

Primer 3:

Kreirati T-SQL skript koji najpre uklanja proceduru **dohvatiPodatke** ukoliko ona postoji, a potom je ponovo kreira tako da se ponaša kao „parametrizovani pogled“ nad tabelom **Radnik**.

```
IF EXISTS(SELECT * FROM sys.objects WHERE type = 'P' AND ime = 'dohvatiPodatke')
    DROP PROCEDURE dohvatiPodatke
```

```
CREATE PROCEDURE dohvatiPodatke
AS
SELECT * FROM Radnik
```

Primer 4:

- a. Kreirati T-SQL proceduru **dodajNovog** kojom se dodaje jedan red u prethodno kreiranu tabelu **Radnik**. Kao argument procedure se prosleđuju ID, grad i oblast zaposlenog. Procedura treba da doda novog zaposlenog samo ako za dati oblast u tabeli sa zaposlenima već postoji bar jedan zaposleni iz tog oblata.
- b. Demonstrirati upotrebu kreirane procedure, najpre na primeru validnih argumenata, a potom na primeru nevalidnih argumenata.

```
DROP PROCEDURE dodajNovog
```

```
CREATE PROCEDURE dodajNovog @ID nvarchar(20), @Grad nchar(50), @OblastID nchar(1)
AS
BEGIN
    DECLARE @Broj int

    SELECT @Broj = COUNT(*)
    FROM Radnik
    WHERE oblast = @OblastID

    IF @Broj < 1
        RAISERROR ('OblastID nije validan. Proverite uneti argument i ponovite zahtev.', 11, 1)
    ELSE
        INSERT INTO Radnik (ID, grad, oblast)
        VALUES (@ID, @Grad, @OblastID)
END
```

```
EXEC dodajNovog @ID=99, @Grad = 'DDD', @OblastID='N'
```

```
EXEC dodajNovog @ID=99, @Grad = 'DDD', @OblastID='A'
```

[SQL] #50000: OblastID nije validan. Proverite uneti argument i ponovite zahtev.