

Podatkovne baze (ali Zbirke podatkov)

doc.dr. Dimitar Hristovski, dipl.ing.rač.

Inštitut za biostatistiko in medicinsko
informatiko

Medicinske fakultete v Ljubljani

Motivacija

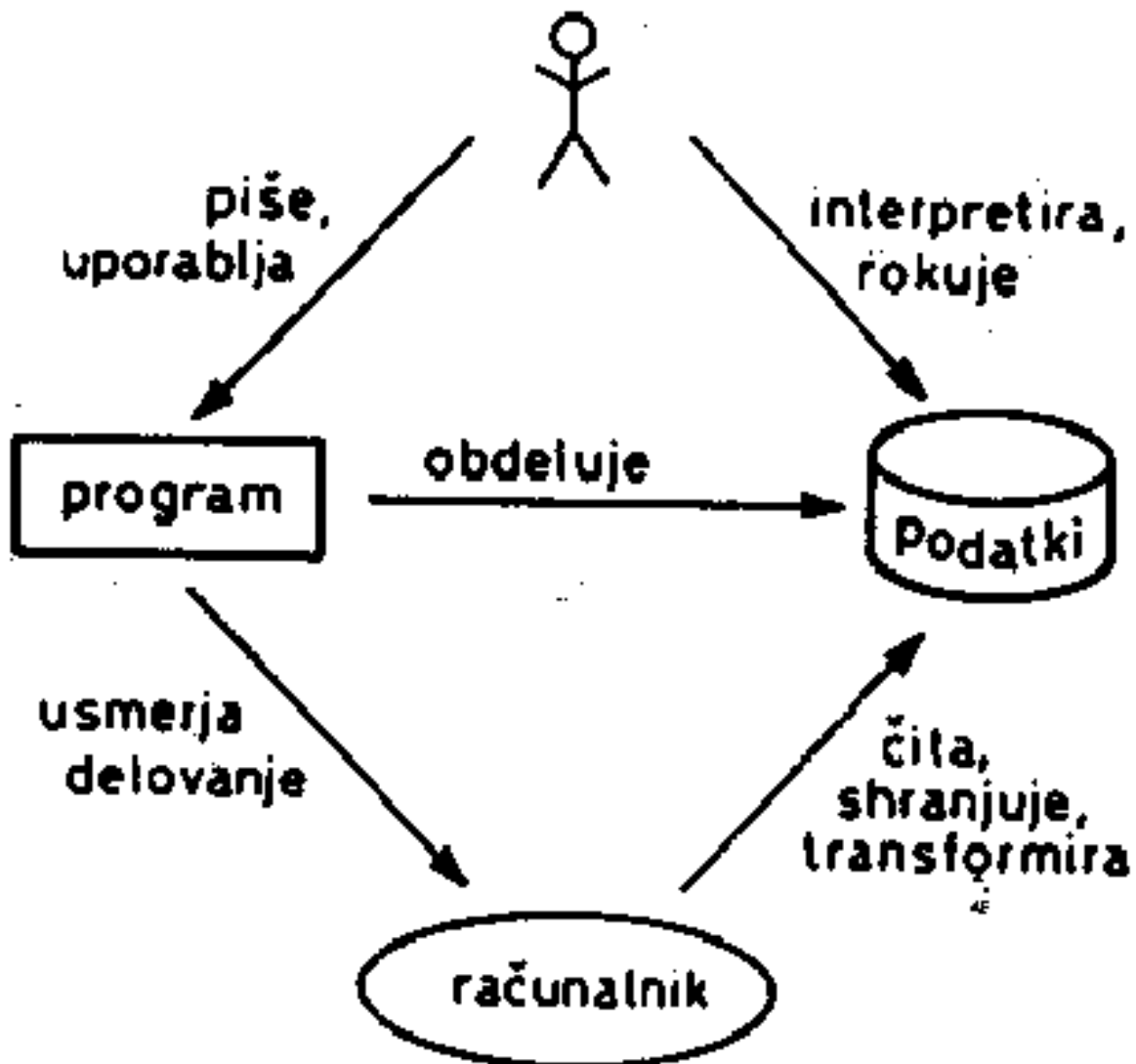
- ❖ Tehnološka osnova za številne biomedicinske aplikacije: elektronski medicinski zapis, zdravstveni informacijski sistemi, ...
- ❖ Osebni nivo (v času študija in pozneje): raziskovalne naloge, magisteriji, doktorati, ...
- ❖ Uporabne tudi drugje:
 - ❖ Banke
 - ❖ Zavarovalnice
 - ❖ Državne ustanove

Računalniško podprt podatkovni sistem

Računalniško podprt **podatkovni sistem** – sistem za obdelavo in rokovanje s podatki. Ima štiri komponente:

- ❖ Človek
- ❖ Program
- ❖ Podatki
- ❖ Računalnik

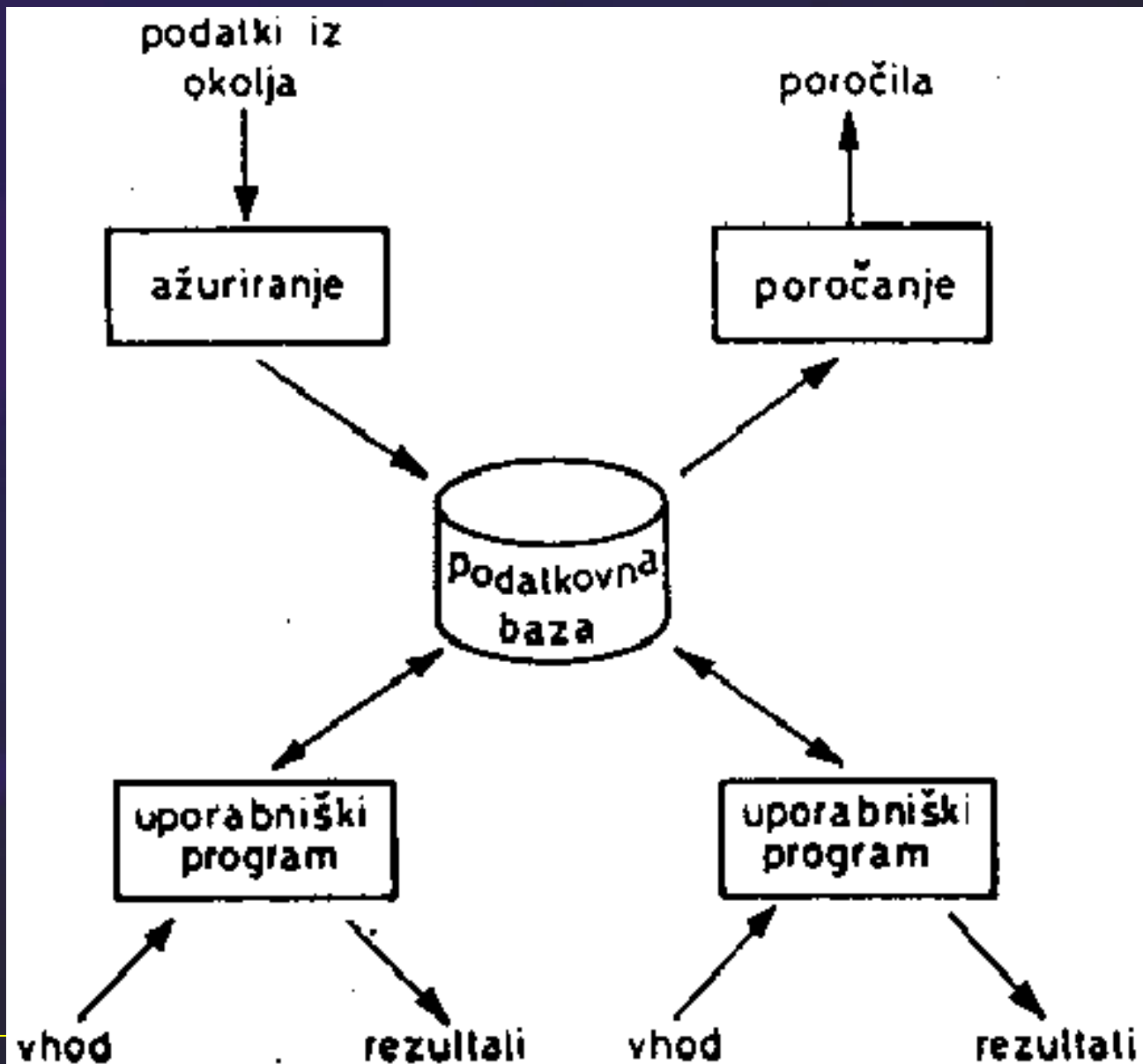
Podatkovni sistem



Podatkovna revolucija

- ❖ Prvo obdobje: središče pozornosti *računalnik*.
- ❖ Drugo obdobje: programska revolucija, središče pozornosti *programi*.
- ❖ Tretje obdobje: podatkovna revolucija, središče pozornosti *podatki*.

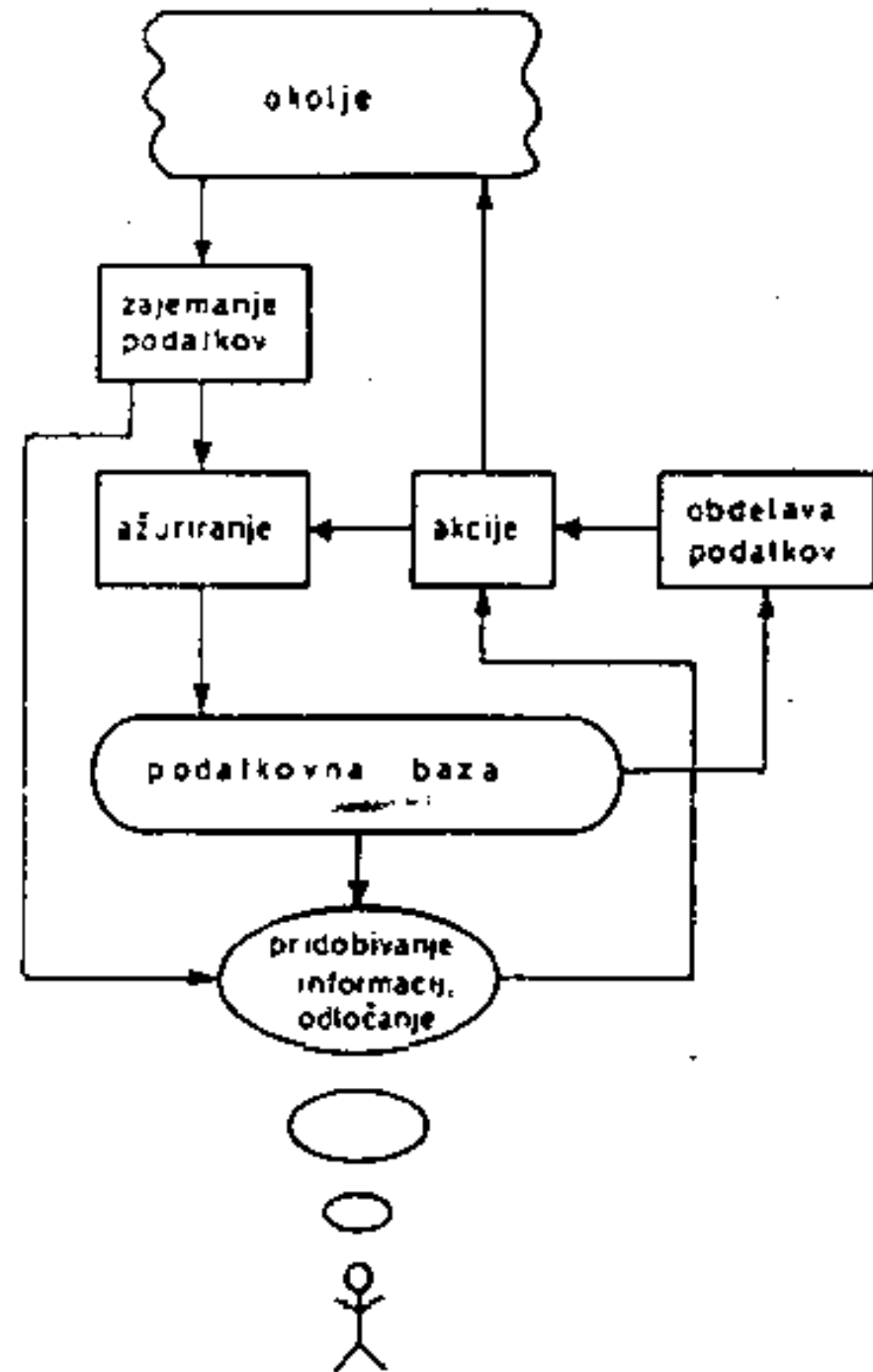
Podatki v središču pozornosti



Podatkovna baza – definicije

- ❖ Podatkovna baza je model okolja, ki služi kot osnova za sprejemanje odločitev in izvajanje akcij.
- ❖ Podatkovna baza je množica med seboj povezanih podatkov, ki so shranjeni v računalniškem sistemu, dostop do njih je centraliziran in omogočen s pomočjo sistema za upravljanje podatkovnih baz (SUPB).
- ❖ Podatkovna baza je računalniško podprta, večuporabniška, formalno definirana in centralno nadzorovana zbirka podatkov.

Podatkovna baza kot model okolja



Upravljanje podatkovne baze

Ker so podatki osnova, na kateri temelji celotno delovanje organizacije, je treba z njimi, enako kot z drugimi sredstvi organizacije, smiselno upravljati, kar zajema:

- ❖ Zagotavljanje razpoložljivosti podatkov
- ❖ Nadzor nad uporabo podatkov

Zagotavljanje razpoložljivosti podatkov

- ❖ Učinkovit dostop
- ❖ Vseh upravnikov
- ❖ Sočasno
- ❖ Do vseh vrst podatkov
- ❖ Ves čas

Nadzor nad uporabo podatkov

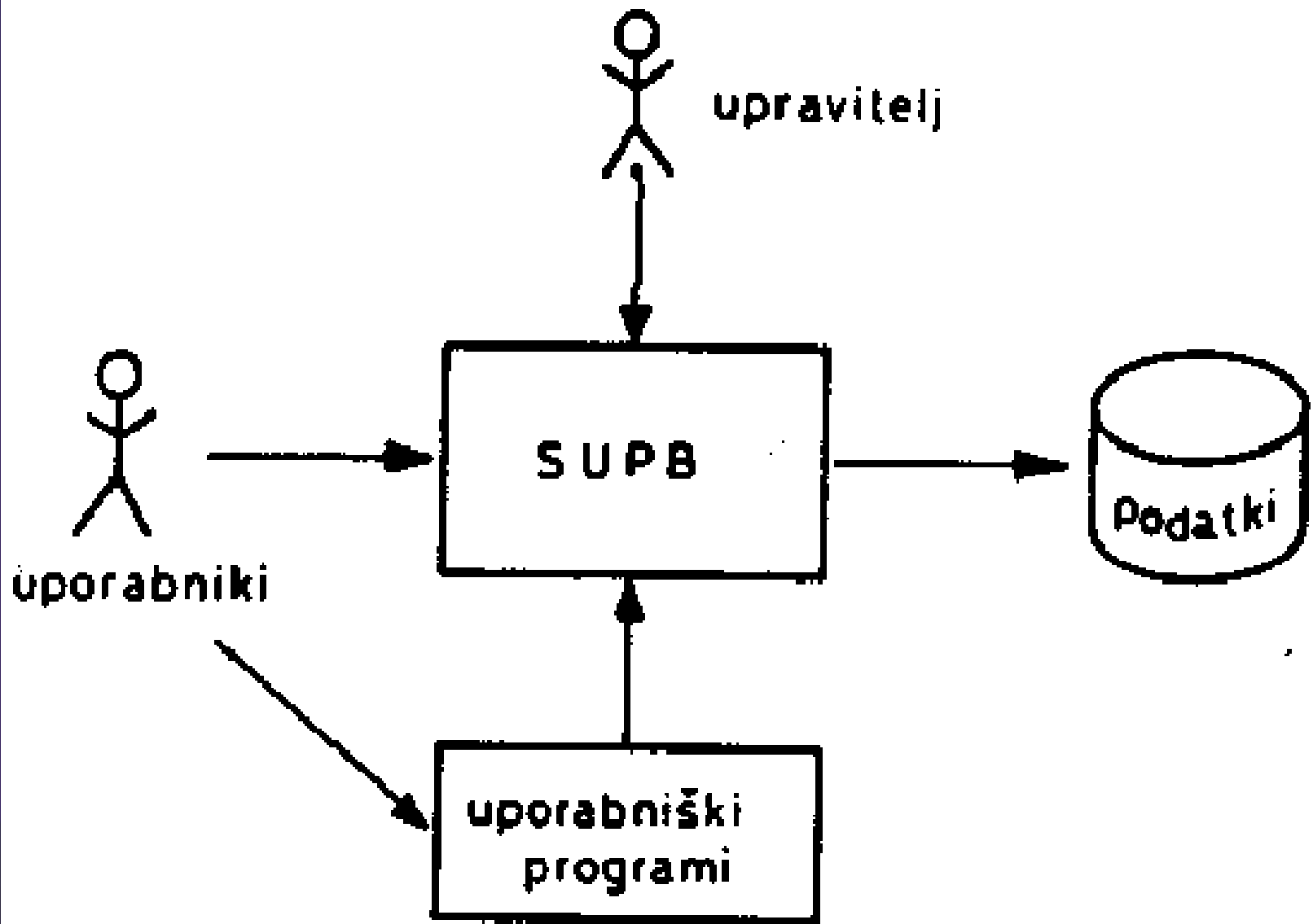
- ❖ Celovitost (integriteta) podatkov:
 - ❖ Obnavljanje podatkovne baze
 - ❖ Nadzor nad sočasnim dostopom do podatkov
 - ❖ Preverjanje vhodnih podatkov
- ❖ Uporaba podatkov v skladu z njihovim namenom:
 - ❖ Pravilno razumevanje podatkov
 - ❖ Dostop do dovoljenih podatkov
- ❖ Uporabnost podatkov tudi v prihodnje

Organizacija podatkovne baze

Podatkovno bazo v širšem smislu sestavljajo:

- ❖ Podatki
- ❖ Uporabniki in uporabniški programi
- ❖ Upravitelj podatkovne baze
- ❖ Sistem za upravljanje podatkovne baze

Struktura podatkovne baze

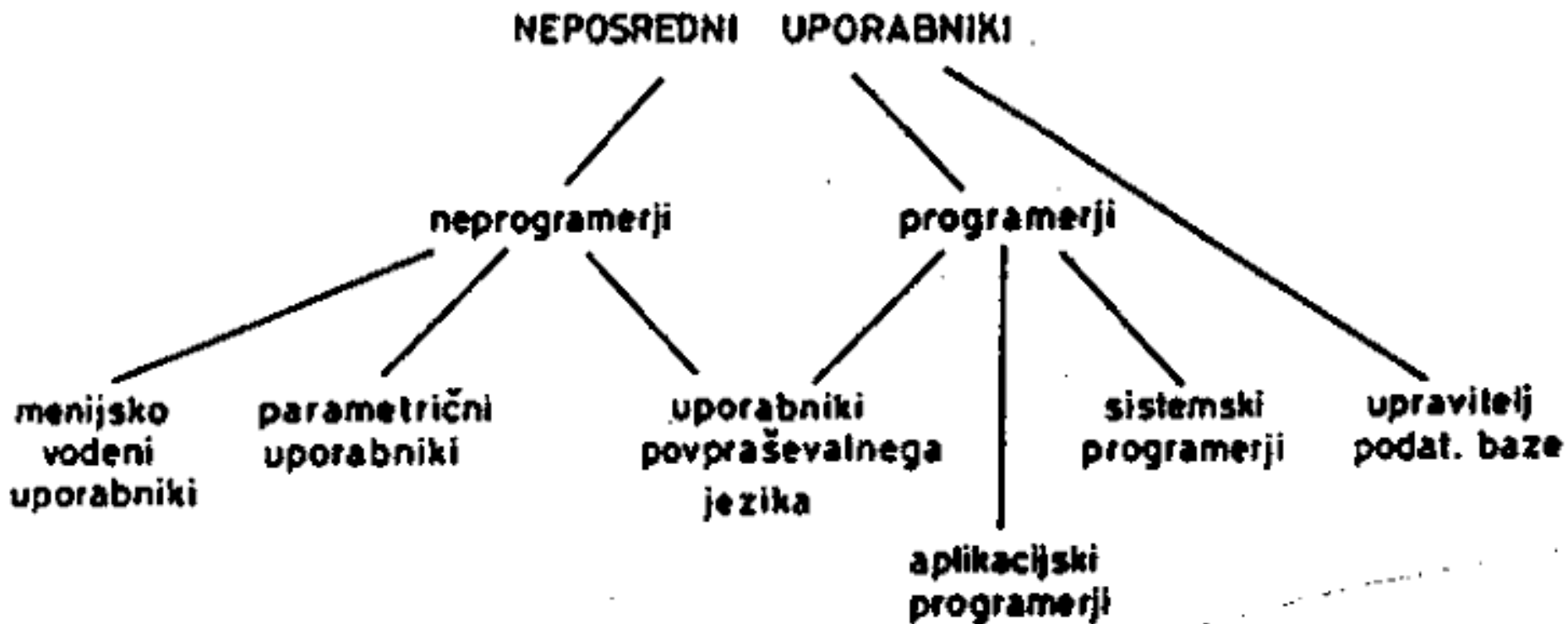


Podatkovna baza z vidika upravljanja

- ❖ Centralizirana podatkovna baza
- ❖ Porazdeljena podatkovna baza

Uporabniki podatkovne baze

- ❖ Neposredni uporabniki
- ❖ Posredni uporabniki



Podatkovni del podatkovne baze

- ❖ Fizična podatkovna baza (FPB) – vrednosti podatkovnih elementov
- ❖ Metapodatkovna baza (MPB) – opisi fizičnih podatkov:
 - ❖ Kako so shranjeni v zunanjem pomnilniku
 - ❖ Kaj pomenijo
 - ❖ Kako so dostopni uporabnikom

MPB – Meta podatkovna baza

Ima trinivojsko zgradbo:

- ❖ Zunanja shema
 - ❖ Uporabnikov pogled: model okolja posameznega uporabnika
- ❖ Konceptualna shema
 - ❖ Globalni model okolja
- ❖ Notranja shema
 - ❖ Zbirka logičnih zapisov

Podatkovna neodvisnost

Spremembe ene sheme ne smejo vplivati na drugo shemo.

- ❖ Fizična podatkovna neodvisnost
- ❖ Logična podatkovna neodvisnost

Podatkovni modeli

Posamezne vrste shem opišemo s pomočjo strukturnega in opisnega mehanizma - **podatkovnega modela**, imenovanega tudi *jezik za opis podatkov*.

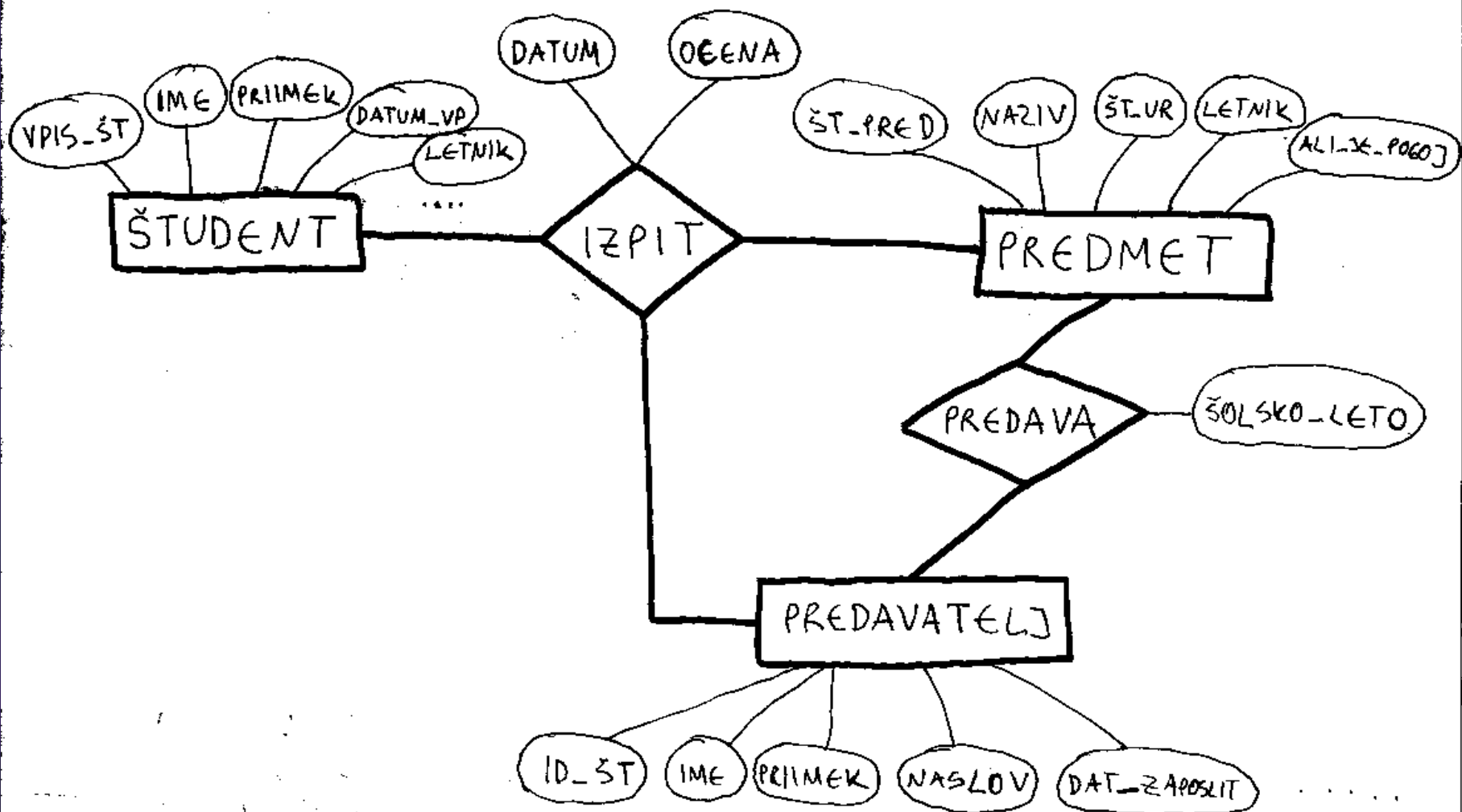
Vrste podatkovnih modelov

- ❖ Površinski:
 - ❖ Relacijski
 - ❖ Mrežni
 - ❖ Hierarhični
- ❖ Globinski:
 - ❖ Eniteta-razmerje
 - ❖ ...

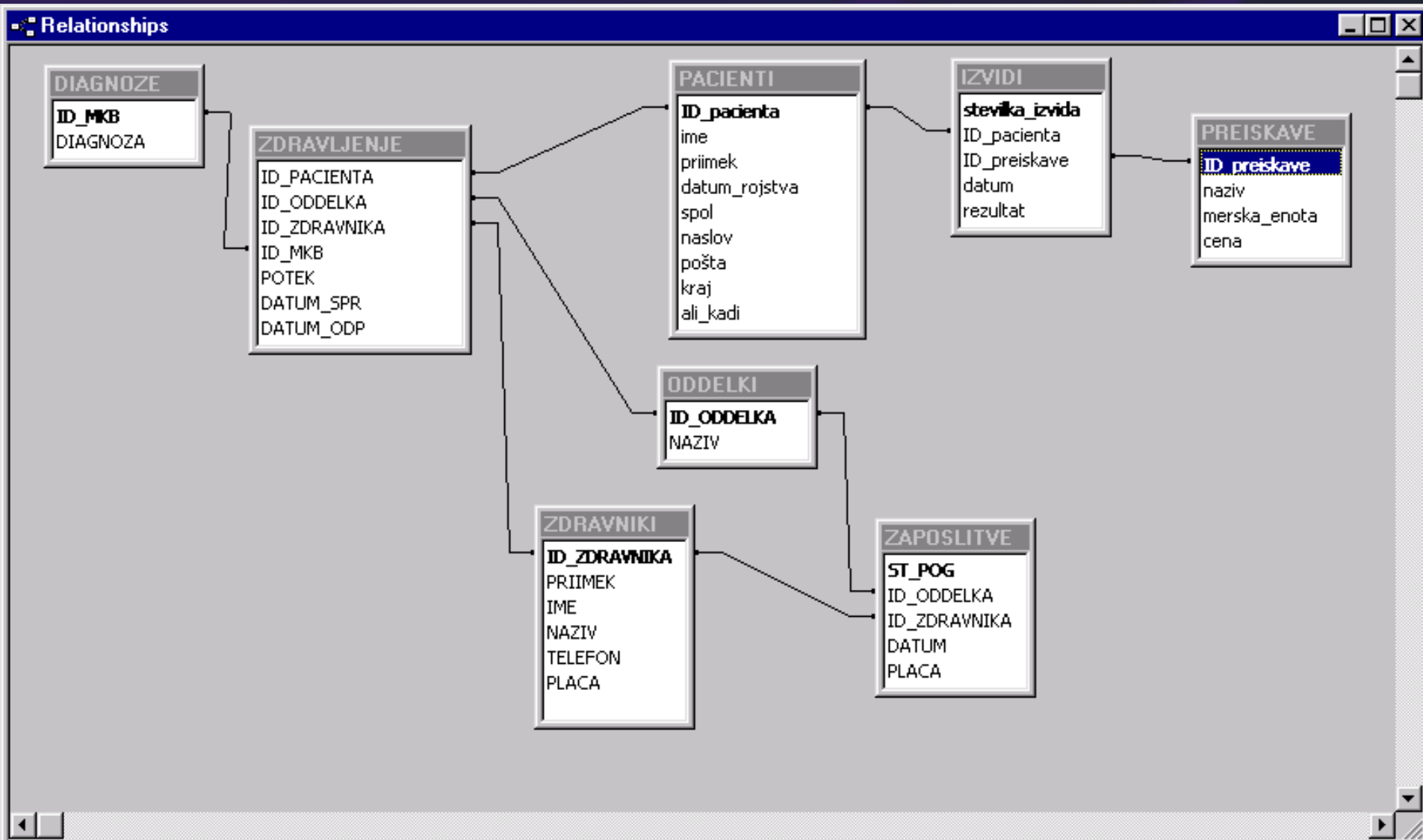
Entiteta razmerje

- ❖ **Entiteta**: katerikoli konkretni ali abstraktni objekt ali dogodek v organizaciji ali okolju, ki ga modeliramo (realni svet) (pr. Pacient, Preiskava, Študent, ...)
- ❖ **Atribut**: nekaj kar hočemo vedeti o kakšni entiteti (pr. Ime, Priimek, Spol, ...)
- ❖ **Vrednosti**: vrednosti atributov entitet (pr. Ime='Janez', Spol='M', ...)
- ❖ **Razmerje**: pomenska povezava med entitetami (pr. Izpit : razmerje med entitetama Študent in Predmet)

Študentska pisarna – diagram entiteta razmerje



Podatkovni model - Bolnica



Primer relacijske tabele - Pacienti

ID_pacienta	ime	priimek	datum_rojstva	spol	naslov	pošta	kraj	ali_kadi
1	JANEZ	NOVAK	13.03.1955	M	TROMOSTOVJE BŠ	1000	LJUBLJANA	<input checked="" type="checkbox"/>
2	ŠPELA	PLANINC	22.02.1944	F	LEPI POT 2	9000	MURSKA SOBOTA	<input type="checkbox"/>
3	MITJA	JEŽ	17.03.1999	M	ZALOŠKA 17	1000	LJUBLJANA	<input type="checkbox"/>
4	KAREL	FRANC	22.02.1966	M	PLAVŽ 1	4000	JESENICE	<input checked="" type="checkbox"/>
5	POLONA	GOSTIŠA	31.05.2023	F	ROŠKA 2	1000	LJUBLJANA	<input type="checkbox"/>
6	DUŠAN	MEKIŠ	24.05.1963	M	NA ČRETI 6	2000	MARIBOR	<input type="checkbox"/>
7	RENATA	VERHOVEC	24.11.1971	F	TRUBARJEVA 6	8210	TREBNJE	<input type="checkbox"/>
8	TOMISLAVA	MIRKOVIČ	27.03.1963	F	PADLIH B. 5	5220	TOLMIN	<input checked="" type="checkbox"/>
9	ASDFJKLK	SDFJKDF				0		<input type="checkbox"/>
*	(AutoNumber)					0		<input type="checkbox"/>

Record: 1 of 9

Struktura tabele Pacienti

The screenshot displays the Microsoft Access interface. At the top, a window titled 'PACIENTI : Table' shows a table structure with the following fields:

Field Name	Data Type	Description
ID_pacienta	AutoNumber	identifikacijska številka pacienta
ime	Text	
priimek	Text	
datum_rojstva	Date/Time	
spol	Text	
naslov	Text	
pošta	Number	
kraj	Text	
ali_kadi	Yes/No	

Below the table structure, the 'Field Properties' dialog box is open, showing the 'General' tab. The properties for the selected field are:

- Field Size: Long Integer
- New Values: Increment
- Format: (empty)
- Caption: (empty)
- Indexed: Yes (No Duplicates)

On the right side of the dialog box, there is a text box with the following message:

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.

Tabela - Preiskave

PREISKAVE : Table				
	ID_preiskave	naziv	merska_enota	cena
▶	1	EKG		500.00 SIT
	2	NATRIJ	mmol/l	20.00 SIT
	3	KALIJ	mmol/l	30.00 SIT
	4	KLORID	mmol/l	40.00 SIT
	5	TEŽA	kg	10.00 SIT
	6	RR	mmHg	100.00 SIT
	7	RTG pc		300.00 SIT
	8	MRI		00,000.00 SIT
*	(AutoNumber)			0.00 SIT


Record:  1 of 8

Tabela (Relacija) - Izvidi

	stevilka_izvida	ID_pacienta	ID_preiskave	datum	rezultat
	1	1	5	3/14/99	77
▶	2	1	6	3/15/99 3:38:14 PM	150/100
	3	2	7	3/15/99 3:39:04 PM	b.p.
	4	1	7	3/15/99 3:52:11 PM	B.P.
	5	4	1	3/22/99 2:41:29 PM	BP
	6	4	6	3/22/99 2:41:51 PM	120/85
	7	5	3	3/22/99 2:42:56 PM	4,7
	8	8	7	3/29/99 2:42:11 PM	DIFUZNO ZASENČENJE BILL.
	9	8	8	3/29/99 2:42:50 PM	ANENCEFALUS
	10	8	1	3/29/99 2:43:32 PM	FIB. VENTRIKLOV
	11	7	5	3/29/99 2:44:37 PM	300 KG
	12	7	6	3/29/99 2:44:55 PM	91/20
	13	7	2	3/29/99 2:45:21 PM	180
	14	6	3	3/29/99 2:46:17 PM	12
	15	6	5	3/29/99 2:46:29 PM	22
	16	6	7	3/29/99 2:46:49 PM	ST. P.PNEUMONEKT. BILL.
*	(AutoNumber)	0	0	4/6/00 3:40:46 PM	

Record: 2 of 16

Zaslonski obrazec - Pacienti in izvidi

PACIENTI IN IZVIDI

ID_pacienta: pošta:
ime: kraj:
priimek: ali_kadi:
datum_rojstva:
spol:
naslov:

IZVIDI

	stevilka_izvida	ID_preiskave	datum	rezultat
▶	14	KALIJ	3/29/99 2:46:17 PM	12
	15	TEŽA	3/29/99 2:46:29 PM	22
	16	RTG pc	3/29/99 2:46:49 PM	ST. P.PNEUMONEKT. BIL
*	(AutoNumber)		4/6/00 3:38:14 PM	

Record: of 3

Record: of 9

Iskanje z primerom (QBE)

Q_Pacienti_Izvidi_Presikave : Select Query

The screenshot shows a query builder interface with three tables: PACIENTI, IZVIDI, and PREISKAVE. The IZVIDI table is selected with an asterisk (*), and the PREISKAVE table is selected with ID_preiskave. The query grid below shows the resulting fields and criteria.

Field:	PACIENTI.*	IZVIDI.*	PREISKAVE.*	spol	ali_kadi	
Table:	PACIENTI	IZVIDI	PREISKAVE	PACIENTI	PACIENTI	
Sort:						
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Criteria:				"M"	Yes	
or:						

Ekvivalenčni SQL ukaz

Q_Pacienti_Izvidi_Presikave : Select Query

```
SELECT PACIENTI.*, IZVIDI.*, PREISKAVE.*
```

```
FROM PREISKAVE INNER JOIN (PACIENTI INNER JOIN IZVIDI ON PACIENTI.ID_pacienta = IZVIDI.ID_pacienta) ON  
PREISKAVE.ID_preiskave = IZVIDI.ID_preiskave
```

```
WHERE (((PACIENTI.spol)="M") AND ((PACIENTI.ali_kadi)=Yes));
```

Poizvedovalni jezik SQL (Structured Query Language)

- ❖ Create – naredi novi objekt
- ❖ Update – ažuriraj podatke
- ❖ Insert – dodaj novi zapis
- ❖ Delete – briši enega ali več zapisov
- ❖ Select – poišči podatke

Create

```
CREATE TABLE `test`.`studenti` (  
  `id` INTEGER UNSIGNED NOT NULL DEFAULT  
    NULL AUTO_INCREMENT,  
  `ime` VARCHAR(45) NOT NULL,  
  `priimek` VARCHAR(45) NOT NULL,  
  `datum_rojstva` DATETIME NOT NULL,  
  `spol` char(1) NOT NULL,  
  `letnik` INTEGER UNSIGNED NOT NULL,  
  PRIMARY KEY (`id`)  
)  
ENGINE = InnoDB;
```


Insert, Update

```
insert into test.studenti(ime,priimek,spol, letnik)  
values ('Janez','Novak','M',1);
```

```
select * from test.studenti;  
1, Janez, Novak, NULL, M, 1
```

```
update test.studenti  
set datum_rojstva = '1990-01-03'  
where id=1;  
1, Janez, Novak, 1990-01-03, M, 1
```

Create

```
CREATE TABLE `test`.`predmeti` (  
  `id` INTEGER UNSIGNED NOT NULL  
    AUTO_INCREMENT,  
  `naziv` VARCHAR(70) NOT NULL,  
  `letnik` INTEGER UNSIGNED NOT NULL,  
  `je_pogoj` BOOLEAN NOT NULL,  
  PRIMARY KEY (`id`)  
)  
ENGINE = InnoDB;
```

Insert

```
insert into test.predmeti(naziv,letnik,je_pogoj)  
values('Informatika',1,FALSE);
```

Create

```
CREATE TABLE `semmed2009`.`izpiti` (  
  `id` INTEGER UNSIGNED NOT NULL  
    AUTO_INCREMENT,  
  `id_studenta` INTEGER UNSIGNED NOT NULL,  
  `id_predmeta` INTEGER UNSIGNED NOT NULL,  
  `datum` DATETIME NOT NULL,  
  `ocena` INTEGER UNSIGNED NOT NULL,  
  PRIMARY KEY (`id`)  
)  
ENGINE = InnoDB;
```

```
ALTER TABLE `test`.`izpiti` ADD CONSTRAINT
  `FK_studenti` FOREIGN KEY `FK_studenti`
  (`id_studenta`)
  REFERENCES `studenti` (`id`)
  ON DELETE RESTRICT
  ON UPDATE RESTRICT,
ADD CONSTRAINT `FK_predmeti` FOREIGN KEY
  `FK_predmeti` (`id_predmeta`)
  REFERENCES `predmeti` (`id`)
  ON DELETE RESTRICT
  ON UPDATE RESTRICT;
```

Prijava na izpit

```
insert into
```

```
    izpiti(id_studenta,id_predmeta,datum)  
values(1,1,'2009-12-15');
```

```
select * from izpiti;
```

```
1, 1, 1, 2009-12-15
```

Vnos ocene izpita

```
update izpiti
```

```
  set ocena = 10
```

```
where id=1;
```

```
select * from izpiti;
```

```
1, 1, 1, 2009-12-15, 10
```