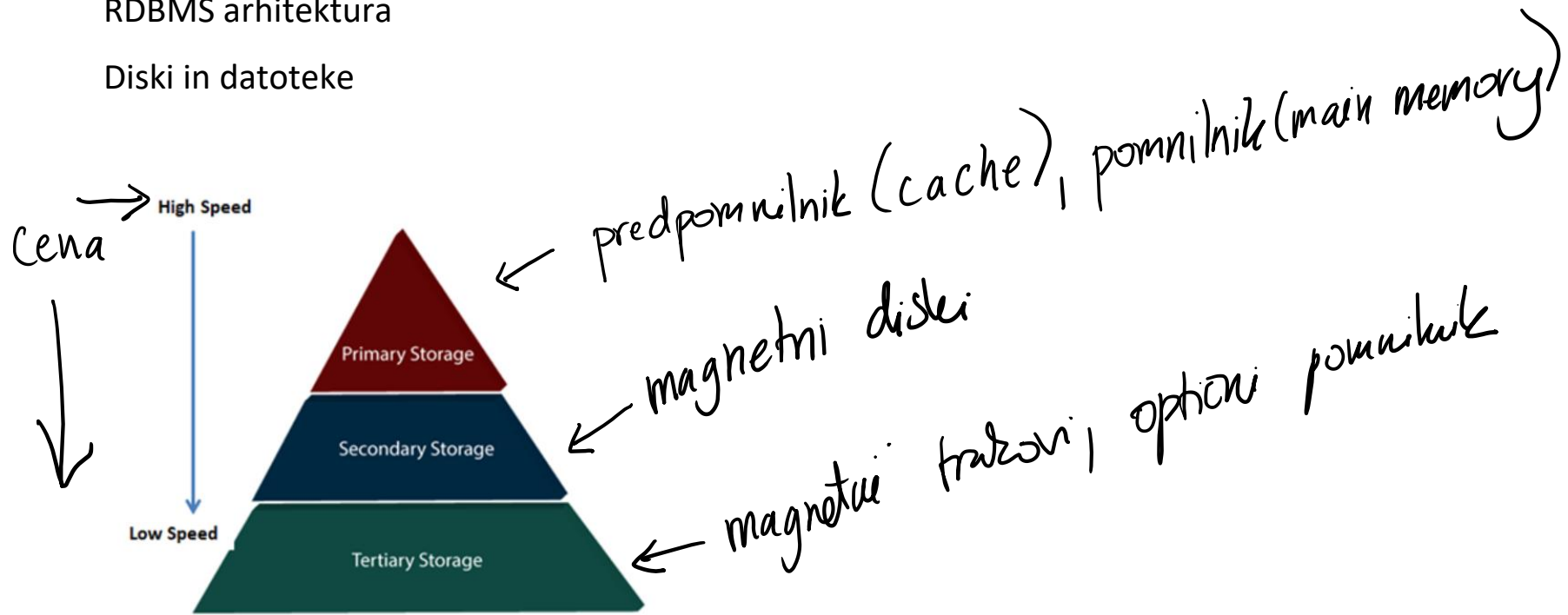


(vir slike: <https://www.javatpoint.com/storage-system-in-dbms#:~:text=A%20database%20system%20provides%20an,for%20accessing%20and%20storing%20data.>)

RDBMS arhitektura

Diski in datoteke



Datoteke in indeksi

- Non-clustered imamo lahko več indeksov
- Clustered

DDL: sql

↓
CREATE:

Sintaksa:

→ **CREATE DATABASE** *databasename*;

ime baze

Pred ustvarjanjem baze podatkov se prepričajte, da imate skrbniške pravice.

DROP:

Sintaksa:

DROP DATABASE *databasename*;

katera

Stavek DROP DATABASE se uporablja za izbris obstoječe baze podatkov SQL.

Primer:

DROP DATABASE testDB;

arhiviraj

↓
BACKUP DATABASE

Sintaksa:

BACKUP DATABASE *databasename*
TO DISK = '*filepath*';

ime baze

!c ...

Stavek `BACKUP DATABASE` se v SQL Serverju uporablja za ustvarjanje popolne varnostne kopije obstoječe baze podatkov SQL.

Vedno izdelajte varnostno kopijo podatkovne baze drugje, kot je dejanska zbirka podatkov. Potem, če pride do zrušitve diska, ne boste izgubili varnostne kopije skupaj z bazo podatkov.


Primer:

```
BACKUP DATABASE testDB  
TO DISK = 'D:\backups\testDB.bak';
```

CREATE TABLE: 

Sintaksa:

```
CREATE TABLE table_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    ....  
);
```



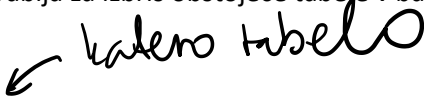
Stavek `CREATE TABLE` se uporablja za ustvarjanje nove tabele v bazi podatkov.

DROP TABLE:

Stavek `DROP TABLE` se uporablja za izbris obstoječe tabele v bazi podatkov.

Sintaksa:

```
DROP TABLE table_name;
```



!!!!!!!!!!!!!! Bodite previdni, preden izbrišete tabelo. Če izbrišete tabelo, boste izgubili vse informacije, shranjene v tabeli!

ALTER TABLE:

- Uporablja za dodajanje, brisanje ali spreminjanje stolpcev v obstoječi tabeli.
- uporablja tudi za dodajanje in opuščanje različnih omejitev v obstoječi tabeli.

Sintaksa za dodajanje stolpca v tabelo:

```
ALTER TABLE table_name  
ADD column_name datatype;
```

ime
Integer (cela številka)
stolpec

NOT NULL:

Privzeto lahko stolpec vsebuje vrednosti NULL.

Omejitev NOT NULL vsiljuje, da stolpec NE sprejema vrednosti NULL.

To uveljavi, da polje vedno vsebuje vrednost, kar pomeni, da ne morete vstaviti novega zapisa ali posodobiti zapisa, ne da bi temu polju dodali vrednost.

Primer:

```
CREATE TABLE Persons (  
  ID int NOT NULL,  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255) NOT NULL,  
  Age int  
);
```

stolpec

TUJI KLJUČ:

TUJI KLJUČ je polje (ali zbirka polj) v eni tabeli, ki se nanaša na PRIMARNI KLJUČ v drugi tabeli.

Tabela s tujim ključem se imenuje podrejena tabela, tabela s primarnim ključem pa referenčna ali nadrejena tabela.

Primer:

```
CREATE TABLE Orders (  
  → OrderID int NOT NULL,  
  → OrderNumber int NOT NULL,  
  → PersonID int,  
  PRIMARY KEY (OrderID),  
  FOREIGN KEY (PersonID) REFERENCES Persons(PersonID)  
);
```

brez vrednosti

aj

tabela

ljaka

SQL CREATE INDEX Statement

Stavek CREATE INDEX se uporablja za ustvarjanje indeksov v tabelah.

Indeksi se uporabljajo za hitrejše pridobivanje podatkov iz baze podatkov kot sicer. Uporabniki ne morejo videti indeksov, uporabljajo se le za pospešitev iskanj/poizvedb.

Sintaksa:

```
CREATE INDEX index_name  
ON table_name (column1, column2, ...);
```

R

atrib

i

Primer:

```
CREATE INDEX idx_lastname  
ON Persons (LastName);
```

atribut

i

INSERT INTO:

R Mesta, Post.št., Država

```
1. → INSERT INTO table_name (column1, column2, column3, ...)  
VALUES (value1, value2, value3, ...);
```

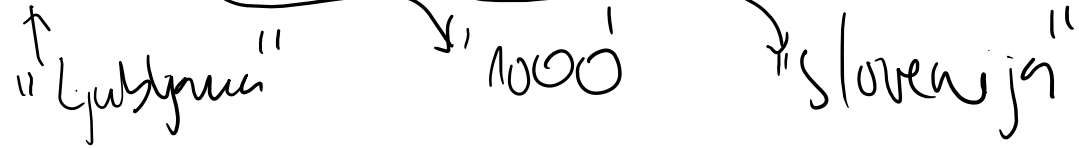


Primer:

```
→ INSERT INTO Customers (CustomerName, ContactName, Address, City, PostalCode, Country)  
VALUES ('Cardinal', 'Tom B. Erichsen', 'Skagen 21', 'Stavanger', '4006', 'Norway');
```



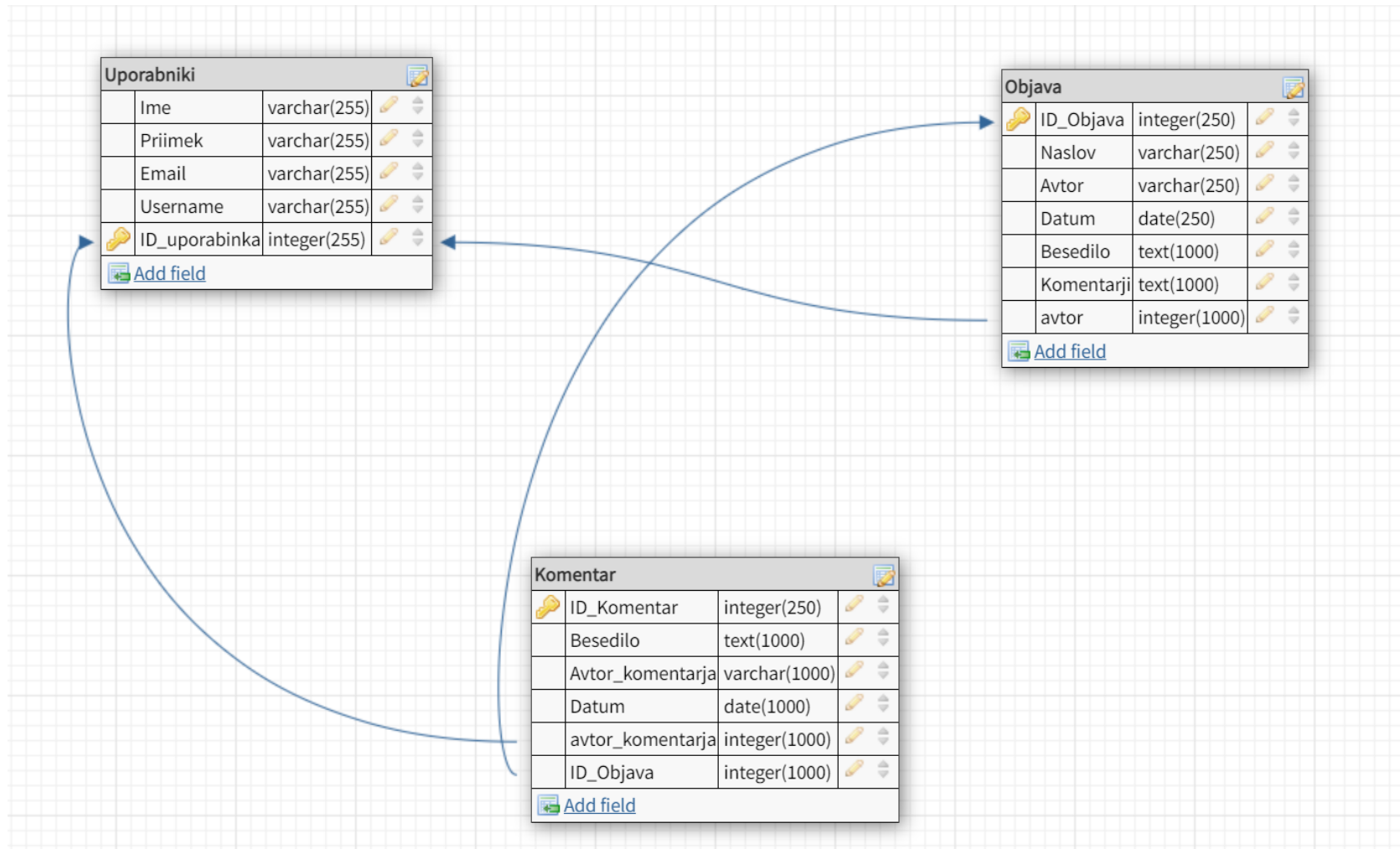
```
2. → INSERT INTO table_name  
VALUES (value1, value2, value3, ...);
```



R	Mesta	Post.št.	Država
	A	B	C

NALOGE:

1. Ustvari ER diagram in shemo za spletno stran. Spletna stran je blog, kjer registrirani uporabniki lahko objavljajo svoje bloge. Blog lahko registrirani uporabniki tudi komentirajo.



2. Ustvari ER diagram in shemo za univerzo

ENTITETE:

- UNIVERZA (Naslov, ID_Univerza, Ime, smeri studija)
- ŠTUDENT (ID-student, Prebivališče, smer, telefonska)
- PROFESORJI (ID-prof, Ime, kaj-ucijo, ocena)

