

UVOD U BAZE PODATAKA



Neđeljko Lekić
Irena Orović

www.etf.ac.me

U OVOJ LEKCIJI

- Informacije o predmetu.
- Što je baza podataka?
- Sistemi za upravljanje bazama podataka (DBMS – database management system).
- Relacioni model.

INFORMACIJE O PREDMETU

BAZE PODATAKA

<i>Status predmeta</i>	<i>Semestar</i>	<i>Broj ECTS kredita</i>	<i>Fond časova</i>
<i>Obavezan</i>	<i>III</i>	<i>6</i>	<i>3P+0V+2L</i>

Uslovljenost drugim predmetima:

Nema formalnih uslova.

Metod nastave i savladanja gradiva:

Predavanja, računske vježbe i vježbe u računarskoj učionici. Učenje i samostalna izrada praktičnih zadataka. Konsultacije.

Ime i prezime nastavnika i saradnika:

Doc. dr Irena Orović, Doc. dr Neđeljko Lekić - nastavnici

Mr Predrag Raković - saradnik

INFORMACIJE O PREDMETU

Sadržaj predmeta:

Pripremna sedmica

Priprema i upis semestra

I sedmica

Uvod; Što je baza podataka? Sistem za upravljanje bazama podataka.

II sedmica

Relacije i relaciona algebra.

III sedmica

Relacioni model podataka.

IV sedmica

Model objekti veze (E/R - Entity-Relationship model).

V sedmica

SQL - Opis podataka (DDL).

VI sedmica

Slobodna sedmica.

VII sedmica

I provjera znanja.

VIII sedmica

SQL – Rukovanje podacima (DML).

IX sedmica

Nedostajući podaci.

X sedmica

Normalizacija.

XI sedmica

Indeksi i optimizacija upita.

XII sedmica

Transakcije i konkurentnost.

XIII sedmica

II provjera znanja.

XIV sedmica

Sigurnost i integritet podataka.

XV sedmica

Moderne baze podataka (Druge vrste baza podataka).

XVI sedmica

SQL u drugim programskim jezicima.

Završna sedmica

Završni ispit, ovjera semestra i upis ocjena

XVIII-XXI sedmica

Dopunska nastava i popravni ispitni rok

INFORMACIJE O PREDMETU

Opterećenje studenata na predmetu

Sedmično

6 kredita x 40/30 = 8 časova

Struktura:

3 časa predavanja

2 časa rada u računarskoj učionici

3 časa samostalnog rada,
uključujući konsultacije

U toku semestra

Nastava i završni ispit: 8 časova x 16 = **128**

Neophodne pripreme prije početka semestra
(administracija, upis, ovjera)

2 x 8 časova = **16 časova**

Ukupno opterećenje za predmet **6x30 = 180 časova**

Dopunski rad za pripremu ispita u popravnom ispitnom roku, uključujući i polaganje popravnog ispita od 0 do 36 časova (preostalo vrijeme od prve dvije stavke do ukupnog opterećenja za predmet 180 časova)

Struktura opterećenja:

128 časova (Nastava)+16 časova (Priprema)+30 časova (Dopunski rad)

Studenti su obavezni da pohađaju nastavu, rade zadatke tokom nastave, odrade laboratorijske vježbe, rade oba kolokvijuma i završni ispit.

INFORMACIJE O PREDMETU

Literatura: Materijal u elektronskoj formi koji studenti dobijaju u toku nastave,
<http://dev.mysql.com> (mysql reference manual),
Poliščuk E. J.: Baze podataka, JEP, Podgorica, 2003,
Literatura sa interneta.

Oblici provjere znanja i ocjenjivanje:

Rad na laboratorijskim vježbama, 10 poena,
Dvije provjere znanja po maksimalno, 20 poena (ukupno maksimalno 40 poena),
Završni ispit se ocjenjuje sa maksimalno, 50 poena.
Prelazna ocjena se dobija ako se kumulativno sakupi najmanje 50 poena.

Posebna naznaka za predmet:

U slučaju da je to potrebno nastava se može izvoditi i na engleskom jeziku.

KRATAK PREGLED SADRŽAJA PREDMETA

■ Nekoliko glavnih tema

- DBMS
- Modeliranje podataka
- Dizajniranje baze
- SQL
- Transakcije
- Konkurentnost
- Administriranje

■ Praktični dio

□ MySQL

- Kreiranje baze podataka
- Rad sa bazom podataka

ISTORIJA



Unos podataka



Składištenje i pretraživanje



Obrada upita (zahtjeva)



Sortiranje

NEKOLIKO DEFINICIJA POJMA BAZE PODATAKA

- "A set of information held in a computer"

Oxford English Dictionary

- "One or more large structured sets of persistent data, usually associated with software to update and query the data"

Free On-Line Dictionary of Computing

- "A collection of data arranged for ease and speed of search and retrieval"

Dictionary.com

ZAŠTO IZUČAVATI BAZE PODATAKA

■ Baze podataka su upotrebljive

- Mnoge računarske aplikacije rade sa velikom količinom podataka
- DBMS obezbeđuje skup alata za skladištenje, pretraživanje i upravljanje tim podacima.

■ Baze podataka u računarskim naukama

- Baze podataka su jezgro računarskih nauka.
- Za diplomiranog studenta računarskih nauka podrazumijeva se poznavanje osnovnih koncepata baza podataka i određeno iskustvo u radu sa sistemima za upravljanje bazama podataka.

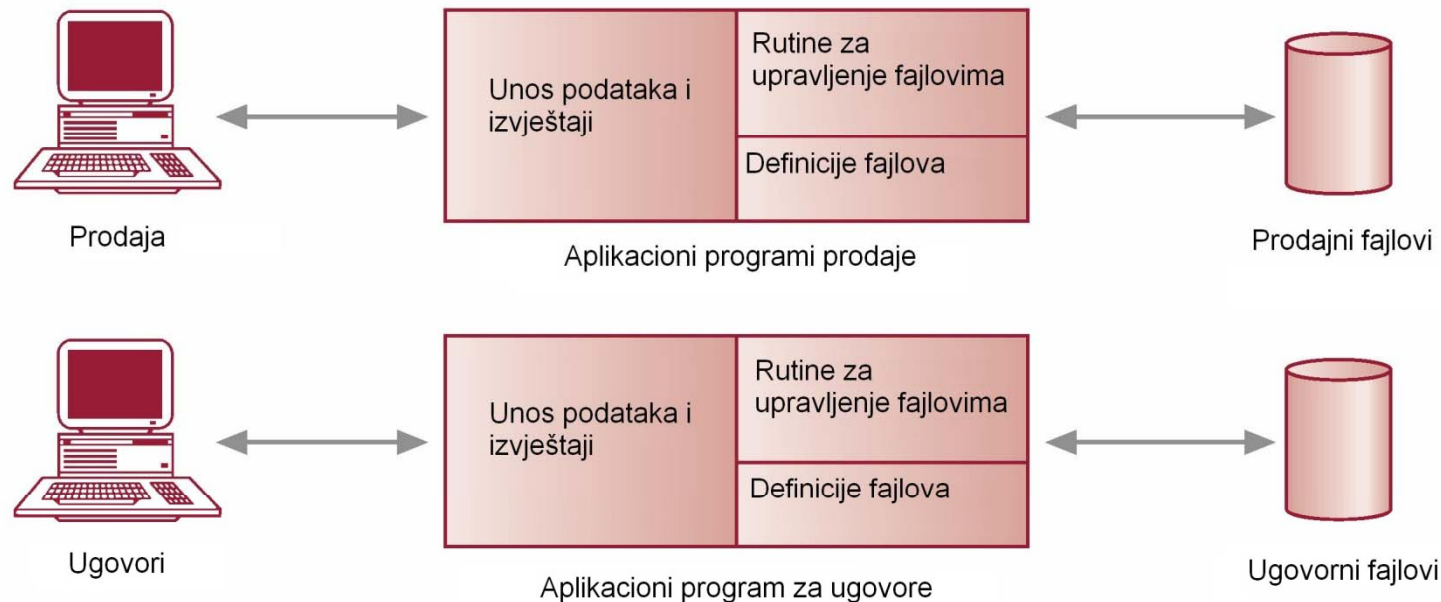
BAZE PODATAKA - PRIMJENE

- Web indeksi
- Bibliotetski katalogi
- Medicinski zapisi
- Bankovni računi
- Kontrola zaliha
- Katalogi proizvoda
- Telefonski imenici
- Red vožnje
- Avio rezervacije
- Kreditne kartice
- Podaci studenata
- Podaci o korisnicima
- Berza
- i tako dalje ...

FAJL SISTEMI

- **Aplikacioni programi koji servisiraju krajnjeg korisnika (npr. izvještaji).**
- **Svaki program definiše i upravlja svojim vlastitim podacima.**

OBRADA PODATAKA U FAJL SISTEMIMA



Prodajni fajlovi

ImovinaZaProdaju (imovinaNo, ulica, grad, poštanskiBroj, vrsta, cijena, vlasnikNo)

VlasniciImovine (vlasnikNo, Ime, Prezime, adresa, telNo)

Klijent (klijentNo, Ime, Prezime, adresa, telNo, vrsta, maxCijena)

Ugovorni fajlovi

Zakup (zakupNo, imovinaNo, klijentNo, renta, nacinPlacanja, depozit, plaćeno, pocetakRente, krajRente, trajanje)

ImovinaZaZakup (imovinaNo, ulica, grad, poštanskiBroj, renta)

Klijent (klijentNo, Ime, Prezime, adresa, telNo)

OGRANIČENJA FAJL SISTEMA

■ **Odvojenost (izoliranost) podataka**

- Svaki program upravlja vlastitim setom podataka.
- Korisnici jednog programa često ne znaju za potencijalno korisne podatke drugog programa.

■ **Dupliranje podataka (redudansa)**

- Isti podaci se čuvaju u različitim programima.
- Rasipanje resursa i potencijalno različite vrijednosti i/ili različiti formati istih pojmova.

OGRANIČENJA FAJL SISTEMA

■ Zavisnost od podataka

- Fajl struktura je definisana u programskom kodu.

■ Nekompatibilni formati fajlova

- Programi su pisani u različitim jezicima, i ne mogu jednostavno pristupati fajlovima drugih programa.

■ Problemi sa novim zahtjevima

- Programi su pisani da zadovolje posebne funkcije.
- Svaki novi zahtjev stvara potrebu za novim programom.

SISTEMI ZASNOVANI NA BAZI PODATAKA

■ Nastali su zbog:

- Definicije podataka su bile ugrađene u aplikacioni program, umjesto da budu smještene odvojeno i nezavisno.
- Nema kontrole nad pristupom i obradom podataka izvan aplikacionog programa.

■ Rezultat:

- Sistem za upravljanje bazom podataka (DBMS).

SISTEMI ZASNOVANI NA BAZI PODATAKA

Sistemi zasnovani na bazi podataka (Database System) objedinjuju u cjelinu:

- Bazu podataka,
- Model podataka i
- Sistem za upravljanje bazom podataka (DBMS)

BAZA PODATAKA - DEFINICIJA

Baza podataka je skup međusobno povezanih podataka, pohranjenih bez nepotrebne redundantnosti, s ciljem da na optimalni način posluže u raznim primjenama.

- Podaci se kreiraju nezavisno od programa koji ih koriste. Zajedničkim pristupom dodaju se novi podaci te mijenjaju i premještaju postojeći.
- Podaci se pohranjuju u bazu podataka na organizovan način koristeći odgovarajući model podataka.

MODEL PODATAKA - DEFINICIJA

Model podataka je formalni sistem koji mora imati barem sljedeće tri komponente:

1. Skup objekata koji su osnovni elementi baze podataka;
2. Skup operacija koje se mogu izvoditi nad objektima definisanim pod 1) i kojima se mogu pretraživati, dodavati i modifikovati podaci o tim objektima;
3. Skup opštih pravila integriteta podataka koja implicitno ili eksplicitno definišu skup konzistentnih stanja podataka ili promjena stanja, ili oboje. Pravila su opšta u smislu da su primjenjiva na bilo koju bazu podataka koja koristi taj model.

ZA ŠTO SE KORISTI MODEL PODATAKA?

- Za razvijanje sistema za upravljanje bazom podataka.
- Za razvijanje programskih jezika za rad s podacima u bazi podataka.
- Za razvijanje opštih teorija oblikovanja baza podataka.
- Za istraživanje svojstava podataka, odnosno svojstava baza podataka.

SISTEMI ZA UPRAVLJANJE BAZOM PODATAKA

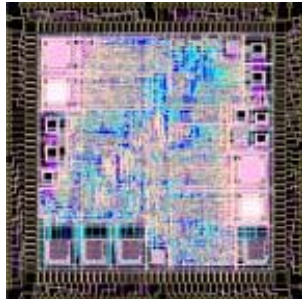
- Baza podataka je kolekcija podataka.
- Sistem za upravljanje bazom podataka (DBMS – Database Management System) je softver koji kontroliše te podatke.

SISTEM ZA UPRAVLJANJE BAZOM PODATAKA

Aplikacija
dolazi ovdje

DBMS

DBMS interfejs
omogućuje da aplikacije
i sistem za upravljanje
podacima budu
izvedeni odvojeno



Sirovi podaci (goli metal)

ŠTO DBMS DONOSI?

- **Obezbjeđuje:**

- **Jezik za opis podataka** (DDL - Data definition language)
- **Jezik za rukovanje podacima** (DML - Data manipulation language)
- **Jezik za kontrolu podataka** (DCL - Data control language)

- **Često se ovi jezici smatraju jednim jezikom – SQL.**

- **DBMS obezbjeđuje**

- Trajnost
- Konkurentnost
- Integritet
- Bezbjednost
- Nezavisnost podataka

- **Rječnik podataka**

- Opisuje samu bazu podataka.

ŠTO DBMS DONOSI?

- **Fizička nezavisnost podataka.** Razdvaja se logička definicija baze od njene stvarne fizičke građe.

Na primjer, ako se fizička građa promijeni (na primjer, podaci se prepisu u druge datoteke na drugim diskovima), to neće zahtijevati promjene u postojećim aplikacijama.

- **Logička nezavisnost podataka.** Razdvaja se globalna logička definicija cijele baze podataka od lokalne logičke definicije za jednu aplikaciju.

Na primjer, ako se logička definicija promijeni (na primjer uvede se novi entitet ili veza), to neće zahtijevati promjene u postojećim aplikacijama.

- **Mogućnost oporavka nakon kvara.** Zaštita baze u slučaju kvara hardvera ili grešaka u radu sistemskog softvera.

ŠTO DBMS DONOSI?

- **Fleksibilnost pristupa podacima.** Korisnik može slobodno pretraživati podatke, i po želji uspostavljati veze među podacima.

U starijim mrežnim i hijerarhijskim bazama, staze pristupanja podacima bile su unaprijed definisane. Korisnik je mogao pretraživati podatke jedino onim redoslijedom koji je bio predviđen u vrijeme projektovanja i implementiranja baze.

- **Istovremeni pristup do podataka.** Mogućnost da veći broj korisnika istovremeno koristi iste podatke. Korisnici ne smiju ometati jedan drugoga.

ŠTO DBMS DONOSI?

- **Zaštita od neovlašćenog korišćenja.** Mogućnost da se korisnicima ograniče prava korišćenja baze.

Svakom korisniku se dodjeljuju ovlašćenja: što on smije, a što ne smije raditi s podacima.

- **Zadovoljavajuća brzina pristupa.** Operacije nad podacima moraju se odvijati dovoljno brzo, u skladu s potrebama određene aplikacije.

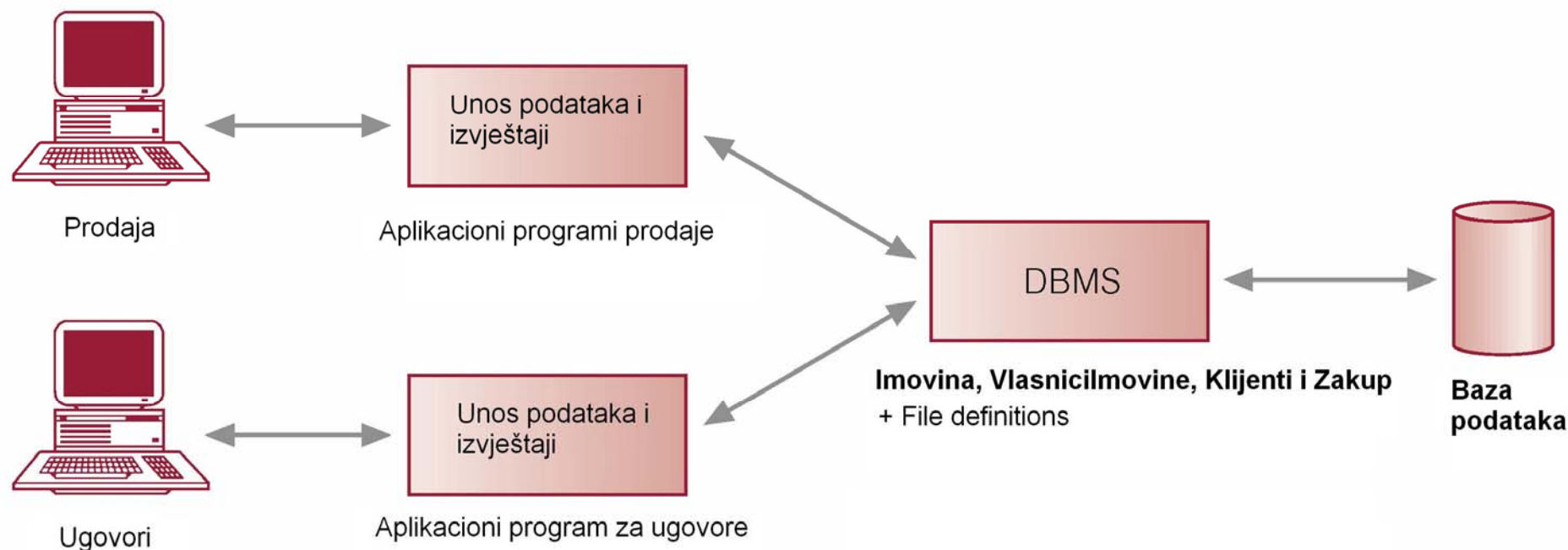
Na brzinu pristupa može se uticati izborom pogodnih fizičkih struktura podataka, te izborom pogodnih algoritama za pretraživanje.

- **Mogućnost podešavanja i kontrole.** Velika baza zahtijeva stalnu brigu: praćenje performansi, mijenjanje parametara u fizičkoj građi, rutinsko smještanje rezervnih kopija podataka.

RJEČNIK PODATAKA - METAPODACI

- Rječnik ili katalog sadrži informacije o samoj bazi podataka.
- Podaci o podacima ili 'metapodaci'.
- Gotovo svaki segment DBMS koristi rječnik.
- Rječnik sadrži:
 - Opis objekata baze podataka (tabela, korisnika, pravila, pogleda, indeksa, ...)
 - Informacije ko koristi koje podatke (brave).
 - Šeme i preslikavanja.

SISTEM ZA UPRAVLJANJE BAZOM PODATAKA



Imovina(imovinaNo, ulica, grad, poštanskiBroj, vrsta, cijena, vlasnikNo)

Vlasnicimovine (vlasnikNo, Ime, Prezime, adresa, telNo)

Klijent (klijentNo, Ime, Prezime, adresa, telNo, vrsta, maxCijena)

Zakup (zakupNo, imovinaNo, klijentNo, renta, nacinPlacanja, depozit, plaćeno, pocetakRente, krajRente, trajanje)

SISTEM ZA UPRAVLJANJE BAZOM PODATAKA

Danas postoji više različitih DBMS-a:

mysql:

www.mysql.org

Open source, dosta moćan

PostgreSQL:

www.postgresql.org

Open source, moćan

Microsoft Access:

Jenostavan sistem sa puno korisnih grafičkih alata.

Komercijalni sistemi:

Oracle 11g (www.oracle.com)

SQL Server (www.microsoft.com/sql)

DB2 (www.ibm.com/db2)

PREDNOSTI SISTEMA SA BAZOM PODATAKA

- Minimalna redudantnost podataka
- Konzistentnost podataka.
- Integracija podataka.
- Dijeljenje podataka.
- Sprovođenje standarda.
- Lakoća razvoja aplikacija.
- Uniformni postupak obezbjeđenja sigurnosti, privatnosti i integriteta.
- Nezavisnost podataka.

KORISNICI BAZE PODATAKA

- **Krajnji korisnici**

- Koriste baze podataka da dodju do potrebne informacije

- **Aplikacioni programeri**

- Pišu softver koji krajnjim korisnicima omogućuje pristup bazi podataka.

- **Administratori baze podataka (DBA)**

- Kreiraju i upravljaju bazom podataka

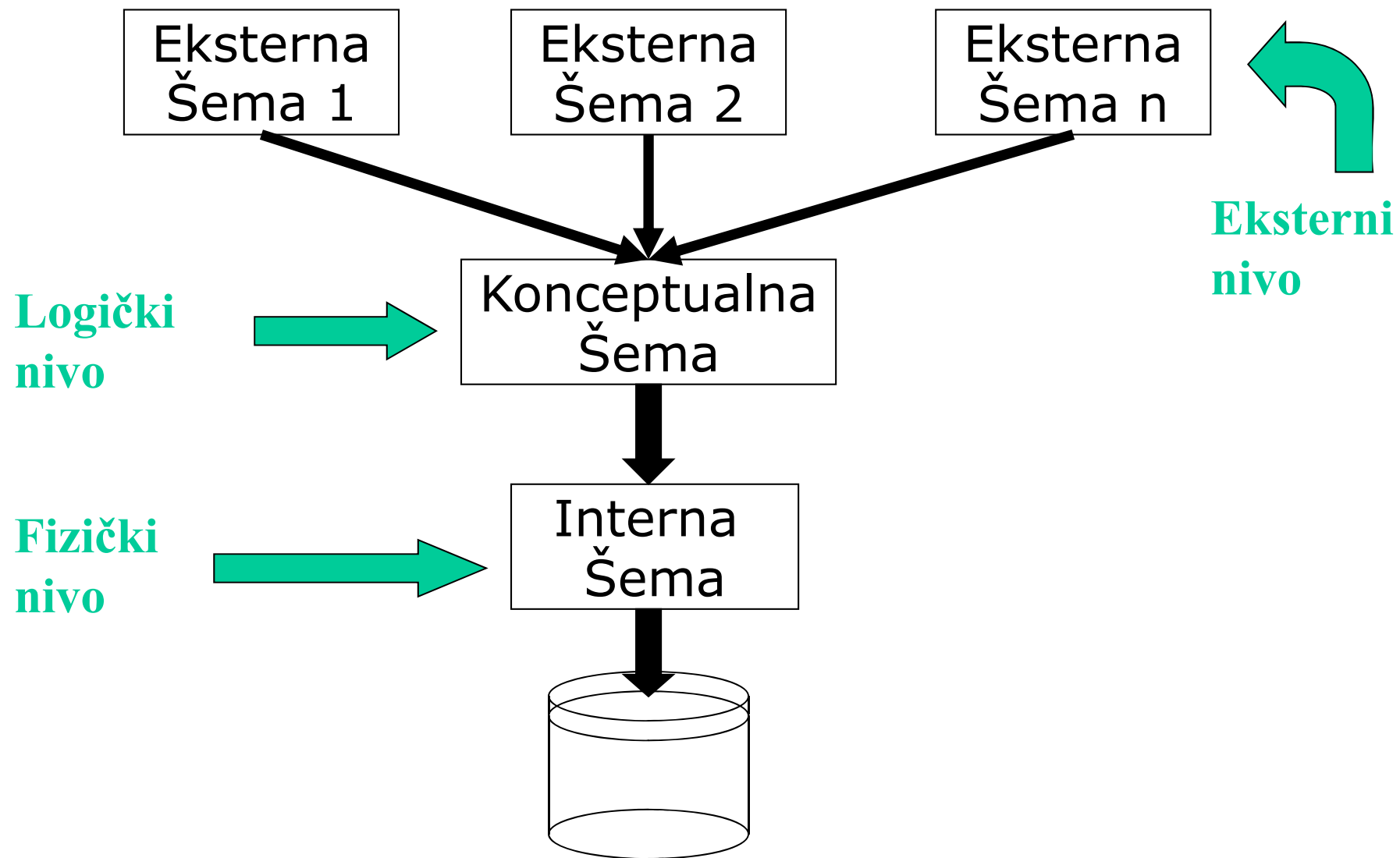
- **DBMS programeri**

- Pišu DBMS softver

ANSI/SPARC Arhitektura

- **ANSI** - American National Standards Institute
- **SPARC** - Standards Planning and Requirements Committee
- **1975** – predložen okvir za baze podataka
- **Tri nivoa arhitekture**
 - **Interni nivo (Fizički nivo):** Za systemske dizajnere
 - **Konceptualni nivo (Logički nivo):** Za dizajnere baze podataka i administratore
 - **Eksterni nivo (Eksterna šema):** Za korisnike baze podataka

TRI NIVOJA ARHITEKTURE



Interni nivo

■ Fizičko smještanje podataka

- Struktura zapisa na disku - fajlovi, stranice, blokovi
- Indeksi i uređivanje zapisa
- Koriste programeri DBMS sistema

■ Interna šema

```
RECORD EMP  
LENGTH=44  
HEADER: BYTE(5)  
        OFFSET=0  
NAME:   BYTE(25)  
        OFFSET=5  
SALARY: FULLWORD  
        OFFSET=30  
DEPT:   BYTE(10)  
        OFFSET=34
```

Konceptualni nivo

- **Organizacija podataka kao cjeline**

- Abstrakcija uvedena u cilju uklanjanja potrebe za detaljima internog nivoa.
- Koriste DBAs i programeri aplikacija

- **Konceptualna šema**

```
CREATE TABLE  
Zaposleni (  
Ime VARCHAR(25),  
Zarada REAL,  
imeOdjeljenja INTEGER)
```

Externi nivo

- **Pogled na bazu podataka skrojen prema potrebi korisnika**

- Dio podataka može ostati skriven
- Podaci su predstavljeni u upotrebljivom obliku
- Koriste krajnji korisnici i programeri aplikacija

- **Eksterna šema**

```
SELECT Ime,  
Zarada FROM  
Zaposleni
```

Preslikavanja

- **Preslikavanja prevode informacije sa jednog nivoa na drugi**

- Externi/Konceptualni
- Konceptualni/Interni

- **Preslikavanja obezbjeđuju nezavisnost podataka**

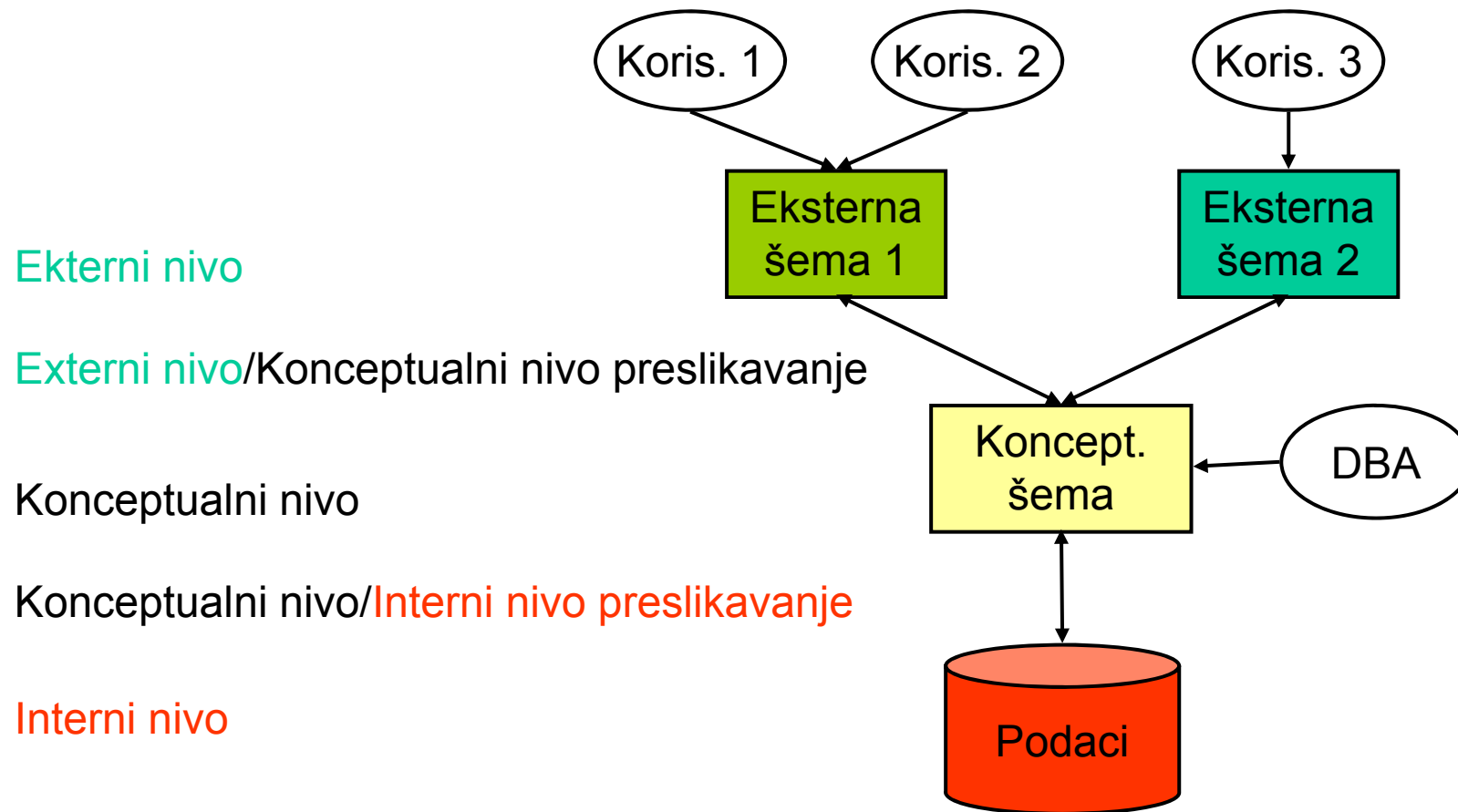
- **Fizička nezavisnost podataka**

- Promjene unutar internog nivoa ne utiču na konceptualni nivo

- **Logička nezavisnost podataka**

- Promjene na konceptualnom nivou ne utiču na eksterni nivo

PRESLIKAVANJA

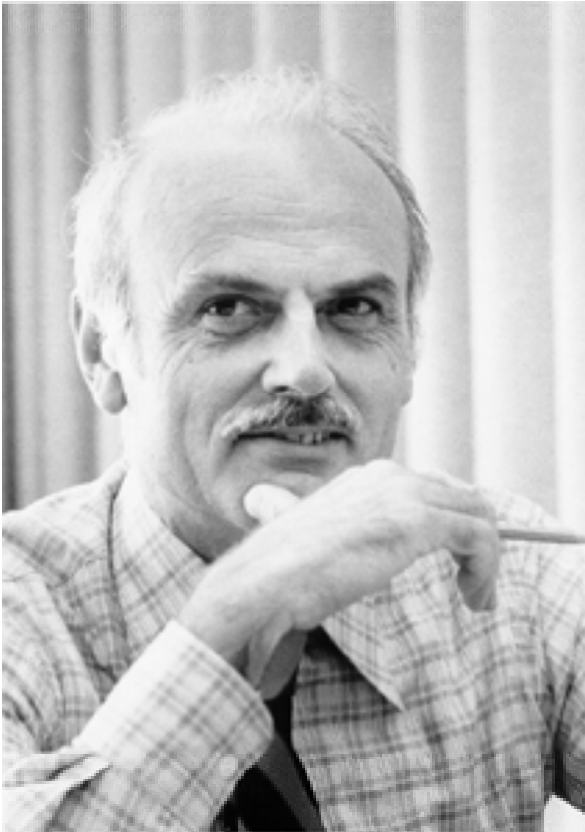


RELACIONI SISTEM BAZE PODATAKA

- Problemi sa ranijim bazama podataka
 - Rad sa zapisima zahtijeva kompleksne programe.
 - Minimalna nezavisnost podataka
 - Nema teoretske osnove
- Onda, 1970, E. F. Codd je napisao "A Relational Model of Data for Large Shared Databanks" i uveo relacioni model.

EDGAR F. CODD

Edgar F. "Ted" Codd (23 Avgust 1923 -18 April 2003) – Britanski naučnik računarskih nauka koji je, dok je radio za IBM, izumio relacione baze podataka.



Rođen je u Portlandu, Dorset. Studirao je matematiku i hemiju na Oxford-u. Tokom drugog svjetskog rata, bio je pilot u Royal Air Force. 1948 zaposlio se u IBM u New York-u kao matematičar programer. Za vrijeme McCarthy-ja prebjegao je u Kanda. Kasnije se vratio u USA i radio doktorat iz oblasti računarskih nauka, na Univerzitetu Michigan u Ann Arbor-u. Zatim je nastavio kao IBM istraživač u San Jose-u.

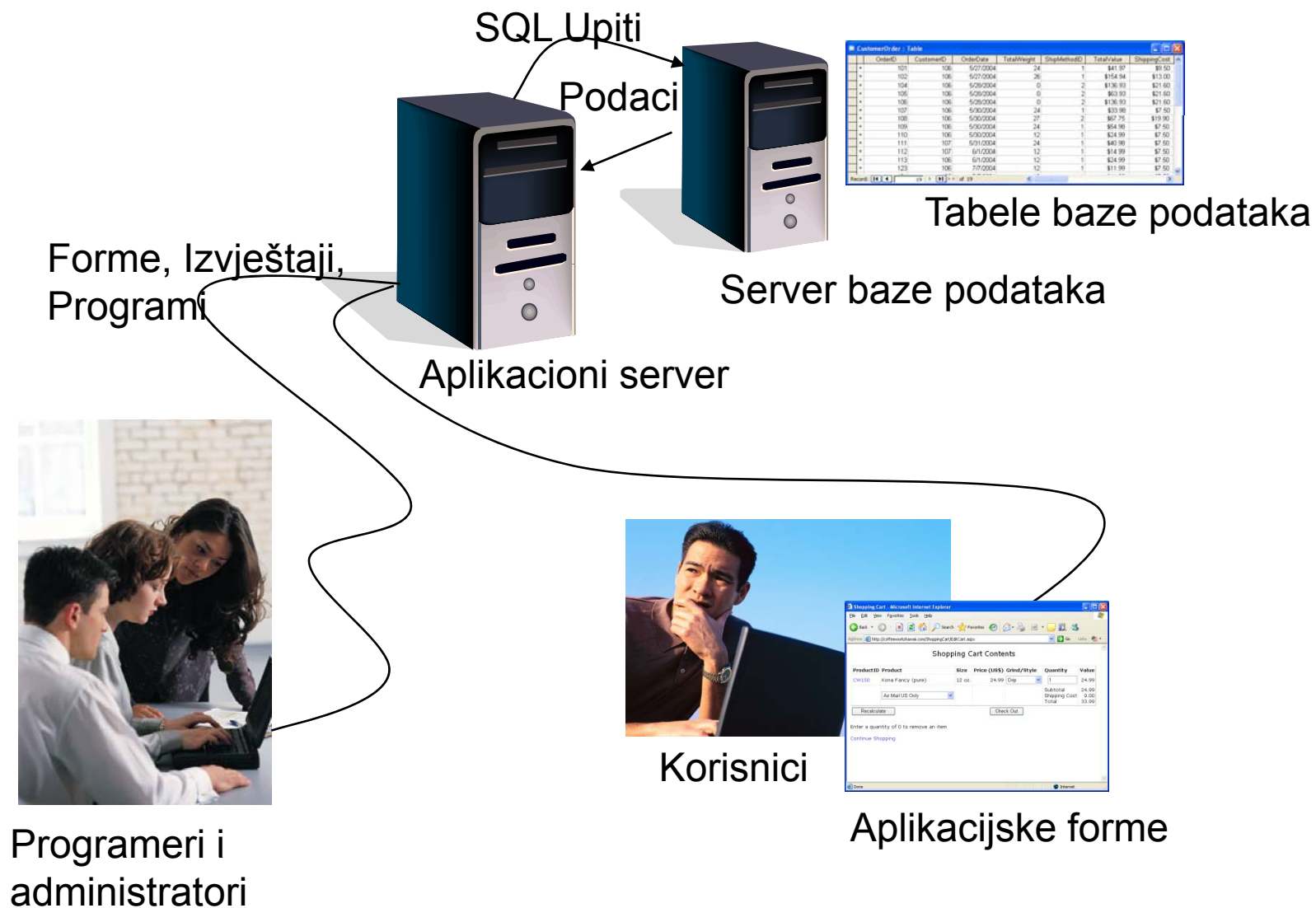
1970, njegov rad "A Relational Model of Data for Large Shared Data Banks" mijenja sve.

Tokom 1990-tih uveo je pojam:
OLAP (On Line Analytical Processing).

RELACIONI SISTEM BAZE PODATAKA

- Informacije se smještaju kao *n-torke* (tuples) ili *zapisi* (records) u *relacijama* ili *tebelama*
- Oslanja se na matematičku relacionu teoriju.
- Najviše modernih DBMS bazirano je na relacionom modelu.
- Relacioni model pokriva 3 područja:
 - Stukturu podataka
 - Integritet podataka
 - Manipulisanje podacima
- Više detalja u sljedećoj lekciji...

RAZVOJ APLIKACIJA SA DBMS



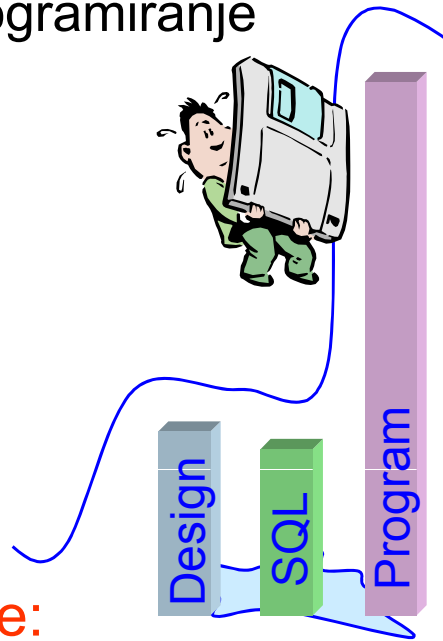
IZRADA POSLOVNE APLIKACIJE

Alati:

Dizajn baze podataka

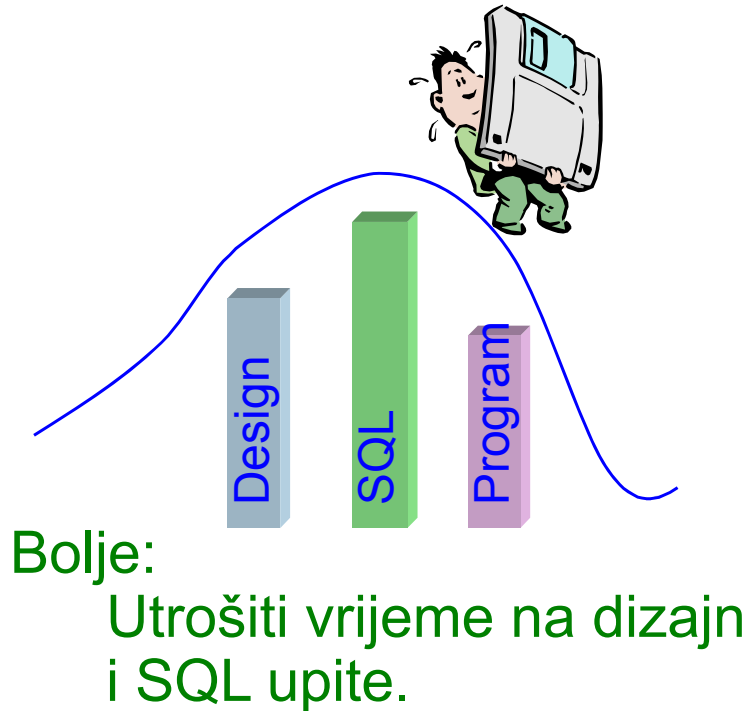
SQL (upiti)

Programiranje



Loše:

Nadoknađivati loš dizajn baze podataka
i nedovoljno moćne SQL upite programiranjem.



DIZAJNIRANJE BAZE PODATAKA

Analiza zahtjeva

Potrebe korisnika; što baza podataka mora zadovoljiti?

Konceptualni dizajn

Opis visokog nivoa apstrakcije; često se koristi E/R model.

Logički dizajn

Prevođenje E/R modela u (tipično) relacionu šemu.

Optimizovanje relacione šeme

Otklanjanje redudanse i anomalija.

Fizički dizajn/optimizacija

Fizička realizacija baze podataka i njena dalja optimizacija

PITANJA IZ OVE LEKCIJE

Što je to baza podataka?

Što predstavlja DBMS?

Koje su primjene baze podataka?

Kako teče obrada podataka u fajl sistemima?

Koji su nedostaci fajl sistema?

Kako teče obrada podataka u sistemima sa bazom podataka?

Koje su prednosti sistema sa bazom podataka?

Koji nivoi postoje u ANSI/SPARC modelu.

Koja je uloga pojedinog nivoa?

Koji korisnici su zainteresovani za koji nivo?

Kako su nivoi međusobno povezani?

Kakav se postupak preporučuje pri dizajniranju baze podataka?