

## 2. Baze podataka

### 2.1. Osnovni pojmovi vezani uz baze podataka

Baza podataka je skup međusobno povezanih podataka pohranjenih u vanjskoj memoriji računala. Podaci su istovremeno dostupni raznim korisnicima i programima. Za ubacivanje, brisanje, promjenu i čitanje podataka svi se obavezno služe zajedničkim i kontroliranim alatima. Korisnici i programi ne moraju u detalje poznavati strukture upotrebljene za pohranjivanje podataka. Sustav za upravljanje bazom podataka - DBMS (Data Sustav Management System) je software koji je u stanju podržavati razne neovisne baze. Podaci u bazi su logički organizirani u skladu s nekim modelom podataka. Model čini osnovu za koncipiranje, definiranje i implementiranje baze.

Današnji DBMS-i podržavaju neki od modela:

- relacijski model - zasnovan je na matematičkom pojmu relacije i zahtjeva da su podaci i veze između njih prikazani "dvodimenzionalnim" tablicama;
- mrežni model - baza je predočena usmjerenim grafom, a čvorovi u tom grafu su podaci (tipovi zapisa), a lukovi definiraju veze između podataka;
- hijerarhijski model - baza je predočena stablima (hijerarhijama) gdje su čvorovi u tim stablima podaci (tipovi zapisa), a hijerarhijski odnos "nadređeni-podređeni" izražava veze.

Komunikacija između programa, odnosno korisnika i DBMS-a odvija se pomoću posebnih jezika. Ti se jezici tradicionalno dijele na tri kategorije:

- Jezik za opis podataka (DDL - Data Description Language) - služi projektantu baze podataka ili administratoru u svrhu zapisivanje sheme ili pogleda;
- Jezik za manipuliranje podacima (DML - Data Manipulation Language) - služi programeru za uspostavljanje veza između aplikacijskih programa. Naredbe DML-a omogućuju manevriranje po bazi, te jednostavne operacije kao što su upis, promjena, brisanje ili čitanje podataka. U nekim DBMS-ima je DML

zapravo biblioteka potprograma, a drugdje se zaista radi o posebnom jeziku. Programer tada piše program u kojem su izmješane naredbe dvaju jezika;

- Jezik za postavljanje upita (QL - Query Language) - ovaj jezik služi korisniku za interaktivno pretraživanje baze. To je jezik visokog nivoa koji podsjeća na govorni (engleski) jezik. Naredbe su neproceduralne, tj. specificiraju rezultat kojeg želimo dobiti, ali ne i algoritam.

Kod relacijskih baza su sva tri jezika objedinjena u jedan sveobuhvatni (npr. SQL).

## 2.2. Relacijsko modeliranje podataka

### 2.2.1. Modeliranje entiteta i veza

Modeliranje entiteta i veza (Entity-Relationship Modelling) je pomoćna faza za oblikovanje jedne manje precizne (konceptualne) sheme koja predstavlja apstrakciju realnog svijeta. Ta tzv. ER-shema se dalje više-manje pretvara u relacijsku. Modeliranje entiteta i veza zahtjeva da se svijet promatra preko tri kategorije:

- entiteti: objekti, događaji, bića, koji su nam od interesa;
- atributi: svojstva entiteta, koja su nam od interesa;
- veze: odnosi među entitetima koji su nam od interesa.

Entitet je nešto o čemu želimo spremati podatke. Entitet može biti neki objekt ili biće (kuća, student, auto), odnosno događaj ili pojava (nogometna utakmica, praznik, servisiranje auta). Entitet je opisan atributima (npr. auto; atribut auta - GOLF, no sam model GOLF ima attribute - broj vrata, snaga...). Ime entiteta s pripadnim atributima određuje tip entiteta. Kandidat za ključ je atribut ili skup atributa čije vrijednosti jednoznačno određuju entitet zadanog tipa. Ako jedan tip entiteta ima više kandidata za ključ, tada biramo jednog od njih i proglašavamo ga primarnim ključem (npr. za tip entiteta STUDENT primarni ključ bi mogao biti BROJ INDEKSA).

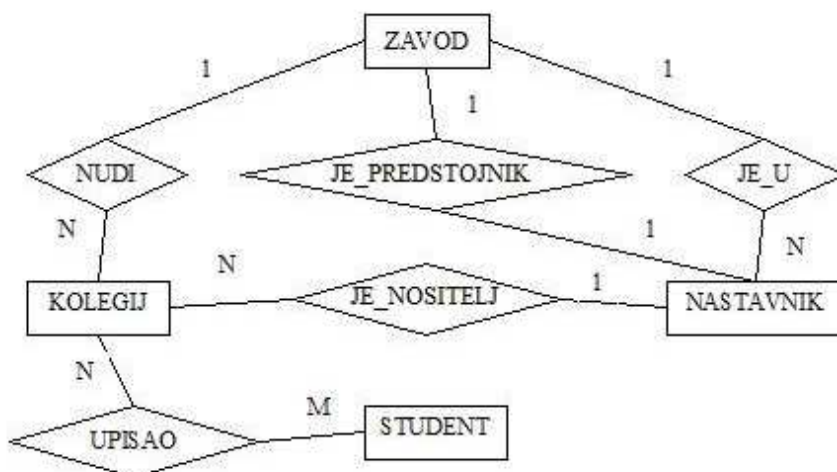
Veze se uspostavljaju između dva ili više tipova entiteta (npr. veza IGRA između tipova IGRAČ i TIM). Zapravo je riječ o imenovanoj binarnoj ili  $k$ -narnoj relaciji između primjeraka entiteta zadanih tipova. Za sad se ograničavamo na dva tipa entiteta. Funkcionalnost veze može biti jedan-naprema-jedan ( $1 : 1$ ), jedan-naprema-mnogo ( $1 : N$ ), mnogo-naprema-mnogo ( $N : M$ ).

$1 : 1$  veza JE\_PROEDSTOJNIK između entiteta NASTAVNIK i ZAVOD (na fakultetu),  $1 : N$  veza JE\_NOSITELJ između tipova entiteta NASTAVNIK i KOLEGIJ,

$N : M$  veza UPISAO između tipova entiteta STUDENT i KOLEGIJ.) Veza može imati svoje atribute, koje ne možemo pripisati ni jednom od tipova entiteta (npr. veza UPISAO može imati atribut DATUM\_UPISA). Ako svaki primjerak entiteta nekog tipa mora sudjelovati u zadanoj vezi, kažemo da tip entiteta ima obvezno članstvo u toj vezi.

ER-shema prikazuje se pomoću dijagrama u kojem su pravokutnici tipovi entiteta, a rombovi veze. Imena tipova entiteta i veza, te funkcionalnost veza uneseni su u dijagram. Veze su povezane bridovima s odgovarajućim pravokutnicima. Posebno se još prilaže lista atributa za svaki entitet, odnosno vezu. U toj listi možemo specificirati obveznost članstva.

**Primjer 2.1.** *Baza podataka o fakultetu*



Slika 2.1.

Tipovi entiteta:

1. ZAVOD s atributima IME\_ZAVODA, ADRESA, ...
2. KOLEGIJ s atributima BR\_KOLEGIJA, NASLOV, SEMESTAR,...
3. STUDENT s atributima BR\_INDEKSA, IME\_STUDENTA, ADRESA, SPOL,...
4. NASTAVNIK s atributima IME\_NASTAVNIKA, BR\_SOBE,...

Veze:

1. JE\_PREDSTOJNIK bez atributa. ZAVOD ima obvezno članstvo
2. JE\_U bez atributa. NASTAVNIK ima obvezno članstvo
3. NUDI bez atributa. KOLEGIJ ima obvezno članstvo
4. UPISAO s atributom DATUM\_UPISA
5. JE\_NOSITELJ bez atributa. KOLEGIJ ima obvezno članstvo.

### 2.2.2. Relacijski model

Relacijski model zahtjeva da se baze sastoje od skupa pravokutnih tablica - tzv. relacija. Svaka relacija ima svoje ime po kojem ih razlikujemo od ostalih u istoj bazi. Jedan stupac relacije sadrži vrijednosti jednog atributa za neki entitet ili vezu. Zato stupac poistovjećujemo s atributom i obrnuto. Svaki stupac-atribut ima svoje ime po kojem ga razlikujemo od ostalih u jednoj relaciji. Vrijednosti jednog atributa su podaci istog tipa, tj. definiran je skup dozvoljenih vrijednosti za atribut - domena atributa. Vrijednost atributa mora biti jednostruka i jednostavna, tj. ne može se rastaviti na dijelove koji bi se mogli promatrati kao posebni podaci u relaciji. Pod nekim uvjetima toleriramo situaciju kad vrijednost atributa nedostaje, tj. nije upisana. Jedan redak relacije predstavlja jedan primjerak entiteta ili bilježi vezu između dva ili više primjeraka. Redak nazivamo *n*-toraka (tuple). U jednoj relaciji ne smiju postojati dvije jednake *n*-torke. Broj atributa zove se stupanj relacije, a broj *n*-torke kardinalnost relacije. Relacija ne propisuje redoslijed svojih *n*-torke i atributa. Permutiranjem redaka i stupaca dobit ćemo drugačiji zapis iste relacije.

**Primjer 2.2.** *Relacija GURMAN s atributima IME, RESTORAN, HRANA*

*GURMAN*

<i>IME</i>	<i>RESTORAN</i>	<i>HRANA</i>
<i>Ivan</i>	<i>Boban</i>	<i>morski plodovi</i>
<i>Marko</i>	<i>Nokturno</i>	<i>tjestenina</i>
<i>Kruno</i>	<i>Stara Sava</i>	<i>pizza</i>
<i>Igor</i>	<i>Baschiera</i>	<i>pizza</i>
<i>Davor</i>	<i>The Movie Pub</i>	<i>tjestenina</i>

Pretvaranje ER-sheme u relacijsku

1. Pretvorba tipova entiteta - svaki tip entiteta prikazuje se jednom relacijom. Atributi tipa entiteta postaju atributi relacije. Jedan primjerak entiteta prikazan je jednom *n*-torkom. Primarni ključ entiteta postaje primarni ključ relacije.

2. Pretvorba binarnih veza - ako tip entiteta  $E_2$  ima obvezno članstvo u ( $N : 1$ ) vezi s tipom  $E_1$ , tada relacija od  $E_2$  treba uključiti primarni atribut od  $E_1$ . Npr. ako svaki kolegij mora biti ponuđen od nekog Zavoda, onda se veza NUDI svodi na to u relaciji KOLEGIJ ubacimo ključ relacije ZAVOD; KOLEGIJ (BR.KOLEGIJA, IME\_ZAVODA, NASLOV, SEMESTAR).

**Primjer 2.3.** Koristeći relacijski model modelirati bazu podataka o kruhu koji se proizvodi u jednoj pekarnici u Zagrebu.

<i>ime kruha</i>	<i>tip</i>	<i>brašno glavna komp.</i>	<i>brašno spor. komp.</i>	<i>masa (kg)</i>
<i>Bijeli Dubravica</i>	<i>bijeli kruh</i>	<i>tip-550</i>	—	0.70
<i>Bijeli specijal</i>	<i>bijeli kruh</i>	<i>tip-550</i>	—	0.70
<i>Sendvič bijeli</i>	<i>bijeli kruh</i>	<i>tip-550</i>	—	0.70
<i>Sendvič polubjeli</i>	<i>polubjeli kruh</i>	<i>tip-850</i>	—	0.70
<i>Lički</i>	<i>pšenični miješani kruh</i>	<i>tip-850</i>	<i>tip-1250</i>	1.30
<i>Kukuruzni miješani</i>	<i>kukuruzni miješani kruh</i>	<i>tip-850</i>	—	0.60
<i>Seljački kukuruzni</i>	<i>kukuruzni miješani kruh</i>	<i>tip-850</i>	<i>tip-1250</i>	0.80
<i>Kukuruzni delikates</i>	<i>kukuruzni miješani kruh</i>	<i>tip-550</i>	—	0.50
<i>Raženi miješani</i>	<i>raženi miješani kruh</i>	<i>tip-850</i>	<i>tip-1250</i>	0.65
<i>Bavarski raženi</i>	<i>raženi kruh</i>	<i>tip-1250</i>	<i>tip-850</i>	0.70
<i>Alpski</i>	<i>raženi miješani kruh</i>	<i>tip-850</i>	<i>tip-1250</i>	0.50
<i>Sovital</i>	<i>kruh sa sjemenkama</i>	<i>tip-850</i>	<i>tip-1250</i>	0.50
<i>Zlatno polje</i>	<i>pšenični miješani kruh</i>	<i>tip-550</i>	—	0.60
<i>Kruh sa suncokretom</i>	<i>kruh sa sjemenkama</i>	<i>tip-850</i>	<i>tip-550</i>	0.50
<i>Bakin kruh</i>	<i>kruh sa sjemenkama</i>	<i>tip-850</i>	—	0.45
<i>Baguette rustik</i>	<i>bijeli kruh</i>	<i>tip-550</i>	—	0.25

Rješenje. Tipovi entiteta:

- KRUH (IME\_KRUHA, TIP)
- BRASNO (IME\_KRUHA, BRASNO\_GLAVNAKOMP, BRASNO\_SPORKOMP)
- PRODAJA (IME\_KRUHA, MASA)

Relacijski model:

KRUH

IME_KRUHA	TIP
Bijeli Dubravica	bijeli kruh
Bijeli specijal	bijeli kruh
Sendvič bijeli	bijeli kruh
Sendvič polubjeli	polubjeli kruh
Lički	pšenični miješani kruh
Kukuruzni miješani	kukuruzni miješani kruh
Seljački kukuruzni	kukuruzni miješani kruh
Kukuruzni delikates	kukuruzni miješani kruh
Raženi miješani	raženi miješani kruh
Bavarski raženi	raženi kruh
Alpski	raženi miješani kruh
Sovital	kruh sa sjemenkama
Zlatno polje	pšenični miješani kruh
Kruh sa suncokretom	kruh sa sjemenkama
Bakin kruh	kruh sa sjemenkama
Baguette rustik	bijeli kruh

BRASNO

IME_KRUHA	BRASNO_GLAVNAKOMP	BRASNO_SPORKOMP
Bijeli Dubravica	tip-550	
Bijeli specijal	tip-550	
Sendvič bijeli	tip-550	
Sendvič polubjeli	tip-850	
Lički	tip-850	tip-1250
Kukuruzni miješani	tip-850	
Seljački kukuruzni	tip-850	tip-1250
Kukuruzni delikates	tip-550	
Raženi miješani	tip-850	tip-1250
Bavarski raženi	tip-1250	tip-850
Alpski	tip-850	tip-1250
Sovital	tip-850	tip-1250
Zlatno polje	tip-550	
Kruh sa suncokretom	tip-850	tip-550
Bakin kruh	tip-850	
Baguette rustik	tip-550	

## PRODAJA

IME_KRUHA	MASA
Bijeli Dubravica	0.70
Bijeli specijal	0.70
Sendvič bijeli	0.70
Sendvič polubjeli	0.70
Lički	1.30
Kukuruzni miješani	0.60
Seljački kukuruzni	0.80
Kukuruzni delikates	0.50
Raženi miješani	0.65
Bavarski raženi	0.70
Alpski	0.50
Sovital	0.50
Zlatno polje	0.60
Kruh sa suncokretom	0.50
Bakin kruh	0.45
Baguette rustik	0.25

## 2.3. Jezici za relacijske baze podataka

### 2.3.1. Relacijska algebra

Relacijska algebra je teoretska matematička notacija i svodi se na izvrednjavanje algebarskih izraza građenih od relacija, te unarnih i binarnih operatora (čiji operandi su relacije, a rezultat je opet relacija). Svaki algebarski izraz predstavlja jedan upit ili pretraživanje baze podataka.

#### Skupovni operatori

Relacije su skupovi  $n$ -torki pa se primjenjuju uobičajene skupovne operacije. Neka  $R$  i  $S$  označavaju relacije. Tada je

$R$  union  $S$  ... skup  $n$ -torki koje su u  $R$  ili u  $S$  ili u obje,

$R$  intersect  $S$  ... skup  $n$ -torki koje su i u  $R$  i u  $S$ ,

$R$  minus  $S$  ... skup  $n$ -torki koje su u  $R$ , a nisu u  $S$ .

Da bi operatore mogli primjenjivati, relacije moraju biti kompatibilne, tj. moraju imati isti stupanj i iste attribute (ista imena i tipove).

#### Selekcija

Unaran operator koji izvlači iz relacije one  $n$ -torke koje zadovoljavaju zadani

Booleanski uvjet. Označavat ćemo selekciju na relaciju  $R$  u skladu s Booleanskim uvjetom  $B$  s:  $R$  where  $B$ .

Uvjet  $B$  je formula koja se sastoji od:

1. operanada koji su ili konstante ili atributi
2. operatora za uspoređivanje  $=, <, >, \leq, \geq, \neq$
3. logičkih operatora and, or, not

#### Projekcija

Unarni operator koji iz relacije "izvlači" zadane attribute. Pritom se u rezultirajućoj relaciji eliminiraju eventualne  $n$ -torke duplikati. Projekciju relacije  $R$  na njene attribute  $A_1, A_2, \dots, A_n$  označavamo s:  $R[A_1, A_2, \dots, A_n]$ . Smatramo da projekcija ima viši prioritet od ostalih operacija.

#### Kartezijev produkt

Neka su  $R$  i  $S$  relacije stupnja  $n_1$ , odnosno  $n_2$ . Tada algebarski izraz  $R$  times  $S$  daje Kartezijev produkt relacija  $R$  i  $S$ , dakle skup svih  $(n_1 + n_2)$ -torki čijih prvih  $n_1$  komponenti čine  $n_1$ -torku u  $R$ , a zadnjih  $n_2$  komponenti čine  $n_2$ -torku u  $S$ .

#### Prirodni spoj

$R$  join  $S$  se sastoji od svih  $n$ -torki dobivenih spajanjem jedne  $n$ -torke iz  $R$  s jednom  $n$ -torkom iz  $S$  koja ima iste vrijednosti zajedničkih atributa.

#### Dijeljenje

$R$  divideby  $S$  je skup svih  $(n - m)$ -torki  $\langle x \rangle$  t.d. se  $n$ -toraka  $\langle xy \rangle$  pojavljuje u  $R$  za sve  $m$ -torke  $\langle y \rangle$  u  $S$ .

### 2.3.2. Relacijski račun

Riječ je o matematičkoj notaciji zasnovanoj na predikatnom računu. Upit se izražava t.d. zadamo predikat kojeg  $n$ -torke moraju zadovoljavati. Postoje dvije vrste relacijskog računa:

- račun orijentiran na  $n$ -torku
- račun orijentiran na domenu

#### Račun orijentiran na $n$ -torke:

Varijable su  $n$ -torke i one poprimaju vrijednosti iz imenovane relacije. Ako je  $t$  varijabla koja prolazi relacijom  $R$ , tada  $t.A$  označava vrijednost atributa  $A$  u toj  $n$ -torki.



### Račun orijentiran na domene:

Varijable prolaze domenama, a ne relacijama. Također imamo uvjete članstva oblika:  $R(A : v_1, B : v_2, C : v_3, \dots)$  gdje su  $A, B, C, \dots$  atributi relacije  $R$ , a  $v_1, v_2, v_3, \dots$  su ili varijable ili konstante.

### 2.3.3. Jezik SQL

Structured Query Language jezik za postavljanje upita. Zasnovan je na relacijskom računu (orijentiranom na  $n$ -torke) s tim da je matematička notacija zamijenjena ključnim riječima nalik na engleski govorni jezik.

#### *Stvaranje i ažuriranje tablica*

Tipovi podataka:

- **VARCHAR**( $l$ )... nizovi podataka (character stringovi) sa max duljinom  $l$
- **DATE**... datumi u formatu dd-Mmm-gg
- **REAL**( $l, d$ )... brojevi od najviše  $l$  znamenki s točno  $d$  decimala
- **DECIMAL**( $l, d$ )... s eventualnim predznakom
- **INTEGER**... cijeli brojevi (32-bitni prikaz)
- **SMALLINT**... cijeli brojevi (16-bitni prikaz)
- **FLOAT**... realni brojevi s "plivajućom" točkom

Uvodi se specijalna "konstanta" NULL koja može biti pridružena atributu bilo kojeg tipa i koja označava praznu (nepostojeću) vrijednost.

Terminologija SQL-a:

relacija		table
$n$ -torka	→	row
atribut		column

**Primjer 2.4.** *Kreiraj bazu podataka o kruhu.*

*Rješenje.*

```
CREATE TABLE KRUH
(IME_KRUHA          VARCHAR(50)  NOT NULL,
 TIP                VARCHAR(50)  NOT NULL,
 PRIMARY KEY       (IME_KRUHA));

CREATE TABLE BRASNO
(IME_KRUHA          VARCHAR(50)  NOT NULL,
 BRASNO_GLAVNAKOMP VARCHAR(20)  NOT NULL,
 BRASNO_SPORKOMP   VARCHAR(20),
 PRIMARY KEY       (IME_KRUHA));

CREATE TABLE PRODAJA
(IME_KRUHA          VARCHAR(50)  NOT NULL,
 MASA               REAL(10,2)  NOT NULL,
 PRIMARY KEY       (IME_KRUHA));
```

Definicija relacije može se naknadno poništiti naredbom DROP:

```
DROP TABLE KRUH;
```

Promjena definicije relacije vrši se s naredbom ALTER.

Upacivanje podataka u tablicu vrši se s naredbom INSERT:

```
INSERT INTO KRUH VALUES ('Graham kruh', 'pšenični miješani kruh');
```

Promjena podataka u relaciji vrši se s naredbom UPDATE:

```
UPDATE BRASNO
SET BRASNO_GLAVNAKOMP = 'tip-850'
WHERE BRASNO_SPORKOMP = 'tip-1250';
```

Brisanje podataka iz relacije vrši se s naredbom DELETE:

```
DELETE FROM PRODAJA WHERE MASA = 0.50;
```

### ***Jednostavni upiti***

Upit se postavlja naredbom SELECT.

Upit 1: "Ispiši podatke o svim kruhovima (ime i tip)"

```
SELECT * FROM KRUH;
```

Pojavljuje se ispis:

IME_KRUHA	TIP
Bijeli Dubravica	bijeli kruh
Bijeli specijal	bijeli kruh
Sendvič bijeli	bijeli kruh
Sendvič polubijeli	polubijeli kruh
Lički	pšenični miješani kruh
Kukuruzni miješani	kukuruzni miješani kruh
Seljački kukuruzni	kukuruzni miješani kruh
Kukuruzni delikates	kukuruzni miješani kruh
Raženi miješani	raženi miješani kruh
Bavarski raženi	raženi kruh
Alpski	raženi miješani kruh
Sovital	kruh sa sjemenkama
Zlatno polje	pšenični miješani kruh
Kruh sa suncokretom	kruh sa sjemenkama
Bakin kruh	kruh sa sjemenkama
Baguette rustik	bijeli kruh

Upit 2: "Ispiši sve tipove brašna koji su glavna komponenta"

```
SELECT DISTINCT BRASNO_GLAVNAKOMP FROM BRASNO;
```

Pojavljuje se ispis:

BRASNO_GLAVNAKOMP
tip-550
tip-850
tip-1250

Upit 3: "Ispiši sva imena kruhova čija je masa 0.70kg"

```
SELECT IME_KRUHA FROM PRODAJA WHERE MASA=0.70;
```

Pojavljuje se ispis:

IME_KRUHA
Bijeli Dubravica
Bijeli specijal
Sendvič bijeli
Sendvič polubijeli
Bavarski raženi

### ***Složeni upiti***

Upit 1: "Koji kruh ima masu veću od 0.50kg, a za sporednu komponentu brašna ima tip 1250? Ispiši ime kruha, masu i glavnu komponentu brašna.

```

SELECT  P.IME_KRUHA, P.MASA, B.BRASNO_GLAVNAKOMP
FROM    PRODAJA P,
        BRASNO B
WHERE   P.IME_KRUHA=B.IME_KRUHA
AND     P.MASA > 0.50
AND     B.BRASNO_SPORKOMP = 'tip-1250';

```

Pojavljuje se ispis:

IME_KRUHA	MASA	B.BRASNO_GLAVNAKOMP
Lički	1.30	tip-850
Seljački kukuruzni	0.80	tip-850
Raženi miješani	0.65	tip-850

Upit 2: "Ispišite sve kruhove koji su tipa 'kukuruzni miješani', a nemaju sporednu komponentu brašna!"

```

SELECT  K.IME_KRUHA
FROM    KRUH K,
        BRASNO B
WHERE   K.IME_KRUHA=B.IME_KRUHA
AND     K.TIP = 'kukuruzni miješani kruh'
AND     B.BRASNO_SPORKOMP = "";

```

Pojavljuje se ispis:

IME_KRUHA
Kukuruzni miješani
Kukuruzni delikates