

Operatori u SQL jeziku

Operatori se mogu podeliti u dve osnovne klase: unarne i binarne operatore. Unarni operatori deluju nad jednim operandom, a binarni nad dva.

Podela prema vrsti akcije koje izvode:

Aritmetički operatori - izvršavaju određene akcije sa operandima brojčanog tipa.

| <i>Operator</i> | <i>Svrha</i> | <i>Primer</i> |
|-----------------|--|--|
| +, - | Definiše predznak broja (unarna operacija) | <code>select * from cena where min_cena < -10;</code> |
| +, - | Sabiranje i oduzimanje brojeva (binarni operator) | <code>select iznos+1000 from prodaja;</code> |
| *, / | Množenje i deljenje brojeva (binarni operator) | <code>select iznos/25.5 from prodaja;</code> |
| MOD,% | Moduo - daje ostatak pri deljenju (binarni operator) | <code>select mod(iznos,100) from prodaja;</code> |

Operator sabiranja (+)

Operator sabiranje se može koristiti na više načina. Jedan od načina je u izračunavanju nekog aritmetičkog izraza nad kolonama tabele.

Primer 1

```
SQL> select proizvod_id, min_cena, min_cena+10  
      from cena  
      where proizvod_id='105123';
```

Izlaz:

```
PROIZVOD_ID  MIN_CENA  MIN_CENA+10  
-----  
      105123         29         39  
      105123         32         42  
      105123         34         44
```

U ovom primeru smo dodali 10 na vrednost minimalne cene (MIN_CENA) proizvoda čija je šifra (PROIZVOD_ID) '105123'.

Primer 2.

```
SQL> select katalogska_cena, min_cena,  
      katalogska_cena+min_cena  
      from cena  
      where proizvod_id='105123';
```

Izlaz:

```
KATALOGSKA_CENA  MIN_CENA  KATALOGSKA_CENA+MIN_CENA  
-----  
          36.3         29         65.3  
          37.7         32         69.7  
          40          34         74
```

Kod ovog primera izvršili smo sabiranje katalogske cene i minimalne cene proizvoda '105123'.

U oba primera došlo je do kreiranja po jedne virtuelne (fiktivne) kolone (MIN_CENE+10 odnosno KATALOSKE_CENE+MIN_CENE). SQL dopušta kreiranje virtuelnih kolona kombinovanjem ili modifikovanjem postojećih kolona. Takođe, svakoj koloni možemo fiktivno i da dodelimo ili promenimo ime. Kod postojećih kolona u tabeli menja se samo naziv kolone u ispisu a nikako u tabeli. Svaku kolonu možemo preimenovati koristeći sintaksu:

```
column_name alias (column_name ime kolone i alias novo ime ili sinonim).
```

Primer 3.

```
SQL> select kataloska_cena, min_cena,  
           kataloska_cena+min_cena suma  
           from cena  
           where proizvod_id='105123';
```

Izlaz:

| KATALOSKA_CENA | MIN_CENA | SUMA |
|----------------|----------|------|
| 36.3 | 29 | 65.3 |
| 37.7 | 32 | 69.7 |
| 40 | 34 | 74 |

U ovom primeru poslednjoj koloni dato je ime SUMA. Obratite pažnju da je reč SUMA u upitu napisana malim slovima. SQL imena kolona obično piše velikim slovima. Moguće je svim kolonama dodeliti nova imena. Ako se novo ime piše u više reči onda se ono mora pisati pod navodnicima (""). Ime kolone napisano pod navodnicima ispisuje se onako kako je napisano u upitu tj. nema konverzije u velika slova.

Primer 4.

```
SQL> select kataloska_cena "Tekuca cena",  
           min_cena "Minimalna cena",  
           kataloska_cena+min_cena suma  
           from cena  
           where proizvod_id='105123';
```

Izlaz:

| Tekuca cena | Minimalna cena | SUMA |
|-------------|----------------|------|
| 36.3 | 29 | 65.3 |
| 37.7 | 32 | 69.7 |
| 40 | 34 | 74 |

Operator oduzimanja (-)

Operator oduzimanja se može koristiti na dva načina: promena znaka broja (unarni operator) i u aritmetičkim izrazima kao operator oduzimanja (binarni operator).

Primer 1.

```
SQL> select -kredit_limit dug  
           from kupac  
           where kupac_id='215'
```

Izlaz:

```
DUG
-----
-8000
```

Primer 2.

```
SQL> select katalogska_cena "Tekuca cena",
       min_cena "Minimalna cena",
       katalogska_cena - min_cena Razlika
from cena
where proizvod_id='105123'
```

Izlaz:

| Tekuca cena | Minimalna cena | RAZLIKA |
|-------------|----------------|---------|
| 36.3 | 29 | 7.3 |
| 37.7 | 32 | 5.7 |
| 40 | 34 | 6 |

Operator množenja (*)

Operator deljenja se koristi u izračunavanju aritmetičkih izraza kao operacija množenja.

Primer 1.

```
SQL> select proizvod_id "Sifra proizvoda",
       katalogska_cena "Tekuca cena",
       0.9*katalogska_cena "Cena sa popustom od 10%"
from cena
where proizvod_id='105123';
```

Izlaz:

| Sifra proizvoda | Tekuca cena | Cena sa popustom od 10% |
|-----------------|-------------|-------------------------|
| 105123 | 36.3 | 32.67 |
| 105123 | 37.7 | 33.93 |
| 105123 | 40 | 36 |

Operator deljenja (/)

Operator deljenje se koristi u operaciji deljena pri izračunavanju aritmetičkih izraza.

Primer 1.

```
SQL> select katalogska_cena, min_cena,
       (katalogska_cena+min_cena)/2 "Srednja vrednost"
from cena
where proizvod_id='105123';
```

Izlaz:

| LIST_PRICE | MIN_PRICE | Srednja vrednost |
|------------|-----------|------------------|
| 36.3 | 29 | 32.65 |
| 37.7 | 32 | 34.85 |
| 40 | 34 | 37 |

Primer 2.

```
SQL> select proizvod_id sifra_proizvoda,
       min_cena "Minimalna cena",
       10*min_cena/100 "Umanjenje (10%)",
       min_cena-10*min_cena/100 ukupno
  from cena
 where proizvod_id='105123';
```

Izlaz:

| SIFRA_PROIZVODA | Minimalna cena | Umanjenje (10%) | UKUPNO |
|-----------------|----------------|-----------------|--------|
| 105123 | 29 | 2.9 | 26.1 |
| 105123 | 32 | 3.2 | 28.8 |
| 105123 | 34 | 3.4 | 30.6 |

Znakovni operatori - izvršavaju određene akcije nad nizom znakova.

SQL jezik koristi operator '||' koji omogućuje spajanje dva niza.

Primer 1:

```
SQL> select ime||' '||prezime from radnik;
```

Izlaz (deo):

```
-----
PAUL ROSS
FRANCIS KING
MARY TURNER
DIANE ADAMS
JENNIFER FORD
GRACE ROBERTS
MICHAEL DOUGLAS
BARBARA MILLER
ALICE JENSEN
JAMES MURRAY
```

32 rows selected.

U ovom primeru izvršili smo spajanje imena i prezimena zaposlenih. Da nebi došlo do spajanja reči ubacili smo jedan prazan znak - blanko (' ') između reči.

Operatori poredjenja

omogućuju poredjenje dve vrednosti koje mogu poticati ili iz kolona baze, ili mogu nastati kao rezultat nekog izraza.

| <i>Operator</i> | <i>Svrha</i> | <i>Primer</i> |
|---------------------|--|---|
| = | poredjenje jednakosti dve strane | <code>select * from radnik where prezime='JAMES';</code> |
| !=, <> | Operator nejednakosti | <code>select ime from kupac where grad<>'NEW YORK';</code> |
| <, > | Operator poredjenja veće, manje | <code>select ime, kredit limit from kupac where kredit limit > 5000;</code> |
| <=, >= | Operatori: mane ili jednako i veće ili jednako | <code>select ime, kredit limit from kupac where kredit limit >= 5000;</code> |
| IN | Ispituje se jednakost sa svakim članom koji se testira u uslovu | <code>select ime, grad from kupac where drzava in ('NY', 'TX');</code> |
| NOT IN | Suprotno od IN operatora | <code>select ime, grad from kupac where drzava not in ('NY', 'TX');</code> |
| (NOT) BETWEEN - AND | Ispituje vrednosti koje se (ne) nalaze unutar traženog intervala | <code>select ime from kupac where kredit limit between 5000 and 10000;</code> |
| (NOT) LIKE | Upoređuje sličnost sa ispitivanim vrednostima tako da one odgovaraju nizu znakova koji je definisan zajedno sa znakom %. | <code>select * from radnik where prezime like 'A%';</code> |

Znak jednakosti (=)

Znak jednakosti smo koristili do sada više puta. Najčešće se pojavljuje iza klauzule WHERE. Njegova upotreba je najpogodniji način za biranje jedne vrednosti iz mnoštva,

Primer 1.

```
SQL> select * from odeljenje where ime='SALES';
```

Izlaz:

```
ODELJENJE_ID  IME                LOKACIJA_ID
-----
          30 SALES                123
          13 SALES                122
          23 SALES                124
          43 SALES                167
```

Operator veće (>) i veće i jednako (>=)

Operator veće (>) radi na sledeći način. Ako želimo da iz table KUPAC izdvojimo sve korisnike čiji kreditni limit premašuje 10000 postavimo sledeći upit:

```
SQL> select ime, kredit_limit
       from kupac
       where kredit_limit > 10000;
```

Izlaz:

```
IME                                KREDIT_LIMIT
-----
THE POWER FORWARD                 12000
```

Ako želimo da vidimo i one korisnika čiji kreditni limit uključuje i 10000, moramo koristiti operator veće ili jednako (>=):

```
SQL> select ime, kredit_limit
      from kupac
      where kredit_limit >= 10000;
```

Izlaz:

```
IME                                KREDIT_LIMIT
-----
TKB SPORT SHOP                    10000
EVERY MOUNTAIN                    10000
WOMENS SPORTS                     10000
STADIUM SPORTS                    10000
REBOUND SPORTS                    10000
THE POWER FORWARD                 12000
WHEELS AND DEALS                  10000
```

7 rows selected.

Primer 1.

Imena svih kupaca koja počinju sa T:

```
SQL> select ime from kupac
      where ime like 'T%';
```

Primer 2.

Imena svih kupaca čije je drugo slovo A:

```
SQL> select ime from kupac
      where ime like '_A%';
```

Logički operatori

| Operator | OPIS | Primer |
|----------|--|---|
| NOT | Negacija. Vraća DA ako je uslov NE | select * from artikal where not (kolicina < 250); |
| AND | Logičko I. Vraća DA ako su svi uslovi koji se ispituju tačni | select * from artikal where aktuelna_cena>9 and kolici- na>250; |
| OR | Logičko ILI. Vraća DA ako je bar jedan od uslova koji se ispituju tačan. | select * from artikal where aktuelna_cena>50 or kolici- na<400; |

Primer 1:

Imena svih kupaca koji u imenu nemaju reč SPORT i čiji je kreditni limit veći od 8000:

```
SQL> select ime from kupac
      where ime not like '%SPORT%'
      and
      kredit_limit>8000;
```

Primer 2:

Imena svih kupaca koji su iz TX i CA:

```
SQL> select ime from kupac
      where drzava in ('CA', 'TX');
```

Primer 3:

Imena i kredit limit svih kupaca čiji je kredit limit između 3000 i 8000:

```
SQL> select ime, kredit_limit from kupac
      where kredit_limit between 3000 and 8000;
```