

Jezik Baze Podataka SQL

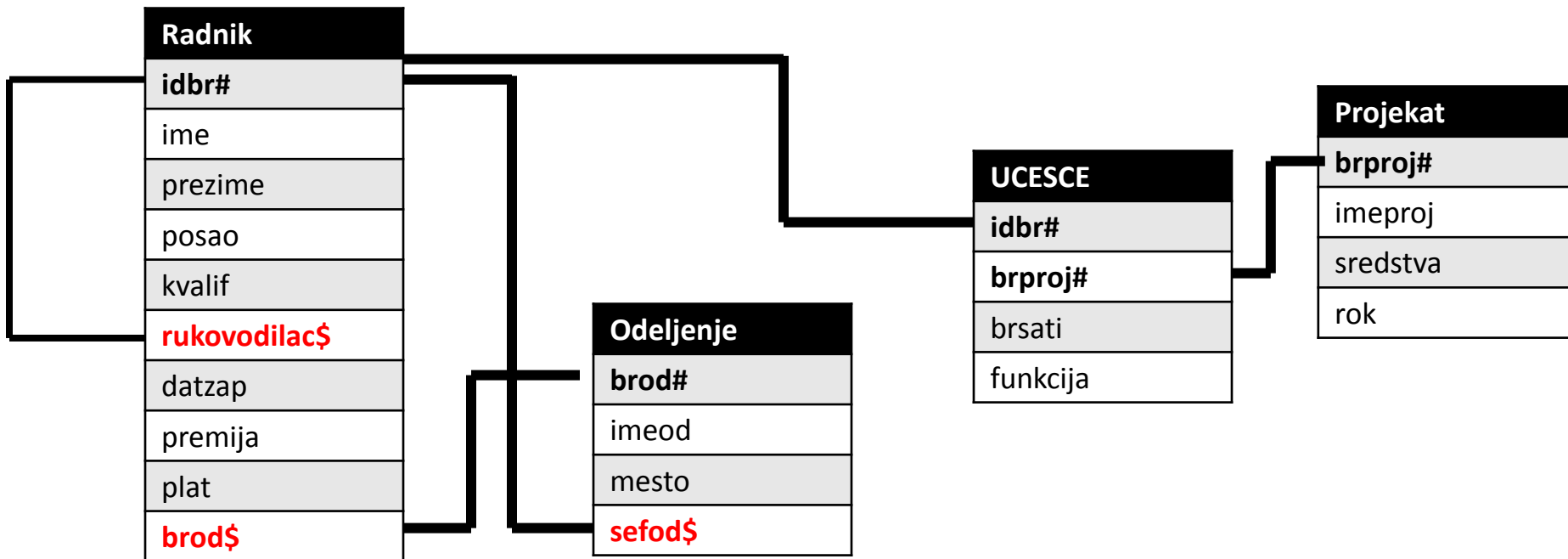
SQL – Baza Kompanija

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap, Premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>



SQL – KREIRANJE TABELA

ODELJENJE <**brod#**, imeod, mesto, **sefod\$**>

RADNIK <**idbr#**, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap. Premija, plata, **brod\$**>

UCESCE <**idbr#**, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

CREATE TABLE RADNIK

```
(idbr INTEGER NOT NULL,  
ime CHAR(25) NOT NULL,  
prezime CHAR(25),  
posao CHAR(10),  
kvalif CHAR(3),  
rukovodilac INTEGER,  
datzap DATETIME,  
premija FLOAT(1),  
plata FLOAT(1) DEFAULT 0);
```

CREATE TABLE PROJEKAT

```
(brproj INTEGER PRIMARY KEY,  
imeproj CHAR(25) NOT NULL UNIQUE,  
sredstva FLOAT(2),  
rok DATETIME);
```

CREATE TABLE ODELJENJE

```
(brod SMALLINT NOT NULL PRIMARY KEY,  
imeod CHAR(5) NOT NULL,  
mesto CHAR(25),  
sefod INTEGER);
```

CREATE TABLE UCESCE

```
(idbr INTEGER NOT NULL,  
brojproj INTEGER NOT NULL FOREIGN KEY REFERENCES PROJEKAT(brproj),  
brsati DATETIME,  
funkcija CHAR(15));
```

KLAUZULA WHERE

ODELJENJE <**brod#**, imeod, mesto, **sefod\$**>

RADNIK <**idbr#**, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap. Premija, plata, **brod\$**>

UCESCE <**idbr#**, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

- Izdvajanje(selekcija) redova koji zadovoljavaju definisan uslov
- Izdvajanje redova koji zadovoljavaju više uslova (**AND**)
- Izdvajanje redova koji zadovoljavaju bar jedan od uslova (**OR**)
- Izdvajanje redova koji zadovoljavaju složene uslove (**AND i OR**)
- Izdvajanje redova čija je vrednost unutar nekih granica (**BETWEEN**)
- Izdvajanje redova čija vrednost pripada nekoj listi vrednosti (**IN**)
- Izdvajanje redova koji ne zadovoljavaju postavljene uslove (**NOT, IS NOT**)
- Izdvajanje redova ako definisana vrednost postoji (**EXISTS**)
- Izdvajanje redova koji mečuju definisan patern (**LIKE**)

KLAUZULA WHERE

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap. premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

Prikazati identifikacioni broj, ime, kvalifikaciju I broj odeljenja zaposlenih koji ne rade u odeljenju 10

```
SELECT idbr, ime, kvalif, brod
FROM RADNIK
WHERE brod <> 10
```

```
SELECT idbr, ime, kvalif, brod
FROM RADNIK
WHERE brod != 10
```

KLAUZULA WHERE

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap. premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

Prikaži ime i posao upravnika i analitičara iz odeljenja 10. Objasniti razliku u upitu

```
SELECT R.ime, R.posao
FROM RADNIK R
WHERE R.posao='upravnik' OR R.posao='analiticar' AND R.brod=10
```

Prikazuje imena i poslove svih upravnika i imena i poslove analitičara koji rade u odeljenju 10

```
SELECT R.ime, R.posao
FROM RADNIK R
WHERE (R.posao='upravnik' OR R.posao='analiticar' ) AND R.brod=10
```

Prikazuje imena i poslove upravnika i analitičara koji rade u odeljenju 10

Operacija **AND** ima viši prioritet od operacije **OR** a operacija **NOT** ima najviši prioritet

KLAUZULA WHERE

ODELJENJE <**brod#**, imeod, mesto, **sefod\$**>

RADNIK <**idbr#**, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap, premija, plata, **brod\$**>

UCESCE <**idbr#**, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

Prikazati ime, datum zaposlenja, platu, premiju i broj odeljenja za zaposlene koji imaju platu između 1000 i 2000 uključujući i te vrednosti

```
SELECT ime, datzap, plata, premija, brod
FROM RADNIK R
WHERE plata BETWEEN 1000 and 2000
```

```
SELECT ime, datzap, plata, premija, brod
FROM RADNIK R
WHERE plata >= 1000 AND plata <= 2000
```

KLAUZULA ORDER BY

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap. premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

Prikazati ime, platu, premiju, posao i broj odeljenja za sve zaposlene. Rezlutate urediti po broju odeljenja u rastućem redosledu a zatim po plati u opadajućem

```
SELECT ime, plata, premija,posao, brod
FROM RADNIK R
ORDER BY brod ASC, plata DESC
```

```
SELECT ime, plata, premija,posao, brod
FROM RADNIK R
ORDER BY brod, plata DESC
```

```
SELECT ime, plata, premija,posao, brod
FROM RADNIK R
ORDER BY 5, 2 DESC
```

Ukoliko atribut sadrži null vrednosti, tada prilikom sortiranja sve null vrednosti se grupišu.

RAD SA TEKSTUALNIM PODACIMA – OPERATOR LIKE

ODELJENJE <**brod#**, imeod, mesto, **sefod\$**>

RADNIK <**idbr#**, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap. premija, plata, **brod\$**>

UCESCE <**idbr#**, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

- | | |
|--------------------------------------|------------------------------------|
| ○ Ime se završava slovom a | WHERE ime LIKE '%a'; |
| ○ Treće slovo imena je V | WHERE ime LIKE '__V%'; |
| ○ U imenu nema slovo N | WHERE ime NOT LIKE '%N%'; |
| ○ Ime je dužine 6 karaktera | WHERE ime LIKE '_____' ; |
| ○ Ime nije dužine od 5 slova | WHERE ime NOT LIKE '_____' ; |
| ○ U imenu je slovo I ispred slova N | WHERE ime LIKE '%I%N%'; |
| ○ Broj ima dve cifre, crticu i cifru | WHERE ime LIKE '[0-9][0-9]-[0-9]'; |

RAD SA TEKSTUALNIM PODACIMA – OPERATOR LIKE

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap. premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

Prikazati ime, posao i broj odeljenja zaposlenih, čija imena počinju slovom M a drugo slovo u imenu je o ili i.

```
SELECT ime, posao, brod  
FROM RADNIK R  
WHERE ime LIKE 'M[o,i]%'
```

```
SELECT ime, posao, brod  
FROM RADNIK R  
WHERE ime LIKE 'Mo%' OR ime LIKE 'Mi%'
```

KLAUZULA GROUP BY

ODELJENJE <**brod#**, imeod, mesto, **sefod\$**>

RADNIK <**idbr#**, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap. premija, plata, **brod\$**>

UCESCE <**idbr#**, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

- Koristi se za grupisanje n torki na osnovu kriterijuma.
- Svi redovi unutar grupe imaju jednake vrednosti atributa po kome se grupišu
- Grup by vraća rezultat koji je karakteristika grupe redova a ne samo jednog reda.

SQL: Agregacija

```
Select A1, A2, ..., An
From R1, R2, ..., Rm
where uslov
Group By kolone
Having uslov
```

“Agregacione” funkcije
min, max, sum, avg, count

Novi iskazi

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata

FUNKCIJE ZA DOBIJANJE GRUPNIH INFORMACIJA

- AGREGATNE FUNKCIJE -

ODELJENJE <**brod#**, imeod, mesto, **sefod\$**>

RADNIK <**idbr#**, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap. premija, plata, **brod\$**>

UCESCE <**idbr#**, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

- Agregatne funkcije daju grupne, sumarne informacije koje se odnose na sve redove u nekoj grupi.
- Grupa može da se formira pomoću klauzule **WHERE** i tada se funkcija primenjuje na sve redove koji zadovoljavaju postavljen kriterijum.
- Agregatne funkcije mogu da se primene i na grupe koje se formiraju grupisanjem vrednosti po željenom atributu klauzulom **GROUP BY**.

- **MAX**
- **MIN**
- **SUM**
- **AVG**
- **COUNT**

FUNKCIJE ZA DOBIJANJE GRUPNIH INFORMACIJA

- AGREGATNE FUNKCIJE -

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap, premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

Prikazati ukupnu platu, srednju platu, maksimalnu i minimalnu platu i broj zaposlenih u celom preduzeću

```
SELECT sum(plata) as [Ukupna plata],  
        avg(plata) as [Srednja plata],  
        max(plata) as [Maksimalna plata],  
        min(plata) as [Minimalna plata],  
        count(*) as [Ukupan broj zaposlenih]  
FROM RADNIK
```

FUNKCIJE ZA DOBIJANJE GRUPNIH INFORMACIJA

- AGREGATNE FUNKCIJE + KLAUZULA WHERE -

ODELJENJE <**brod#**, imeod, mesto, **sefod\$**>

RADNIK <**idbr#**, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap. premija, plata, **brod\$**>

UCESCE <**idbr#**, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

Prikazati ukupnu platu svih radnika koji rade kao analitičari

```
SELECT sum(plata) as [Ukupna plata],  
FROM RADNIK  
WHERE posao='analiticar'
```

FUNKCIJE ZA DOBIJANJE GRUPNIH INFORMACIJA

- AGREGATNE FUNKCIJE + KLAUZULA GROUP BY -

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap. premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

- Sa izuzetkom COUNT(*) agregatne funkcije, sve druge ignorišu NULL vrednosti i mogu da ispred argumenta imaju ključnu reč DISTINCT (različite vrednosti bez duplikata).
- **Grupne funkcije** su dozvoljene u okviru **SELECT liste** (glavnog upita ili podupita) i u **HAVING klauzuli**

Prikazati za svaki posao ukupnu platu radnika koji ga obavljaju. Rezultate urediti po ukupnim primanjima u opadajućem redosledu

```
SELECT posao, SUM(plata) as [Ukupna plata],  
FROM RADNIK  
GROUP BY posao  
ORDER BY SUM(plata) DESC
```


FUNKCIJE ZA DOBIJANJE GRUPNIH INFORMACIJA

- AGREGATNE FUNKCIJE + KLAUZULA GROUP BY -

- Redosled izvršavanja operacija prilikom agregacije nad atributom je sledeći:
 - Izdvoje se zapisi koji zadovoljavaju kriterijum na osnovu WHERE klauzule
 - Grupišu se izdvojeni zapisi po atributu ili atributima iz GROUP BY klauzule
 - Izvršava agregatna funkcija nad svakom izdvojenom grupom

Prikazati brojeve odeljenja i srednju platu u svakom od njih. Iz proračuna isključiti analitičare i upravnike. Rezultate urediti po prosečnim primanjima u rastućem redosledu

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap. premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

```
SELECT brod, AVG(plata) as [Srednja plata],
FROM RADNIK
WHERE posao NOT IN ('analiticar', 'upravnik')
GROUP BY brod
ORDER BY AVG(plata)
```

GRUPISANJE PO VIŠE ATRIBUTA

- AGREGATNE FUNKCIJE + KLAUZULA GROUP BY -

- Grupisanje se može vršiti po više kolona i tada svaka različita kombinacija kolona predstavlja jednu grupu

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap. premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

Prikazati kvalifikaciju, posao i ukupnu platu radnika sa istom kvalifikacijom koji obavljaju određeni posao. Rezultate urediti po kvalifikaciji u rastućem redosledu

```
SELECT kvalif, posao, SUM(plata) as [Ukupna plata]
FROM RADNIK
GROUP BY kvalif, posao
ORDER BY kvalif
```

FILTRIRANJE GRUPA

- AGREGATNE FUNKCIJE + GROUP BY + HAVING -

- Klauzula HAVING se koristi za filtriranje grupa samo ukoliko u upitu postoji GROUP BY klauzula
 - WHERE klauzula se primenjuje na sve redove pre nego što oni postanu deo agregacije dok HAVING se primenjuje na grupu tj. nakon grupisanja
 - Brzina dobijanja odgovora nije ista, uvek je bolje ukoliko je to moguće filtriranje uraditi u WHERE klauzuli.
 - HAVING koristiti kada se radi filtriranje na osnovu vrednosti agregacionih funkcija jer se u WHERE klauzuli agregacione funkcije ne mogu koristiti.

ODELJENJE <brod#, imeod, mesto, **sefod\$**>

RADNIK <idbr#, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap. premija, plata, **brod\$**>

UCESCE <idbr#, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

Prikazati koje poslove obavlja više od jednog radnika unutar svakog odeljenja

```
SELECT brod, posao, count(*) as [Broj zaposlenih]
FROM RADNIK
GROUP BY brod, posao
HAVING count(*) >1
```

FILTRIRANJE GRUPA

- AGREGATNE FUNKCIJE + GROUP BY + HAVING -

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap, premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

Prikazati šifru, ime, platu radnika, broj časova angažovanja na projektima i broj projekata na kojima rade za radnike koji rade na dva ili više projekata

```
SELECT R.idbr, ime, plata, SUM(brsati) as [Ukupan Broj Sati], COUNT(*) as [Broj projekata]
FROM RADNIK R, UCESCE U
WHERE R.idbr=U.idbr
GROUP BY R.idbr, R.ime, R.plata
HAVING count(*) >1
```

FUNKCIJE ZA DOBIJANJE GRUPNIH INFORMACIJA

- AGREGATNE FUNKCIJE -

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap, premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

- SQL može da sadrži i aritmetičke izraze sastavljene od funkcija, imena kolona koje su povezane aritmetičkim operatorima +, -, *, /

Odrediti srednju godišnju platu unutar svakog odeljenja ne uzimajući u obzir plate direktora i upravnika

```
SELECT brod, avg(plata)*12
FROM RADNIK
WHERE posao NOT IN ('direktor', 'upravnik')
GROUP BY brod, posao
HAVING count(*) >1
```

RAD SA NULL VREDNOSTIMA

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap. premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

- Prikazivanje nepostojećeg podatka čija vrednost nije definisana.
- Atribut premija može da ima NULL vrednost za određene radnike. Ukoliko se očekuje da samo nekoliko radnika ima premiju dok većina nema treba razmisliti da se za premiju kreira nova tabela koja bi se zvala PREMIJA<idbr, premija>
- Funkcija **NVL** služi da se u upitu dodeli neka vrednost poljima u koloni koja imaju NULL vrednost.
 - Ukoliko se kolona koristi za sabiranje, potrebno je NULL poljima dodeliti neutralnu vrednost a to je u ovom slučaju 0.

RAD SA NULL VREDNOSTIMA

ODELJENJE <**brod#**, imeod, mesto, **sefod\$**>

RADNIK <**idbr#**, ime, prezime, posao, kvalif, **rukovodilac\$**, datzap. premija, plata, **brod\$**>

UCESCE <**idbr#**, **brproj#**, brsati, funkcija>

PROJEKAT <**brproj#**, imeproj, sredstva, rok>

Za sve zaposlene prikazati njihova imena, broj odeljenja u kome rada i ukupna primanja

```
SELECT ime, brod, plata + IFNULL(premija,0) as 'Ukupna primanja'  
FROM RADNIK
```

Ukoliko ne bismo upisali nulu za NULL vrednosti prilikom sabiranja ne bi se prikazale kolone koje sadrže platu ali ne sadrže premiju.

UPOTREBA ISNULL, IFNULL ILI NVL FUNKCIJE

ODELJENJE <brod#, imeod, mesto, sefod\$>

RADNIK <idbr#, ime, prezime, posao, kvalif, rukovodilac\$, datzap, premija, plata, brod\$>

UCESCE <idbr#, brproj#, brsati, funkcija>

PROJEKAT <brproj#, imeproj, sredstva, rok>

- Ako se grupna funkcija primeni na atribut koji ima null vrednosti, u izveštaju neće biti uključeni oni redovi kod kojih je vrednost tog atributa NULL.
- Preporuka je da se u tom slučaju koloni koja može da sadrži NULL vrednosti dodeli neutralna vrednost 0.
- Dodela neutralne vrednosti vrši se primenom:
 - ISNULL(atribut,vrednost) ---- SQL
 - IFNULL(atribut, vrednost) ---- MySQL
 - NVL(atribut, vrednost) ---- Oracle

Odrediti srednju godišnja primanja unutar svakog odeljenja ne uzimajući u obzir plate direktora i upravnika

```
SELECT brod, AVG(plata + ISNULL(premija,0)*12 AS 'Godisnja primananja unutar svakog odeljenja')
FROM RADNIK
WHERE posao NOT IN ('direktor', 'upravnik')
GROUP BY brod
```