

OBLIKOVANJE BAZA PODATAKA

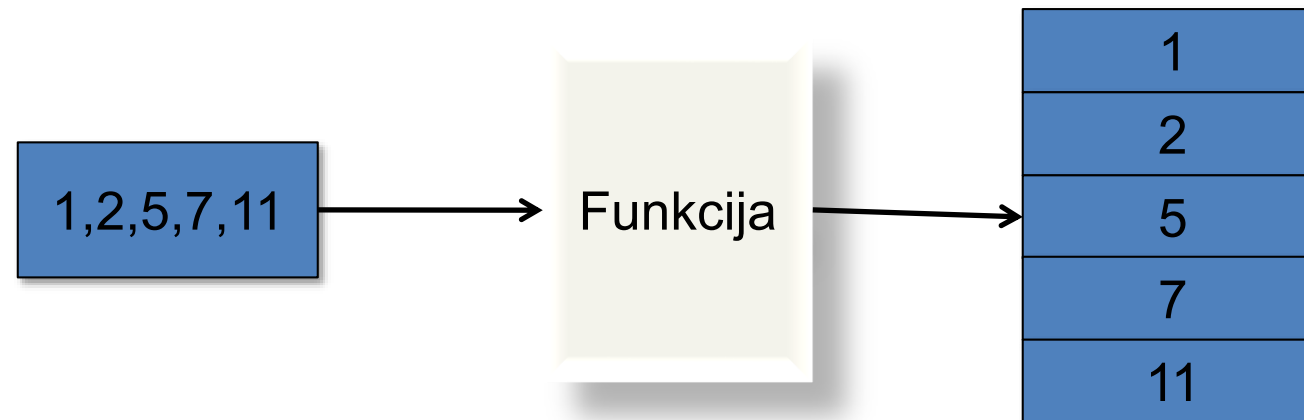
Vježbe 09

Iterativna metoda - uvod

- Iterativna metoda – prosljeđivanje više vrijednosti unutar stringa u proceduru
- Ideja:
 - Procedura će primiti jedan parametar, vjerojatno nvarchar(max)
 - Više vrijednosti odvojenih nekim separatorom
 - Unutar procedure ćemo proslijeđene vrijednosti izvaditi iz string parametra i staviti u tablicu
 - Svaki redak u tablici će biti jedna vrijednost iz stringa
 - Obično posebna funkcija

Vađenje vrijednosti iz stringa

- Kod odabira separatora paziti da se odabrani separator ne može pojaviti kao vrijednost



Sistemske funkcije za rad sa stringovima

LEN (*string*)

- Vraća broj znakova u zadanom stringu

LEN('Ana') - vraća 3

SUBSTRING (string, početak, duljina)

- Vraća dio zadanog stringa

SUBSTRING('Miroslav', 1, 4) - vraća 'Miro'

CHARINDEX (što, gdje [, početak])

- Vraća prvu poziciju stringa *što* u stringu *gdje* veću od *početak*

CHARINDEX ('a', 'Katarina') - vraća 2

CHARINDEX ('a', 'Katarina', 5) - vraća 8

CHARINDEX ('f', 'Katarina', 1) - vraća 0

Rad sa stringovima - zadaci

1. Napisati funkciju koja prima string i vraća novi string sastavljen od: druga polovica stringa + prva polovica stringa.
Ako ulazni string ima neparan broj znakova, neka u prvoj polovici bude manji broj znakova (primjerice, za "Slon" treba vratiti "onSl", za "Mirko" treba "rkoMi").
 - Pozvati funkciju s nekim parametrom i ispisati rezultat.
 - Ispisati obrnuti naziv svakog proizvoda iz tablice Proizvod.
2. Napisati funkciju koja prima string i koja vraća broj pojavljivanja slova 'a' u tom stringu.
 - Pozvati funkciju s nekim parametrom i ispisati rezultat.
 - Dohvatiti sve proizvode iz i pokraj svakog ispisati broj pojavljivanja slova 'a' u nazivu.
3. Napisati funkciju koja prima string i koja vraća broj samoglasnika u tom stringu.
 - Pozvati funkciju s nekim parametrom i ispisati rezultat.
 - Dohvatiti sve proizvode iz i pokraj svakog ispisati broj samoglasnika u nazivu.

XML - uvod

- Tip podataka XML čuva u sebi ispravni XML dokument ili fragment
- Primjer:

```
DECLARE @Adrese xml
```

```
SET @Adrese = '
```

```
<Adrese>
```

```
<Adresa Grad="Zagreb" Pbr="10000">Sunčana 17</Adresa>
```

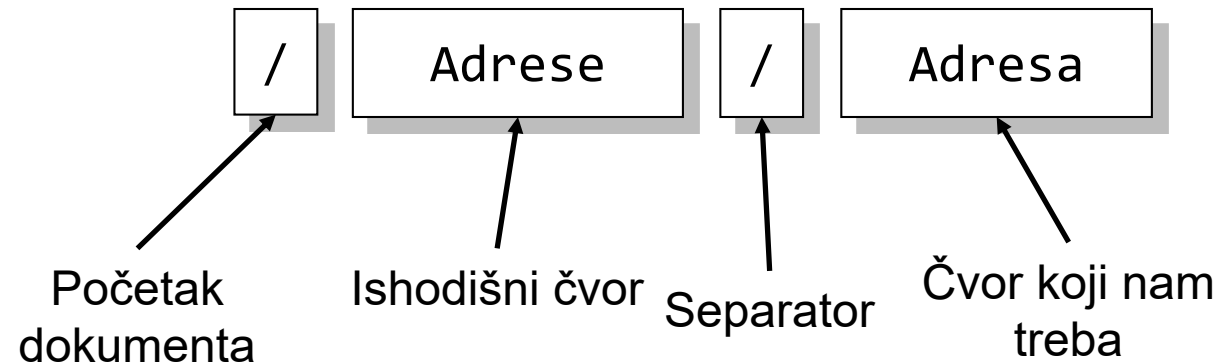
```
<Adresa Grad="Varaždin" Pbr="42000">Široka 25</Adresa>
```

```
<Adresa Grad="Zabok" Pbr="49210">Trg slobode 4</Adresa>
```

```
</Adrese>'
```

Operacije nad XML tipom podataka

- XML sadržava tekstualnu prezentaciju podataka, a nama je za SQL potrebna relacijska prezentacija, tj. tablica
- SQL Server sadrži metode definirane na tipu podataka xml koje služe za dobivanje tablične prezentacije: NODES i VALUE
- Za dohvaćanje važno znati osnovnu XQuery sintaksu:



Kako koristiti xml podatak

- Plan je sljedeći:
 1. Procedura će primiti jedan XML parametar
 2. Napisat ćemo SELECT koji će XML parametar pretvoriti u tablicu
 - a. Metoda NODES() će XML izrezati u retke
 - b. Metode VALUE() će odabrati stupce
 3. Tablicu ćemo dalje koristiti za rješavanje poslovnog problema

Metoda nodes()

- Koristimo je u FROM dijelu
- Iz XML-a vraća tablicu u čijim su retcima dijelovi XML-a
 - Tablica **uvijek sadrži jedan stupac**
 - Moramo toj tablici i stupcu dati nazive pomoću aliasa
- Primjerice:

```
FROM @Adrese.nodes ( '/Adrese/Adresa' ) AS tbl ( stupac )
```
- Dobivamo tablicu **tbl** sa stupcem **stupac** i 3 retka:

stupac
<Adresa Grad="Zagreb" Pbr="10000">Sunčana 17</Adresa>
<Adresa Grad="Varaždin" Pbr="42000">Široka 25</Adresa>
<Adresa Grad="Zabok" Pbr="49210">Trg slobode 4</Adresa>

Metoda value()

- Služi uzimanju SQL vrijednosti iz XML-a i koristimo je u SELECT listi za definiranje stupaca
- Metoda prima dva string parametra:
 - Prvi parametar govori koji dio XML-a želimo uzeti
 - Točka '.' označava vrijednost trenutnog elementa
 - Dvije točke '..' označavaju roditeljski element
 - Atribut dohvaćamo **navođenjem** '@' i njegovog naziva
 - Drugi parametar je tip podataka u koji želimo pretvoriti vrijednost
- Primjerice:

```
SELECT  
tbl.stupac.value('@Grad', 'nvarchar(50)')  
tbl.stupac.value('.', 'nvarchar(50)')
```

XML - zadaci

4. Napisati proceduru koja prima XML dokument oblika:

```
<Gradovi>
```

```
  <Grad>Karlovac</Grad>
```

```
  ...
```

```
</Gradovi>
```

Neka procedura vrati tablicu s nazivima svih gradova. Pozvati proceduru s nekim parametrom.

5. Proširiti prethodni XML dokument tako da svaki grad sadržava atribut "PostanskiBroj". Promijeniti proceduru tako da uz naziv grada ispisuje i poštanski broj.

XML - zadaci

6. Napisati proceduru koja prima XML dokument oblika:

```
<Kupci>  
  <Kupac>  
    <Ime>Mirko</Ime>  
    <Prezime>Mirkić</Prezime>  
    <Grad>Osijek</Grad>  
  </Kupac>  
  ...  
</Kupci>
```

Neka procedura vrati dva skupa redaka:

- Prezimena svih kupaca poslagana abecednim redom
- Nazive gradova. Ako ima više kupaca iz istog grada, grad treba ispisati samo jednom.

XML - zadaci

7. Napisati proceduru koja prima XML dokument oblika:

```
<Osobe>  
  <Osoba ID="1">Ana Anić</Osoba>  
  <Osoba ID="2">Mira Mirić</Osoba>  
  ...  
</Osobe>
```

Neka procedura vrati imena i prezimena te ID-eve svih osoba. Svaki ID treba nadopuniti s nulama s lijeve strane na duljinu od 5 znakova (Primjerice, za ID = 1 treba ispisati "00001", za ID = "176" treba ispisati "00176")