

Predmet:
Uvod u baze podataka

SEMINARSKI RAD

Tema:
Biblioteka

Studenti:

Neira Piranić
Dajana Prašo
Emina Palalić
Amila Zaimović

Predmetni nastavnik:

Doc. Dr. Damir Omerašević dipl. ing.

Sadržaj

UVOD	3
Sadržaj baze biblioteka	4
Opis tabela	4
ER DIJAGRAM.....	6
Fizičko kreiranje baze.....	7
Kreiranje tabela	7
Unos podataka u bazu.....	9
ERD model.....	9
Prikaz sadržaja tabele	10
SELECT iskaz	10
AS operator	11
ORDER BY klauzula.....	12
LIKE operator	12
IS NULL & IS NOT NULL operator	12
Agregatne funkcije.....	12
Join-ovi.....	13
Inner join.....	13
Left join	15
Pogledi (Views)	16
Kreiranje procedure.....	16
Trigeri	17
Normalizacija baze podataka	18
Prva normalna forma	18
Druga normalna forma.....	20
ERD model nakon normalizacije.....	24
Literatura.....	25

UVOD

Od postanka ljudske civilizacije pa sve do danas ljudi su nalazili način da izučavaju različite sfere života te pišu od istim. Od najstarijeg književnog djela ,Ep o Gilgamešu, kroz historiju se mijenjao oblik knjige, način na koji se piše, teme o kojima se piše ali i način na koji se knjige nabavljaju te općenite informacije o knjigama.

Nakon razvoja industrije došlo je do razvoja informacijskog doba koji je obilježen samom brzinom kretanja informacije. To je sve omogućilo da je jedan od najvećih prioriteta upravo bude sama pohrana informacija, njihova organizacija te lakši i efikasniji pristup istim. Tada na scenu dolaze baze podataka koje omogućavaju da sve prethodno navedeno bude na dohvat ruke.

Što se tiče baza podataka koje se bave bibliotekama one se ne razlikuju mnogo od obične biblioteke koja vodi evidenciju na neki drugi način jer svi imaju isti cilj ali baze podataka tu pokazuju svoju prednost u brzini i efikasnosti.

Ideja naše baze podataka jeste da pokaže osnovne informacije o knjiga, korisnicima biblioteke ali i o radnicima pomoću kojih vrlo lahko možemo naći ko je i kad zadužio neku knjigu kod kojeg radnika to jest bibliotekara.

Sadržaj baze biblioteka

Prije kreiranja baze podataka, trebamo osmisлити sam izgled i sadržaj baze. S obzirom da se u biblioteci nalaze knjige, a te knjige raspoređuju i izdaju radnici koji tu rade, članovima koji su učlanjeni u biblioteku za biblioteku imamo osnovne tabele: knjige, radnici, članovi, izdavanje.

Opis tabela

Tabela knjige čuva podatke o knjigama koje se nalaze u biblioteci. Njeni atributi su: id_knjige, koji je jedinstven za svaku knjigu, naziv_knjige, autor, godina_izdanja, žanr, jezik (na kojem je pisana knjiga ili na koji je prevedena), ukupna_kolicina (broj dostupnih knjiga), broj_zaduzenih, broj_citaone i broj_police.

Tako da bi entitet tabele „knjige” izgledao ovako:

knjige	
PK(primarni ključ)	*id_knjige
	naziv_knjige
	autor
	godina_izdanja
	žanr
	jezik
	ukupna_kolicina
	broj_zaduzenih
	broj_citaone
	broj_police

Tabela radnici čuva podatke o radnicima koji su zaposleni u biblioteci, tačnije, čuva njihove informacije, kao što su ime i prezime radnika, kontakt, adresa i slično. Njeni atributi su: id_radnika, koji je jedinstven za svakog radnika, ime, prezime, spol, datum_rodjenja, datum_zasposlenja, posao_koji_obavlja, email_adresa i broj_telefona.

Tako da bi entitet tabele „radnici” izgledao ovako:

radnici	
PK(primarni ključ)	*id_radnika
	ime
	prezime
	spol
	datum_rodjenja
	datum_zasposlenja
	posao_koji_obavlja
	email_adresa
	broj_telefona

Tabela članovi čuva podatke o ljudima koji su učlanjeni u biblioteku, tačnije, čuva informacije kao što su ime i prezime člana, njegova adresa, kontakt i slično. Atributi ove tabele su: id_clana, koji je jedinstven za sve članove, ime, prezime, grad, postanski_broj, adresa_stanovanja i telefonski_broj.

Tako da bi entitet tabele „*clanovi*” izgledao ovako:

clanovi	
PK(primarni ključ)	*id_clana
	ime
	prezime
	grad
	postanski_broj
	adresa_stanovanja
	telefonski_broj

Ostaje nam još tabela izdavanje. Cilj tabele izdavanje je da čuva podatke o tome koja je knjiga izdata, kojem članu i koji radnik, tj. bibliotekar ju je izdao. Njeni atributi su jednostavni: id_izdavanje, datum_iznajmljivanja, datum_vracanja. S obzirom da čuva informacije o knjizi, članu i radniku, ova tabela povezuje te tri tabele preko njihovih primarnih ključeva (koji su u tabeli izdavanje strain ključevi). Znači moramo dodati još: id_knjige, id_clana i id_radnika.

Tako da bi entitet tabele „izdavanje” izgledao ovako:

izdavanje	
PK(primarni ključ)	*id_izdavanja
	datum_iznajmljivanja
	datum_vracanja
FK(strain ključ)	id_knjige
FK(strain ključ)	id_radnika
FK(strain ključ)	id_clana

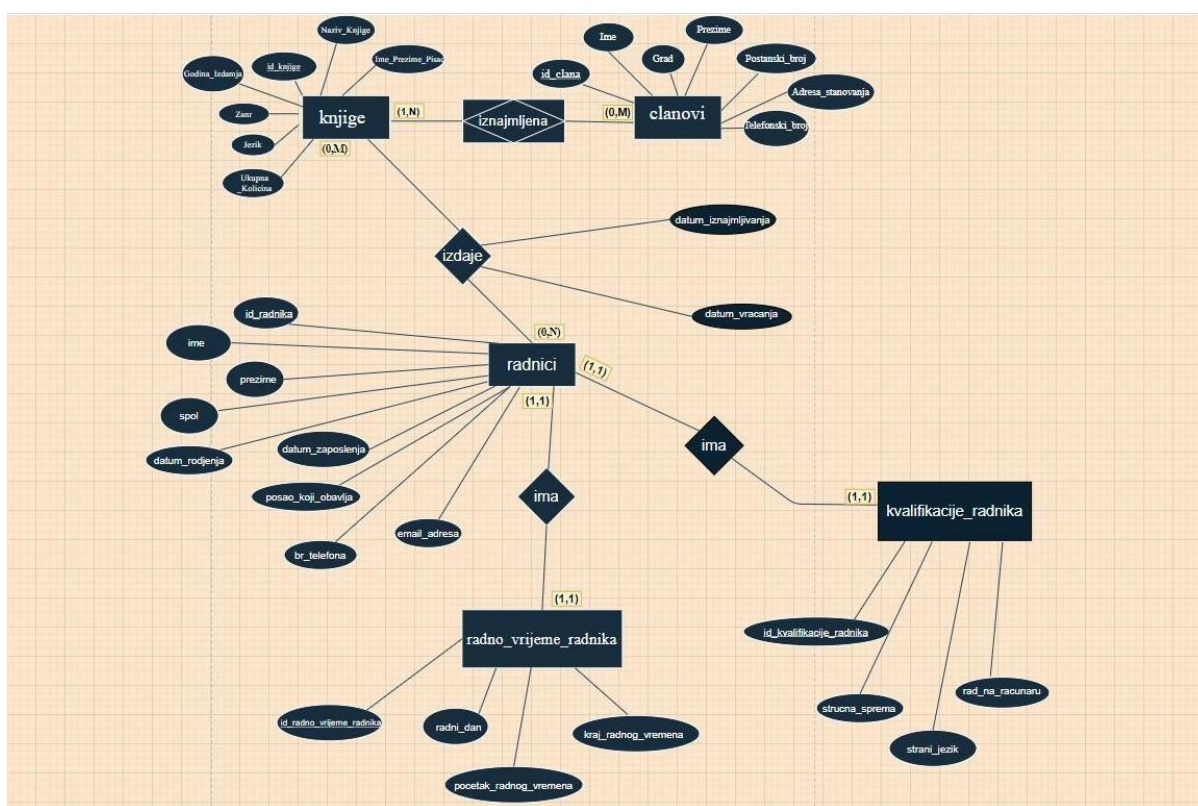
Možemo još ubaciti tabele kvalifikacije_radnika i radno_vrijeme_radnika. Te dvije tabele će nam, pored informacija o kvalifikacijama i radnom vremenu, čuvati strain ključ id_radnika, kako bi bile povezane sa tabelom radnici.

Atributi tabele kvalifikacije_radnika su: id_kvalifikacije_radnika, strucna_sprema, poznavanje_stranog_jezika, poznavanje_rada_na_racunar, id_radnika. Atributi tabele radno_vrijeme_radnika su: id_radno_vrijeme_radnika, radni_dan, pocetak_radnog_vremena, kraj_radnog_vremena, id_radnika.

ER DIJAGRAM

Sa ovim smo završili sadržaj tabela. Prije fizičkog kreiranja baze, bilo bi dobro prvo napraviti ER dijagram(entity relation diagram). ER dijagram bi nam još bliže pojasnio veze između tabele. U ovom slučaju nemamo puno tabela (za sada), ali je dobra praksa, tako da ćemo to uraditi.

Za kreiranje ER dijagrama postoje pravila označavanja entiteta, atributa i veza. Entiteti predstavljaju objekat, tj. to su sami nazivi tabela, atributi predstavljaju njihov opis, tj. kolone koje se nalaze u tim tabelama, dok veze predstavljaju veze između tih tabela (preko ključeva). Entiteti se obilježavaju pravougaonikom, atributi elipsom, a veze rombom.

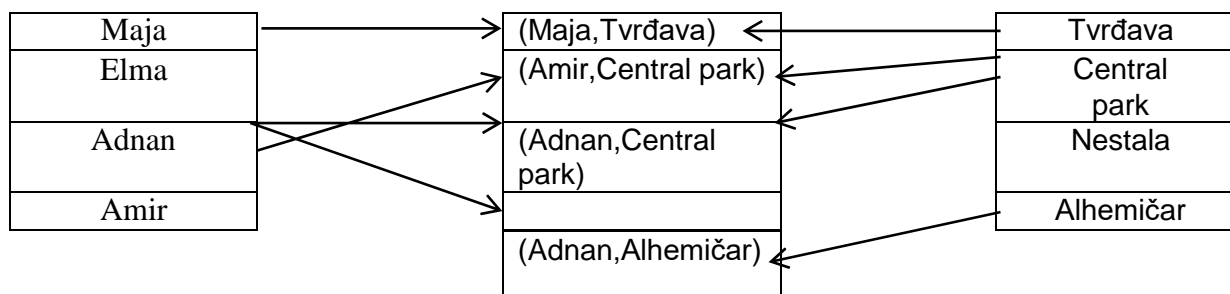


Slika 1. ER dijagram baze

Na ovom dijagramu je prikazan sadržaj baze biblioteka i veze između tabela. Da bi bilo jasnije šta se tačno nalazi na dijagramu, možemo objasniti vezu između radnika i knjiga, jer smo entitete i njihove attribute već opisali. Veza između radnika i knjiga je N:M (višena prema više):



Zamislamo sad da imamo sljedeću situaciju:



Da pojasnimo: ne mora svaki radnik izdati knjigu (s obzirom da nisu svi radnici bibliotekari), ali jedan radnik može izdati više knjiga. Tako da u ovom slučaju imamo da Elma nije izdala niti jednu knjigu, ali je Adnan izdala dvije. Za tabelu knjige: ne mora svaka knjiga biti izdata, ali jedna knjiga (tj. knjiga pod nazivom Central park), može biti izdata više članova, jer imamo više primjeraka te knjige.

Fizičko kreiranje baze

Za kreiranje baze u SQL koristimo naredbu **CREATE DATABASE** database_name. Tako da za kreiranje baze biblioteka imamo: **create database biblioteka;**. Nakon što smo kreirali bazu, da bismo radili u toj bazi moramo navesti komandu: **use biblioteka;**.

Kreiranje tabela

Pri kreiranju tabela, prije svega, moramo paziti na tipove podataka u tabeli. Za tabele knjige, radnici, članovi i izdavanje imamo sljedeće tipove podataka:

knjige		
Tip ključa	Naziv kolone	Tip podatka
PK(primarni ključ)	*id_knjige	Int(11)
	naziv_knjige	Varchar(30)
	autor	Varchar(30)
	godina_izdanja	Int(4)
	zanr	Varchar(20)
	jezik	Varchar(20)
	ukupna_kolicina	Int(11)
	broj_zaduzenih	Int(11)
	broj_citaone	Int(11)
	broj_police	Int(11)

radnici		
Tip ključa	Naziv kolone	Tip podataka
PK(primarni ključ)	*id_radnika	Int(11)
	ime	Varchar(20)
	prezime	Varchar(20)
	spol	Int(11)
	datum_rođenja	Date
	datum_zasposlenja	Date
	posao_koji_obavlja	Varchar(20)
	email_adresa	Varchar(50)
	broj_telefona	Varchar(20)

clanovi		
Tip ključa	Naziv kolone	Tip podataka
PK(primarni ključ)	*id_clana	Int(11)
	ime	Varchar(15)
	prezime	Varchar(20)
	grad	Varchar(20)
	postanski_broj	Int(5)
	adresa_stanovanja	Varchar(30)
	telefonski_broj	Varchar(20)

Tip podatka i veličina stranog ključa se mora podudarati sa tipom i veličinom podatka primarnog ključa, da bismo mogli ostvariti relaciju između njih.

izdavanje		
Tip ključa	Naziv kolone	Tip podatka
PK(primarni ključ)	*id_izdavanja	Int(11)
	datum_iznajmljivanja	Date
	datum_vracanja	Date
FK(strain ključ)	id_knjige	Int(11)
FK(strain ključ)	id_radnika	Int(11)
FK(strain ključ)	id_clana	Int(11)

Za kreiranje tabele u SQL koristimo naredbu `CREATE TABLE table_name(entity_1 attribute_1, attribute_2..., entity_2 attribute_1, attribute_2...);`. Uraditi ćemo primjer za kreiranje tabele `clanovi`, za ostale tabele radimo analogno.

Primjer:

```
CREATE TABLE clanovi(id_clana int not null auto increment primary key, ime varchar(15), prezime varchar(20), grad varchar(20) not null, postanski_broj int, adresa_stanovanja varchar(30), telefonski_broj varchar(20) not null);
```


Unos podataka u bazu

Nakon što smo završili kreiranje tabela i provjerili da li su svi tipovi podataka uredni, možemo početi sa unosom podataka. Unos podataka vrši se pomoću naredbe: `INSERT INTO table_name VALUES("value_1", "value_2", "value_3"....);`

Sada ćemo prikazati unos podataka u tabelu u našoj bazi. Koristimo naredbu:

```
INSERT INTO clanovi VALUES (NULL, "Meho", "Mehić", "Sarajevo", "71000",
"Olimpijska 20", "61342165"), (NULL, "Mirnesa", "Nuhović", "Sarajevo", "71000",
"Kralja Tvrtka 4", "62454845"), (NULL, "Emina", "Efendić", "Sarajevo", "71000",
"Kolodvorska 5", "63458124");
```

Sada u tabeli imamo sljedeće:

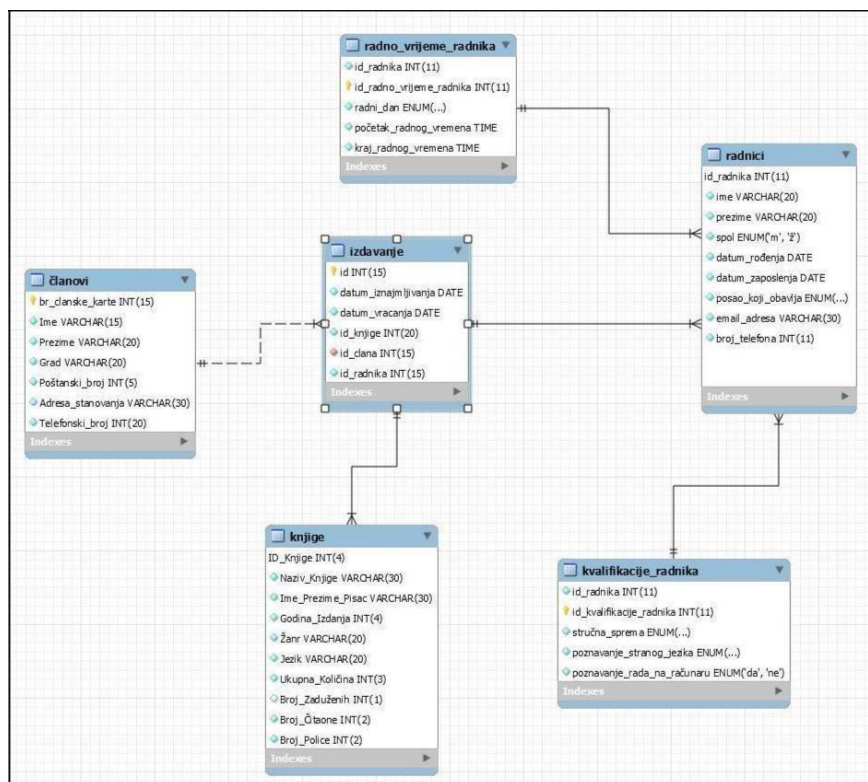
id_clana	ime	prezime	grad	postanski_broj	adresa_stanovanja	telefonski_broj
1	Meho	Mehić	Sarajevo	71000	Olimpijska 20	61342165
2	Mirnesa	Nuhović	Sarajevo	71000	Kralja Tvrtka 4	62454845
3	Emina	Efendić	Sarajevo	71000	Kolodvorska 5	63458124

Slika 2. Prikaz umetnutih podataka u tabelu clanovi

ERD model

Na slici ispod imamo ERD model naše baze koji je napravljen u Workbench-u. Na modelu su prikazane veze između 6 tabela od kojih se sastoji naša baza podataka :

- radnici
- radno_vrijeme_radnika
- kvalifikacije_radnika
- članovi
- knjige
- izdavanje



Slika 3. ERD model baze

Prikaz sadržaja tabele

SELECT iskaz

Kako smo dobili ovaj sadržaj tabele? Za ispis podataka iz tabele koristimo upit `SELECT * FROM table_name`. Za prikaz slike 2. Koristili smo upit `SELECT * FROM clanovi`. Nakon što smo unijeli podatke u ostale tabele, možemo raditi razne select upite nad njima.

Primjer: Napisati upit koji prikazuje sve knjige koje su izdate 1966. godine.

To ćemo uraditi na sljedeći način: `SELECT * FROM knjige WHERE godina_izdanja="1966"`; Dobijemo sljedeće:

id_knjige	naziv_knjige	autor	godina_izdanja	zanzr	jezik	ukupna_kolicina	broj_zaduzenih	broj_citaone	broj_police
1	Uбивство u "Orient Expressu	Agatha Christie	1966	Kriminalistički	Engleski	99	3	1	1
5	Derviš i smrt	Meša Selimović	1966	Novela	Bosanski	200	2	2	5
10	Razgovori na Nilu	Nagib Mahfouz	1966	Psihološki	Arapski	30	3	1	6
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Slika 4. Ispis knjiga čija je godina izdanja 1966.

AS operator

AS koristimo za preimenovanje tabele. Recimo da imamo tabelu . Na primjer, u tabeli radno_vrijeme_radnika imamo kolone pocetak_radnog_vremena i kraj_radnog_vremena.

pocetak_radnog_vremena	kraj_radnog_vremena
08:00:00	16:00:00
08:00:00	16:00:00
08:00:00	16:00:00
16:00:00	18:00:00
08:00:00	16:00:00
08:00:00	16:00:00
16:00:00	18:00:00
08:00:00	16:00:00
08:00:00	16:00:00
08:00:00	16:00:00

Slika 4. Prikaz kolona pocetak_radnog_vremena i kraj_radnog_vremena

S obzirom da se one nalaze u tabeli radno_vrijeme, njih možemo prikazati kao pocetak I kraj, jer se zna da se odnosi na radno vrijeme. To možemo uraditi na sljedeći način:

```
select pocetak_radnog_vremena as pocetak,
       kraj_radnog_vremena as kraj
from radno_vrijeme_radnika;
```

Sada imamo ovakav ispis:

pocetak	kraj
08:00:00	16:00:00
08:00:00	16:00:00
08:00:00	16:00:00
16:00:00	18:00:00
08:00:00	16:00:00
08:00:00	16:00:00
16:00:00	18:00:00
08:00:00	16:00:00
08:00:00	16:00:00
08:00:00	16:00:00

Slika 5. Prikaz kolona pocetak_radnog_vremena I kraj_radnog_vremena kao pocetak I kraj

ORDER BY klauzula

Order by upit koristimo kada ono što smo odabrali za prikaz želimo sortirati po nekom redoslijedu ili kriteriju.

Primjer:

```
SELECT * FROM knjige WHERE godina_izdanja="1966" order by naziv_knjige;
```

LIKE operator

Like koristimo kada nismo sigurni za tačan sadržaj ćelije. Na primjer: ako želimo da izlistamo sve članove čije ime počinje slovom A.

```
SELECT * FROM clanovi WHERE ime like "A%".
```

IS NULL & IS NOT NULL operator

Is null (is not null) koristimo kada želimo da izlistamo sve ćelije koje su prazna ili koje nisu prazne, tj. gdje je unešena vrijednosti. Na primjer, u našoj tabeli, ako želimo izlistati sve članove koji nemaju unesen grad:

```
SELECT * FROM clanovi WHERE grad is null;
```

id_clana	ime	prezime	grad	postanski_broj	adresa_stanovanja	telefonski_broj
31	Neira	Pipić	NULL	NULL	Kralja Tvrtka 8	61248871

Slika 6. Prikaz članova koji nisu
unijeli grad u kojem žive

Agregatne funkcije

U agregatne funkcije spadaju: AVG (prosjek), MIN (ispisuje minimalnu vrijednost), MAX (ispisuje maksimalnu vrijednost), SUM (zbir vrijednosti redova koje odaberemo) i COUNT (broji redove).

Primjer: želimo da prebrojimo sve članove koji su iz Sarajeva. To ćemo uraditi na sljedeći način:

```
Select count(id_clana),ime from clanovi where="Sarajevo" group by id_clana;
```

Join-ovi

Join služi za uzimanje podataka iz više tabela, koje su povezane preko ključeva. Tipovi joina su: inner join, left join, right join, selft join i cross join.

Inner join

Sintaksa inner join-a:

```
SELECT  
  select_lis  
t FROM  
t1  
INNER JOIN t2 ON  
join_condition1 INNER JOIN t3
```

Primjer1 za našu tabelu:

```
select r1.id_radnika,  
       r1.prezime,  
       r1.ime,  
       r1.posao_koji_obavlja,  
       r2.radni_dan,  
       r2.pocetak_radnog_vremena,  
       r2.kraj_radnog_vremena  
from radnici r1  
innerjoin radno_vrijeme_radnika r2  
on r1.id_radnika=r2.id_radnika  
order by r1.prezime;
```

Ovim upitom dobili smo sljedeće:

id_radnika	prezime	ime	posao_koji_obavlja	radni_dan	pocetak_radnog_vremena	kraj_radnog_vremena
4	Ajselović	Ajsel	higijeničar/ka	ponedjeljak, srijeda i petak	08:00:00	16:00:00
6	Dajić	Dajana	sekretar/ka	ponedjeljak, srijeda i petak	08:00:00	16:00:00
7	Danijelović	Danijel	higijeničar/ka	ponedjeljak, srijeda i petak	08:00:00	16:00:00
8	Emić	Naida	bibliotekar/ka	utorak, četvrtak i subota	08:00:00	16:00:00
5	Eminović	Emina	bibliotekar/ka	ponedjeljak, srijeda i petak	08:00:00	16:00:00
9	Farić	Faris	sekretar/ka	utorak, četvrtak i subota	08:00:00	16:00:00
2	Knjiga	Mahir	bibliotekar/ka	utorak, četvrtak i subota	08:00:00	16:00:00
10	Nejrić	Nejra	bibliotekar/ka	utorak, četvrtak i subota	08:00:00	16:00:00
3	Samirović	Samir	sekretar/ka	utorak, četvrtak i subota	16:00:00	18:00:00
1	Sekretarović	Maja	sekretar/ka	ponedjeljak, srijeda i petak	16:00:00	18:00:00

Primjer2 za našu tabelu:

```
SELECT i.id_clana AS 'ID ČLANA',
       c.ime AS 'IME ČLANA',
       k.naziv_knjige AS 'NAZIV KNJIGE',
       k.autor AS 'AUTOR',
       i.datum_iznajmljivanja AS ' DATUM IZNAJMLJIVANJA KNJIGE'
FROM izdavanje i
INNER JOIN knjige k
ON i.id_knjige=k.id_knjige
INNER JOIN clanovi c
ON i.id_clana=c.id_clana
WHERE k.naziv_knjige LIKE "A%" OR k.naziv_knjige LIKE "R%"
ORDER BY i.datum_iznajmljivanja;
```

Prikaz upita:

ID ČLANA	IME ČLANA	NAZIV KNJIGE	AUTOR	DATUM IZNAJMLJIVANJA KNJIGE
3	Emina	Ringišpil	Jelena Bačić Alimpić	2020-11-01
8	Mirnesa	Ringišpil	Jelena Bačić Alimpić	2020-11-05
13	Dženan	Anđeli i demoni	Dan Brown	2020-11-07
20	Amina	Razum i osjećajiu	Jane Austin	2020-11-10
21	Saira	Ana Karenjina	Lav Tolstoy	2020-11-10

Slika 7. Prikaz upita join

Left join

Sintaksa left join-a:

```
SELECT
select_list
FROM
t1
LEFT JOIN t2 ON
join_condition;
```

Primjer left join-a za našu tabelu:

```
Select c.ime,
       c.prezime,
       c.id_clan
a,
       i.datum_iznajmljivanja
a from clanovi c
left join izdavanja i
```

Ovim upitom smo dobili sljedeće:

ime	prezime	id_clana	datum_iznajmljivanja
Meho	Mehić	1	2020-11-01
Mirnesa	Nuhović	2	2020-11-01
Emina	Efendić	3	2020-11-01
Meho	Mehović	4	2020-11-02
Meho	Mehović	4	2020-11-02
Mersad	Homo	5	2020-11-03
Ema	Medonjić	6	2020-11-04
Mirnes	Mirvić	7	2020-11-05
Mirnes	Mirvić	7	2020-11-05
Mirnesa	Mivić	8	2020-11-05
Mirnesa	Mivić	8	2020-11-05
Nermina	Silajdžić	9	2020-11-06
Dženana	Lovrić	10	NULL
Tea	Dragan...	11	2020-11-07
Eman	Srna	12	2020-11-07
Eman	Srna	12	2020-11-07
Dženana	Pindžo	13	2020-11-07

Slika 8. Prikaz upita left

Pogledi (Views)

Pisati svaki put jedan te isti kod može biti naporno i trajati dugo. Zbog toga koristimo pogleda. Recimo da imamo neki upit koji nam često treba, kao što je primjer za inner join (slika 7.). Pomoću pogleda možemo sebi olakšati. Sitnaks kreiranja pogleda je:

```
CREATE VIEW view_name AS select_clause;
```

Sada ćemo kreirati pogled za primjer joina:

```
create view radnici_info
as
select r1.id_radnika,
       r1.prezime,
       r1.ime,
       r1.posao_koji_obavlja,
       r2.radni_dan,
       r2.pocetak_radnog_vremena,
       r2.kraj_radnog_vremena
from radnici r1
inner join radno_vrijeme_radnika r2
on r1.id_radnika=r2.id_radnika
order by r1.prezime;
```

Sada da bismo dobili ispis kao u prethodnom primjeru za inner join, potrebno je samo da uradimo sljedeće:

```
Select * from radnici_info;
```

Možemo još kreirati upit izdavanje_info, koji bi nam izlistao neke osnovne informacije o izdavanju, kao što su: naziv_knjige, autora, ukupnu_kolicinu, datum_iznajmljivanja, datum_vracanja i id_clana.

To ćemo uraditi na sljedeći način:

```
create view izdavanje_info
as
select k.naziv_knjige,
       k.autor,
       k.ukupna_kolicina as broj_knjiga,
       i.datum_iznajmljivanja,
       i.datum_vracanja,
       c.id_clana
from izdavanje i
inner join knjige k
on i.id_knjige=k.id_knjige
inner join clanovi c
on i.id_clana=c.id_clana;
```

Kada izvršimo upit `select * from izdavanje_info`, dobijemo sljedeće:

naziv_knjige	autor	broj_knjiga	datum_iznajmljivanja	datum_vracanja	id_clana
Ana Karenjina	Lav Tolstoy	150	2020-11-10	NULL	21
Anđeli i demoni	Dan Brown	60	2020-11-07	NULL	13
Bilježnica	Nikolas Sparks	120	2020-11-01	2020-11-05	1
Božanstvena komedija	Dante Alighieri	90	2020-11-02	2020-11-06	4
Braća Karmazovi	Fyodor Dostoevsky	80	2020-11-11	NULL	23
Braća Karmazovi	Fyodor Dostoevsky	80	2020-11-09	NULL	15
Crveni zmaj	Thomas Harris	20	2020-11-09	NULL	13
Da Vinčijev Kod	Dan Brown	60	2020-11-02	2020-11-06	4
Derviš i smrt	Meša Selimović	200	2020-11-07	NULL	13
Djevojka u vozu	Paula Hawking	65	2020-11-11	NULL	24
Evgenij Onjegin	Aleksandr Pushkin	99	2020-11-01	2020-11-06	2
Evgenij Onjegin	Aleksandr Pushkin	99	2020-11-11	NULL	21
Germinal	Emile Zola	110	2020-11-10	NULL	17
Gospodar prstenova	J.J.R Tolkien	40	2020-11-07	NULL	12
Gromovo đule	Skender Kulenović	250	2020-11-07	NULL	12
Gromovo đule	Skender Kulenović	250	2020-11-07	NULL	11
Hamlet	William Shakespeare	77	2020-11-06	NULL	9
Harry Potter (paket)	J.K.Rowling	100	2020-11-05	NULL	7
Idiot	Fyodor Dostoevsky	70	2020-11-09	NULL	16
Isijavanje	Stephen King	90	2020-11-03	2020-11-07	5
Isijavanje	Stephen King	90	2020-11-10	NULL	19
Jadnici	Victor Hugo	67	2020-11-09	NULL	16
Jadnici	Victor Hugo	67	2020-11-11	NULL	22
Kad jaganjci utihnu	Thomas Harris	100	2020-11-10	NULL	25

Slika 9. Prikaz ispisa pogleda
izdavanje_info

Procedure (Procedures)

Procedure predstavljaju potprogram koji je pohranjen u samoj bazi. U ovoj bazi koristili smo ih za osvježavanje podataka za tabelu izdavanje, npr. kada član iznajmi knjigu, ne moramo ručno dodavati tu knjigu u tabelu izdavanje i mijenjati količinu knjiga koja nam je ostala, dovoljno je samo da pozovemo procedure.

Procedure koje smo napravili su izdaj_knjigu i vrati_knjigu.

Kreiranje procedure

```
DELIMITER //
CREATE PROCEDURE izdaj_knjigu (
in id_knjige1 int,
in id_clana1 int,
in id_radnika1 int,
in datum_izdavanja1 date
)
BEGIN
declare ukupna_kolicina1 int default 0;
set ukupna_kolicina1 = (select ukupna_kolicina from knjige where id_knjige=id_knjige1
);
insert into izdavanje( datum_iznajmljivanja, id_knjige, id_clana, id_radnika) values (
datum_izdavanja1, id_knjige1, id_clana1, id_radnika1 );
update knjige
set ukupna_kolicina=(ukupna_kolicina1-1)
where id_knjige = id_knjige1;
END //
DELIMITER ;
```

Datim upitom smo kreirali procedure izdaj_knjigu. Sada ako želimo izdati neku knjigu uradimo sljedeće:

```
5 • call izdaj_knjigu ('4', '2', '2', '2020-01-01');
```

Analogno za vraćanje knjige:

```
DELIMITER //
CREATE PROCEDURE vrati_knjigu (
in naziv_knjige1 varchar(40),
in id_clana1 int,
in datum_vracanja1 date,
in datum_izdavanja1 date
)
BEGIN
declare ukupna_kolicina1 int default 0;
declare id_knjige1 int default 0;
set id_knjige1 = (select id_knjige from knjige where naziv_knjige = naziv_knjige1);
set ukupna_kolicina1 = (select ukupna_kolicina from knjige where id_knjige = id_knjige1);
update knjige
set ukupna_kolicina=(ukupna_kolicina1 + 1)
where id_knjige=id_knjige1;
update izdavanje
set datum_vracanja = datum_vracanja1
where id_knjige = id_knjige1 and id_clana = id_clana1 and datum_iznajmljivanja =
datum_izdavanja1;
END //
DELIMITER ;
```

```
10 • call vrati_knjigu ('Sofijin izbor', '2', '2020-02-02', '2020-01-01');
```

Trigeri

Trigeri predstavljaju kod koji se izvršava kada se pokrene (triggers) neke upit za umetanje ili ažuriranje podataka. U našoj bazi napravili smo trigger koji se pokreće prije umetanja podataka.

```
□ DELIMITER //
□ CREATE TRIGGER bosanski
□ before insert
□ on knjige FOR EACH ROW
□ BEGIN
□     if (new.jezik = ' Bosanski ') then
□     set new.naziv_knjige = upper (new.naziv _knjige );
□     end if;
□ END //
□ DELIMITER;
```

Ovaj trigger se odnosi na tabelu knjige i on pretvara ime knjige u velika, svaki put kada neko unese da je jezik knjige bosanski.

Normalizacija baze podataka

Normalizacija je postupak uklanjanja anomalija i suvišnosti iz dizajna baze podataka. Postoje tri posebne vrste anomalija koje se mogu pojaviti u oblikovanju baze podataka:

- Anomalije umetanja
- Anomalije ažuriranja
- Anomalije brisanja

Tokom godina stručnjaci za baze podataka razvili su niz "normalnih oblika". Svaka forma je dizajnirana kako bi se eliminirala jedna ili više ovih anomalija. Uobičajeni oblici su sljedeći:

- Prva normalna forma
- Druga normalna forma
- Treća normalna forma

Prva normalna forma

Prva normalna forma [1NF] uključuje uklanjanje ponavljajućih grupa ili nizova. Svaki atribut bi trebao da sadrži samo jednu vrijednost jednog tipa. To znači nekoliko stvari. Kao prvo, sve vrijednosti pod atributom trebaju biti približno ista stvar. Atribut nazvan "email_adresa", na primjer, trebao bi sadržati samo e-mail, bez brojeva telefona ili slično. Drugo značenje je da svaka vrijednost pohranjena pod atribut treba biti jedna vrijednost, a ne niz ili popis vrijednosti. Bilo bi pogrešno, na primjer, za pohranu dvije ili tri e-adrese iste osobe odvojene zarezima.

Entitet je u prvom normalnom obliku ako

- Svaki atribut predstavlja samo jednu vrijednost.
- Ne postoje grupe ili nizovi koji se ponavljaju.
- Svaki je red jedinstven.

Pogledajmo naše tabele:

id_knjige	naziv_knjige	autor	godina_izdanja	zanr	jezik	ukupna_kolicina	broj_zaduzenih	broj_citaone	broj_police
1	Ubistvo u "Orient Expressu"	Agatha Christie	1966	Kriminalistički	Engleski	100	3	1	1
2	Ožalošćena porodica	Branislav Nušić	1934	Komedija	Srpski	50	1	2	2
3	Ježeva kućica	Branko Ćopić	1949	Dječja literatura	Srpski	200	2	3	3
4	Sofijin izbor	William Styron	1979	Drama	Engleski	100	2	1	4
5	Derviš i smrt	Meša Selimović	1966	Novela	Bosanski	200	2	2	5
6	Četrdeset pravila ljubavi	Elif Shafak	2009	Novela	Engleski	100	1	2	5
7	Harry Potter (paket)	J.K.Rowling	1997	Dječja literatura	Engleski	100	1	3	3
8	Suludo bogati Aziji	Kevin Kvan	2019	Komedija	Engleski	80	1	2	2
9	Zvonar crkve Notre-Dame	Victor Hugo	1831	Drama	Francuski	75	3	1	4
10	Razgovori na Nilu	Nagib Mahfouz	1966	Psihološki	Arapski	30	3	1	6

Slika 9. Tabela knjige

id_radnika	ime	prezime	spol	datum_rođenja	datum_zaposlenja	posao_koji_obavlja	email_adresa	broj_telefona
1	Maja	Sekretarović	ž	1987-03-25	2019-07-20	sekretar/ka	amila.sekretarovic@gmail.com	62546378
2	Mahir	Knjiga	m	1985-09-12	2018-05-05	bibliotekar/ka	knjiga_mahir@hotmail.com	65382901
3	Samir	Samirović	m	1990-12-09	2019-12-12	sekretar/ka	samir.ovic@gmail.com	63819767
4	Ajsel	Ajselović	ž	1989-05-22	2018-03-03	higijeničar/ka	ajsel.ovic@gmail.com	64928376
5	Emina	Eminović	ž	1990-07-06	2018-10-10	bibliotekar/ka	eminovic89@hotmail.com	62198735
6	Dajana	Dajić	ž	1977-03-12	2013-04-10	sekretar/ka	dajana.dajic@gmail.com	603101922
7	Danijel	Danijelović	m	1985-09-23	2015-05-14	higijeničar/ka	danijel.dani@hotmail.com	64938762
8	Naida	Emić	ž	1989-08-20	2017-07-10	bibliotekar/ka	emic_naida@gmail.com	61029874
9	Faris	Farić	m	1980-03-03	2014-05-20	sekretar/ka	faris.faric@gmail.com	64738294
10	Nejra	Nejrić	ž	1976-07-19	2010-05-06	bibliotekar/ka	nejric76@hotmail.com	61928765

Slika 10. Tabela radnici

id_clana	ime	prezime	grad	postanski_broj	adresa_stanovanja	telefonski_broj
1	Meho	Mehić	Sarajevo	71000	Olimpijska 20	61342165
2	Mirnesa	Nuhović	Sarajevo	71000	Kralja Tvrtka 4	62454845
3	Emina	Efendić	Sarajevo	71000	Kolodvorska 5	63458124
4	Meho	Mehović	Mostar	88000	Zagrebacka 18	61526372
5	Mersad	Homo	Mostar	88000	Zagrebačka 9	62567487
6	Ema	Medonjić	Banja Luka	78000	12 beba	63567876
7	Mirnes	Mirvić	Travnik	75000	Kalinska	64154245
8	Mirnesa	Mivić	Travnik	75000	Kalinska	64152345
9	Nermina	Silajdžić	Sarajevo	71000	Trg Barcelone 6	61736467
10	Dženana	Lovrić	Sarajevo	71000	Valtera Perića 8	61767876

Slika 11. Tabela clanovi

id_izdavanja	datum_iznajmljivanja	datum_vracanja	id_knjige	id_clana	id_radnika
1	2020-11-01	2020-11-05	16	1	2
2	2020-11-01	2020-11-06	39	2	2
3	2020-11-01	2020-11-06	17	3	2
4	2020-11-02	2020-11-06	29	4	5
5	2020-11-02	2020-11-06	36	4	5
6	2020-11-03	2020-11-07	26	5	8
7	2020-11-04	2020-11-08	19	6	5
8	2020-11-05	NULL	7	7	2
9	2020-11-05	NULL	2	7	2
10	2020-11-05	NULL	11	8	8

Slika 12. Tabela izdavanje

id_kvalifikacije_radnika	strucna_sprema	poznavanje_stranog_jezika	poznavanje_rada_na_racunaru	id_radnika
1	viša školska sprema	engleski, njemački i francuski	da	10
2	visoka stručna sprema	engleski, njemački i turski	da	2
3	viša školska sprema	engleski i njemački	da	7
4	srednja stručna sprema	engleski	ne	1
5	visoka stručna sprema	engleski i turski	da	9
6	visoka stručna sprema	engleski i njemački	da	8
7	nekvalifikovani radnik	engleski	ne	4
8	visoka stručna sprema	engleski i švedski	da	6
9	viša školska sprema	njemački	da	5
10	visoka stručna sprema	engleski, njemački i švedski	da	3

Slika 13. Tabela kvalifikacije_radnika

id_radno_vrijeme_radnika	radni_dan	pocetak_radnog_vremena	kraj_radnog_vremena	id_radnika
1	ponedjeljak, srijeda i petak	08:00:00	16:00:00	4
2	utorak, četvrtak i subota	08:00:00	16:00:00	10
3	ponedjeljak, srijeda i petak	08:00:00	16:00:00	6
4	utorak, četvrtak i subota	16:00:00	18:00:00	3
5	ponedjeljak, srijeda i petak	08:00:00	16:00:00	5
6	utorak, četvrtak i subota	08:00:00	16:00:00	2
7	ponedjeljak, srijeda i petak	16:00:00	18:00:00	1
8	utorak, četvrtak i subota	08:00:00	16:00:00	9
9	ponedjeljak, srijeda i petak	08:00:00	16:00:00	7
10	utorak, četvrtak i subota	08:00:00	16:00:00	8

Slika 14. Tabela radno_vrijeme_radnika

Vidimo da svaka tabela ima kolonu koja predstavlja identifikaciju, tj. id, i da sve ostale kolone zavise od id-a, da nema ponavljanja atributa i da je svaki red jedinstven, što znači da su tabele u prvoj normalnoj formi.

Druga normalna forma

Druga normalna forma uklanja funkcionalnu zavisnost. Recimo, u tabeli kvalifikacije_radnika (Slika 13.) imamo kolonu strucna_sprema u kojoj imamo opširnije predmete. Da bismo ovu tabelu prilagodili drugoj normalnoj formi, razbit ćemo je na zasebne cjeline, tj. strucnu sprema, poznavanje stranog jezika i poznavanje rada na računaru ćemo prebaciti u novu tabelu.

	id	strucna_sprema
▶	1	visoka strucna sprema
	2	visa skolska sprema
	3	srednja strucna sprema
	4	nekvalifikovani radnik

Slika 15. Tabela

id	poznavanje_rada_na_racunaru
1	'da'
2	'ne'

Slika 16. Table
poznavanje_rada_na_racunaru

id	poznavanje_stranog_jezika
1	engleski
2	njemacki
3	turski
4	francuski
5	svedski
6	engleski i njemacki
7	engleski i turski
8	engleski i francuski
9	engleski i svedski

Slika 17. Tabela
poznavanje_stranog_jezika

Sada naša tabela kvalifikacije_radnika izgleda ovako:

id_radnika	id_kvalifikacije_radnika	strucna_sprema	poznavanje_stranog_jezika	poznavanje_rada_na_računaru
10	1	2	5	1
2	2	1	6	1
7	3	2	4	1
1	4	3	1	2
9	5	1	6	1
8	6	1	8	1
4	7	4	6	2
6	8	1	2	1
5	9	2	3	1
3	10	1	7	1

Slika 18. Tabela radno_vrijeme_radnika u 2NF

Također, i tabelu knjige (Slika 9.) možemo normalizovati. Tj., napraviti ćemo novu tabelu autor_knjige, koja će držati informacije o autoru, jezik_knjige i zanr_knjige, na isti način kao što smo uradili za tabelu kvalifikacije.

Nakon normalizovanja, tabela knjige izgleda odavko:

id_knjige	naziv_knjige	autor	godina_izdanja	zanr	jezik	ukupna_kolicina	broj_zaduzenih	broj_citaone	broj_police
1	Ubistvo u "Orient Expressu"	1	1966	1	1	100	3	1	1
2	Ožalošćena porodica	2	1934	2	2	50	1	2	2
3	Ježeva kućica	3	1949	3	2	200	2	3	3
4	Sofjin izbor	4	1979	4	1	100	2	1	4
5	Derviš i smrt	5	1966	5	3	200	2	2	5
6	Četrdeset pravila ljubavi	6	2009	5	1	100	1	2	5
7	Harry Potter (paket)	7	1997	3	1	100	1	3	3
8	Suludo bogati Azijci	8	2019	2	1	80	1	2	2
9	Zvonar crkve Notre-Dame	9	1831	4	4	75	3	1	4
10	Razgovori na Nilu	10	1966	6	5	30	3	1	6

Slika 19. Tabele knjige u 2NF

Isto ćemo uraditi i za tabelu clanovi za kolonu grad.

id	ime_grada
1	Sarajevo
2	Tuzla
3	Banja Luka
4	Mostar
5	Travnik
6	Jajce
7	Gradiška

Slika 20. Tabela grad

Sada tabela clanovi izgleda ovako:

id_clana	ime	prezime	grad	postanski_broj	adresa_stanovanja	telefonski_broj
1	Meho	Mehić	1	71000	Olimpijska 20	61342165
2	Mirnesa	Nuhović	1	71000	Kralja Tvrtka 4	62454845
3	Emina	Efendić	1	71000	Kolodvorska 5	63458124
4	Meho	Mehović	4	88000	Zagrebačka 18	61526372
5	Mersad	Homo	4	88000	Zagrebačka 9	62567487
6	Ema	Medonjić	3	78000	12 beba	63567876
7	Mirnes	Mirvić	5	75000	Kalinska	64154245
8	Mirnesa	Mivić	5	75000	Kalinska	64152345
9	Nermina	Silajdžić	1	71000	Trg Barcelone 6	61736467
10	Dženana	Lovrić	1	71000	Valtera Perića 8	61767876

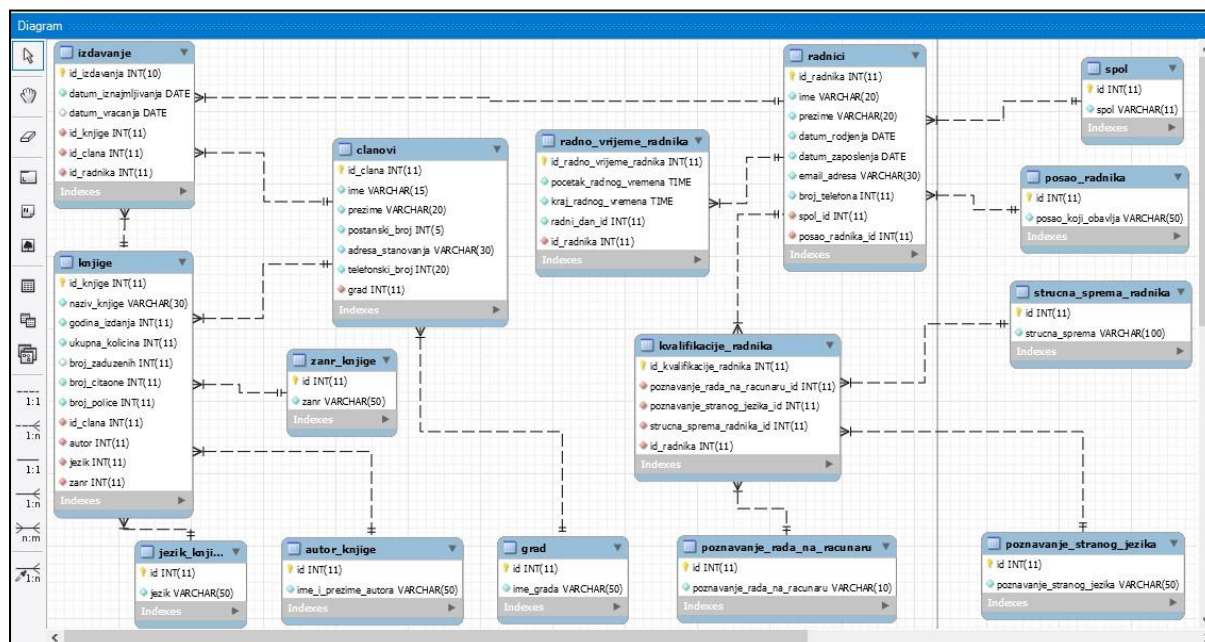
Slika 21. Tabela clanovi u 2NF

Za tabelu radno_vrijeme_radnika ubacili smo nove tabele radni_dan.

id_radno_vrijeme_radnika	radni_dan	pocetak_radnog_vremena	kraj_radnog_vremena	id_radnika
1	1	08:00:00	16:00:00	4
2	2	08:00:00	16:00:00	10
3	1	08:00:00	16:00:00	6
4	2	16:00:00	18:00:00	3
5	1	08:00:00	16:00:00	5
6	2	08:00:00	16:00:00	2
7	1	16:00:00	18:00:00	1
8	2	08:00:00	16:00:00	9
9	1	08:00:00	16:00:00	7
10	2	08:00:00	16:00:00	8

Slika 22. Tabela
radno_vrijeme_radnika u 2NF

ERD model nakon normalizacije



Slika 23. ER model nakon normalizacije

Literatura

1. Steve Conger: Hands-on database