



CENTAR ZA
mlade talente

Primena OOP u izradi složenih aplikacija (programski jezik C#)

Uvod u baze podataka

- ☐ Većina aplikacija vrši obradu podataka
- ☐ Najčešći način čuvanja tih podataka i njihovog korišćenja jeste baza podataka
- ☐ Podaci mogu biti različitog tipa – tekstualni, numerički, slike, audio i video zapisi i sl.

Definicija

- ❑ Baza podataka predstavlja organizovan skup podataka koji se odnosi na slične pojmove ili objekte i njihove veze
- ❑ Primeri
 - Kompanija
 - Banka
 - Fakultet
 - Biblioteka
 - Supermarket

Primer – CMT baza podataka

❑ Obuhvata podatke o:

- **Objektima** realnog sveta

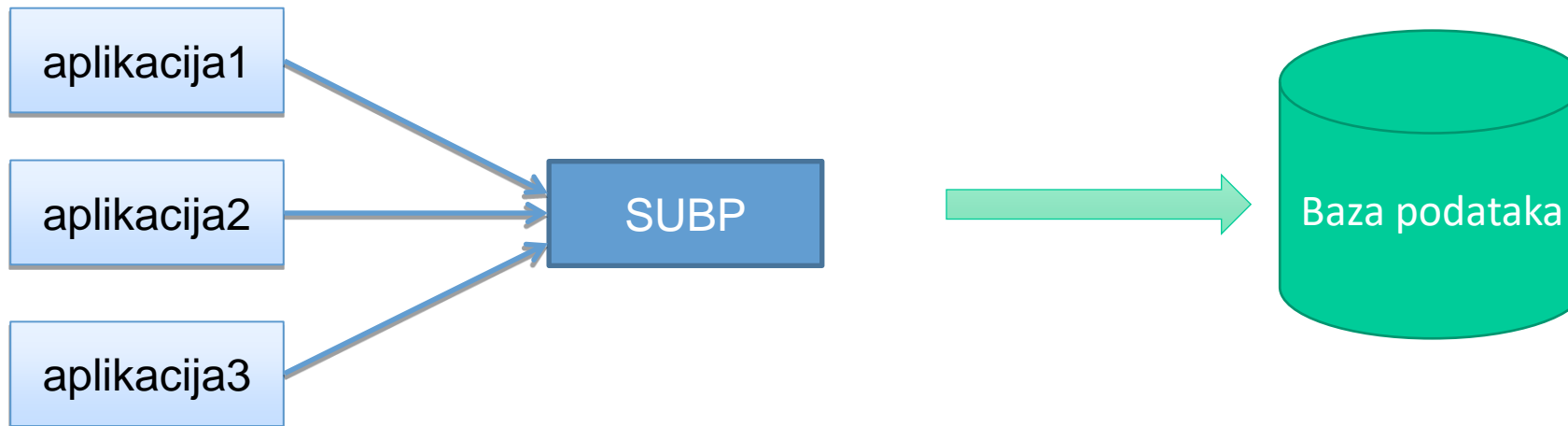
- škole
- polaznici
- kursevi
- učionice
- predavači

- **Vezama** među objektima

- *polaznik ide u školu*
- *kurs se izvodi u učionici*
- *predavač drži kurs*

Sistem za upravljanje bazama podataka (SUBP)

- Skup programskih modula koji omogućavaju pristup podacima u bazi podataka
 - Neki od SUBP-ova – *Microsoft SQL Server, Microsoft Access, Oracle, MySQL i IBM DB2*



- Jedan od načina organizovanja podataka u bazi jeste po principu tabela
 - Jedna **tabela (relacija)** predstavlja skup podataka o nekom objektu, fenomenu ili procesu
 - Podaci su podeljeni po **kolonama** (koje odgovaraju **atributima**), a u svakom **red**u (**vrsti**) date tabele su smešteni podaci o konkretnom objektu, fenomenu ili procesu
 - Redosled kolona i redova ne utiče na informacioni sadržaj tabele

Primer tabele

□ Tabela “škola”

- **Atributi**

- osobine objekta realnog sveta – npr. *naziv*, *adresa*

- **Vrste**

- predstavlja jedan slog podataka o konkretnom objektu – npr. podaci o školi *Gimnazija Isidora Sekulić*

skolald	naziv	adresa
1	Gimnazija Svetozar Marković	Njegoševa 22
2	Gimnazija Isidora Sekulić	Vladike Platona 2
3	Tehnička škola Pavle Savić	Šajkaška 34

Vrsta

Atribut

Primarni i strani ključ u tabeli

- ❑ Ne mogu da postoje dva identična reda u jednoj tabeli
 - **Primarni ključ** – atribut pomoću kojeg se jedinstveno identifikuju redovi u tabeli
- ❑ Između tabela sa podacima može da postoji **relaciona veza**
 - **Spoljni ključ** – opcioni atribut preko kojeg se ostvaruje veza za drugim tabelama

Primer veze između tabela

□ Tabela “skola”

skolald	naziv	adresa
1	Gimnazija Svetozar Marković	Njegoševa 22
2	Gimnazija Isidora Sekulić	Vladike Platona 2
3	Tehnička škola Pavle Savić	Šajkaška 34

□ Tabela “polaznik”

polaznikld	ime	prezime	skolald
1	Marko	Marković	1
2	Jovana	Jovanić	1
3	Dragan	Draganić	3

Primer veze između tabela (2)

Tabela "kurs"

kursId	naziv	jezik
1	P-K4-ISA	C#
2	P-K3-OOP	C++
3	P-K2-SPL	C

Tabela "predavac"

predavacId	ime	prezime
1	Nikola	Nikolić
2	Jelena	Jelić
3	Petar	Petrović

Tabela "predaje"

kursId	predavacId
1	1
2	2
2	3

SQL – Structured Query Language

- ❑ Programski jezik namenjen za upravljanje podacima u sistemima za upravljanje relacionim bazama podataka
 - Deklarativni jezik – definiše **ŠTA** treba da se dobije kao rezultat, za razliku od imperativnih jezika koji definišu način **KAKO** će nešto da se izvrši
- ❑ Osnovne naredbe SQL-a za rad sa bazom podataka su:
 - **SELECT** naredba – vraća skup redova tabela
 - **INSERT** naredba – upisuje nove redove u tabele
 - **DELETE** naredba – briše redove u tabelama
 - **UPDATE** naredba – vrši izmenu redova u tabelama

SELECT naredba

```
SELECT [atributi koji se selektuju]  
FROM [tabela iz koje se selektuje]  
WHERE [kriterijum na osnovu kog se selektuje]
```

INSERT naredba

INSERT INTO

[tabela u koju se vrsi insert]

([redosled atributi koji se dodaju])

VALUES [vrednosti atributa koji se dodaju]

Ukoliko se atributi nabroje redosledom koji se nalazi u bazi podataka onda nema potrebe pisati attribute na samom pocetku.

DELETE naredba

```
DELETE FROM [tabela iz koje se brise]  
WHERE [uslovi po kojima se brise]  
AND [ukoliko ima vise uslova odvajaju se sa and]
```

UPDATE naredba

UPDATE [tabela]

SET [vrednost koja se menja]

WHERE [uslov po kom se pronalazi podatak koji se modifikuje]

SQL – Primeri

- ❑ Predavači koji se prezivaju Jović poređani po abecednom redu
 - **SELECT** * **FROM** predavac **WHERE** prezime = 'Jović'
ORDER BY ime;
- ❑ Ukupan broj kurseva u bazi
 - **SELECT COUNT**(*) **FROM** kurs;
- ❑ Upis škole sa nazivom Jovan Jovanović Zmaj
 - **INSERT INTO** skola (naziv, adresa)**VALUES** ('Gimnazija Jovan Jovanović Zmaj', 'Zlatne grede 4');
- ❑ Brisanje polaznika Dragana Draganića
 - **DELETE FROM** polaznik **WHERE** ime = 'Dragan' **AND** prezime = 'Draganić';
- ❑ Menjanje imena polazniku sa id-em 2
 - **UPDATE** polaznik **SET** ime = 'Aca' **WHERE** polaznikId = 2;

Spajanje dve tabele – osnovno spajanje

skolald	naziv		polaznikld	ime	skolald
1	Svetozar Marković	↔	1	Marko	1
2	Isidora Sekulić	↔	2	Jovana	1

❑ **SELECT** * **FROM** skola, polaznik;

skolald	naziv	polaznikID	Ime	skolald
1	Svetozar Marković	1	Marko	1
1	Svetozar Marković	2	Jovana	1
2	Isidora Sekulić	1	Marko	1
2	Isidora Sekulić	2	Jovana	1

Spajanje dve tabele po jednakosti atributa

- Prikazati id i ime polaznika kao i naziv škole koju pohađaju

polaznikID	ime	skolald	skolald	naziv
1	Marko	1	1	Svetozar Marković
2	Jovana	1	2	Isidora Sekulic

- **SELECT** p.polaznikId, p.ime, s.naziv **FROM** polaznik p, skola s **WHERE** p.skolaId = s.skolaId;

ucenikId	ime	naziv
1	Marko	Svetozar Marković
2	Jovana	Svetozar Marković

Spajanje dve tabele po jednakosti atributa

- Prikazati nazive kurseva i imena njihovih predavača

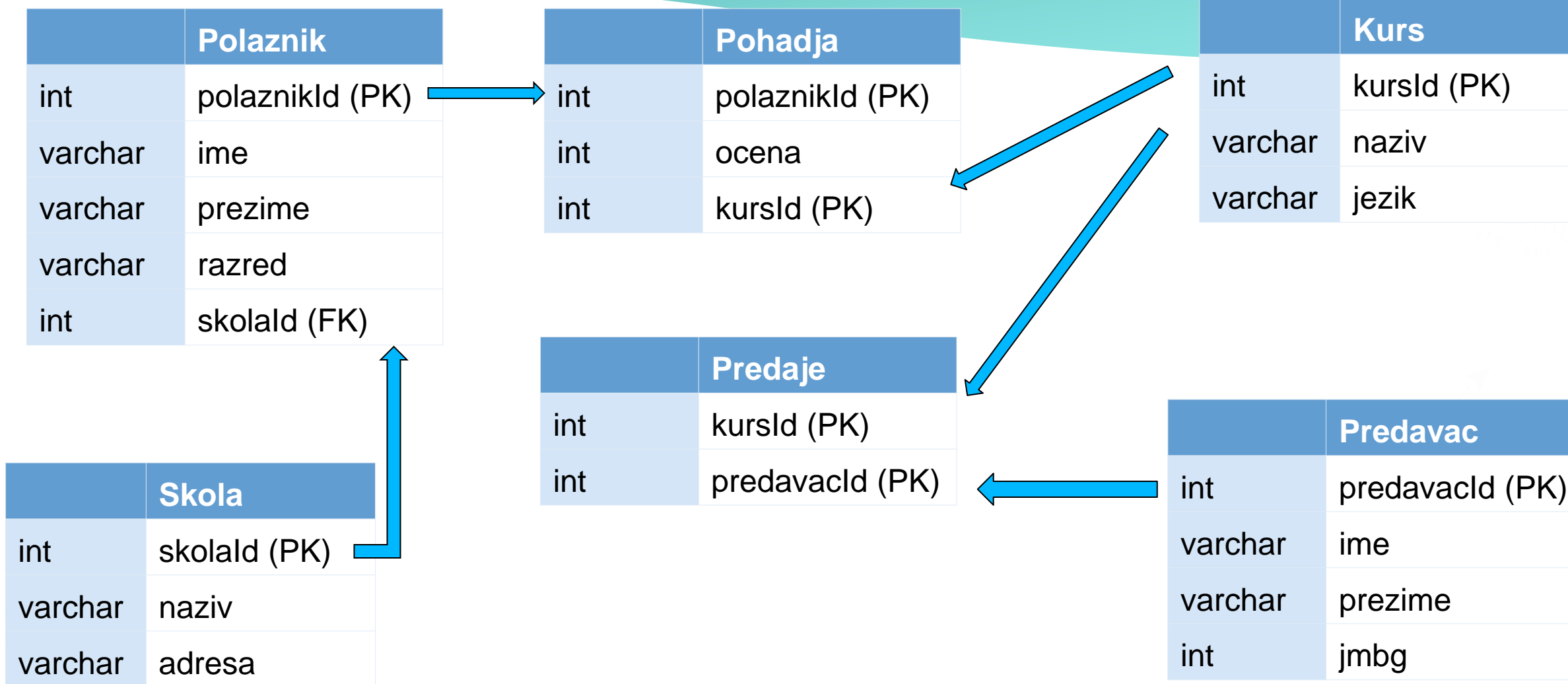
Tabela "kurs"		Tabela "predaje"		Tabela "predavac"	
kursId	nazivKurs	kursId	predId	predId	ime
1	P-K4-ISA	1	1	1	Nikola Nikolić
2	P-K3-OOP	2	2	2	Jelena Jelić
3	P-K2-SPL	2	3	3	Petar Petrović

Rezultat

- ```
SELECT k.nazivKurs, p.ime
FROM kurs k, predavac p,
 predaje pr
WHERE k.kursId = pr.kursId
AND pr.predId = p.predId;
```

| nazivKurs | ime            |
|-----------|----------------|
| P-K4-ISA  | Nikola Nikolić |
| P-K3-OOP  | Jelena Jelić   |
| P-K3-OOP  | Petar Petrović |

# Tabele u nasoj CMT bazi podataka



# Hvala na pažnji!



CENTAR ZA  
mlade talente