

Baze podataka II

Modul 7 – Jezik SQL

DDL komande

Rad sa bazom podataka i tabelama



Summary

- Rad sa bazom podataka
- Tipovi podataka
- Rad sa tabelama
- Ograničenja

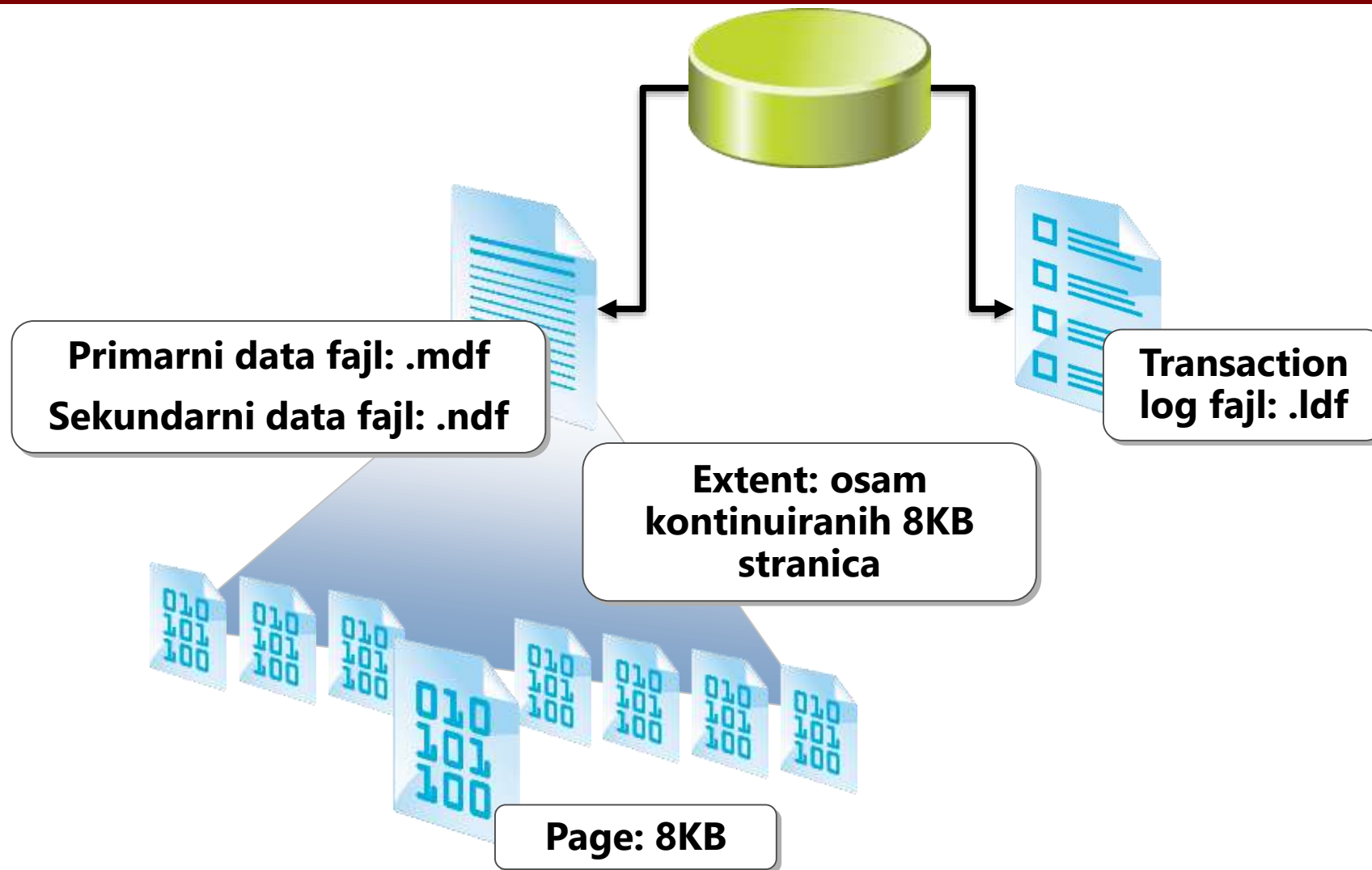


Lekcija 1: Rad sa bazom podataka

- Kako su podaci pohranjeni u bazi
- Arhitektura pohrane
- Planiranje kapaciteta
- Lokacija DATA i LOG fajla
- Grupe fajlova
- Kreiranje korisničke baze
- Izmjena i brisanje baze podataka



Arhitektura pohrane



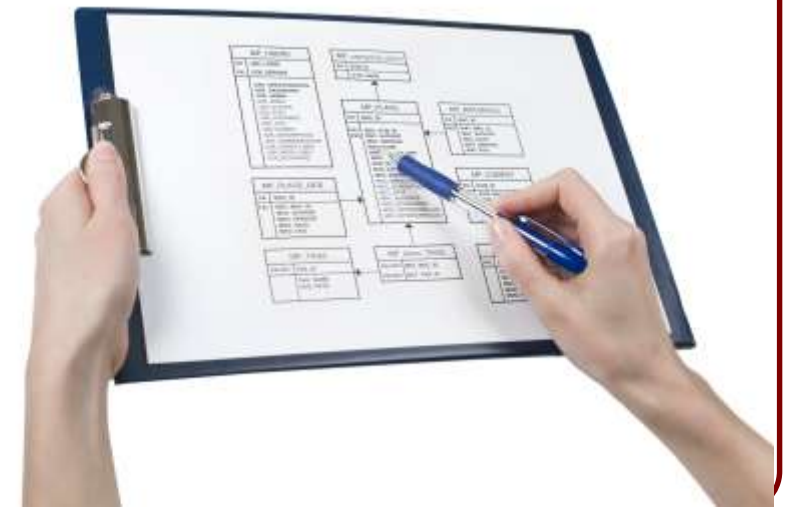
Kako su podaci pohranjeni u bazi

- Baza podatak se sastoji od minimalno dva fajla
- Data (MDF) – primarni data fajl;
 - Može biti i sekundarni (NDF)
- Log (LDF) – log fajl baze podataka;
- Za kreiranje nove baze se koristi kopija “model” baze podataka
- Podaci se pohranjuju u 8 KB blokove koji se zovu stranice (pages);
- 8 stranica u kontinuitetu se naziva extent;
 - Služe za pohranu tabela i indeksa
 - Manje tabele mogu da dijele „extent“ sa drugim objektima baze podataka;
- Log fajl sadrži informacije neophodne za restauraciju baze u slučaju greške;
 - Nije organizovan na principu stranica
- DATA i LOG fajl imaju svoje fizičko ime (na disku) i logičko koje možemo koristiti u SQL kodu;



Planiranje kapaciteta

- **Elementi koji utiču na kvalitetnu procjenu veličine baze su:**
 - Veličina model baze podataka;
 - Arhitektura sistema: web, desktop, mobile, cloud, hibrid;
 - Broj korisnika;
 - Procjena rasta baze podataka tokom vremena;
 - Indeksiranje tabela;
 - Veličina Transaction Log fajla;



Lokacija DATA i LOG fajla

- Zbog WAL, DATA i LOG se trebaju izolirati na odvojene fizičke diskove
- Broj DATA fajlova zavisi od inicijalne procjene kapaciteta i rasta baze podataka
- Poželjno je koristiti RAID polja
- Minimalno RAID 0 i RAID 1

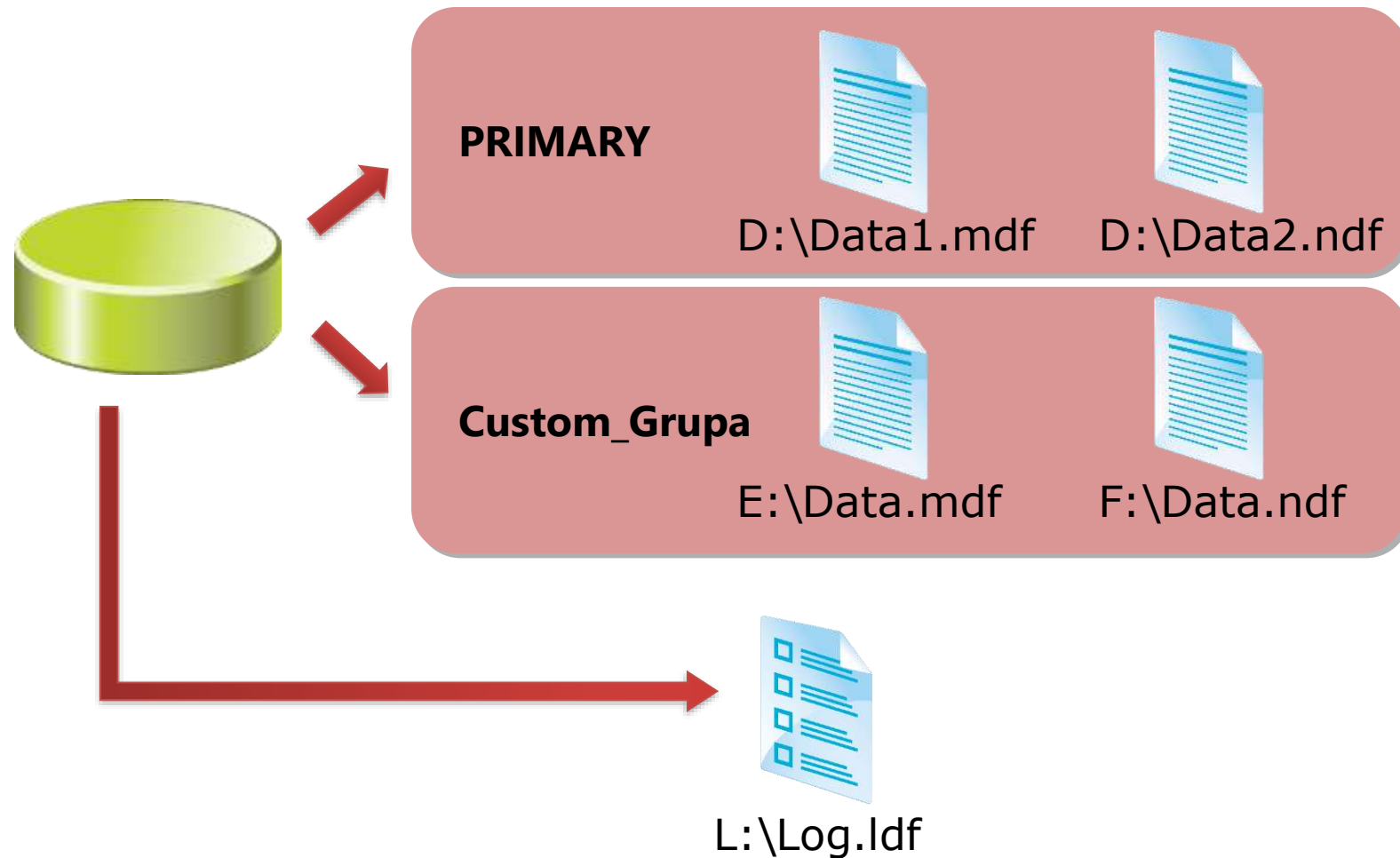
RAID 0



RAID 1



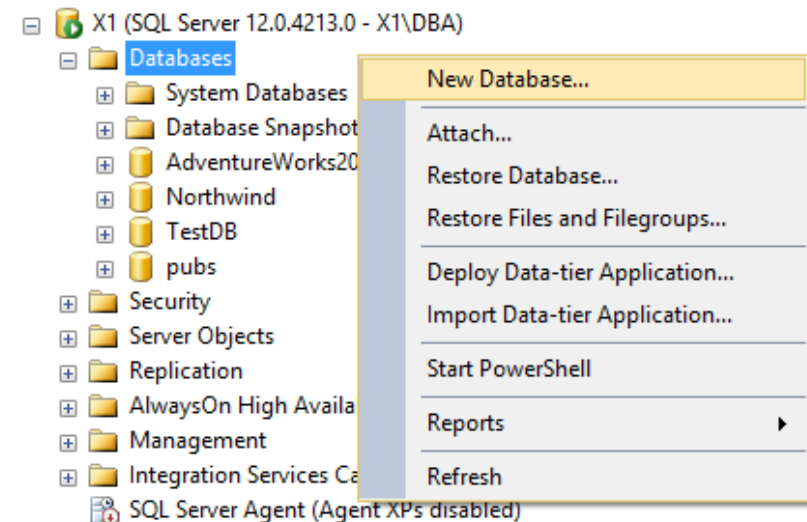
Grupe fajlova



Kreiranje korisničke baze

- GUI (SQL Server Management)
- SQL kod (CREATE DATABASE komanda)

```
CREATE DATABASE Prodaja  
GO  
-- default postavke servera
```



```
CREATE DATABASE Prodaja  
ON  
    (NAME = Prodaja_dat, FILENAME = 'D:\Data\Prodaja.mdf', SIZE = 100MB, MAXSIZE = 500MB,  
      FILEGROWTH = 20% )  
LOG ON  
    (NAME = Prodaja_log, FILENAME = 'E:\Logs\Prodaja.ldf', SIZE = 20MB, MAXSIZE =  
      UNLIMITED, FILEGROWTH = 10MB );
```

Izmjena i brisanje baze podataka

```
ALTER DATABASE Prodaja  
SET READ_ONLY;
```

```
DROP DATABASE Prodaja  
GO
```

Lekcija 2: Tipovi podataka

- Osnove tipova podataka
- Egzaktni numerički tipovi
- Aproksimacijski numerički tipovi
- Tipovi za datum i vrijeme
- Jedinstveni identifikator
- Tekstualni tipovi podataka
- Konverzija tipova podataka
- Ostali tipovi



Osnove tipova podataka

- Tipovi podataka predstavljaju ograničenja u smislu vrstu i opsega podataka koju možete pohraniti u kolone;
- Odluka o tipovima podataka je jedna od najvažnijih u procesu dizajniranja sloja podataka;
- Nije moguće kvalitetno dizajnirati tabele bez poznavanja ove oblasti;
- Pogreške mogu dovesti do:
 - Pada performansi;
 - Gubitka podataka;
 - Pogrešnog prikaza podataka
 - Itd.
- **Tri vrste**
 - Sistemski, Alias-i i korisnički definisani tipovi

Egzaktni numerički tipovi

Data Type	Notes
tinyint	8 bits (0 to 255)
smallint	16 bit integer (-32768 to 32767)
int	32 bit integer (-2147483648 to 2147483647)
bigint	64 bit integer (-2^{63} to $2^{63} - 1$)
decimal	fixed precision and scale ($-10^{38+1} - 10^{38-1}$)
numeric	functionally equivalent to decimal
smallmoney	fixed scale of 4 decimal places in 32 bits
money	fixed scale of 4 decimal places in 64 bits
bit	values of 1, 0, or NULL

Aproksimacijski numerički tipovi

- Samo dva tipa Float i Real
- Rijetko se koriste u poslovanju iz razloga što nisu precizni

Data Type	Notes
float	approximate value with scale of up to 53 digits
real	ISO standard floating point with fixed storage of 32 bits

Tipovi za datum i vrijeme

- Bogata palata opcija za pohranu datuma i/ili vremena
- Pažljivo birati u zavisnosti od poslovnih potreba i procesa

Data Type	Notes
date	0001-01-01 to 9999-12-31 with accuracy of 1 day
datetime2	same date storage as date data type plus time storage as per time data type
datetime	1753-01-01 to 9999-12-31 with accuracy of 3.33 milliseconds
datetimeoffset	as per datetime2 plus timezone offset of -14:00 to +14:00
smalldatetime	1900-01-01 to 2079-06-06 with 1 minute accuracy
time	00:00:00.0000000 to 23:59:59.9999999 with 100 nanosecond accuracy

Jedinstveni identifikator

- Jedinstveni identifikator se obično koristi za pohranu GUID vrijednosti
- To nije IDENTITY
- GUID (globally unique identifier)
- 128 bitni cijeli broj
 - Nisu dozvoljene standardne aritmetičke operacije
- =, <>, <, >, <=, >= su podržane skupa sa NULL i NOT NULL provjerama
- Generišu se preko NEWID() funkcije

Tekstualni tipovi podataka

- Fiksne i varijabilne dužine sa podrškom za CLOB
- Podrška za Unicode tipove podataka

Tipovi podataka	Veličina	Opseg
char (n)	1-8 bajta	8000 karakteri (ANSI)
nchar(n)	2-8 bajta	4000 karakteri(Unicode)
varchar(n)	1-8 bajta	8000 karakteri (ANSI)
varchar(max)	do 2 GB	1,073,741,824 karakteri
nvarchar(n)	2-8 bajta	4000 karakteri(Unicode)
nvarchar(max)	do 2 GB	536,870,912 (Unicode)
text	do 2 GB	1,073,741,824 karakteri
ntext	do 2 GB	536,870,912 (Unicode)

```
DECLARE @Hello  
nvarchar(20);
```

```
SET @Hello = N'Hello';  
SELECT @Hello  
SET @Hello = N'你好';  
SELECT @Hello  
SET @Hello = N'السلام عليكم';  
SELECT @Hello
```

Konverzija tipova podataka

- **CAST, CONVERT i PARSE** mijenjaju tipove podataka
- **CAST** je baziran na ANSI SQL standardu
- **CONVERT** omogućava izbor stila
- **PARSE** izbor „kulture“

```
SELECT CAST(SYSDATETIME() AS nvarchar(30));
```

```
SELECT CONVERT(varchar(8), SYSDATETIME(), 112);
```

```
SELECT CONVERT(char(8), 0x4E616d65, 0)  
AS 'Stil 0, binarni u tekstualni';
```

```
SELECT PARSE('Monday, 13 December 2010' AS datetime2 USING 'en-US');
```

Ostali tipovi

Data Type	Notes
binary	Is a fixed length binary data
varbinary	Is a variable length binary data
varbinary(max)	Is a long variable length binary data
image	Is a long variable length binary data – deprecated – use varbinary(max) instead
hierarchyid	Represents a position in a tree hierarchy
sql_variant	Stores values of various other data types
xml	Is XML data stored in an internal format
cursor	Is used for variables that need to hold a reference to a cursor
table	Holds an entire resultset
geometry, geography	Are used to hold spatial data

Lekcija 3: Rad sa tabelama

- **Kreiranje tabele**
- **Brisanje tabele**
- **Izmjena tabele**
- **Privremene tabele**
- **Izračunate kolone**



Kreiranje tabele

- Tabele se kreiraju sa **CREATE TABLE** komandom
- Neophodno je specificiranje kolona
- Poželjno je da tabela ima PK, kao bi izbjegli „heap“ strukturu
 - Loše po performanse

```
CREATE TABLE Kupci  
( KupacID int IDENTITY(1,1),  
  Prezime nvarchar(30) NOT NULL,  
  Ime nvarchar(30) NOT NULL,  
  Telefon nvarchar (30) NULL  
);
```

Brisanje tabele

- Tabele se uklanjaju sa **DROP TABLE** komandom
- Referencirane tabele (veze preko FK) se neće automatski brisati
- SQL kod referenciran za tabelu koju brišemo se neće obrisati automatski
 - Procedure
 - Funkcije

```
DROP TABLE Kupci;  
GO
```

Izmjena tabele

- Tabela se modifikuje sa **ALTER TABLE** komandom
- Dodaje i uklanja kolone i ograničenja
- Podaci ostaju u tabeli osim u slučaju brisanja kolone sa podacima

```
ALTER TABLE Kupci  
ADD Email nvarchar(100) NOT NULL;  
GO
```

```
ALTER TABLE Kupci  
DROP COLUMN Telefon;  
GO
```

Privremene tabele

- **Privremene sesijske tabele su vidljive jedino njihovim vlasnicima (kreatorima) unutar iste sesije ili podseseije**
 - Počinju sa znakom #
 - Brišu se automatski prilikom diskonektovanja
 - Bolje je brisanje uraditi u kodu nego da zavisi od sesije
- **Globalne privremene tabele su vidljive svim korisnicima**
 - Počinju sa znakom ##

```
CREATE TABLE #Squares
(  NumberID int,
    NumberSquared int
);
GO
```


Izračunate kolone

- Izračunate kolone se generišu na osnovu drugih kolona i/ili funkcija
- Često se koriste u cilju sprečavanja denormalizacije strukture
- Opcija **PERSISTED** može da poboljša performanse **SELECT** komande nad izračunatim kolonama

```
CREATE TABLE Kupci
( KupacID int IDENTITY(1,1),
  Prezime nvarchar(30) NOT NULL,
  Ime nvarchar(30) NOT NULL,
  Telefon nvarchar (30) NULL,
  DatumRodjenja date NOT NULL,
  GodinaRodjenja AS DATEPART (year, DatumRodjenja) PERSISTED
);
```

Lekcija 4: Ograničenja

- Ograničenja primarnog ključa
- UNIQUE ograničenja
- Ograničenja spoljnjeg ključa



Ograničenja primarnog ključa

- Podatak za jedinstveno označavanje zapisa u tabeli
- Mora biti jedinstvena i poznata vrijednost
- Može uključivati više kolona (kompozitni ključ)

```
CREATE TABLE Kupci
( KupacID int IDENTITY(1,1) CONSTRAINT PK_Kupac PRIMARY KEY,
  Prezime nvarchar(30) NOT NULL,
  Ime nvarchar(30) NOT NULL,
  Telefon nvarchar (30) NULL,
  DatumRodjenja date NOT NULL,
  GodinaRodjenja AS DATEPART (year, DatumRodjenja) PERSISTED
);
```

UNIQUE ograničenja

- Koriste se kada je potrebno da osiguramo jedinstvenost, ali bez upotrebe PK
- Mora biti jedinstven podatak, ali jedan od njih može biti NULL
- Može uključivati više kolona

```
CREATE TABLE Kupci
(
  KupacID int IDENTITY(1,1) CONSTRAINT PK_Kupac PRIMARY KEY,
  Prezime nvarchar(30) NOT NULL,
  Ime nvarchar(30) NOT NULL,
  Telefon nvarchar (30) NULL,
  Email nvarchar (100) NOT NULL CONSTRAINT UQ_Email UNIQUE,
  DatumRodjenja date NOT NULL,
  GodinaRodjenja AS DATEPART (year, DatumRodjenja) PERSISTED
);
```

Ograničenja spoljnjeg ključa

- Osiguravanje referencijalnog integriteta između tabela
- Mora biti referenca između PRIMARY KEY ili UNIQUE kolona
- Može biti NULL

```
CREATE TABLE Kupci
(
  KupacID int IDENTITY(1,1) CONSTRAINT PK_Kupac PRIMARY KEY,
  Prezime nvarchar(30) NOT NULL,
  Ime nvarchar(30) NOT NULL,
  Telefon nvarchar (30) NULL,
  Email nvarchar (100) NOT NULL CONSTRAINT UQ_Email UNIQUE,
  DatumRodjenja date NOT NULL,
  GradID int NOT NULL CONSTRAINT FK_Kupac_Gradovi FOREIGN KEY REFERENCES
  Gradovi (GradoviID),
  GodinaRodjenja AS DATEPART (year, DatumRodjenja) PERSISTED
);
```

Pitanja

