

TRAŽENO RJEŠENJE: **pacijent** id, **pacijent** ime\_i\_prezime, **pacijent** spol, sos\_broj\_posjeta, broj\_ostalih\_posjeta

KOD ZA UPIT:

SELECT pacijent.id, CONCAT(pacijent.ime, ' ', pacijent.prezime) AS ime\_i\_prezime,

pacijent.spol, COALESCE(sos.sos\_broj\_posjeta, 0) AS sos\_broj\_posjeta,

COALESCE(ukupno.ukupna\_posjeta, 0)-COALESCE(sos.sos\_broj\_posjeta, 0) AS broj\_ostalih\_posjeta

    FROM pacijent

LEFT JOIN

    (SELECT posjeta.id\_pacijent, COUNT(posjeta.id) AS sos\_broj\_posjeta

        FROM posjeta, sos\_kontakt

        WHERE posjeta.ime=sos\_kontakt.ime AND posjeta.prezime=sos\_kontakt.prezime

        GROUP BY id\_pacijent) AS sos ON pacijent.id=sos.id\_pacijent

LEFT JOIN

    (SELECT posjeta.id\_pacijent, COUNT(posjeta.id) AS ukupna\_posjeta

        FROM posjeta

        GROUP BY id\_pacijent) AS ukupno ON pacijent.id=ukupno.id\_pacijent;

OPIS UPITA:

Za rješavanje ovoga upita koristit ću tri tablice. Unutar tablice **posjeta** možemo provjeriti koliko je posjeta imao pojedini pacijent. U kombinaciji tablica **sos\_kontakt** i **posjeta** možemo provjeriti koliko je od tih posjeta bilo od strane SOS kontakata pacijenta, a unutar tablice **pacijent** možemo naći podatke o pojedinom pacijentu.

* Prvi korak je pronaći broj SOS posjeta, a to sam učinila na način da sam u FROM dijelu upita navela tablice **posjeta** i **sos\_kontakt** i time kreirala kartezijev produkt.
* U WHERE dijelu upita navela sam dva uvjeta. Prvi uvjet garantira jednakost atributa **ime** u objema tablicama, a drugi uvjet atributa **prezime**.
* Sve skupa sam grupirala prema atributu **id\_pacijent** naredbom GROUP BY.
* U SELECT dijelu upita navela sam atribut **id\_pacijent** (koji će mi biti potreban za daljnje povezivanje podataka o pacijentu) i atribut **sos\_broj\_posjeta** koji sam dobila prebrojavajući atribut **id** iz tablice **posjeta** naredbom COUNT.
* Drugi korak je pronaći ukupan broj posjeta. Tablicu **posjeta** grupirala sam naredbom GROUP BY prema atributu **id\_pacijent**.
* U SELECT dijelu upita navela sam atribut **id\_pacijent** (koji mi je potreban za daljnje povezivanje) i atribut **ukupna\_posjeta** koji sam dobila koristeći naredbu COUNT na atributu **id**.
* Kako bi povezala agregirane podatke sa podacima vezanim uz pojedinog pacijenta prvo sam pomoću naredbe LEFT JOIN pod imenom **sos** i uvjetom jednakosti **id** atributa iz tablice **pacijent** sa atributom **id\_pacijent** iz tablice **sos** dodala prvi rezultat unutar tablice **pacijent**.
* Zatim sam na isti način dodala i drugi rezultat samo pod nazivom **ukupno**.
* U SELECT dijelu navela sam atribute **id**, **ime\_i\_prezime** i **spol** iz tablice **pacijent**, te **sos\_broj\_posjeta** i **broj\_ostalih\_posjeta**.
* Atribut **ime\_i\_prezime** objedinila sam koristeći naredbu CONCAT od atributa **ime** i **prezime** iz tablice **pacijent**.
* Kako bi zamjenila NULL vrijendosti sa nulama, u svrhu računanja, koristila sam naredbu COALESCE na atributima **ukupna\_posjeta** iz tablice **ukupno** i **sos\_broj\_posjeta** iz tablice **sos**.
* Atribut **broj\_ostali\_posjeta** dobila sam oduzimanjem atributa **sos\_broj\_posjeta** od atributa **ukupna\_posjeta** u SELECT dijelu upita.

RJEŠENJE:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | ime\_i\_prezime | spol | sos\_broj\_posjeta | broj\_ostalih\_posjeta |
| 400 | Alen Kolić | M | 1 | 0 |
| 401 | Ivan Rupčić | M | 1 | 0 |
| 402 | Lissa Ivić | Ž | 1 | 0 |
| 403 | Klara Zenzerović | Ž | 1 | 0 |
| 404 | Sebastijan Milošević | M | 1 | 0 |
| 405 | Lucas Perić | M | 1 | 0 |
| 406 | Paola Marić | Ž | 1 | 0 |
| 407 | Ema Knežević | Ž | 0 | 1 |
| 408 | Tomi Ivković | M | 0 | 1 |
| 409 | Stefan Markulinčić | M | 0 | 1 |
| 410 | Mia Stepančić | Ž | 0 | 1 |
| 411 | Bruno Marušić | M | 0 | 1 |
| 412 | Iris Orbanić | Ž | 0 | 1 |
| 413 | Nora Marjanović | Ž | 0 | 1 |
| 414 | Borna Karlović | M | 0 | 0 |
| 415 | Oliver Kinić | M | 0 | 0 |
| 416 | Toni Belić | M | 0 | 0 |
| 417 | Petar Dragojević | M | 0 | 0 |
| 418 | Boris Vlašić | M | 0 | 0 |
| 419 | Matija Mladenović | M | 0 | 0 |
| 420 | Filip Perišić | M | 0 | 0 |