

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
V A R A Ž D I N

Elena Kržina

**IZRADA BAZE I SKLADIŠTA PODATAKA
ZA PROCES VRAĆANJA KNJIGA U
KNJIŽNICI**

PROJEKT

Varaždin, 2023.

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
V A R A Ž D I N

Elena Kržina

Matični broj: 0016135585

Studij: Baze podataka i baze znanja

IZRADA BAZE I SKLADIŠTA PODATAKA ZA PROCES VRAĆANJA
KNJIGA U KNJIŽNICI
PROJEKT

Mentor:

Mag. educ. inf. Maja Cerjan

Varaždin, svibanj 2023.

Elena Kržina

Izjava o izvornosti

Izjavljujem da je moj završni rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor/Autorica potvrdio/potvrdila prihvaćanjem odredbi u sustavu FOI-radovi

Sažetak

Projektni se zadatak bavi izradom skladišta podataka uzimajući pritom podatke iz modificiranog postojećeg seta podataka za knjige koji je povezan s vlastito generiranim podacima kako bi se napravila cijela baza posudbe knjiga u knjižnici. Iz 6 relacijskih tablica stvara se dimenzijski model ili model zvijezde za pripremu za rad u OLAP alatu za vizualnu analizu podataka. Prvo dio rada objašnjava postavljanje dimenzijskog modela u alatu MariaDB uz dodatak teorijske podloge, dok se drugi dio bavi sa samom analizom podataka u alatu Microsoft Power BI. Projekt služi za pripremu i pregled podataka u skladištima podataka te za vizualni prikaz korištenog seta podataka i pomoć u odlučivanju.

Ključne riječi: skladište podataka; baza podataka; OLAP; dimenzijski model; relacijska baza podataka; baza podataka vozila; poslovna inteligencija

Sadržaj

1. Uvod	1
2. Metode i tehnike rada	2
3. Priprema seta podataka u SUBP-u MariaDB	3
3.1. Opis domene i seta podataka.....	3
3.2. Izrada dimenzijskog modela i ETL.....	5
4. Analiza i vizualizacija podataka u alatu Microsoft Power BI.....	13
4.1. Učitavanje i povezivanje podataka u Microsoft Power BI.....	13
4.2. Prikazivanje podataka u Microsoft Power BI	14
5. Zaključak	28
Popis literature	29
Popis slika	30
Popis isječaka kôda.....	31
1. Prilog Python kod za generiranje vrijednosti	32

1. Uvod

Skladišta su podataka specijalno dizajnirane baze podataka pogodne za čitanje i učitavanje podataka te služe za detaljan, redundantan način zapisivanja podataka te njegovo kasnije čitanje i analiziranje uz pomoć nekog *business intelligence* (BI) alata ili alata poslovne inteligencije (Rabuzin, 2018). Skladišta podataka posebice su korisna zbog ETL procesa koji se izvršava nad podacima.

ETL kratica je od *extract*, *transform* i *load*, odnosno označava odabir i ekstrahiranje podataka iz različitih izvora, njihovu modifikaciju i čišćenje te učitavanje u skladište. Proces vraća čiste, korisne podatke koji su izrazito pogodni za dubinske analize i za pomoć u odlučivanju. U ovom će se radu izraditi mala baza podataka nad čijim će se podacima provesti ETL proces, te se kreira skladište podataka za specifičan proces vraćanja knjiga u knjižnicu. Cilj je stoga izraditi bazu, provesti ETL proces, kao rezultat dobiti skladište podataka, te nad podacima provesti analizu u alatu za poslovnu inteligenciju.

Rad je podijeljen na 5 glavnih dijela. Prvi je „Uvod“ u kojem je predstavljeno skladište podataka i cilj kojim se teži prilikom izrade projektnog rada. Sljedeće je poglavlje „Metode i tehnike rada“ u kojem su predstavljene metode i tehnike i alati na temelju kojih se rad piše. Sljedeće poglavlje je „Priprema seta podataka u SUBP-u MariaDB“, u kojem se opisuje način izrade baze i u kojem se podaci pripremaju za učitavanje u skladište te izrada samog skladišta podataka. Slijedi naslov „Analiza i vizualizacija podataka u Microsoft Power BI-u“ u kojem se podaci iz skladišta učitavaju u BI alat te su ovdje prikazane razne vizualizacije podataka. Na kraju se nalazi „Zaključak“ kao rezime na cijeli rad. Slijedi popis literatura, slika, isječaka koda te prilozi.

2. Metode i tehnike rada

Za izradu je projekta potrebno odabrati set podataka na kojim će se raditi, alate kojim će se izraditi ER dijagram baze podataka i skladišta podataka, sustav za upravljanje bazama podataka u kojim će se prikazati i po potrebi promijeniti podaci danog seta podataka te alat za poslovnu inteligenciju kojim će se vizualizirati i analizirati ti podaci.

Set podataka načinjen je od preuzimanja jedne tablice dataseta, generiranjem podataka online te generiranjem uz izradu skripte. Izabran je „BX-Books.csv“ iz *Data.World* na adresi <https://data.world/divyanshi/users-books-dataset> zbog velike količine podataka o knjigama. Set podataka prikazuje nazive, ISBN, autore, godinu izdavača i slike knjiga. Za potrebe izrade skladišta podataka izmijenjen je te su mu dodani dodatni stupci za detaljnije podatke o knjigama te je smanjen broj redaka tablice. Tablice članova i zaposlenika generiran je na stranici Mockaroo na <https://www.mockaroo.com/>, tablica žanrova izrađena je ručno, a podaci tablice posudba knjiga i njihovih ocjeni izrađene su i generirane samostalno u Python skripti prikazanoj u prilogima.

Za analizu i pripremu seta podataka za daljnje korištenje u alatu za poslovnu inteligenciju korišten je sustav za upravljanje bazama podataka MariaDB. MariaDB je relacijski *open source* sustav za upravljanje bazama podataka dizajniran od originalnih developera MySQL-a koji se bazira na dobrim performansama, stabilnosti i otvorenosti (MariaDB.org, 2023). Za prikaz rada s podacima korišten je i HeidiSQL, alat za prikaz i promjenu podataka nad sustavima poput MariaDB, MySQL-a, PostgreSQL-a i SQLite (Becker, n.d.). BI alat u kojim se vizualiziraju podaci je Microsoft Power BI. Specifično je korišten Microsoft Power BI Desktop.

Za izradu ERA modela korišten je *MySQL Workbench* kako bi se prikazala struktura postojećeg seta podataka. Željene strukture novog seta podataka koji će se koristiti u BI alatu prikazana je u Diagrams.net, *open source* alatu za prikazivanje dijagrama i modela. Za generiranje podataka korišten je Python programski jezik.

3. Priprema seta podataka u SUBP-u MariaDB

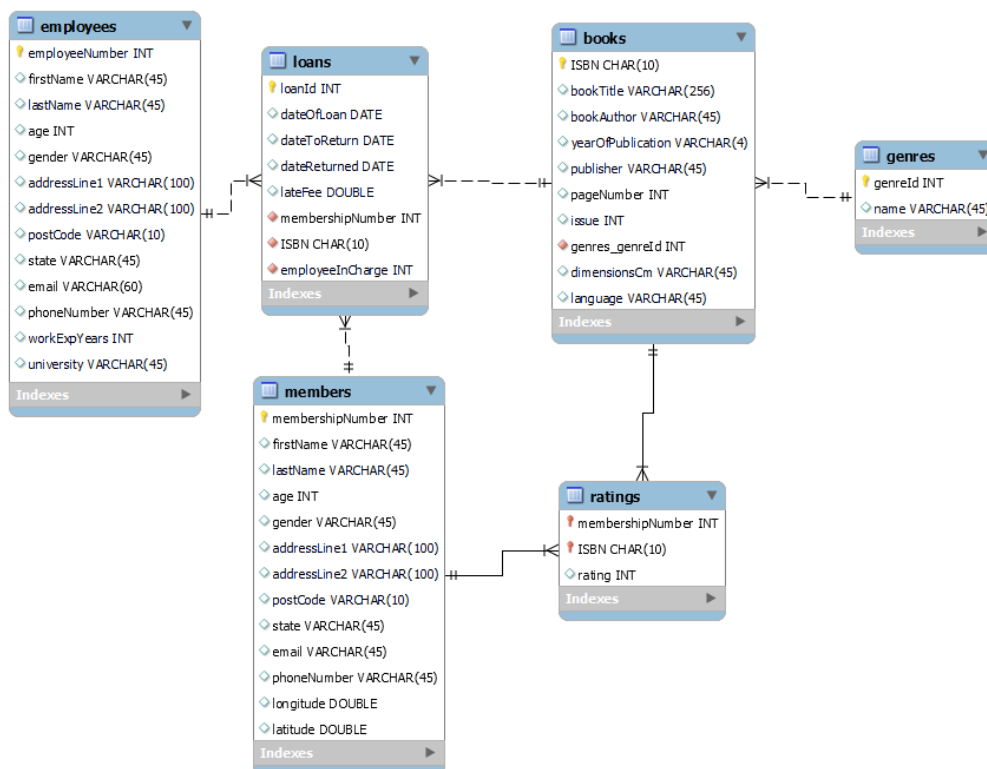
U ovom se dijelu rada prikazuje struktura baze podataka za posudbu knjiga u knjižnici te se opisuju njezini dijelovi za razumijevanje što se treba učiniti kako bi podatke pripremili za korištenje u alatu za poslovnu inteligenciju. Drugi dio poglavlja odnosi se na pripremu podataka i same baze odnosno njezinu pretvorbu u čiste podatke i pretvorbu baze u model zvijezde.

3.1. Opis domene i seta podataka

Domena za koju se izrađuje baza podataka i skladište podataka je posudba knjiga u knjižnici, odnosno specifično vraćanje knjiga u knjižnicu za analizu podataka o knjigama koje su vraćene. Pitanja na koja se pritom želi odgovoriti koristeći skladište podataka i povezani OLAP alat su:

- Koliko je korisnika posudilo neku knjigu?
- Koji žanr je najpopularniji?
- Koje su knjige najčitanije?
- Koja je prosječna ocjena knjige?
- Koliki je zbroj svih ocjena po knjizi?
- Kada se knjige najčešće posuđuju?
- Koji je trend posuđivanja knjiga u nekoj godini?
- Koji korisnici vraćaju knjige sa zakašnjenjem?
- Kolika je ukupna cijena zakašnjenja za određenu knjigu?
- Koje knjige je potrebno naručiti?
- Odakle su naši korisnici?

Na donjoj je slici prikazan model same baze podataka. Sastavljena je od 6 povezanih tablici od koje je svaka prerađena da ima što više atributa kako bi zadovoljila potrebe skladišta podataka.



Slika 1. ER dijagram krajnje verzije posudbe knjiga (Izvor: vlastiti uradak)

Tablica „genres“ sastoji se od primarnog ključa tipa integer te od imena samog žanra. Tablica je napravljena i popunjena ručno s 52 redaka.

Ona je povezana s tablicom books koja je originalno bila skinuta sa stranice *Data.World*, no bila je modificirana kako bi zadovoljila određene kriterije. Atributi u tablici bili su ISBN, naslov knjige, autor knjige, godina izdanja, izdavač, te tri linka na slike knjige koje za potrebe projekta nisu bile potrebne, pa su se maknule. Dodan je broj stranica, izdanje, dimenzija knjige u centimetrima, povezani žanr i jezik knjige. Većina novih atributa generirana je samostalno, preko RAND() funkcije dostupne u MariaDB.

Tablica sadrži oko 29500 redaka podataka. Analizom datoteke, anomalija koja se može primijetiti je da neke knjige za godinu izdanja sadrže vrijednost 0. Umjesto nule, napisat ćemo „N/A“ kao oznaka da godina nije priložena. To možemo učiniti putem sljedećeg upita:

Isječak kôda 1. Mijenjanje vrijednosti godina izdanja (Izvor: vlastiti uradak)

```
UPDATE books SET yearOfPublication = 'N/A' WHERE yearOfPublication=0;
```

Isto tako, u naslovima knjiga, autora i izdavača postoje nepoznati znakovi koji su zamijenjeni kako bi se prikazali čistima. To se izvelo ručno, putem REPLACE() funkcije i putem FindAll funkcije u HeidiSQL sučelju.

„Members“ i „Employees“ slične su tablice koje sadrže identifikacijski broj člana ili zaposlenika knjižnice, ime, prezime, godine, spol, adrese, poštanski broj, saveznu državu, email i broj telefona. Zaposlenici također imaju podatak o radnom stažu i završenom fakultetu, dok članovi knjižnice dodatno sadrže zemljopisnu širinu i dužinu savezne države iz koje dolaze. Podaci unutar tih tablica generirani su preko Mockaroo stranice. Članova je 1000, dok knjižnica ima 24 zaposlenika na šalteru.

Baza podataka također sadrži tablicu koja bilježi posudbu knjiga i tablicu za bilježenje ocjena koje čitatelji mogu ostaviti za određenu knjigu. Tablica „loans“ sadrži oko 17000 podataka o posuđenim knjigama, tko ih je posudio, koji zaposlenik je napravio posudbu, kada je knjiga posuđena, kada se ona vraća, kada se vratila i je li se vratila te se plaća zakašnjenje knjige ukoliko se ne vrati na vrijeme.

Tablica ocjeni „ratings“ sadrži složeni primarni ključ vanjskih ključeva na tablicu članova i knjiga i ocjene knjiga koje su vraćene. Sadrži oko 3200 redaka podataka. Skripta korištena prilikom punjenja podataka ovih tablici prikazana je u prilogu. „lsbns.txt“ je datoteka generirana SELECT naredbom nad svim ISBN kodovima unutar books tablice.

3.2. Izrada dimenzijskog modela i ETL

Dimenzijski je model shema baze podataka koja je posebno prilagođena za učitavanje i čitanje podataka unutar baze. Nije normalizirana i nije namijenjena za rad s naredbama UPDATE i DELETE već se uglavnom koristiti za SELECT jer je u tom aspektu brža od normaliziranih shema, što je pogodno za skladišta podataka (Rabuzin, 2018).

Dimenzijski se model sastoji od činjeničnih i dimenzijskih tablica, pri čemu činjenična tablica sadrži 90% podataka same baze. Ona se sastoji od vanjskih ključeva na sve dimenzijske tablice i od mjera, dok dimenzijske tablice služe za opisivanje činjeničnih i sadrže konkretne vrijednosti za pojedinu kategoriju entiteta. Ovdje se uglavnom nalaze tekstualni podaci, ali mogu biti i brojčane koje opisuju samo proizvod (Rabuzin, 2018).

Prema Rabuzinu (2018), dimenzijsko se modeliranje izvodi u 4 koraka:

- Odabiranje poslovnog procesa koji se modelira
- Određivanje zrnatosti procesa
- Odabiranje dimenzija koje se odnose na redak u činjeničnoj tablici
- Identifikacija činjenica kojima će se popuniti činjenična tablica

Poslovni proces koji u kontekstu skupa podataka ima najviše smisla modelirati je vraćanje knjige. Iako je i sama posudba dobar kandidat, vraćanje knjige osigurava da se može zabilježiti podatak kad je knjiga vraćena, je li vraćanje kasno ili pravovremeno, te je moguće da se knjizi ostavi ocjena, dok prilikom same posudbe te dvije činjenice nije moguće zabilježiti. To će ujedno biti i činjenice na kojima se temelji velik broj pitanja na koja želimo odgovoriti prilikom izrade skladišta podataka.

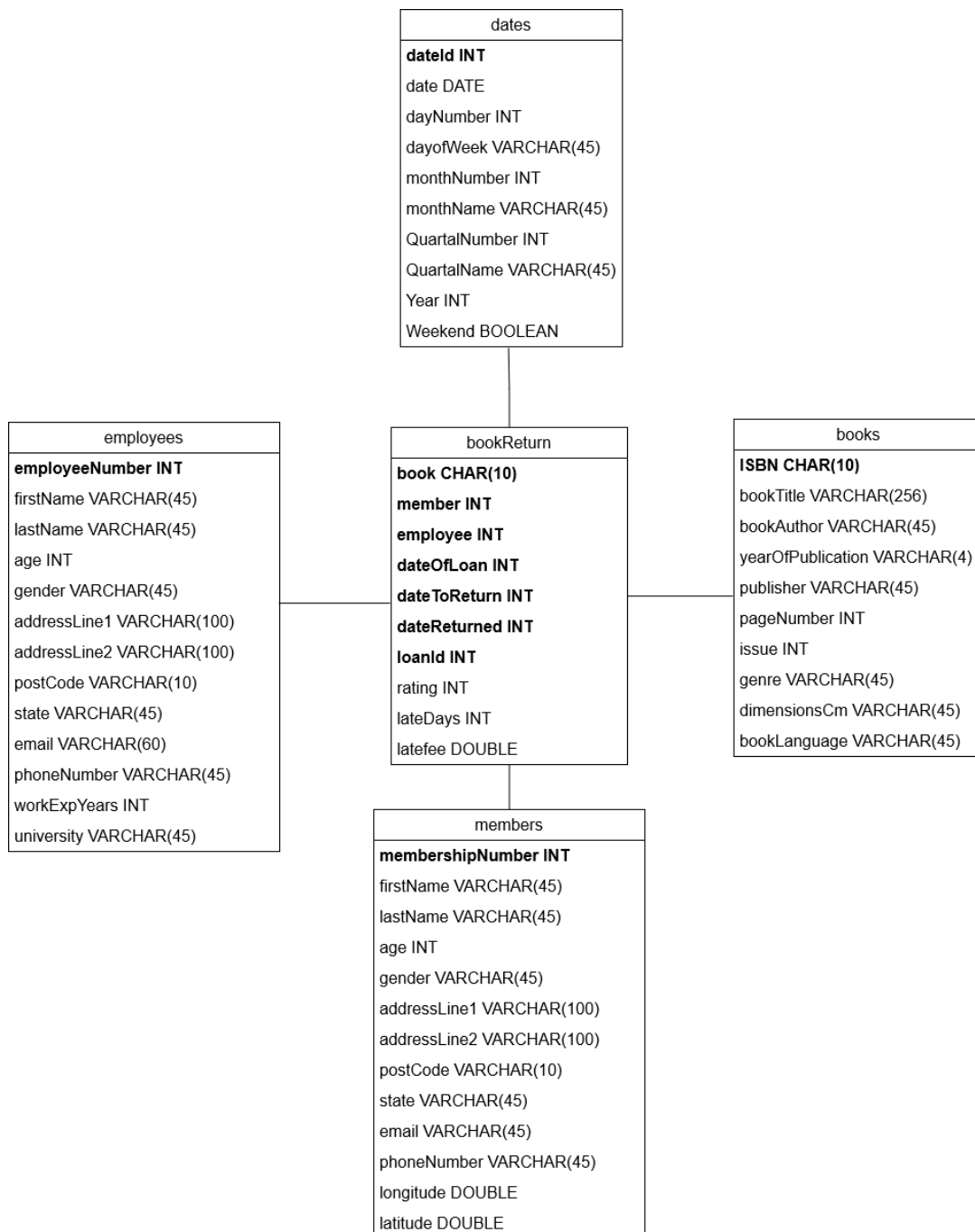
Pritom će se koristiti svi atributi navedeni u već obogaćenim, čišćenim i modificiranim tablicama. Dimenzije koje će se izraditi su dimenzija člana, zaposlenika, knjige i datuma koje će biti povezane na činjeničnu tablicu za vraćanje knjige, u kojoj će se dodatno gledati neobavezna ocjena vraćene knjige, broj dana kašnjenja knjige i cijene zakašnjenja.

Tablice „members“ i „employees“ već su modificirane kako bi bile dimenzije u modelu zvijezde. Tablica „books“ će umjesto vanjskog ključa na tablicu žanrova sada sadržavati samo ime žanra kojeg je dio kako se podatak ne bi izgubio. Dodaje se dimenzija „dates“ koja sadrži datume korištene u bazi u raznim formatima.

Činjenična tablica sadrži jedan složeni primarni ključ načinjen od ključeva na sve dimenzijske tablice i od degenerativnog ključa „loanId“ koji pokazuje na posudbu o kojoj je riječ. Ključevi su „book“ na dimenziju knjiga, „member“ na dimenziju članova, „loanId“ za posudbu, „employee“ na dimenziju zaposlenika te tri posebnih ključeva na dimenziju „dates“ jer je potrebno zabilježiti kad je knjiga posuđena, kad se ona trebala vratiti i kad je ona vraćena.

Na temelju toga u ocjenu, broj dana zakašnjenja i cijenu zakašnjenja zapisuju se podaci o vraćanju knjige. Ocjena, broj dana zakašnjenja i cijena zakašnjenja služe kao mjere ove činjenične tablice.

Slijedi slika izgleda skladišta podataka za proces vraćanja knjiga nakon provedenih svih potrebnih promjena.



Slika 2. Dimenzijski model vraćanja knjiga (Izvor: vlastiti uradak)

Originalna tablica „books“ mijenja ime u „books_old“ kako bi mogli premjestiti podatke iz jedne tablice u drugu. Sljedeći kod prikazuje kreiranje dimenzije „books“ te umetanje svih podataka iz tablice „books_old“ u dimenziju „books“ koristeći INSERT INTO i ugniježdeni SELECT unutar kojeg se nalazi jedan JOIN za očuvanje žanra knjige.

Isječak kôda 2. Kreiranje dimenzije "books" i njezina populacija (Izvor: vlastiti uradak)

```
CREATE TABLE books (

ISBN CHAR(10) PRIMARY KEY,

bookTitle VARCHAR(256),

bookAuthor VARCHAR(45),

yearOfPublication VARCHAR(4),

publisher VARCHAR(128),

pageNumber INT,

issue INT,

dimensionsCm VARCHAR(45),

bookLanguage VARCHAR(45),

genre VARCHAR(45)

);


INSERT INTO books (

SELECT distinct b.ISBN, b.bookTitle, b.bookAuthor,

b.yearOfPublication, b.publisher,

b.pageNumber, b.issue, b.dimensionsCm,

b.bookLanguage, g.name FROM books_old b JOIN

genres g ON g.genreId = b.genreId JOIN loans ON b.ISBN = loans.ISBN

WHERE loans.dateReturned is not null

);
```

Sljedeće se kreira dimenzija datuma. Potrebno je zabilježiti datume posudbe knjige, datum kojeg je potrebno vratiti knjigu te datum kad je knjiga vraćena, pri čemu se u skladište samo kopiraju podaci vezani uz vraćene knjige, odnosno one kojima „dateReturned“ nije NULL.

MariaDB za rad s datumima ima velik broj ugrađenih funkcija, od kojih će se u projektu koristiti DATEDIFF(), DAY(), DAYNAME(), MONTH(), MONTHNAME(), QUARTER(), YEAR() (*Date & Amp; Time Functions*, n.d.).

Način na koji će se popuniti dimenzija 'dates' biti će da će se uzeti svi datumi dateOfLoan iz tablice „loans“ te će se zapisati u dimenziju 'dates'. Nakon toga uzet će se dateToReturn, i naposljetku dateReturned, s time da se uzimaju samo oni datumi gdje su knjige vraćenje, odnosno dateReturned nije null.

Kreira se dimenzija „dates“.

```
CREATE OR REPLACE TABLE dates
(
    dateId INT PRIMARY KEY AUTO_INCREMENT,
    date DATE,
    dayNumber INT,
    dayOfWeek VARCHAR(45),
    monthNumber INT,
    monthName VARCHAR(45),
    quartalNumber INT,
    quartalName VARCHAR(45),
    year INT,
    weekend BOOLEAN
);
```

Dalje je prikazan unos podataka u „dates“ za podatak dateOfLoan. Ista se stvar radi i za vrijednosti dateToReturn i dateReturned. Još jedan način na koji se tablica može popuniti je generiranjem svih vrijednosti datuma između dva određena datuma.

Isječak kôda 3. Unos u dimenziju "dates" (Izvor: vlastiti uradak)

```
INSERT INTO dates (date, dayNumber, dayOfWeek, monthNumber, monthName,
    quartalNumber, year) SELECT dateOfLoan, DAY(dateOfLoan),
    DAYNAME(dateOfLoan), MONTH(dateOfLoan), MONTHNAME(dateOfLoan),
    QUARTER(dateOfLoan), year(dateOfLoan) FROM loans WHERE dateReturned IS NOT
    NULL ORDER BY loanid;
```

Sada se izbacuju sve ponavljajuće vrijednosti koje se mogu otkriti uz DISTINCT klauzulu ili pomoću GROUP BY i HAVING te se dodaju nedostajući dijelovi koda. Dolje je prikazano postavljanje vikenda i imena kvartala ažuriranjem preko SQL upita s uvjetom.

Isječak kôda 4. Ažuriranje vrijednosti u dimenziji "dates" (Izvor: vlastiti uradak)

```
UPDATE dates SET quartalName = 'First' WHERE quartalNumber = 1;

UPDATE dates SET quartalName = 'Second' WHERE quartalNumber = 2;

UPDATE dates SET quartalName = 'Third' WHERE quartalNumber = 3;

UPDATE dates SET quartalName = 'Fourth' WHERE quartalNumber = 4;

UPDATE dates SET Weekend = true WHERE dayOfWeek = 'Sunday';

UPDATE dates SET Weekend = false WHERE dayOfWeek != 'Sunday';
```

Dolje je prikazana rezultanta dimenzijska tablica čije su vrijednosti uzlazno sortirane prema datumu. Tablica sadri 1750 redaka.

libraries.dates: 1,750 rows total (approximately) » Next

dateId	date	dayNumber	dayOfWeek	monthNumber	monthName	quartalNumber	quartalName	year	weekend
555	2018-01-01	1	Monday	1	January	1	First	2,018	0
585	2018-01-02	2	Tuesday	1	January	1	First	2,018	0
300	2018-01-03	3	Wednesday	1	January	1	First	2,018	0
1,431	2018-01-04	4	Thursday	1	January	1	First	2,018	0
216	2018-01-05	5	Friday	1	January	1	First	2,018	0
252	2018-01-06	6	Saturday	1	January	1	First	2,018	0
761	2018-01-07	7	Sunday	1	January	1	First	2,018	1
1,106	2018-01-08	8	Monday	1	January	1	First	2,018	0
1,244	2018-01-09	9	Tuesday	1	January	1	First	2,018	0
920	2018-01-10	10	Wednesday	1	January	1	First	2,018	0
1,400	2018-01-11	11	Thursday	1	January	1	First	2,018	0
1,051	2018-01-12	12	Friday	1	January	1	First	2,018	0
658	2018-01-13	13	Saturday	1	January	1	First	2,018	0
472	2018-01-14	14	Sunday	1	January	1	First	2,018	1
1,276	2018-01-15	15	Monday	1	January	1	First	2,018	0
1,001	2018-01-16	16	Tuesday	1	January	1	First	2,018	0
1,398	2018-01-17	17	Wednesday	1	January	1	First	2,018	0
1,256	2018-01-18	18	Thursday	1	January	1	First	2,018	0
830	2018-01-19	19	Friday	1	January	1	First	2,018	0
290	2018-01-20	20	Saturday	1	January	1	First	2,018	0
1,332	2018-01-21	21	Sunday	1	January	1	First	2,018	1
509	2018-01-22	22	Monday	1	January	1	First	2,018	0
1,037	2018-01-23	23	Tuesday	1	January	1	First	2,018	0
722	2018-01-24	24	Wednesday	1	January	1	First	2,018	0
150	2018-01-25	25	Thursday	1	January	1	First	2,018	0

Slika 3. Dimenzijska tablica "dates" (Izvor: vlastiti uradak)

Sljedeće je potrebno napraviti činjeničnu tablicu koja će povezivati sve vrijednosti tablice i sadržati sve potrebne mjere napomenute u ranijem dijelu rada. Kreiranje činjenične tablice prikazano je u nastavku. Primarni ključ sastavljen je od svih vanjskih ključeva povezanih dimenzijskih tablica te od loanId kao degenerativni ključ za očuvanje podatka o šifri posudbe.

Isječak kôda 5. Kreiranje činjenične tablice (Izvor: vlastiti uradak)

```
CREATE TABLE bookReturn(  
    book char(10),  
    member int,  
    employee int,  
    dateofLoan int,  
    dateToReturn int,  
    dateReturned int,  
    loanId int,  
    rating int,  
    lateDays int,  
    latefee double,  
    CONSTRAINT PRIMARY KEY (book, member, employee, dateOfLoan, dateToReturn,  
    dateReturned, loanID));
```

Za dohvaćanje datuma posudbe, datuma potrebnog vraćanja i datuma vraćanja načinjena je sljedeća funkcija:

Isječak kôda 6. Funkcija za dohvaćanje datuma (Izvor: vlasiti uradak)

```
CREATE FUNCTION getDateId (givenDate date)  
  
    RETURNS INT  
  
    RETURN (SELECT d.dateId FROM dates d WHERE d.date = givenDate);
```

Unosimo sve potrebne podatke za činjeničnu tablicu. Koristit ćemo se kombinacijom INSERT, SELECT, JOIN i kreirane funkcije.

Isječak kôda 7. Popunjavanje činjenične tablice (Izvor: vlastiti uradak)

```
INSERT INTO bookReturn (book, MEMBER, employee, dateOfLoan, dateToReturn,
dateReturned, loanId, rating, lateDays, latefee)

SELECT r.isbn, r.membershipNumber, l.employeeInCharge,
getDateId(l.dateOfLoan), getDateid(l.dateToReturn),
getdateId(l.dateReturned),

l.loanId, r.rating, DATEDIFF(l.dateReturned, dateToReturn), l.latefee from
loans l join ratings r on r.isbn = l.isbn WHERE dateReturned IS NOT NULL;
```

Na kraju ćemo umjesto negativnog broja dana koje daje DATEDIFF() funkcija koja oduzima datum vraćene knjige od datuma kad je knjiga trebala biti vraćena postaviti na 0 kako bi ukazivalo na to da član knjižnice nije kasnio s predajom knjige putem UPDATE upita. Činjenična je tablica izgrađena, ima 4941 podataka te je njezin izgled prikazan na idućoj slici.

libraries.bookreturn: 4,941 rows total (approximately), limited to 1,000 » Next

book	member	employee	dateofLoan	dateToReturn	dateReturned	loanId	rating	lateDays	latefee
0001048473	305	19	512	1,030	744	9,680	8	0	0
0002159767	508	1	302	753	322	22,069	4	29	4.93
0002159767	508	22	921	623	1,454	15,325	4	0	0
0002212064	440	2	1,200	179	365	11,197	5	303	51.51
0002212064	440	19	100	1,380	1,168	11,563	5	173	29.41
0002234440	110	10	327	890	176	4,021	6	0	0
0002234440	110	18	1,050	371	1,475	18,107	6	0	0
0002243687	394	5	17	106	1,387	9,041	8	254	43.18
0002243687	394	13	776	960	1,436	6,524	8	0	0
0002243776	238	12	857	1,214	1,230	3,675	7	0	0
0002258579	313	8	1,355	98	683	14,703	5	0	0
0002258579	313	13	14	62	1,123	18,659	5	0	0
0002258579	313	13	867	759	365	14,651	5	214	36.38
0002258579	765	8	1,355	98	683	14,703	10	0	0
0002258579	765	13	14	62	1,123	18,659	10	0	0
0002258579	765	13	867	759	365	14,651	10	214	36.38
0002261189	864	4	985	1,095	1,152	21,469	9	0	0
0002726874	624	9	323	754	389	2,599	7	0	0
0002726874	624	24	1,473	23	1,508	16,009	7	86	14.62
0003135004	988	1	1,141	793	1,176	21,962	5	0	0
0003135004	988	3	527	490	969	9,977	5	0	0
0003135004	988	15	957	1,381	1,385	10,758	5	0	0
0003701387	234	15	1,317	1,139	935	19,500	0	106	18.02
0004724119	122	21	789	438	490	15,887	10	0	0
0006176607	583	18	381	42	1,750	21,978	9	112	19.04

Slika 4. Činjenična tablica (Izvor: vlastiti uradak)

Sada se eksportiraju sve tablice koje su potrebne u skladištu podataka i može se započeti analiza u programu za poslovnu inteligenciju.

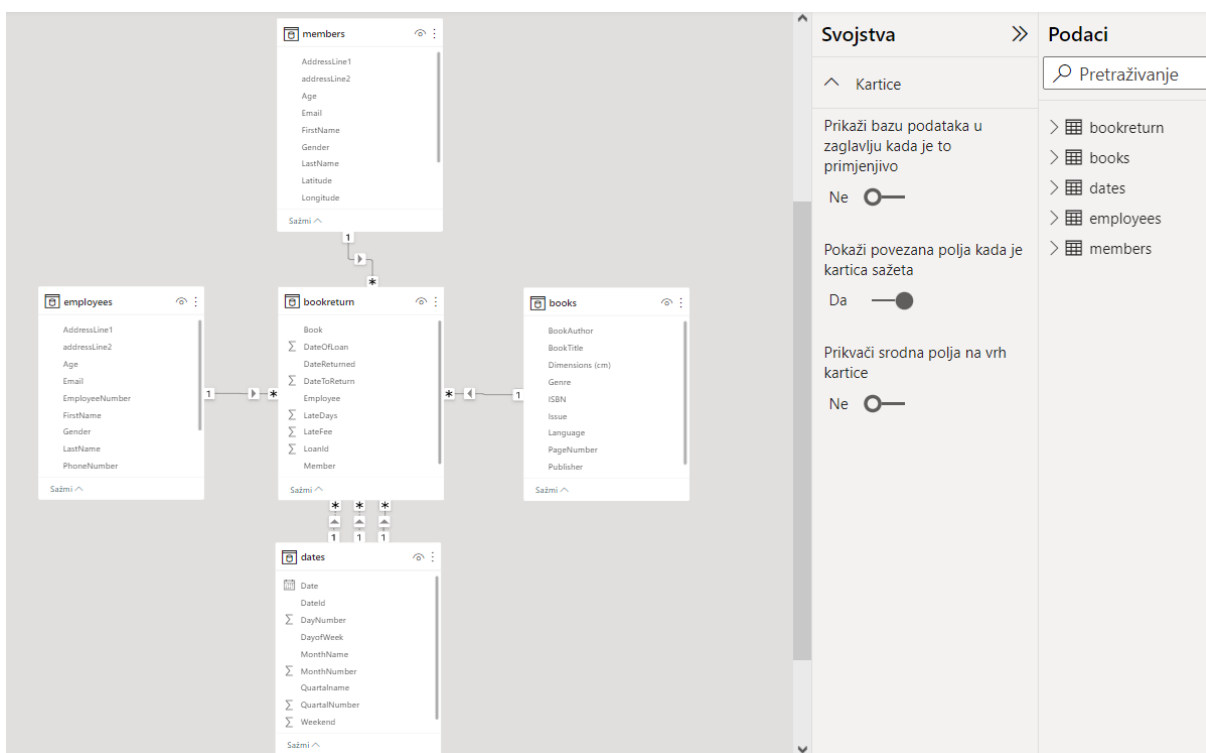
4. Analiza i vizualizacija podataka u alatu Microsoft Power BI

Fokus ovog dijela rada je prikazati načinjene podatke u alatu Microsoft Power BI kako bi se oni analizirali, vizualizirali i kako bi mogli donijeti određene zaključke o podacima koji se nalaze u načinjenom skladištu podataka.

4.1. Učitavanje i povezivanje podataka u Microsoft Power BI

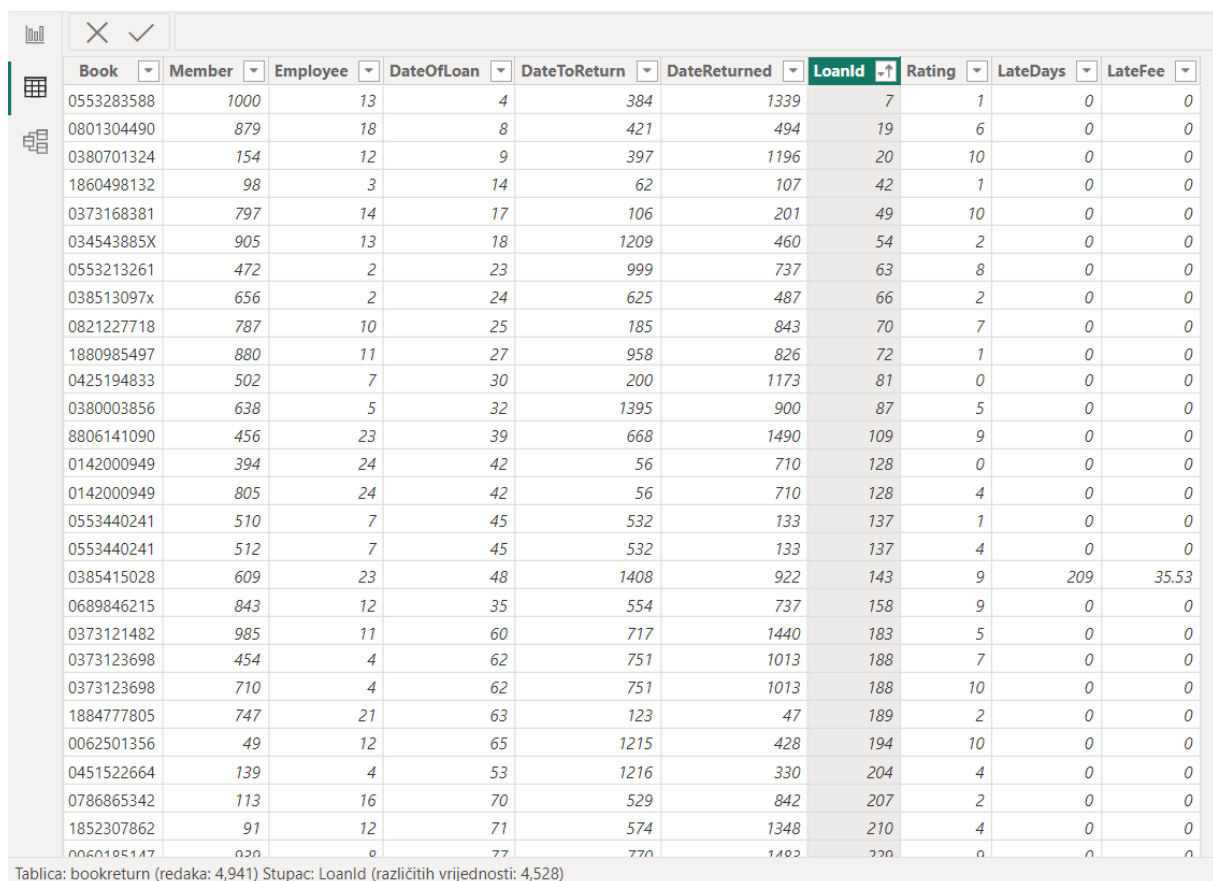
Učitavanje podataka može se izvesti putem naredbe Dohvati podatke, pri čemu se odabire opcija „Tekstne/CSV datoteke“ te se učitaju svi izvori.

Za njihovo međusobno povezivanje, potrebno je ići u „Upravljanje odnosima“ gdje se postavljaju veze između tablica. To se nalazi u kartici „Prikaz modela“. Izgled povezanih tablica je prikazan na donjoj slici. Vidimo da su veze između činjenične tablice i dimenzijskih N:1, a osim toga, na „dates“ tablicu povezano je 3 veza za datum posuđivanja, datum potrebe vraćanja i datum kad je knjiga vraćena, no trenutno je aktivna samo veza za datum kad je knjiga vraćena.



Slika 5. Postavljanje veza između tablica (Izvor: vlastiti uradak)

Sa strane je moguće odabrati karticu „Prikaz podataka“ koji nudi tablični pregled podataka naših tablica. Na donjoj je slici prikazan pregled podataka tablice „bookreturns“.



Book	Member	Employee	DateOfLoan	DateToReturn	DateReturned	LoanId	Rating	LateDays	LateFee
0553283588	1000	13	4	384	1339	7	1	0	0
0801304490	879	18	8	421	494	19	6	0	0
0380701324	154	12	9	397	1196	20	10	0	0
1860498132	98	3	14	62	107	42	1	0	0
0373168381	797	14	17	106	201	49	10	0	0
034543885X	905	13	18	1209	460	54	2	0	0
0553213261	472	2	23	999	737	63	8	0	0
038513097x	656	2	24	625	487	66	2	0	0
0821227718	787	10	25	185	843	70	7	0	0
1880985497	880	11	27	958	826	72	1	0	0
0425194833	502	7	30	200	1173	81	0	0	0
0380003856	638	5	32	1395	900	87	5	0	0
8806141090	456	23	39	668	1490	109	9	0	0
0142000949	394	24	42	56	710	128	0	0	0
0142000949	805	24	42	56	710	128	4	0	0
0553440241	510	7	45	532	133	137	1	0	0
0553440241	512	7	45	532	133	137	4	0	0
0385415028	609	23	48	1408	922	143	9	209	35.53
0689846215	843	12	35	554	737	158	9	0	0
0373121482	985	11	60	717	1440	183	5	0	0
0373123698	454	4	62	751	1013	188	7	0	0
0373123698	710	4	62	751	1013	188	10	0	0
1884777805	747	21	63	123	47	189	2	0	0
0062501356	49	12	65	1215	428	194	10	0	0
0451522664	139	4	53	1216	330	204	4	0	0
0786865342	113	16	70	529	842	207	2	0	0
1852307862	91	12	71	574	1348	210	4	0	0
0060195147	020	0	77	770	1402	220	0	0	0

Tablica: bookreturn (redaka: 4,941) Stupac: LoanId (različitih vrijednosti: 4,528)

Slika 6. Tablični prikaz "bookreturns" u Power BI (Izvor: vlastiti uradak)

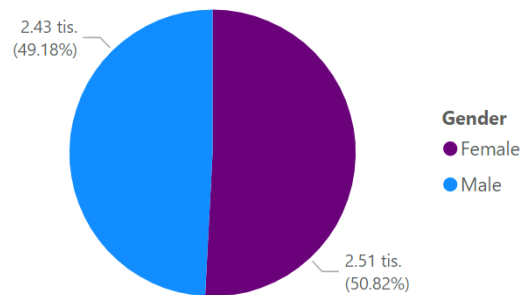
4.2. Prikazivanje podataka u Microsoft Power BI

Prikazivanje se podataka u alatu može napraviti koristeći karticu „Prikaz izvješća“. Odabiru se polja podataka, željena vrsta vizualizacije i filtri koje je moguće postaviti nad podacima.

Recimo da želimo vidjeti odnos spolova po posuđivanju knjiga. Za to možemo odabrati podatke „gender“ u dimenziji članova i LoanId u dimenziji posudba. Za vrstu vizualizacije možemo odabrati tortni grafikon.

Rezultat vizualizacije prikazan je na donjoj slici. Iz njega lagano možemo iščitati da je postotak ljudi koji posuđuje (vraća) knjige pretežito izjednačen.

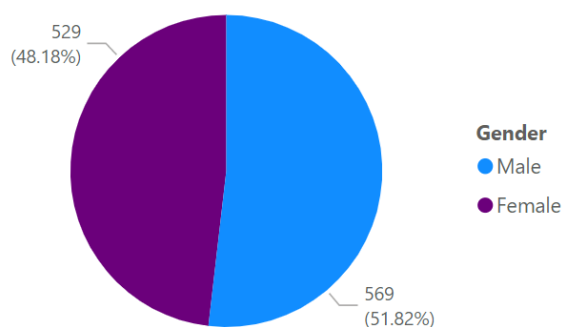
Broj: LoanId kategorije Gender



Slika 7. Usporedba rodova pri posuđivanju knjiga (Izvor: vlastiti uradak)

Dodat ćemo filter koji provjerava postotak muškaraca i žena koji knjige ne predaju na vrijeme tako da vučemo filter LateDays u polje filtra i postavljamo da je dan kašnjenja veći od 0 odnosno članovi kasne s predajom knjiga. Slika dijagrama prikazana je dolje. 51,82% članova koji kasne s predajom su muškarci, dok je 48,18% žena koje kasne s predajom knjiga. Iz ovih podataka možemo zaključiti da nema velike razlike između spolova kad se govori o posuđivanju i vraćanju knjiga.

Broj: LoanId kategorije Gender

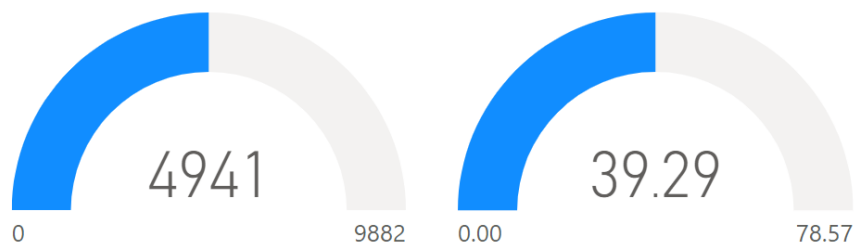


Slika 8. Usporedba rodova pri vraćanju knjiga sa zakašnjenjem (Izvor: vlastiti uradak)

Grafom mjerila možemo prikazati prosječan broj kašnjenja posuđene knjige s postavljanjem vrijednosti na prosjek.

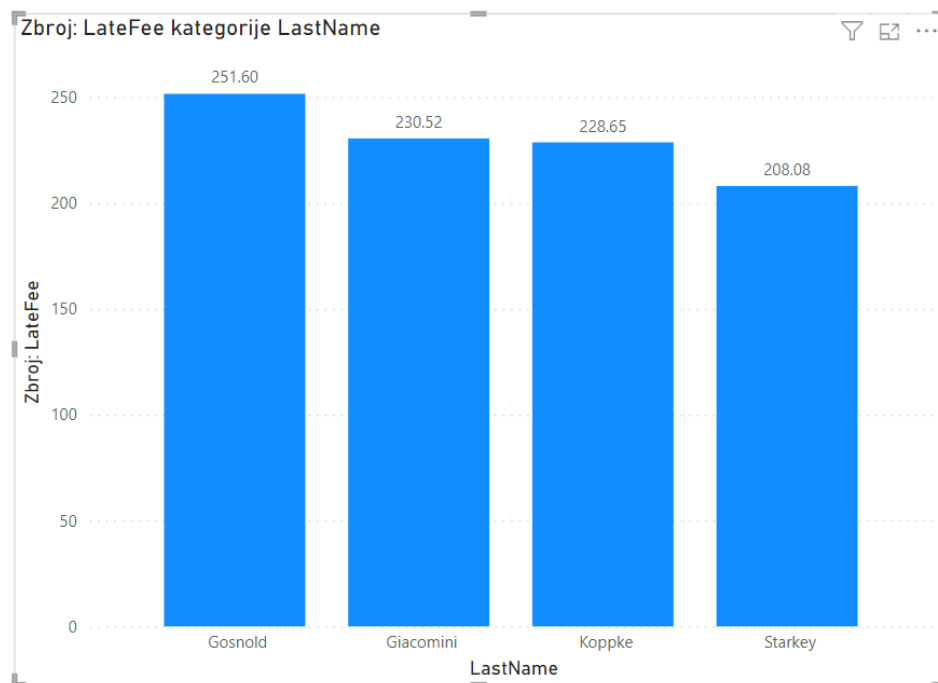
Broj: LoanId

Prosjek: LateDays



Slika 9. Broj posudba i prosječan broj dana kašnjenja s vraćanjem (Izvor: vlastiti uradak)

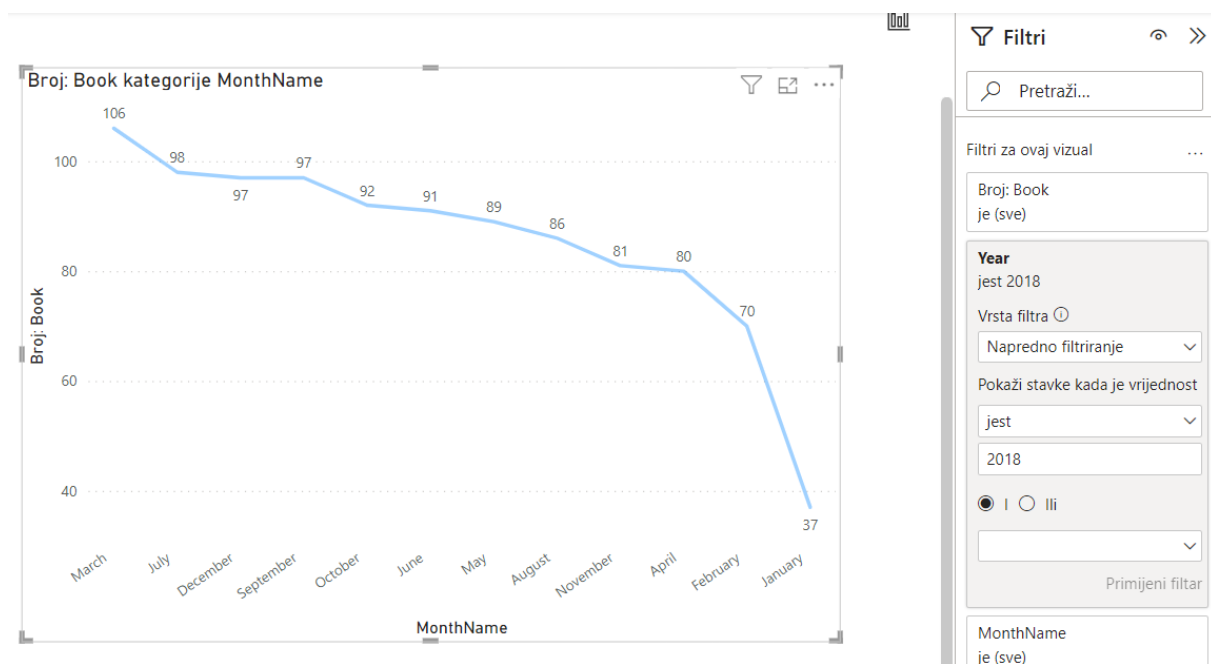
Prikažimo sad prezimena osoba koji kasne s predajom i prikažimo koliko je ukupna cijena koju su morali platiti prilikom vraćanja knjiga. Kako bi broj korisnika bio poprilično velik, postaviti ćemo filtar da se odabiru oni članovi kojima je ukupni zbroj cijene kašnjenja veći od 200 dolara. U stupčastom grafikonu dolje možemo vidjeti da su 4 člana knjižnice ukupno platili više od 200 dolara za zakašnjenje knjige. Označen je član sa šifrom 834 čiji je zbroj svih troškova na kasno vraćanje knjige jednak 208.08 dolara.



Slika 10. Ukupni plaćeni iznosi članova iznad 200 dolara uz grupirana imena i prezimena te kalkulacije (Izvor: vlastiti uradak)

Sljedeće ćemo prikazati broj knjiga vraćen po mjesecu u 2018. postavljajući napredni filter za 2018. godinu.

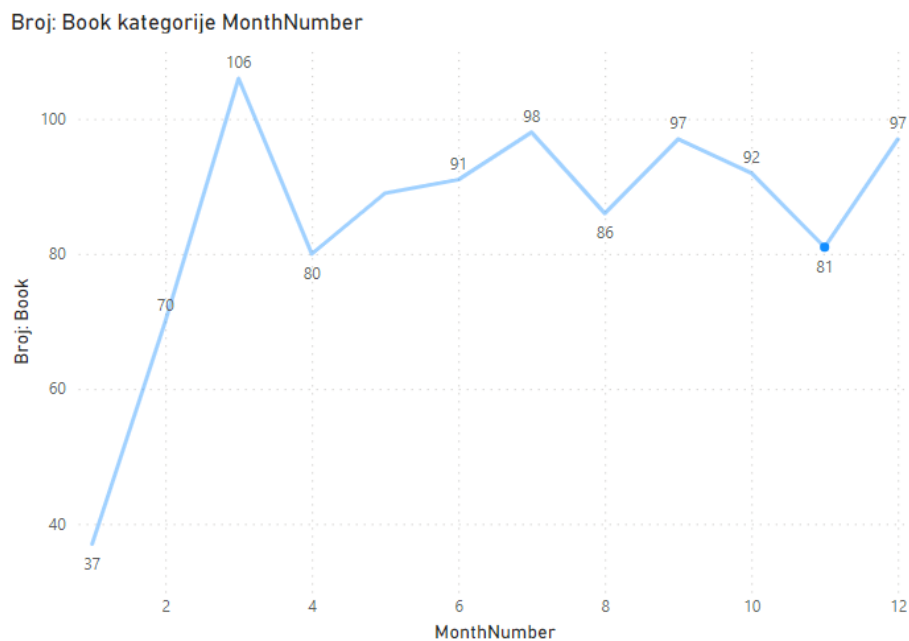
Koristeći linijski dijagram moguće je prikazati mjesece na način da broj posuđenih i vraćenih knjiga opada. Na dijagramu dolje prikazano je da je najveći broj vraćenih knjiga u ožujku (106), pa je u srpnju (98), zatim u prosincu (97), pa redom u listopadu, lipnju, svibnju, kolovozu, studenom, travnju, veljači i na kraju u siječnju s vraćenih 37 knjiga.



Slika 11. Linijski dijagram vraćenih knjiga sortiran prema broju vraćenih knjiga u mjesecima (Izvor: vlastiti uradak)

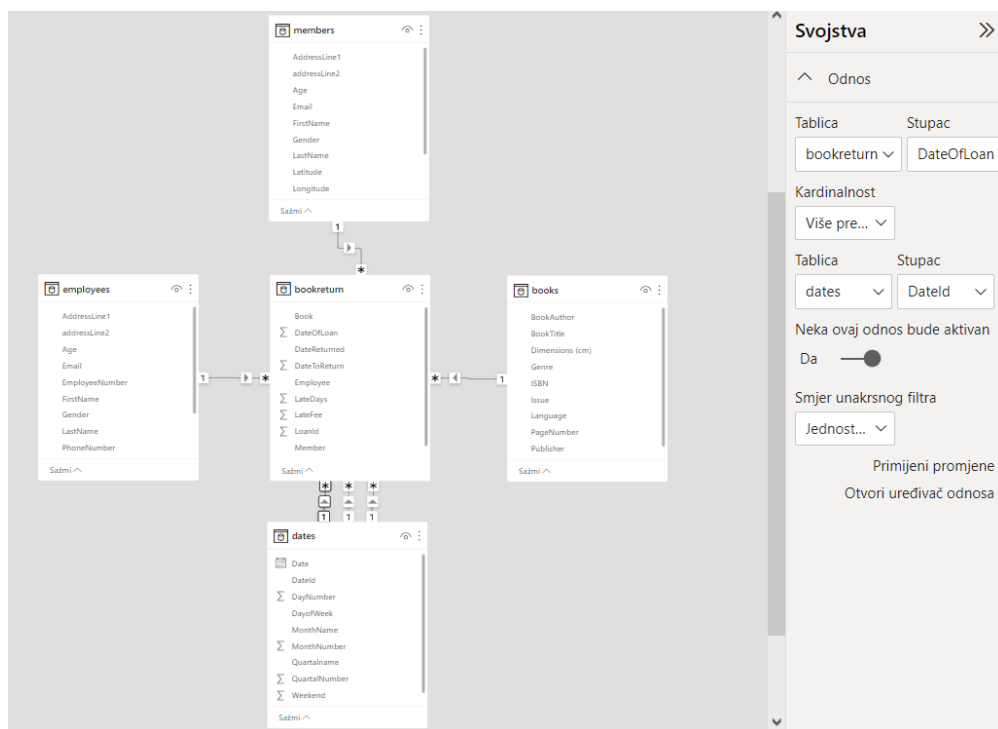
Iste ćemo podatke sad prikazati putem broja mjeseci, pri čemu ćemo mjesece uzlazno. Graf za svaki mjesec po redu izgleda kao na slici prikazanoj dolje.

Iz donjeg prikaza vidi se trend kretanja vraćanja knjiga u 2018. godini. Najviše knjiga vratilo se u trećem mjesecu, dok se najmanji broj knjiga vratio u siječnju te vraćanje knjiga naizmjenično raste i pada u sljedećim mjesecima.



Slika 12. Linijski dijagram vraćenih knjiga po mjesecima (Izvor: vlastiti uradak)

Ukoliko bismo htjeli vidjeti ukupan broj posuđenih knjiga umjesto ukupan broj vraćenih po mjesecu, vraćamo se u karticu „Prikaz modela“ i odabiremo opciju da „dateOfLoan“ bude aktivan te se primijene promjene.

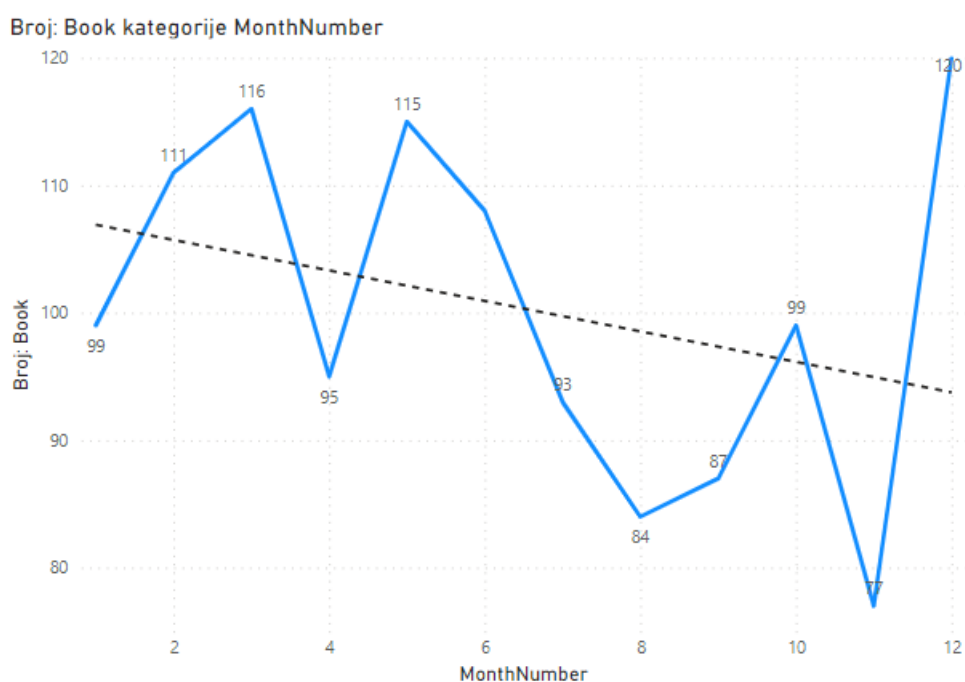


Slika 13. Mijenjanje aktivnosti odnosa (Izvor: vlastiti uradak)

Isti dijagram za datum posudbe izgledat će kao na donjoj slici. Označit ćemo i liniju trenda kojim se knjige posuđuju uz odabir u kartici „Vizualizacija“.

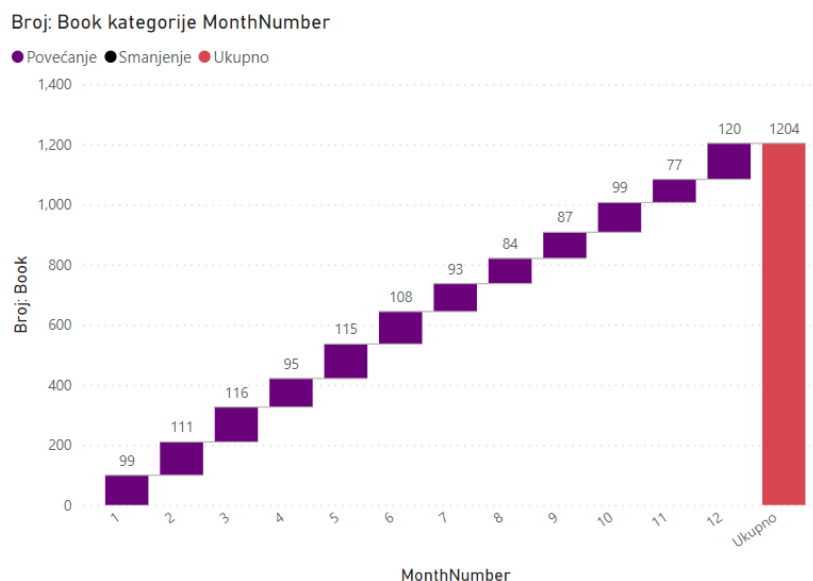
Dijagram pokazuje da se najveći broj knjiga posuđuje u dvanaestom mjesecu, dok se najmanji broj knjiga posuđuje u jedanaestom. Od prvog do jedanaestog mjeseca broj posuđenih knjiga raste i opada periodično te se na kraju godine broj posudba znatno podiže.

Označena je linija trenda posudba knjige. Zbog znatnog pada posudba knjiga u knjižnici od lipnja do studenog, linija trenda opada, no lagano zbog stršila prosinca.



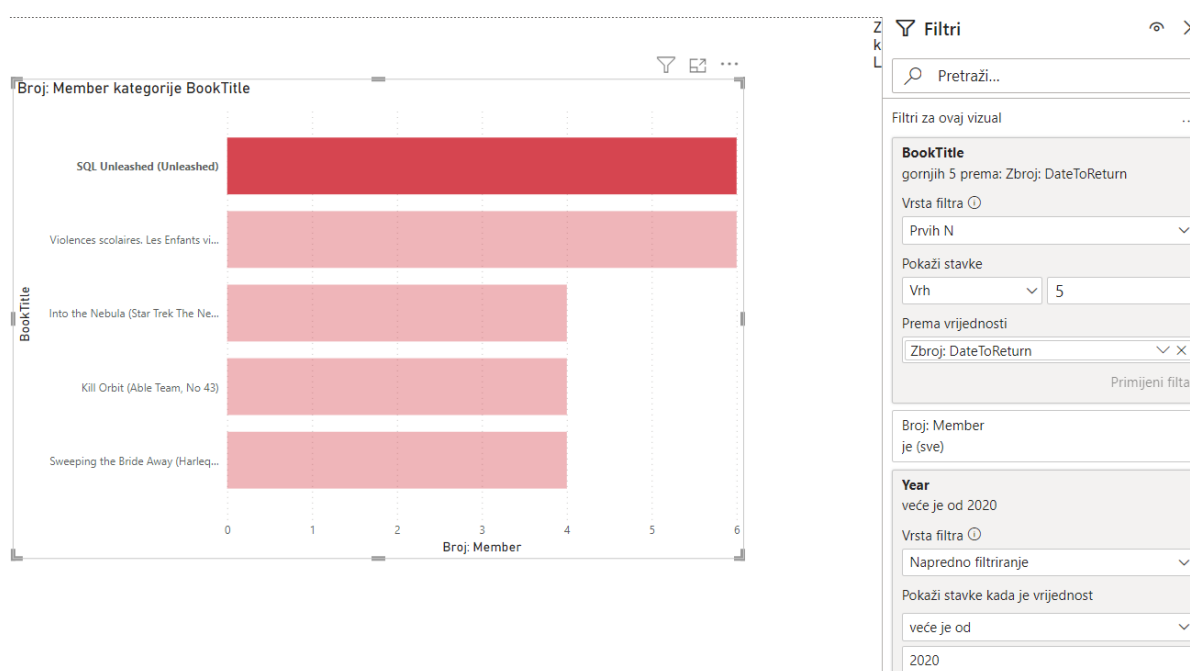
Slika 14. Linijski dijagram posuđenih knjiga po mjesecima uz liniju trenda (Izvor: vlastiti uradak)

Grafikonom vodopada moguće je i vidjeti broj knjiga posuđen po mjesecu te ukupan broj knjiga koji se posudio u 2018. godini. Ukupan broj knjiga vraćeno u ovoj godini je jednaka 1204. Graf je sortiran uzlazno po osi x, odnosno po broju mjeseca.



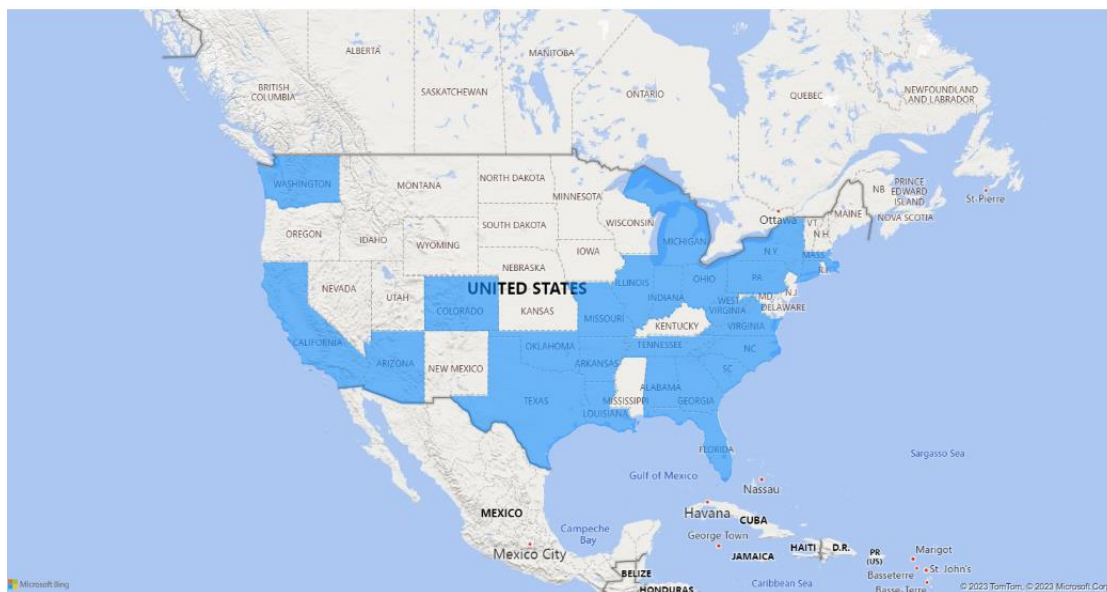
Slika 15. Prikaz broja knjiga posuđeno po mjesecima i ukupan broj knjiga u 2018. godini (Izvor: vlastiti uradak)

Sljedeće se prikazuju 5 najposuđenijih knjiga nakon 2020. godine. To se postiže dodavanjem većeg broja filtra. Uzimaju se prvih 5 vrijednosti za najposuđenije knjige po datumu te uzimamo godine veće od 2020. Kao najposuđenije knjige ističu se SQL Unleashed i Violences scolaires sa šest posuđenih primjeraka te slijede Into the Nebula, Kill Orbit i Sweeping the Bride Away s 4 primjeraka.

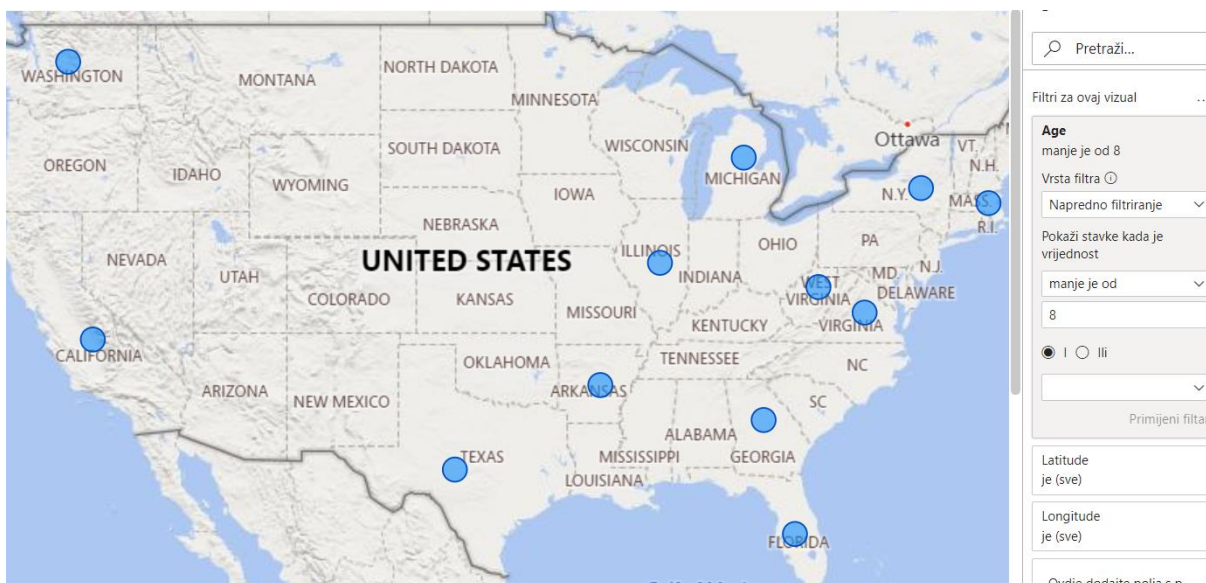


Slika 16. 5 najposuđenijih knjiga 2020. godine (Izvor: vlastiti uradak)

U Microsoft Power BI je pomoću geografske širine i dužine i lokacije moguće označiti geografske lokacije koje sudjeluju u setu podataka. Označit ćemo sve savezne države u kojima se nalaze članovi knjižnice skupa podataka. Na prvoj je slici prikazana karta označena po saveznoj državi, dok je za drugu sliku korištena geografska dužina i širina uz filter da član mora biti mlađi od 8 godina.



Slika 17. Savezne države članova knjižnice (Izvor: vlastiti uradak)

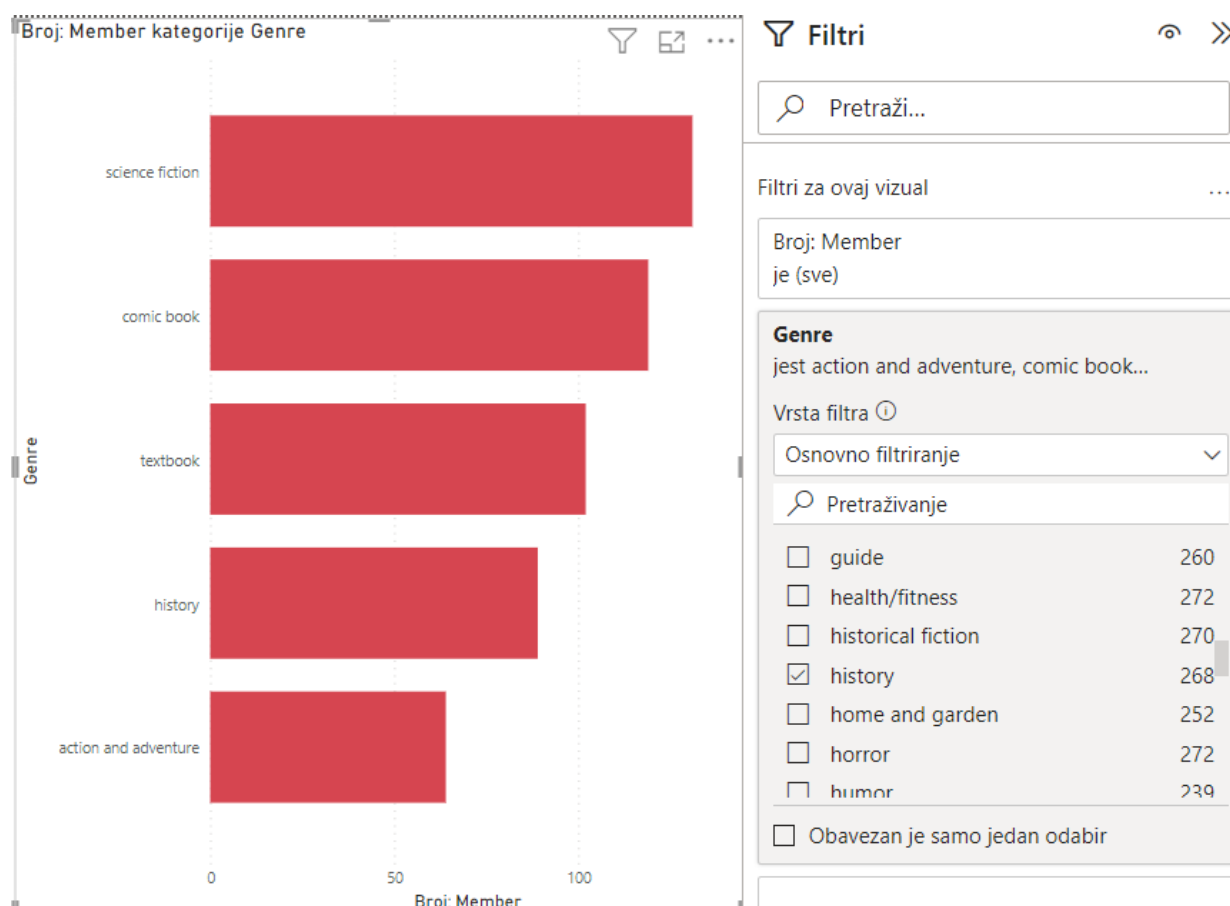


Slika 18. Korištenje geografske širine i dužine za prikaz država članova (Izvor: vlastiti uradak)

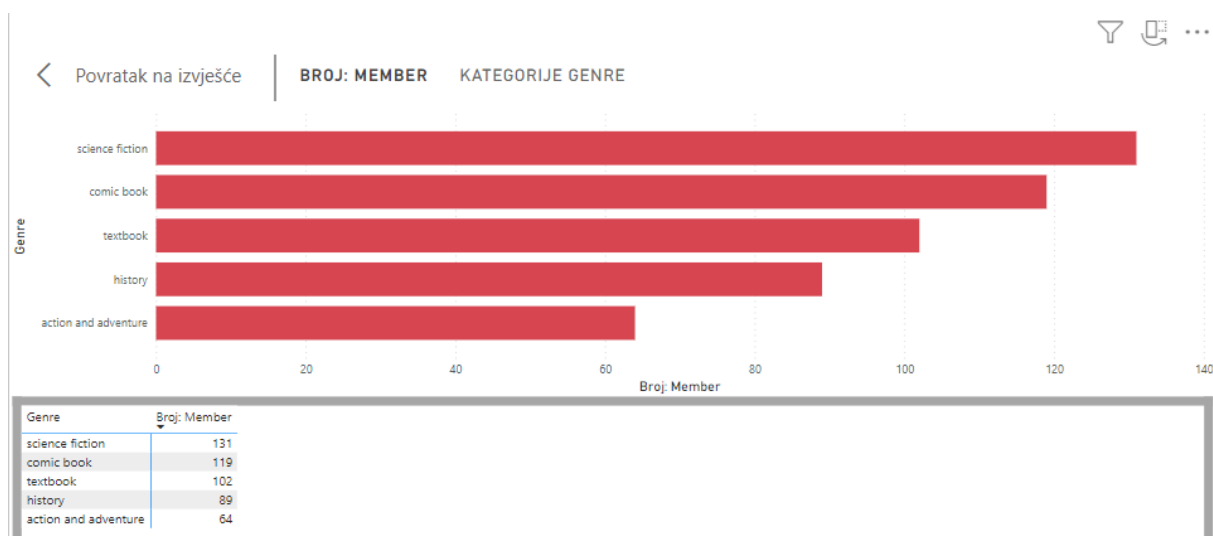
Sljedeće se prikazuje ukupni broj knjiga po određenom žanru. Kako postoji velik broj žanrova, označit ćemo samo specifične tako da postavimo osnovni filter nad podacima označavajući samo žanrove koje želimo prikazati.

Odabrat ćemo žanrove *science fiction*, *comic book*, *textbook*, *history* i *action and adventure*. Gornja slika na sljedećoj stranici prikazuje označavanje konkretnih vrijednosti u osnovnom filteru, dok donja strana prikazuje Microsoft Power BI-ov način prikaza u tablici (opcija „Prikaži kao tablicu“).

Grafovi prikazuju broj žanrova po knjizi koja se posuđuje. Jedna od popularnijih žanrova u ovoj specifičnoj knjižnici je znanstvena fantastika, dok se manje posuđuju knjige žanra akcije i avanture.

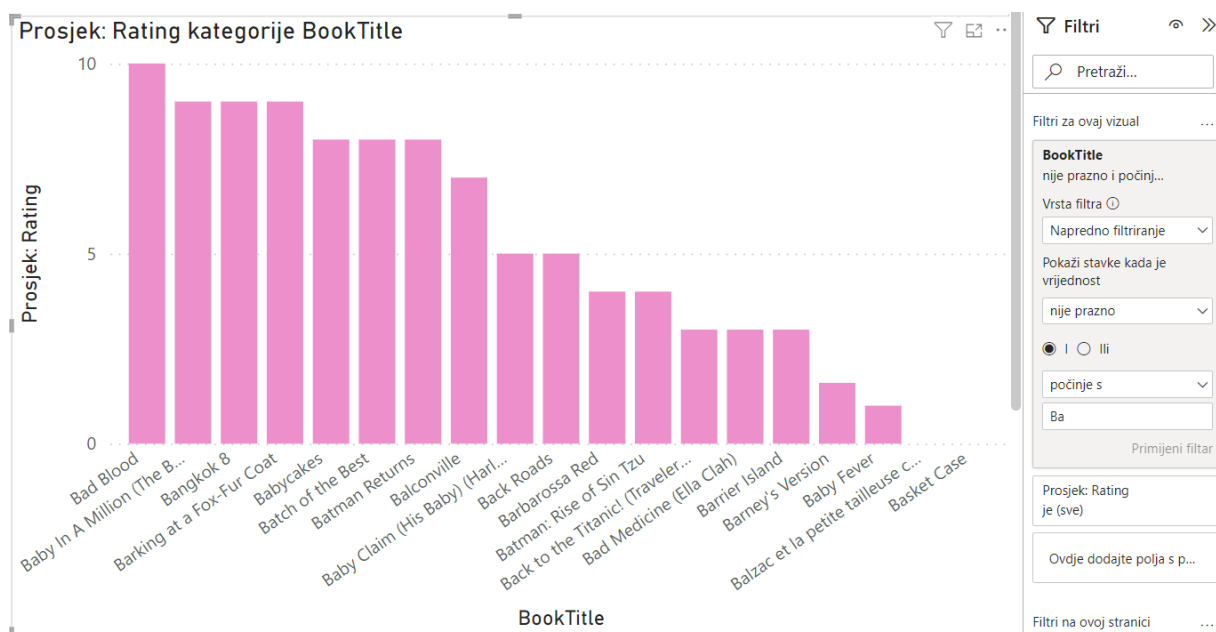


Slika 19. Osnovni filter (Izvor: vlastiti uradak)



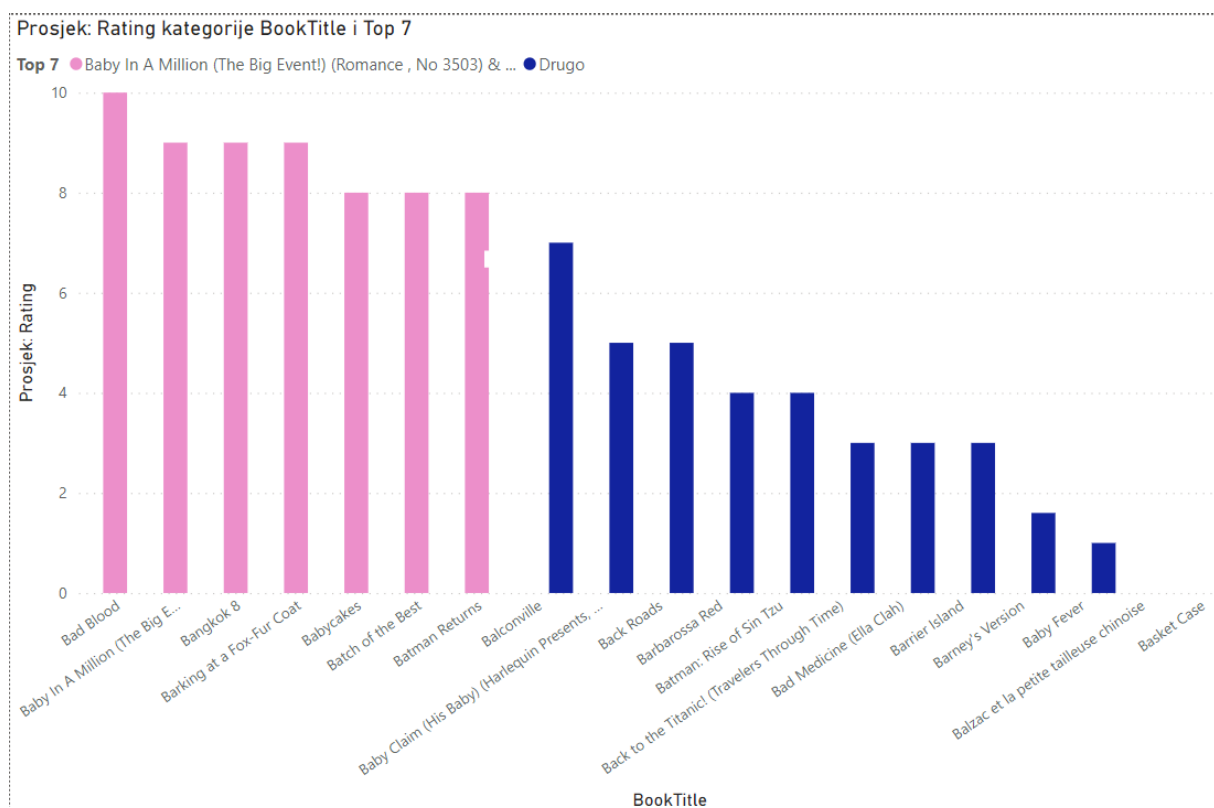
Slika 20. Tablični prikaz odabranih žanrova (Izvor: vlastiti uradak)

Sljedeća slika prikazuje dohvaćanje prosječnih ocjena po knjigama koje počinju s „Ba“. Vidi se da knjiga Bad Blood ima prosječnu ocjenu 10, Balconville 7, a Basket Case 0. Pritom smo s filterom označili da knjiga mora biti ocijenjena, odnosno vrijednost nije prazna.



Slika 21. Prosječna ocjena po knjizi (Izvor: vlastiti uradak)

Vrijednosti je moguće i grupirati tako da se označe vrijednosti koje se žele grupirati u jednu skupinu. Kreira se grupa Top 7, u kojoj će biti knjige koje započinju sa slovima „Ba“ i čiji su prosjeci ocjena jednaki ili veći od 7.

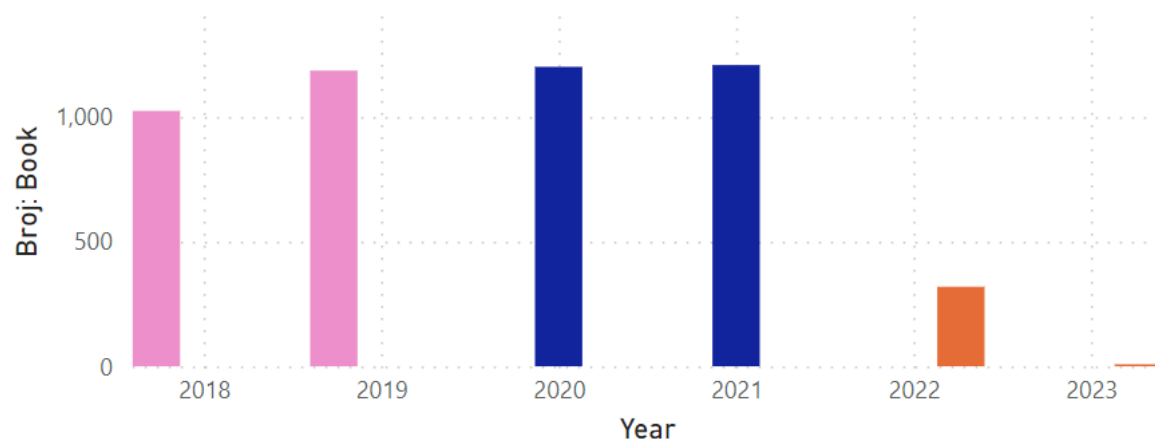


Slika 22. Grupirane vrijednosti 7 najbolje ocijenjenih knjiga uz filter (Izvor: vlastiti uradak)

Za prikaz grupe određene veličine, kreiraju se tzv. kutije ili košare. Kreira se graf vraćenih knjiga po godinama. To će kreirati tri grupe podataka od godina 2018. do 2023. koje će na grafu biti označene drugim bojama.

Broj: Book kategorije Year i Year (košare)

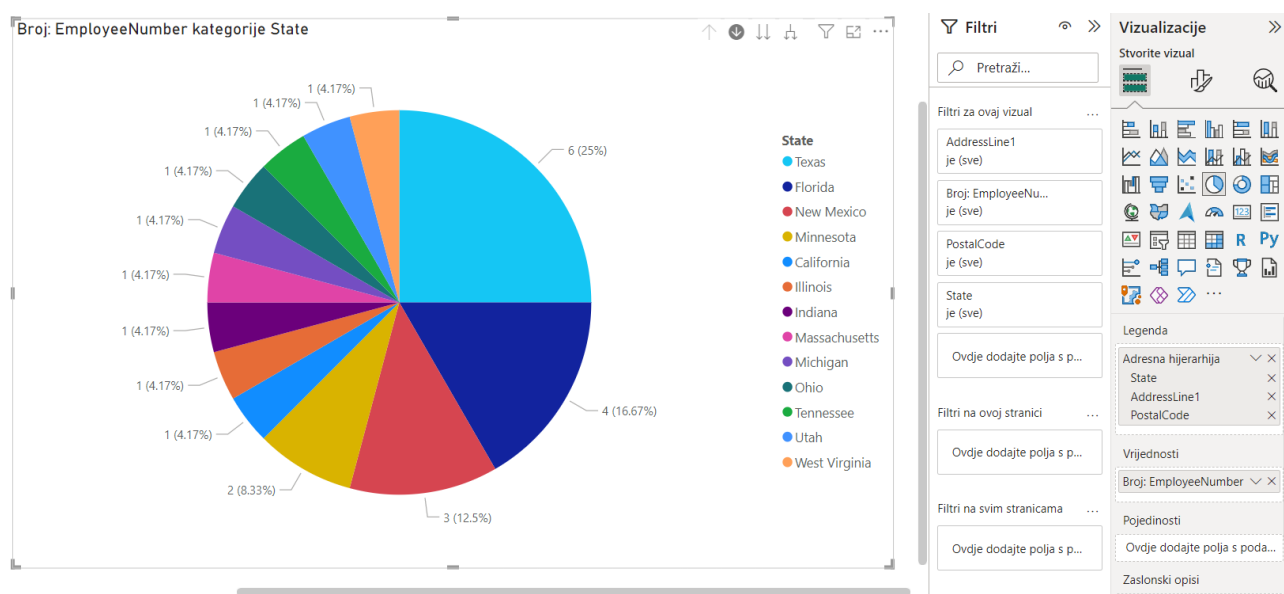
Year (košare) ● 2018 ● 2020 ● 2022



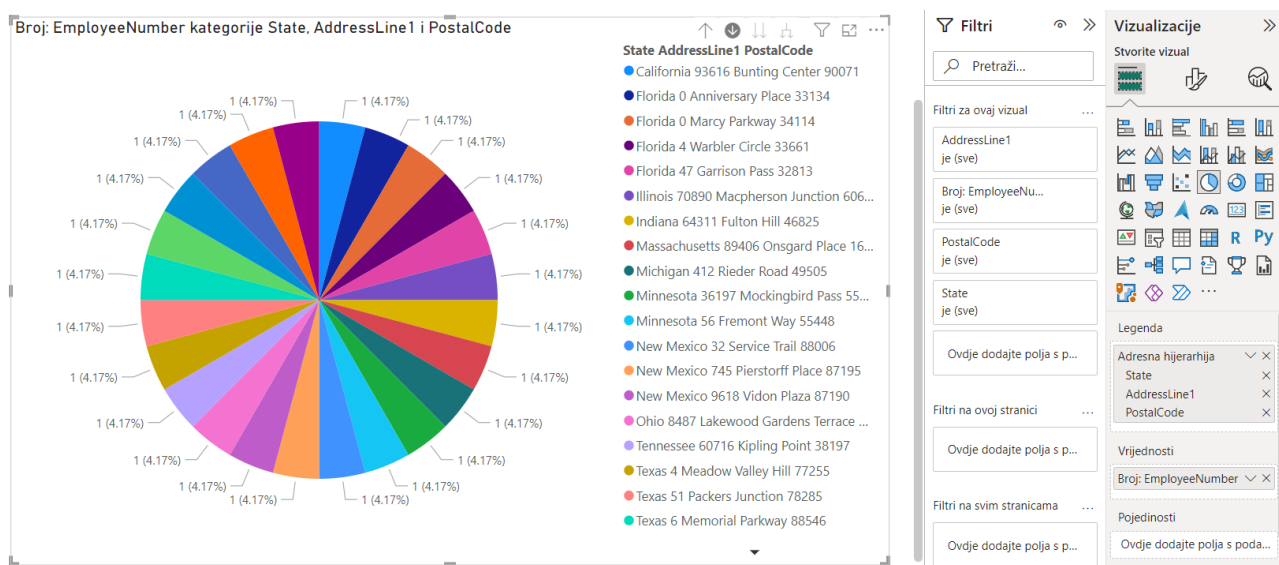
Slika 23. Kutije nad godinama vraćanja knjiga (Izvor: vlastiti uradak)

Sljedeće dvije slike prikazuju korištenje hijerarhija u Microsoft Power BI alatu. Napraviti ćemo hijerarhiju State->AddressLine1->PostalCode u kartici Podaci te ćemo napraviti tortni dijagram koji prikazuje broj zaposlenika koji žive na određenoj adresi. Navigiranjem kroz strjelice na grafu jednostavno je moguće kretati se po razinama hijerarhije.

Na prvoj slici vidljivo je da je 25% zaposlenika iz Texasa, 4 zaposlenika iz Floride i tako dalje, dok je na drugoj slici vidljivo da svi stanuju na različitim adresama jer smo na najduljoj, najpotpunijoj adresi.

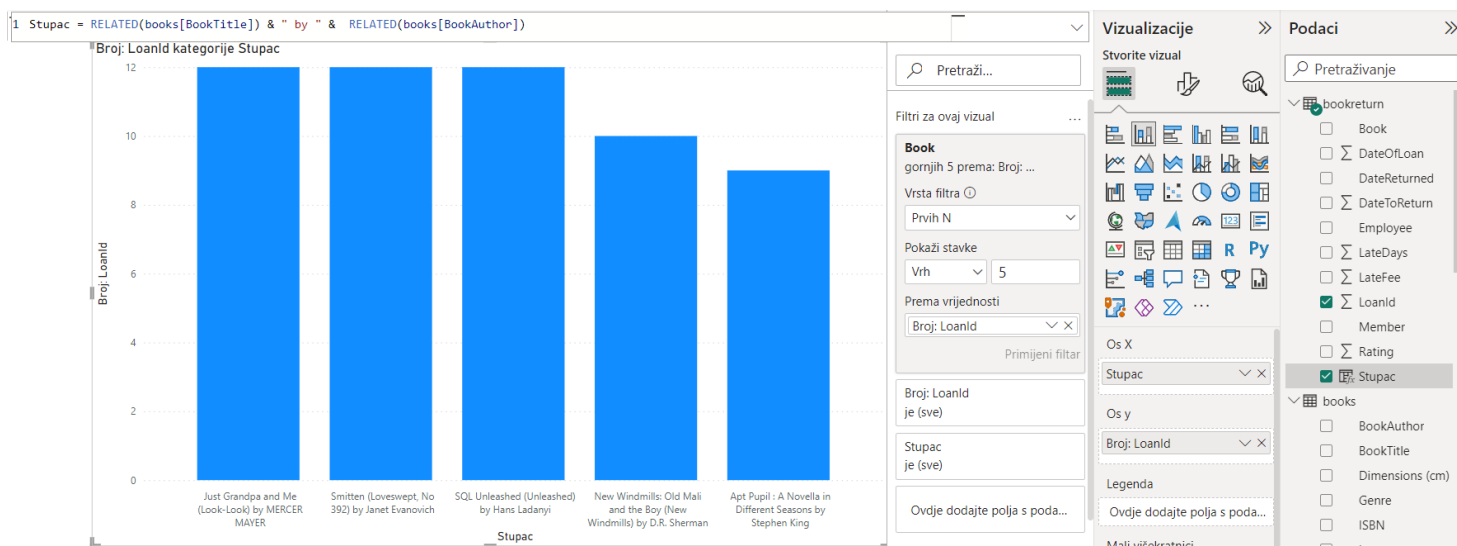


Slika 24. Hijerarhija zaposlenika – najviša razina (Izvor: vlastiti uradak)



Slika 25. Hijerarhija zaposlenika – najniža razina (Izvor: vlastiti uradak)

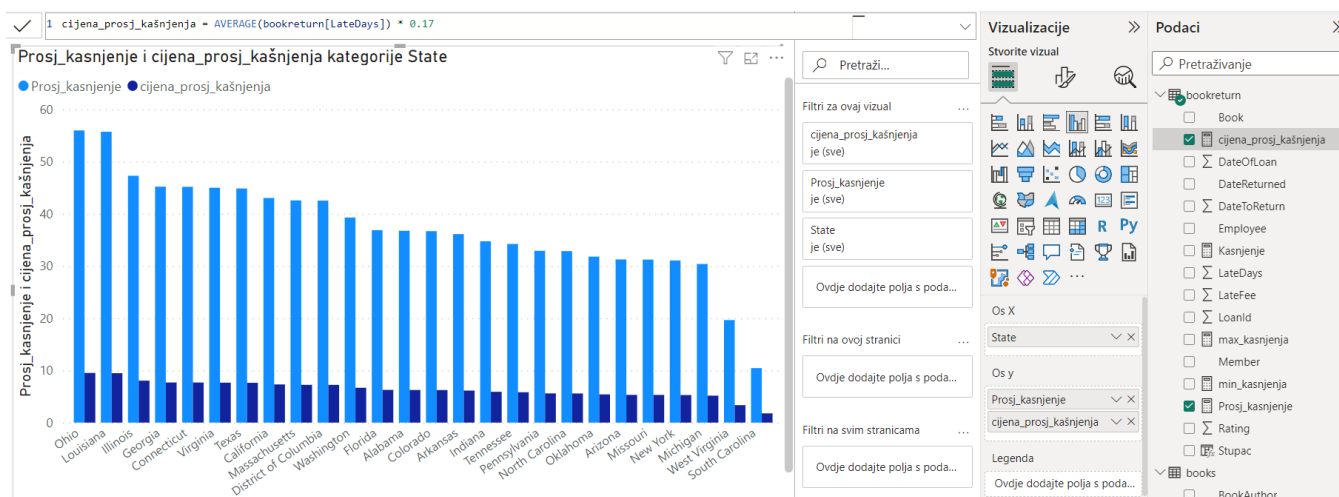
Dalje se prikazuje korištenje izračunatih stupca i izračunatih mjera za detaljnije nazivlje, dublju analizu i za način grupiranja podataka u Power BI uz korištenje kalkulacija. Prva slika prikazuje broj posudba prvih gornjih 5 knjiga. Iako je dohvaćen samo ISBN knjige i ID posudbe u činjeničnoj tablici, na grafu je ISBN moguće prikazati kao NASLOV_KNJIGE by AUTOR_KNJIGE. To se postiže funkcijom RELATED koja traži vrijednosti u povezanoj tablici.



Slika 26. Izračunati stupac (Izvor: vlastiti uradak)

Za vlastite mjere koristi se opcija „Nova mjera“. Na sljedeće dvije slike će se prikazati korištenje izračunatih mjera. Prva slika prikazuje prosječni broj kašnjenja vraćanja knjiga i cijenu prosječnog broja kašnjenja vraćenih knjiga koristeći izračunate mjere, dok druga slika prikazuje maksimalnu ocjenu knjige gdje je prosječna ocjena veća od 5. Kako bi se rezultati prikazali na manjem skupu podataka, rezultati će se filtrirati tako da naslovi sadrže riječ „Flower“.

U prvoj slici možemo vidjeti odnos između saveznih država i cijene koje se plaćaju za knjige koje kasne s predajom. U drugoj slici su naslovi i autori pet knjiga izračunatu putem IF uvjeta, zbroja svih ocjena po knjizi podijeljeno s brojem svih ocjena po knjizi, te uz selektiranje maksimalne vrijednosti tih ocjena.



Slika 27. Izračunata mjera za trošak prosječnog kašnjenja (Izvor: vlastiti uradak)



Slika 28. Mjera maksimalne ocjene knjige čiji je prosjek ocjena veći od 5 (Izvor: vlastiti uradak)

5. Zaključak

Skladišta podataka i alat za poslovnu inteligenciju služe za detaljnu vizualizaciju i analizu podataka unutar neke baze podataka za pomoć pri određivanju trendova, filtriranju i očitavanju podataka i odlučivanju. U radu se kreirala MariaDB baza podataka knjižnice uz jedan javno dostupan dataset i samostalno generirane podatke nad kojom se provelo ekstrahiranje, čišćenje i učitavanje podataka u skladište podataka dimenzijskog oblika.

Podaci se potom detaljno pregledavaju i analiziraju te je prikazan rad različitih mogućnosti unutar alata Microsoft Power BI Desktop kao što su prikaz podataka, osnovno i napredno filtriranje, sortiranje, grupiranje, kreiranje kutija, označavanje linije trenda, tablični i geografski prikaz podataka i kreiranje dodatnih izračunatih stupaca i mjera putem kalkulacija.

Pritom je 80% vremena za izradu projekta otpalo na ETL proces i generiranje potrebnih podataka. Jednom kada su podaci izvezeni, alat za poslovnu inteligenciju čini rad s njima mnogo lakši, brži i pregledniji.

Popis literature

Becker, A. (n.d.). HeidiSQL - MariaDB, MySQL, MSSQL, PostgreSQL and SQLite made easy. <https://www.heidisql.com/>

Date & Time Functions. (n.d.). MariaDB KnowledgeBase. <https://mariadb.com/kb/en/date-time-functions/>

MariaDB.org. (2023). Documentation - MariaDB.org. <https://mariadb.org/documentation/#entry-header>

Rabuzin, K. (2018). *Skladište podataka i poslovna inteligencija*. Moodle. <https://elf.foi.hr/mod/resource/view.php?id=9659>

Popis slika

Slika 1. ER dijagram krajnje verzije posudbe knjiga (Izvor: vlastiti uradak)	4
Slika 2. Dimenzijski model vraćanja knjiga (Izvor: vlastiti uradak).....	7
Slika 3. Dimenzijska tablica "dates" (Izvor: vlastiti uradak).....	10
Slika 4. Činjenična tablica (Izvor: vlastiti uradak)	12
Slika 5. Postavljanje veza između tablica (Izvor: vlastiti uradak).....	13
Slika 6. Tablični prikaz "bookreturns" u Power BI (Izvor: vlastiti uradak)	14
Slika 7. Usporedba rodova pri posuđivanju knjiga (Izvor: vlastiti uradak)	15
Slika 8. Usporedba rodova pri vraćanju knjiga sa zakašnjenjem (Izvor: vlastiti uradak)	15
Slika 9. Broj posudba i prosječan broj dana kašnjenja s vraćanjem (Izvor: vlastiti uradak)	16
Slika 10. Ukupni plaćeni iznosi članova iznad 200 dolara uz grupirana imena i prezimena te kalkulacije (Izvor: vlastiti uradak)	16
Slika 11. Linijski dijagram vraćenih knjiga sortiran prema broju vraćenih knjiga u mjesecima (Izvor: vlastiti uradak).....	17
Slika 12. Linijski dijagram vraćenih knjiga po mjesecima (Izvor: vlastiti uradak)	18
Slika 13. Mijenjanje aktivnosti odnosa (Izvor: vlastiti uradak)	18
Slika 14. Linijski dijagram posuđenih knjiga po mjesecima uz liniju trenda (Izvor: vlastiti uradak)	19
Slika 15. Prikaz broja knjiga posuđeno po mjesecima i ukupan broj knjiga u 2018. godini (Izvor: vlastiti uradak)	20
Slika 16. 5 najposuđenijih knjiga 2020. godine (Izvor: vlastiti uradak).....	20
Slika 17. Savezne države članova knjižnice (Izvor: vlastiti uradak)	21
Slika 18. Korištenje geografske širine i dužine za prikaz država članova (Izvor: vlastiti uradak)	21
Slika 19. Osnovni filter (Izvor: vlastiti uradak)	22
Slika 20. Tablični prikaz odabranih žanrova (Izvor: vlastiti uradak).....	23
Slika 21. Prosječna ocjena po knjizi (Izvor: vlastiti uradak)	23
Slika 22. Grupirane vrijednosti 7 najbolje ocijenjenih knjiga uz filter (Izvor: vlastiti uradak)	24
Slika 23. Kutije nad godinama vraćanja knjiga (Izvor: vlastiti uradak)	24
Slika 24. Hijerarhija zaposlenika – najviša razina (Izvor: vlastiti uradak)	25
Slika 25. Hijerarhija zaposlenika – najniža razina (Izvor: vlastiti uradak).....	25
Slika 26. Izračunati stupac (Izvor: vlastiti uradak)	26
Slika 27. Izračunata mjera za trošak prosječnog kašnjenja (Izvor: vlastiti uradak)	27
Slika 28. Mjera maksimalne ocjene knjige čiji je prosjek ocjena veći od 5 (Izvor: vlastiti uradak)	27

Popis isječaka kôda

Isječak kôda 1. Mijenjanje vrijednosti godina izdanja (Izvor: vlastiti uradak)	4
Isječak kôda 2. Kreiranje dimenzije "books" i njezina populacija (Izvor: vlastiti uradak)	8
Isječak kôda 3. Unos u dimenziju "dates" (Izvor: vlastiti uradak)	9
Isječak kôda 4. Ažuriranje vrijednosti u dimenziji "dates" (Izvor: vlastiti uradak)	10
Isječak kôda 5. Kreiranje činjenične tablice (Izvor: vlastiti uradak)	11
Isječak kôda 6. Funkcija za dohvaćanje datuma (Izvor: vlastiti uradak)	11
Isječak kôda 7. Popunjavanje činjenične tablice (Izvor: vlastiti uradak)	12

1. Prilog Python kod za generiranje vrijednosti

```
#program for mocking loans data

import random

import datetime


#function returns random isbn of text file

def give_isbn():

    isbns = content.split('\n')

    return random.choice(isbns)


#function to return random date

def give_date():

    start_date = datetime.date(2018, 1, 1)

    end_date = datetime.date(2022, 2, 1)

    time_between_dates = end_date - start_date

    days_between_dates = time_between_dates.days

    random_number_of_days = random.randrange(days_between_dates)

    random_date = start_date +
datetime.timedelta(days=random_number_of_days)

    return random_date


#function to return random date between two given

def give_date_b(start_date, end_date):

    time_between_dates = end_date - start_date

    days_between_dates = time_between_dates.days

    random_number_of_days = random.randrange(days_between_dates)

    random_date = start_date +
datetime.timedelta(days=random_number_of_days)

    return random_date
```

```

f = open('isbns.txt', 'r')

content = f.read()

latefee = 0.17


#on repeat
for i in range(1000):

    #random employee from 1 to 24
    employee = random.randint(1,24)

    #picking random number between 1 and 1000
    member = random.randint(1,1000)

    #picking isbn
    chosen_isbn = give_isbn()

    #dates
    date_of_loan = give_date()
    date_to_return = date_of_loan + datetime.timedelta(days=31)


    #late or not + latefee
    yesno = random.randint(0,10)

    if(yesno < 8):

        date_returned = give_date_b(date_of_loan, date_to_return)
        latefee = 0

    else:

        days_late = random.randint(1,356)

        date_returned = date_to_return +
datetime.timedelta(days=days_late)

        latefee = latefee * days_late
        latefee = round(latefee, 2)


    #returned or not

    sql_statement = "INSERT INTO loans (dateOfLoan, dateToReturn,
dateReturned, lateFee, membershipNumber, ISBN, employeeInCharge) VALUES('" +

```

```

str(date_of_loan) + "','" + str(date_to_return) + "','" + str(date_returned)
+ "','" + str(latefee) + "','" + str(member) + "','" + str(chosen_isbn) + "','" +
str(employee) + "');"

    print(sql_statement);

#ratings 1 out of 5
rates = random.randint(1,5)

if(rates < 2):

    #random rating from 0 to 10
    rate = random.randrange(11)

    rate_statement = "INSERT INTO ratings VALUES (" + str(member)
+ ", " + str(chosen_isbn) + ", " + str(rate) + ");"

    print(rate_statement)

#set late fee back to 0.17
latefee=0.17

f.close();

```