**Dodavanje novih podataka**

Za dodavanje podataka u tabelu koristi se SQL naredba **INSERT...INTO**. Naredba **INSERT...INTO** dodaje nove vrste u tabelu relacione baze podataka. Vrednosti kolona se definišu zadavanjem vrednosti u obliku konstanti ili korišćenjem rezultata SQL upita. U zavisnosti od načina zadavanja vrednosti kolona postoje različiti oblici **INSERT..INTO** naredbe.

Svi primeri u nastavku će biti razmatrani za slučaj tabele PROJEKAT. Ova tabela je kreirana korišćenjem sledeće **CREATE TABLE** naredbe:

**CREATE TABLE** PROJEKAT

(

Broj       **Int**,

Naziv      **VarChar**(25) **NOT NULL**,

Lokacija   **VarChar**(15) **DEFAULT** "Niš",

Sektor     **Integer** **NOT NULL**,

**CONSTRAINT** ProjekatPK **PRIMARY KEY** (Broj),

**CONSTRAINT** NadlezanFK **FOREIGN KEY** (Sektor) **REFERENCES** SEKTOR(Broj)

);

U nastavku je dat oblik **INSERT..INTO** naredbe koji se koristi u situacijama kada se vrednosti kolona zadaju korišćenjem konstanti.

**INSERT INTO** <ime\_tabele>

[(<ime\_kolone1> [,<ime\_kolone2>]...)]

**VALUES** (<vrednost\_kolone1> [{, <vrednost\_kolone2>}...]) ;

Iz definicije možemo zaključiti da je lista kolona opciona. Ukoliko je lista kolona izostavljena, u listi vrednosti kolona moraju se navesti vrednosti za svaku kolonu koja postoji u tabeli u koju se dodaje nova vrsta. U tom slučaju lista vrednosti kolona mora da odgovara redosledu kojim su kolone navedene prilikom kreiranja tabele (u **CREATE TABLE** naredbi). Takođe, vrednosti kolona, moraju biti kompatibilne po tipu sa tipovima podataka koji su za kolone navedeni prilikom kreiranja tabele.

**Primer 1**

SQL naredba kojom se dodaje nova vrsta u tabelu PROJEKAT. Dodaju se informacije o projektu čiji je broj 1, zove se ProizvodX, lociran je u Nišu i za njega je zadužen sektor čiji je broj 5.

**INSERT INTO** PROJEKAT

**VALUES**( 1, "ProizvodX", "Niš", 5);

**Primer 2**

Ukoliko se u **INSERT...INTO** naredbi zadaje lista kolona, moguće je promeniti redosled zadavanja kolona u odnosu na onaj koji je specificiran u **CREATE TABLE** naredbi.

Naredba **INSERT...INTO** kod koje je specificirana lista kolona.

**INSERT INTO** PROJEKAT

(Naziv, Broj, Sektor, Lokacija)

**VALUES**("ProizvodX", 1, 5, "Niš");

Lista kolona u pojedinim slučajevima ne mora biti kompletna. Iz liste se mogu izostaviti kolone kod kojih nije definisano **NOT NULL** ograničenje i kolone koje imaju definisano **DEFAULT** ograničenje. Za istovljene kolone se upisuje podrazumevana vrednost definisana **DEFAULT** ograničenjem ili se upisuje **NULL** vrednost ukoliko **DEFAULT** ograničenje ne postoji.

**Primer 3**

U narednom primeru iz liste kolona je izbačena kolona Lokacija. Pošto je u tabeli za tu kolona definisano **DEFAULT** ograničenje, prilikom dodavanja nove vrste u tu kolonu će biti upisana vrednost 'Niš'. Da **DEFAULT** ograničenje ne postoji u kolonu Lokacija bi bila upisana vrednost **NULL**.

**INSERT INTO** PROJEKAT

(Naziv, Broj, Sektor, Lokacija)

**VALUES**('ProizvodX', 1, 5, 'Niš');

Prilikom dodavanja nove vrste u tabelu biće proverena sva ograničenja koja su definisana nad tabelom: primarni ključ, **NOT NULL**, **CHECK**... Ukoliko bar jedno ograničenje nije zadovoljeno DBMS će prijaviti poruku o grešci.

Dodavanje novih podataka korišćenjem upita

Ukoliko se vrednosti kolona zadaju korišćenjem upita, naredba **INSERT...INTO** ima nešto drugačiji oblik.

  **INSERT INTO** <ime\_tabele>

                [(<ime\_kolone1> [,<ime\_kolone2>]...)]

                <upit>;

U ovom slučaju za listu kolona važe ista pravila kao i u prethodnim situacijama. Vrednosti kolona se sada ne zadaju kako konstante već se zadaju kao rezultujuća tabela nekog upita. Redosled i tip kolona u rezultujućoj tabeli upita mora da odgovara redosledu i tipu kolona u listi.

**Primer 4**

**INSERT...INTO** naredba koja koristi SQL upit da bi u tabelu PROJEKAT dodala podatke iz fiktivne tabele PROJEKAT1.

**INSERT INTO** PROJEKAT

(Naziv, Broj, Sektor, Lokacija)

**SELECT** Naziv, Broj, Sektor, Lokacija

**FROM** PROJEKAT1;

**Primer 5**

U nastavku su date **INSERT...INTO** naredbe za punjenje test podacima tabele RADNIK.

INSERT INTO RADNIK
(MatBr, Ime, Prezime, DatRodj, Pol, Plata, Rukovodilac, Sektor)
VALUES( 123456789, "Đorđe", "B", "Savić", "09/01/1955", "M", 3000, 333445555, 5);

INSERT INTO RADNIK
(MatBr, Ime, Prezime, DatRodj, Pol, Plata, Rukovodilac, Sektor)
VALUES( 3334455555, "Voja", "T", "Vasić", "08/12/1945", "M", 6000, 888665555, 5);

INSERT INTO RADNIK
(MatBr, Ime, Prezime, DatRodj, Pol, Plata, Rukovodilac, Sektor)
VALUES( 999887777, "Ana", "J", "Petrović","19/07/1958", "Ž", 3500, 987654321, 4);

INSERT INTO RADNIK
(MatBr, Ime, Prezime, DatRodj, Pol, Plata, Rukovodilac, Sektor)
VALUES( 987654321, "Jelena", "S", "Marković", "20/06/1965", "Ž", 5300, 888665555, 4);

INSERT INTO RADNIK
(MatBr, Ime, Prezime, DatRodj, Pol, Plata, Rukovodilac, Sektor)
VALUES( 666884444, "Ratko", "K", "Veljković", "15/09/1952", "M", 3800, 333445555, 5);

INSERT INTO RADNIK
(MatBr, Ime, Prezime, DatRodj, Pol, Plata, Rukovodilac, Sektor)
VALUES( 456456456, "Milica", "A", "Stojić", "31/07/1962", "Ž", 2300, 3334455555, 5);

INSERT INTO RADNIK
(MatBr, Ime, Prezime, DatRodj, Pol, Plata, Rukovodilac, Sektor)
VALUES( 987987987, "Jovan", "V", "Janković", "29/03/1959", "M", 2500, 987654321, 4);

INSERT INTO RADNIK
(MatBr, Ime, Prezime, DatRodj, Pol, Plata, Rukovodilac, Sektor)
VALUES( 888665555, "Ivan", "E", "Perić", "09/01/1955", "M", 5500, NULL, 5);