SEMINARSKI RAD  
Iz predmeta *„UBI“*

„Library managment system“

STUDENTICA  
  
EMINA KOĆIĆ

Tuzla, SEPTEMBAR 2019. godine

Sadržaj

[1. Baza i tabele 4](#_Toc19929647)

[2. Autentifikacija Baze Podataka 5](#_Toc19929648)

[3. Autorizacija Baze Podataka 8](#_Toc19929649)

[3.1 Kreiranje prijavnih naloga 8](#_Toc19929650)

[4. Oporavak Podataka 20](#_Toc19929651)

[5. Indeksi, Uskladištene Procedure i Okidači 23](#_Toc19929652)

[5.1 Indeksi 23](#_Toc19929653)

[5.2 Uskladištene Procedure 27](#_Toc19929654)

[5.3 Okidači 31](#_Toc19929655)

[6.Export Podataka u Microsoft Access 34](#_Toc19929656)

**Uvod**

*Administracija baze podataka odnosi se na cijeli niz aktivnosti koje provodi administrator baze podataka kako bi osigurao da je baza podataka uvijek dostupna po potrebi. Ostali usko povezani zadaci i uloge su sigurnost baze podataka, nadgledanje baze podataka i rješavanje problema te planiranje budućeg rasta.*

*Administracija baze podataka važna je funkcija u svakoj organizaciji koja ovisi o jednoj ili više baza podataka.*

*Primarna uloga administracije baze podataka je osiguravanje maksimalnog vremena za bazu podataka tako da je uvijek dostupna kad je potrebno. To će obično uključivati ​​proaktivno periodično praćenje i rješavanje problema. To zauzvrat uključuje i neke tehničke vještine DBA-e. Pored detaljnog poznavanja baze podataka o kojoj je riječ, DBA će također trebati znanje i možda obuku na platformi (motor i operativni sustav baze podataka) na kojoj se baza podataka izvodi.*

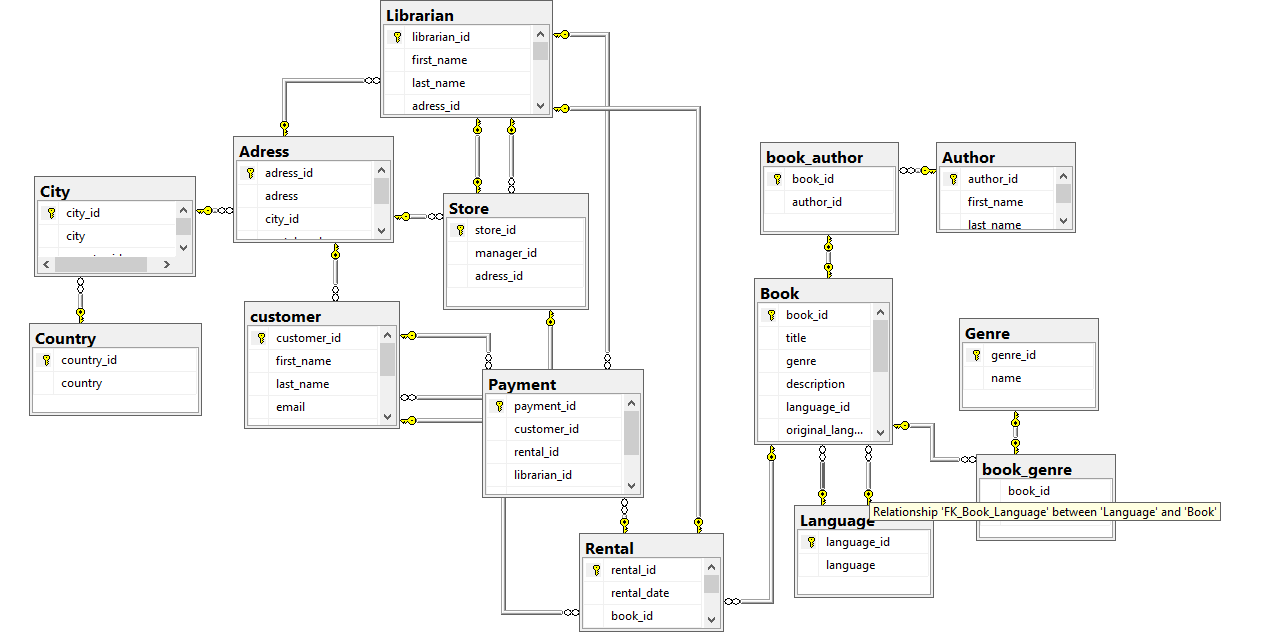
*DBA je obično odgovoran i za ostale sporedne, ali još uvijek kritično važne zadatke i uloge. Neki od njih uključuju:*

* *Sigurnost baze podataka: Osiguravanje da samo ovlašteni korisnici imaju pristup bazi podataka i štite je protiv bilo kojeg vanjskog, neovlaštenog pristupa.*
* *Podešavanje baze podataka: podešavanje bilo kojeg od nekoliko parametara za optimizaciju performansi, kao što su raspoređivanje memorije poslužitelja, fragmentacija datoteke i upotreba diska.*
* *Sigurnosna kopija i oporavak: Uloga DBA je osigurati da baza podataka ima primjerene procedure izrade sigurnosnih kopija i oporavka za oporavak od slučajnog ili namjernog gubitka podataka.*
* *Izrada izvješća iz upita: DBA-ovi se često pozivaju na generiranje izvještaja pisanjem upita koji se zatim izvode prema bazi podataka.*

*U nastavku je prikazana baza podataka koja je kreirana u svrhu zadovoljavanja potreba sistema biblioteke, u kojoj se nalazi svi potrebni podaci za rad aplikacije.*

# 1. Baza i tabele

Baza podataka “Library” je predstavljena sa 14 tabela, kao što je prikazano na dijagramu na sljedećoj slici:



Registrovani korisnici mogu koristiti knjige iz biblioteke. Tabela korisnik (Customer) sadrži podatke o korisniku : njegov ID, ime, prezime, adresu stanovanja, e-mail adresu, datum registrovanja, broj posuđenih knjiga, te poslovnicu u kojoj je registrovan korisnik.

Tabela knjiga sadži sve potrebne podatke o određenoj knjizi : ID knjige, naslov, autora knjige, žanr, kratak opis, jezik, te jezik s kojeg je prevedena, datum kad je izdata, cijenu za period na koji je knjiga izdata.

U tabeli autor nalaze se podaci o autoru : ime autora, njegovo prezime, te bigrafija.

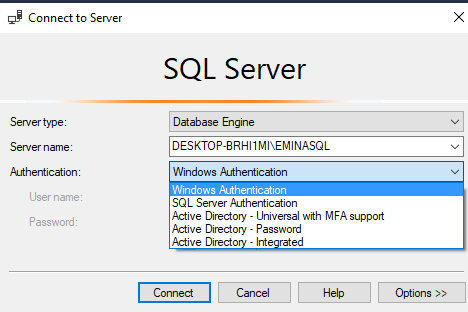
Tabela žanr sadrži žanrove, a tabele „book\_genre“ i „book\_author“ imaju svrhu da omoguće relacije „Many-to\_many između navedenih tabela.

Tabela „Librarian“ (Bibliotekar/ka)sadrži podatke o bibliotekaru, njegov ID, ime, prezime, sliku, podatke za pristup sistemu, i odjel u kojem je zaposlen.

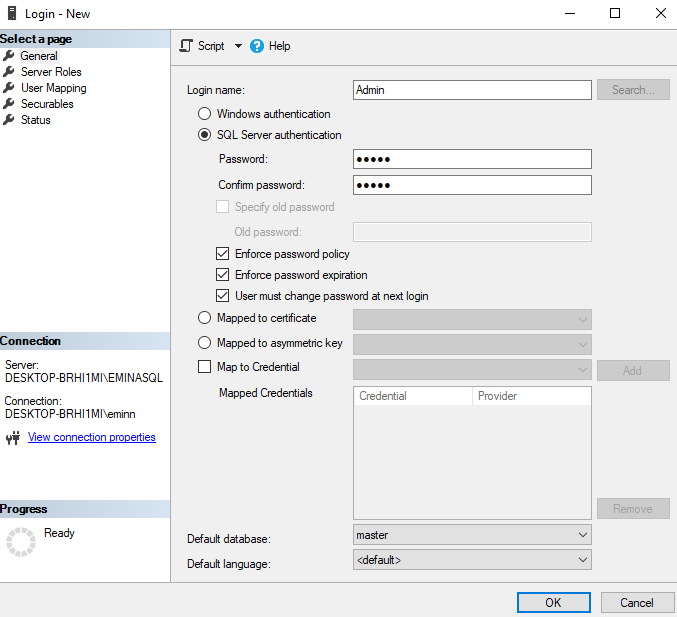
Tabela „Rental“ sadrži podatke o izdavanju knjige, datum izdavanja, predviđeni datum za vraćanje knjige, ID knjige, ID korisnika, te ID bibliotekara. U tabeli „Payment“ sadržani su podaci o cijeni izdavanja.

# 2. Autentifikacija Baze Podataka

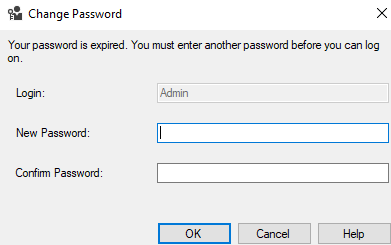
Za konektovanje na server korištena je Windows autentifikacija, što znači da je korisnik autentificiran od strane Windows operativnog sistema. Ova vrsta autentifikacije se naziva i „trusted“ iz razloga što sql server vjeruje Windowsu, pa nakon prijavljivanja na sam Windows, server prepoznaje korisnika, te ne zahtjeva ponovni unos identifikacijskih podataka.



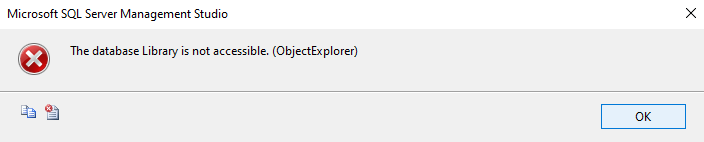
Kreiraću Windows lokalni nalog sa imenom Admin i nekim jednostavnim passwordom koji će se morati pri prvom prijavljivanju korisnika Admin promijeniti, jer je uključeno „Enforce password policy“ i „User must change password at next login“.



Pri prvom prijavljivanju mijenjamo password :



Ovako kreiran račun neće značiti pristup određenoj bazi podataka, već samo pristup na srver. Kada se ulogujemo sa korisnickim nalogom Admin nećemo moći pristupiti svojoj bazi podataka, pokazaće nam sljedeću grešku :



# 3. Autorizacija Baze Podataka

Autorizacija je sigurnosni mehanizam koji se koristi za određivanje privilegija korisnika / klijenta ili razine pristupa koji se odnose na resurse sustava, uključujući računalne programe, datoteke, usluge, podatke i značajke programa. Autorizaciji obično prethodi provjera autentičnosti za provjeru identiteta korisnika. Administratorima sustava (SA) obično su dodijeljene razine dozvola koje pokrivaju sve sistemske i korisničke resurse.

Tokom autorizacije sustav provjerava pravila pristupa autoriziranom korisniku i odobrava ili odbija pristup resursima.

To je process provjere šta korisnik smije da radi u sistemu.

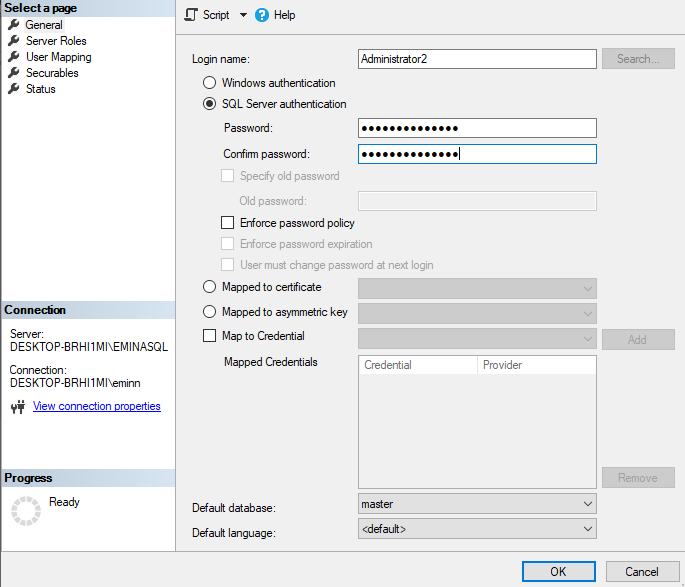
## 3.1 Kreiranje prijavnih naloga

Registracije se moraju mapirati na korisničke račune baze podataka da bi se moglo raditi s objektima baze. Korisnici se mogu dodati ulogama baze podataka, nasljeđujući sve skupove dozvola povezane s tim ulogama.

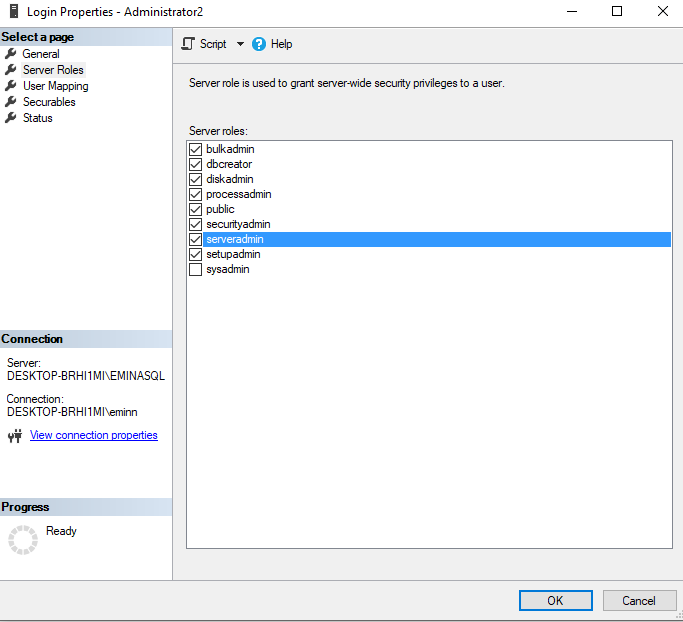
U nastavku ću prikazati pet različitih prijavnih naloga za pet različitih korisnika ove baze podataka, sa određenim pravima. Odrediću autorizacijska prava, pa tako će određeni korisnik imati pravo da uređuje, dodaje, briše i upravlja bazama podataka, može obavljati bilo koju aktivnost na poslužitelju.

Kako se samoj prijavi preko Windowsa dodjeljuje uloga sysadmina ona ima i sva autorizacijska prava na sesrver, kao i na baze podataka.

Kreiraću još jedan nalog sa loginom na SQL Server, koji će biti zadužen za upravljanje serverom i podacima onda kada sistemski admin nije u mugoćnosti.Login ime ću postaviti “Administrator2 i dodijeliću mu password :

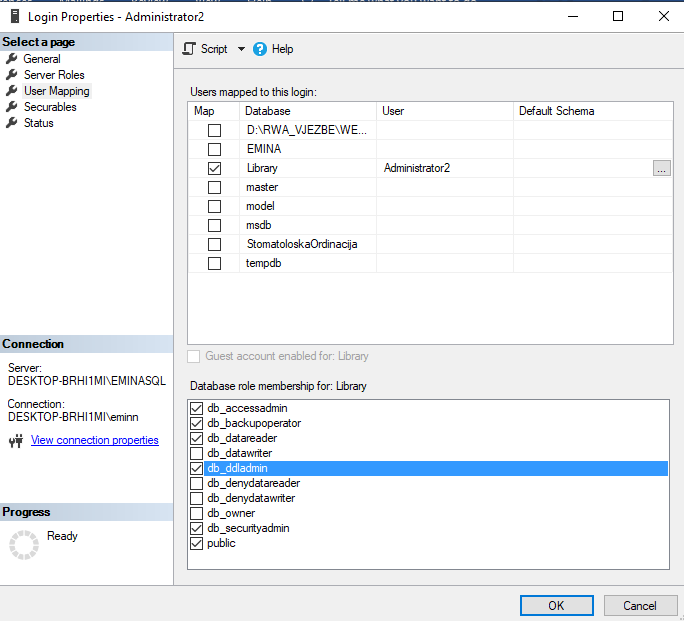


Zatim dodjeljujem serveska fixna prava navedenom korisniku :



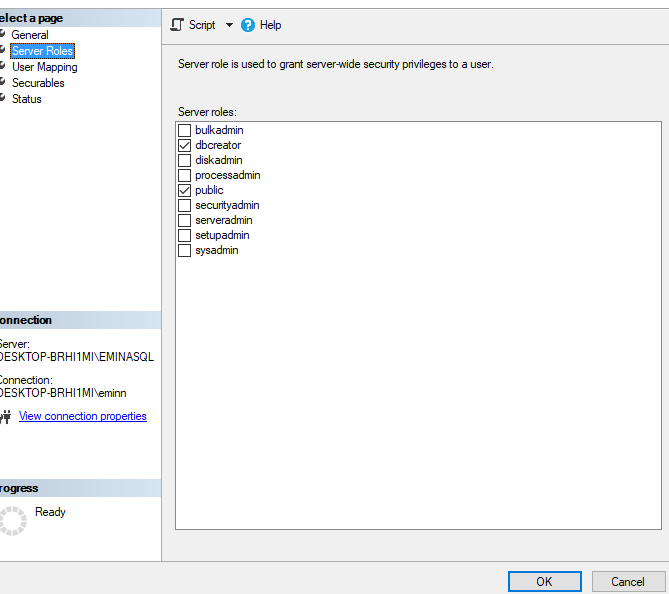
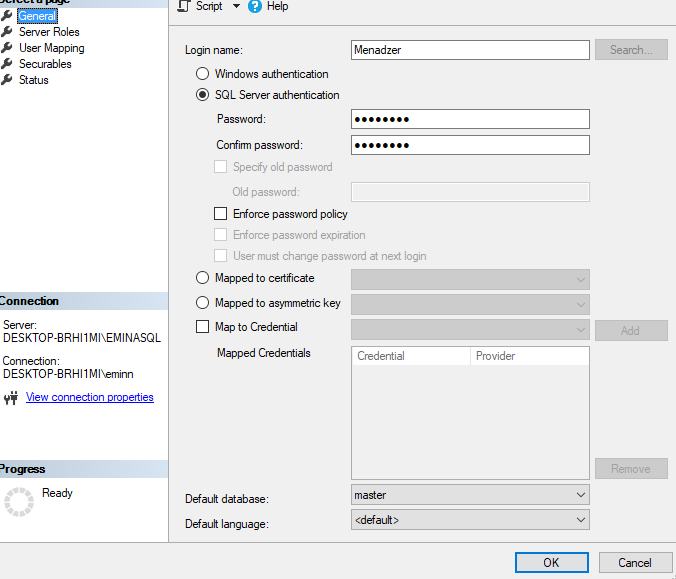
Dat ću mu sva prava, osim sysadmin. Imaće pravo da upravlja prijavama i njihovim svojstvima. Može dodijeliti dopuštenja, i oduzeti dozvole na razini servera. Uz to, može resetovati lozinke za prijavu na SQL Server.

Zatim, dodjeljujem korisniku prava na nivou baze :

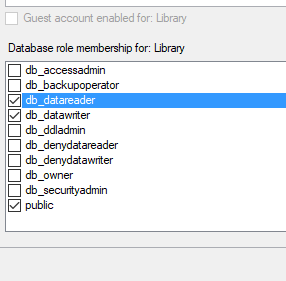


Ovaj korisnik može upravljati dozvolama kao i dodavati, mijenjati i brisati podatke iz naše baze podataka.

Sljedeće, kreirat ću prijavni nalog za menadžera biblioteke. Dodjeljujem mu serversko pravo „dbcreator“. Moći će da dodaje, uređuje, briše i upravlja bazama podataka.



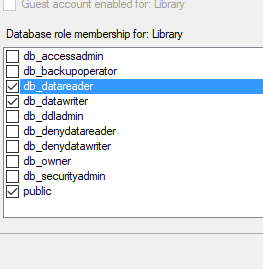
Od prava na bazi podataka dodjeljujem mu sljdeća prava :



Menadžer će moći dodavati, brisati ili mijenjati podatke u svim tabelama baze Library, ali nema pristup ostalim bazama na serveru.

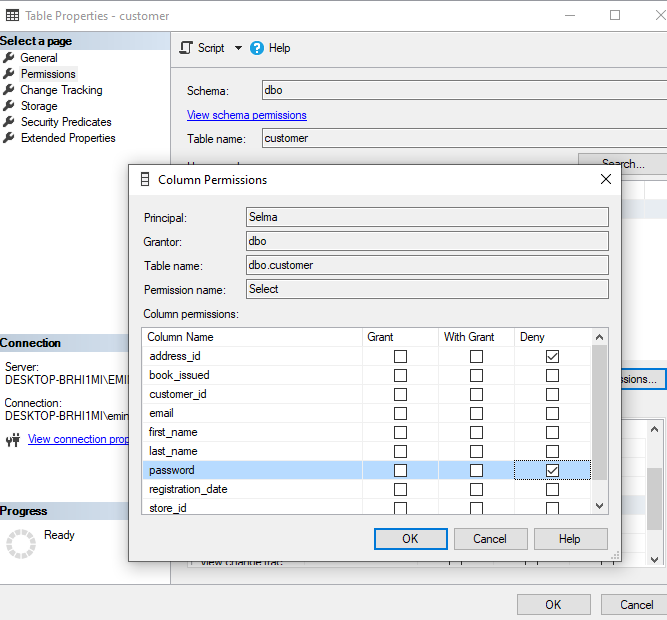
Sljedeći korak je kreirati račun za jednu od bibliotekarki, koja će moći čitati neke podatke, dodavati učlanjene korisnike biblioteke, ali neće moći upravljati serverom,ili vidjeti podatke drugih korisnika.

Jedan od načina je da kreiram račun s imenom Selma, i postavim password na Selma, na isti način kao što sam kreirala prethodne račune. Server roles će biti onemogućeni a od prava na bazu dodaću za sad datareader i datawriter, za bazu Library.



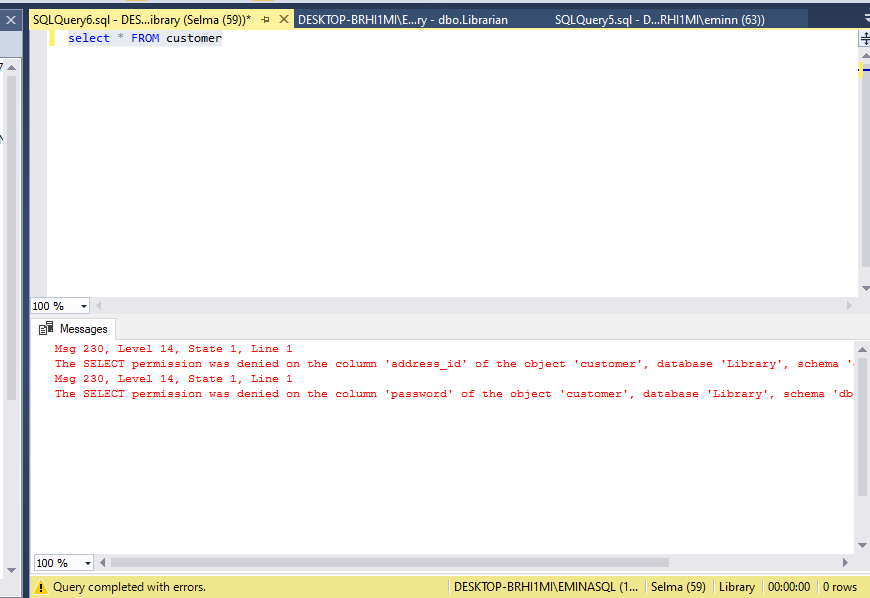
Omogućila sam čitanje podataka iz tabela i kreiranje novih. Međutim biblioterka ne bi trebala vidjeti sve podatke kao što su login podaci drugih zaposlenika, ili login podaci korisnika biblioteke, kao ni uređivati podatke koji su vezani za zaposlene, pa ću sljedećim koracima zabraniti navedeno.

U tabeli „Customers“ u postavkama urediću Permissions na način da prvo unesem korisnika za kojeg želim promijeniti permisije. Unosim ime Selma, pa u permisijama označavam deny za kolone password i adress.



Na isti način onemogućit ću da vidi username, password, e-mail i adress\_id iz tabele Librarian.

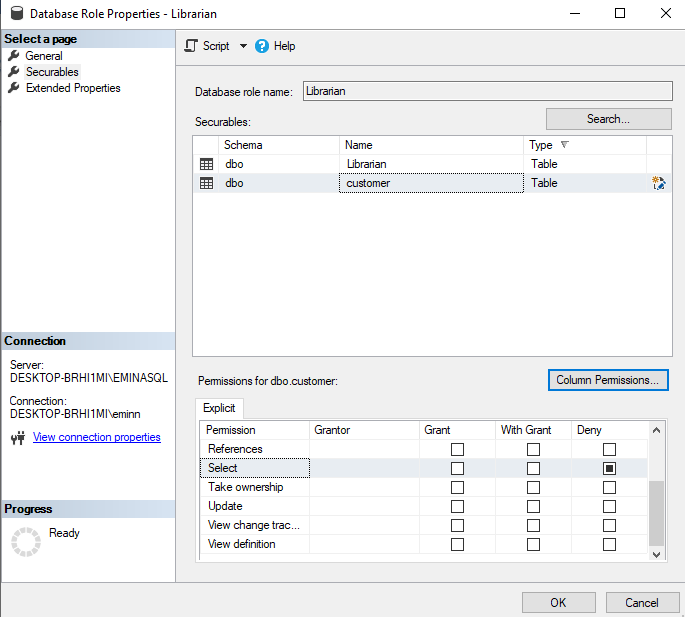
Ako pokušamo selektovat sve kolone iz tabele customer, pri čemu smo ulogovani na račun Selma, pojaviće nam se sljedeći error :



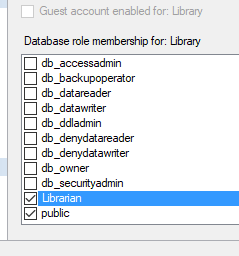
Bolji način je da prvo kreiram Database role u Security opciji za svoju bazu Library. Ovaj role će važiti za sve bibliotekare.

Onemogućit ću Select iz tabele Librarian za kolone adress\_id, username, password i e-mail, te za kolonu password iz tabele Customer.

Za sve ostale tabele ću omogućiti uređivanje, dodavanje, brisanje, te čitanje iz kolona.



Nakon toga ću dodati novog bibliotekara, kreirati username : Emir i password : Emir i dodijeliti mu role Librarian.

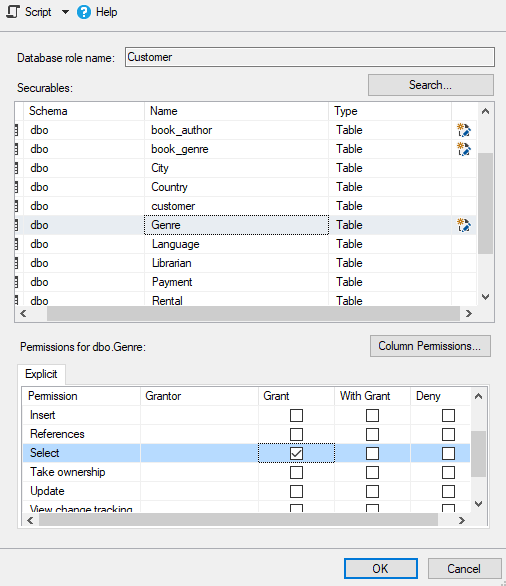


Na isti način moći ćemo dodavati i druge bibliotekare, sa istim permisijama.

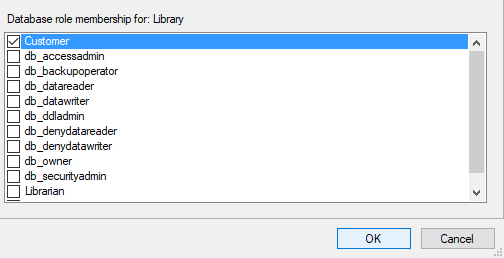
Naredni korak je da kreiramo nalog za regitrovanog korisnika biblioteke, odnosno customera. On neće imati pravo da mijenja podatke iz baze, briše, ili uređuje, niti da čita podatke iz pojedinih tabela. Moći će samo da čita podatke o knjigama, i autorima knjiga.

Najprije ću kreirati Database role sa imenom Customer.

Dodijeliću mu prava na SELECT iz tabela : „Book“, „Author“ i „Genre“. Odnosno, moći će samo da čita podatke iz navedenih tabela.



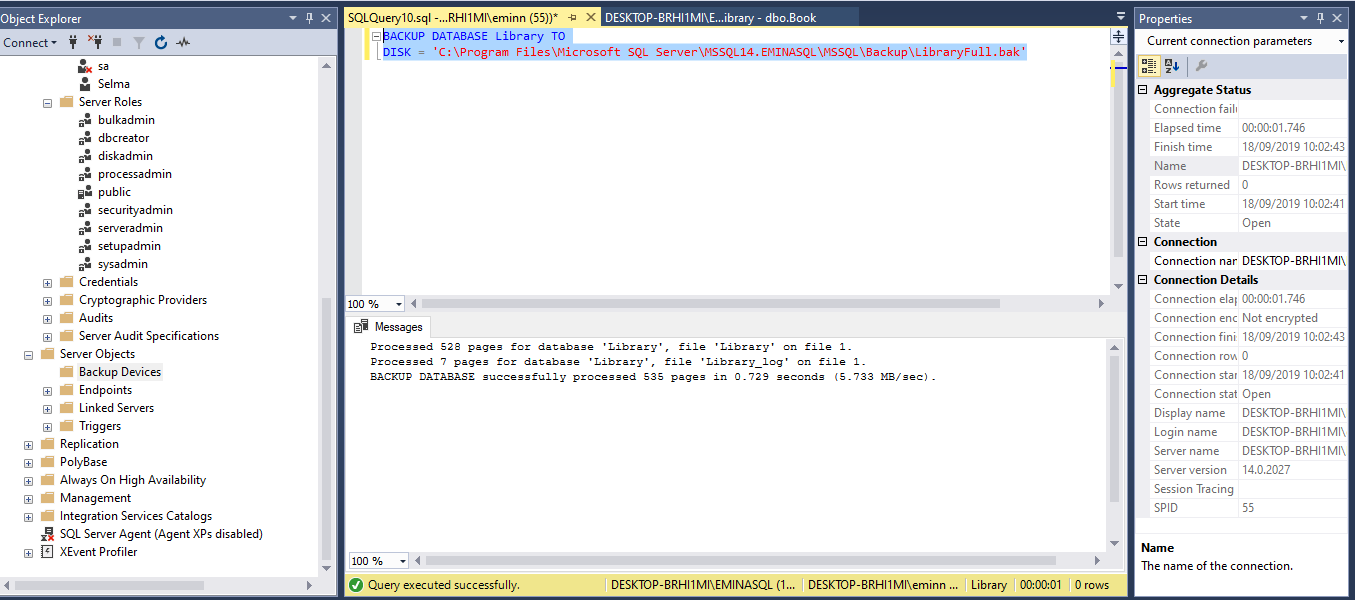
Nakon toga, kreiraću jednom customera sa imenom Gia i passwordom Gia123, te ću joj od prava na Library bazu dodijeliti Customer database role. Serverska prava neće imati.



# 4. Oporavak Podataka

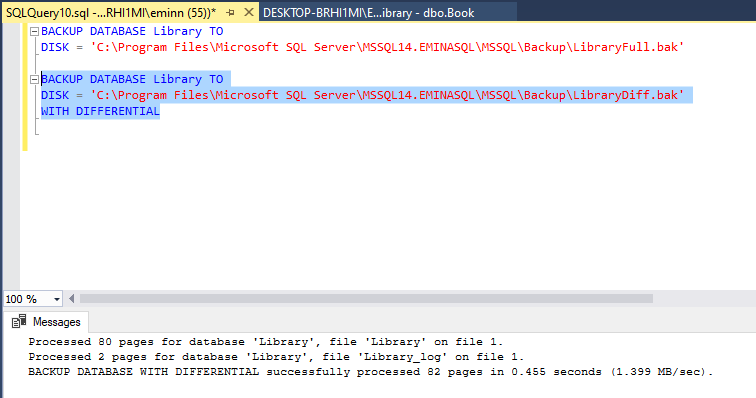
U ovom poglavlju potrebno je uraditi backup baze podataka, te simulirati crash I oporavak iste.

Najprije ću uraditi full backup podataka iz baze Library, na sljedeći način :

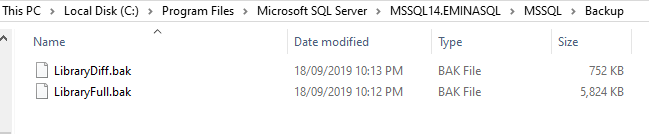


Nakon što je backup urađen u dokumentima se nalazi fajl sa nazivom LibraryFull.bak.

Izmjenit ćemo neke podatke iz tabele “Book”, a nakon toga ćemo odraditi diferencijalni backup, što znači zadnje izmjene od full backupa.



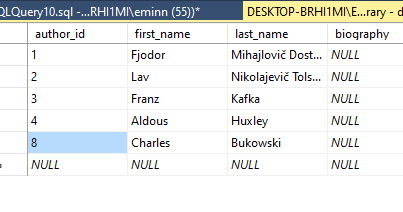
Sada se na računaru pojavljuju 2 fajla, jedan od potpunog, a drugi od diferencijalnog backupa :



Nakon uspješnih backupa izbrisaćemo neke podatke iz tabela :



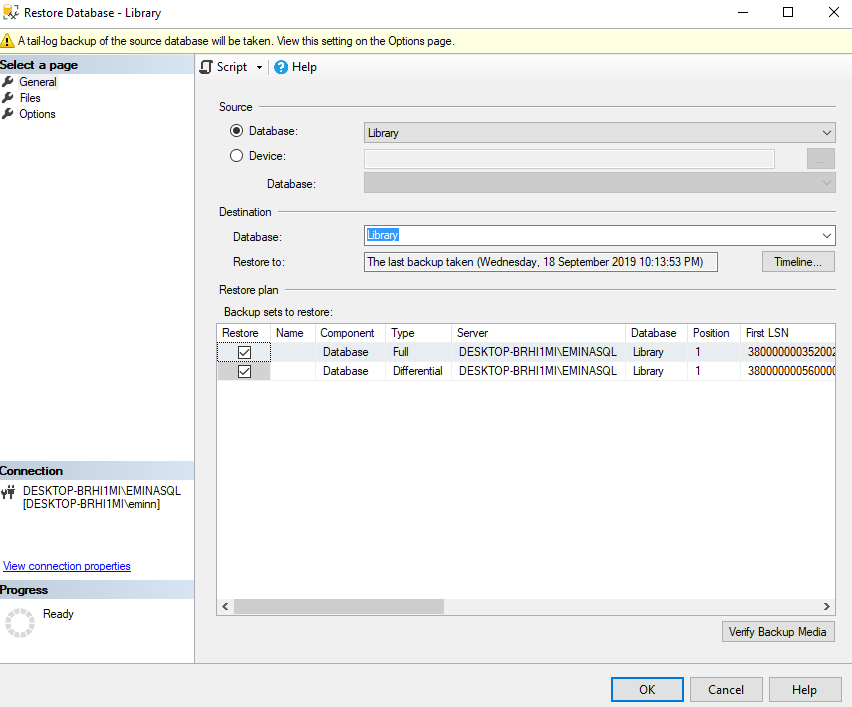
Poslije brisanja :



Kako vratiti izgubljene podatke ?

Nakon što uradimo recovery baze podaci iz prikazane tabele će se vratiti.

Desnim klikom na bazu podataka odabiremo opciju Restore, nakon čega nam se otvara prozor kao na slijedećoj slici :



Prikazuje nam dva backup-a koje smo odradili Full I Differential buckup kao i vrijeme kad je urađen zadnji backup, a opcija timeline nam pokazuje sve backup-e, gdje možemo odrediti koje tačno vrijeme želimo vratiti. Odabiremo Full restore.

Restore je uspješno izvršen I svi izbrisani podaci iz tabele Author su vraćeni !

# 5. Indeksi, Uskladištene Procedure i Okidači

## 5.1 Indeksi

Kako SQL Server pristupa podacima ?

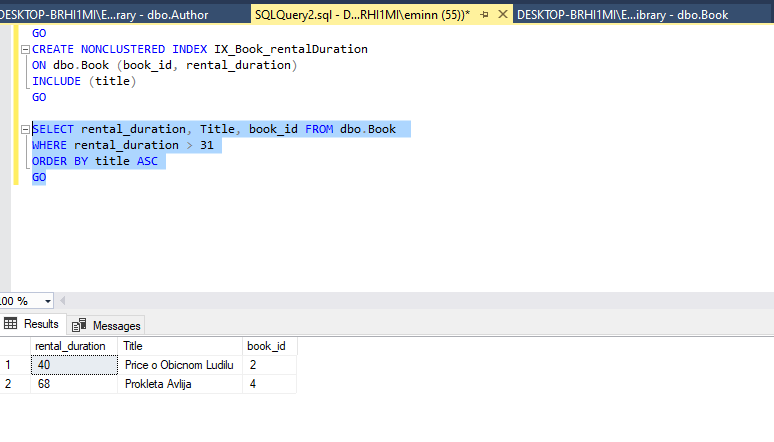
1. SQL Server čita sve stranice u tabeli.
2. SQL Server koristi inekse kako bi pronašao zapise.

Kako bismo što brže pretražili skup podataka u našoj bazi, bilo bi dobro da indeksiramo podatke, kako bismo značajno olakšali pretragu.

Indexi mogu biti „Clustered“ i „NonClustered“ Svaka tabela može imati jedan clustered index, što je najčešće primarni ključ tabele, i 250 nonclustered indexa.

Najprije ću napraviti nonclustered index za tabelu „Book“. Olakšaću pretragu za knjigama koje su predugo iznajmljene, i trebale su biti vraćene.

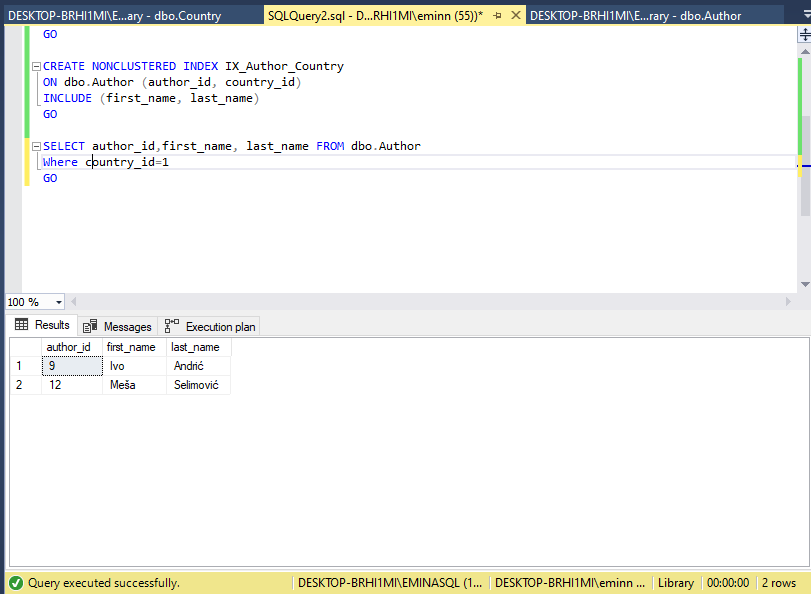
Uradila sam to na sljedeći način :



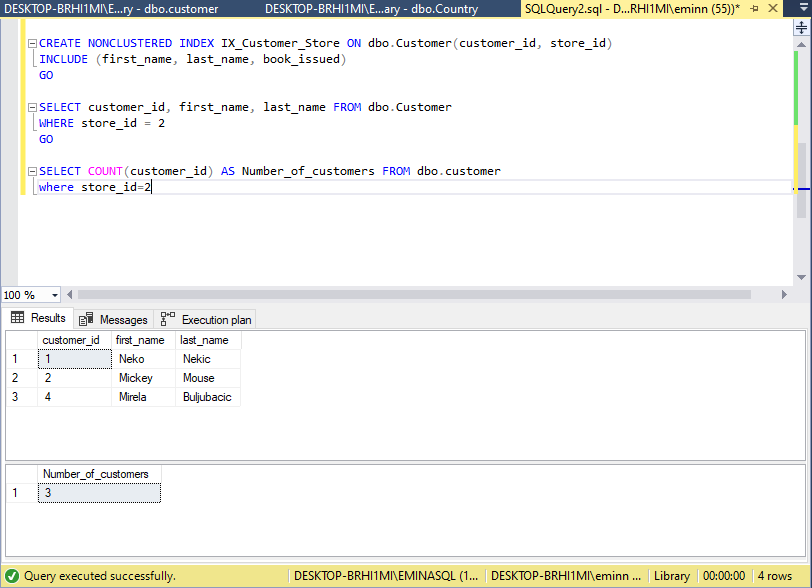
U kreirani index **IX\_Book\_rentalDuration** uključila sam i Title knjige, da bi bilo vidljivo o kojoj knjizi je riječ. Korist ovog indexa se vidi operacijom SELECT.

Sljedeći index koji sam kreirala je **IX\_Author\_Country** koji će nam olakšati pretragu, ako npr. želimo izlistati sve autore iz naše države. Koristila sam tabelu Author, i uključila kolone ime autora i prezime. Nakon toga sam selektovala kolone ID autora, njegovo ime i prezime gdje je država iz koje potiču pod ID-om 1, odnosno BiH.

Ovi koraci su prikazani na sljedećoj slici :

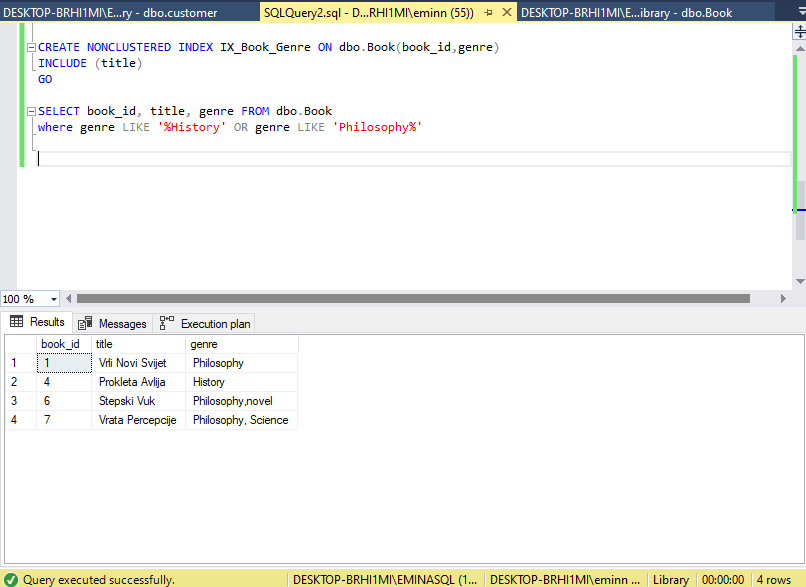


Kreiraću i indeks **IX\_Customer\_Store** za tabelu „Customer“ :



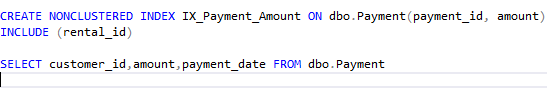
Indeks će nam pomoći da predražimo korisnike koji su učlanjeni baš u našoj biblioteci, kao i da odredimo broj korisnika koji su tu učlanjeni.

Ako želim da pretražim knjige koje pripadaju određenom žanru,a ne želim da gledam naslov po naslov žanr svake knjige, uradiću to na sljedeći način:



Kreirala sam index **IX\_Book\_Genre**, preko kog možemo da pretražimo literaturu željenog žanra.

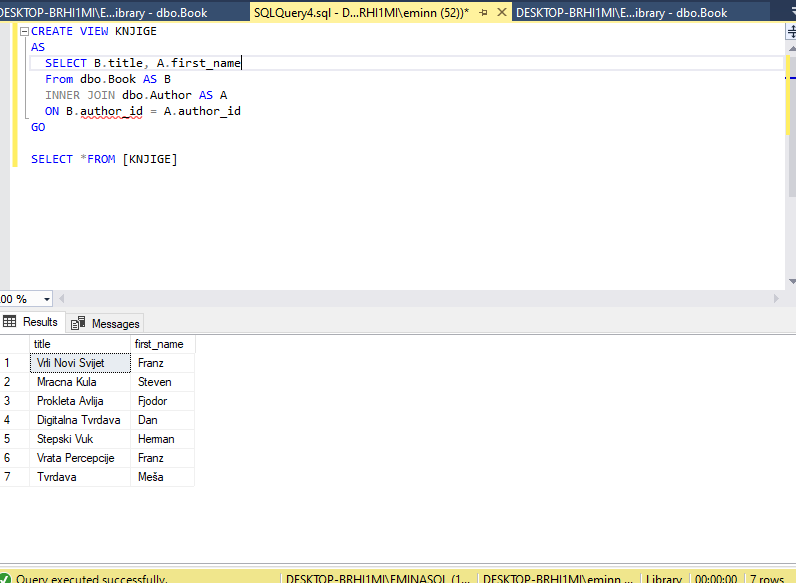
Kreirat ću još jedan index **IX\_Payment\_Amount** gdje za određenog customera mogu vidjeti podatke o iznosu koji treba da plati, ako bih recimo hjela izdati račun.



## 5.2 Uskladištene Procedure

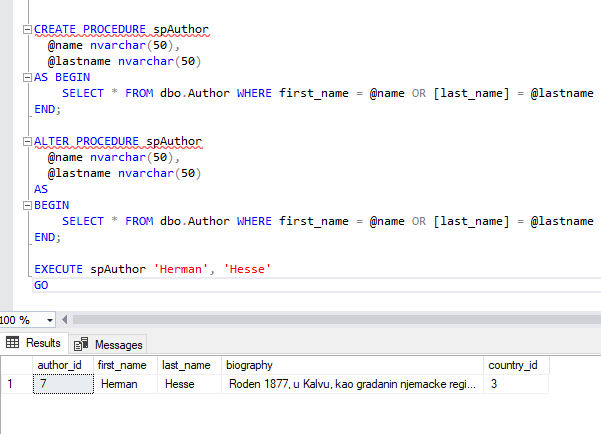
Interakcija aplikacije sa serverom baze podataka se može izvršiti komandama pohranjenim na serveru, koje se nazivaju storne procedure, i kao takve pozivaju se iz aplikacije. Storne procedure omogućavaju nam bolju sigurnost, modularnost, odvojenost logike baze podataka od aplikacije, bolje preformanse...

Napraviću prvo jedan pogled (View) nad tabelama autor i knjige. Ovaj pogled je jako koristan kako bi se sakrile nepotrebne i suvišne informacije od korisnika, i mogao bi samo da lista nazive knjiga i njihove autore

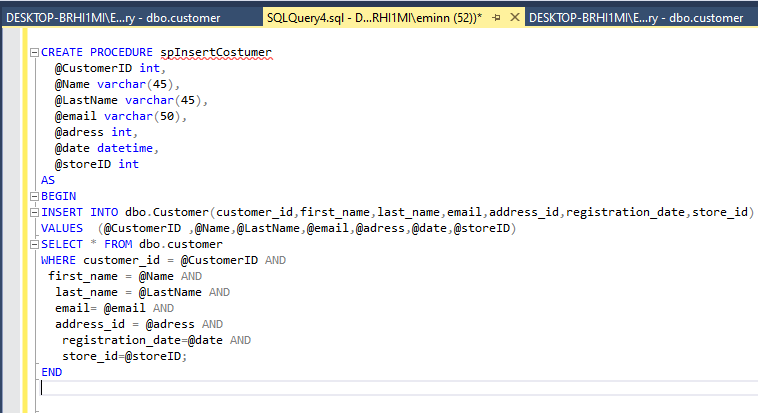


Zatim ću napraviti nekoliko stornih porecdura u nastavku...

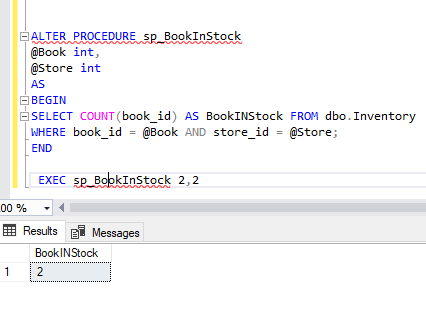
Napravit ću proceduru za pretragu autora po imenu ili prezimenu na sljedeći način :



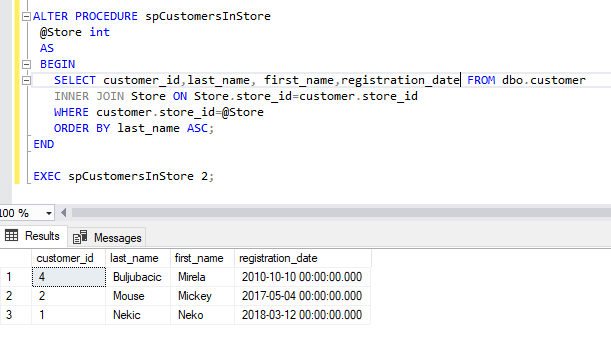
Zatim, proceduru koja će mi omogućiti da unesem nove korisnike biblioteke :



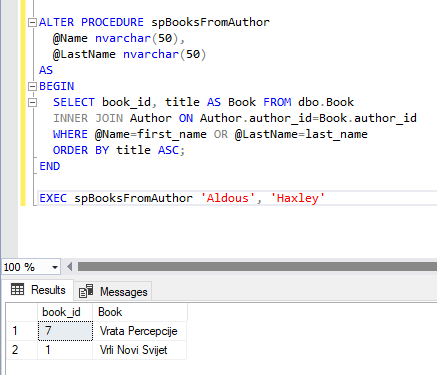
Biblioteka može imati puno kopija jedne knjige, u jednom ili više odjela. Sljedeća procedura će nam omogućiti da vidimo koliko knjiga određenog naslova ima u određenom odjelu :



Kada želimo da vidimo da li je neki korisnik učlanjen u našem odjelu, uradićemo to na sljedeći način :



Želimo li vidjeti npr. koje to sve naslove imamo od određenog autora na stanju u svim odjelima, uradićemo to ovako :

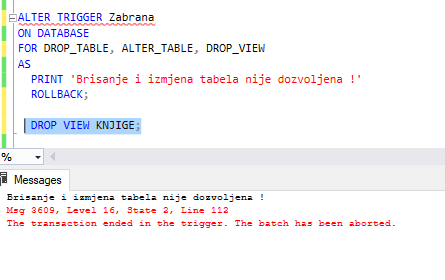


## 5.3 Okidači

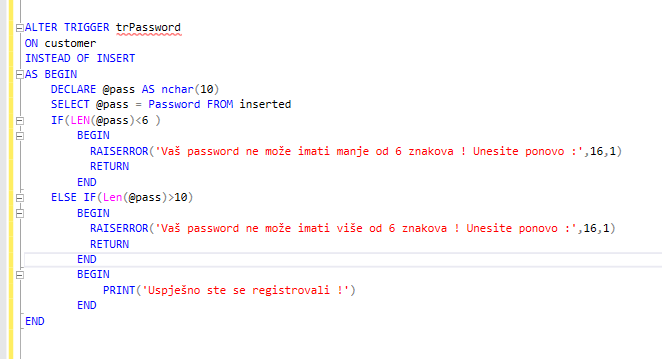
Okidač je dio koda koji se automatski izvršava kao odgovor na određeni događaj koji se dogodio na tabeli u bazi podataka.

Predstaviću primjer nekoliko triggera izvršenih na svojoj bazi podataka.

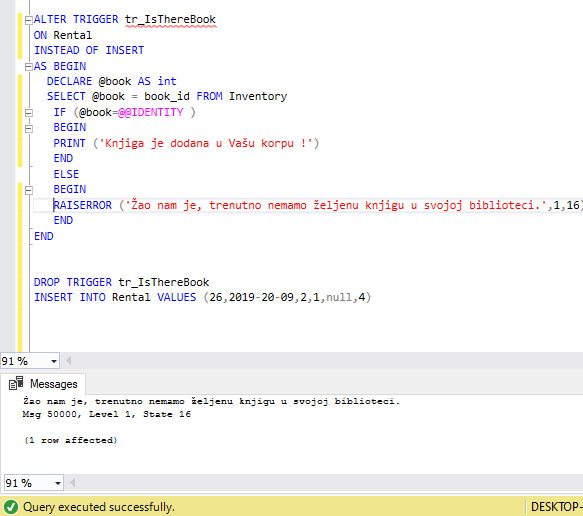
Prvo ću kreirati jedan trigger na bazi, koji bi preventivno zabranio brisanje ili uređivanje tabela, i brisanje pogleda :



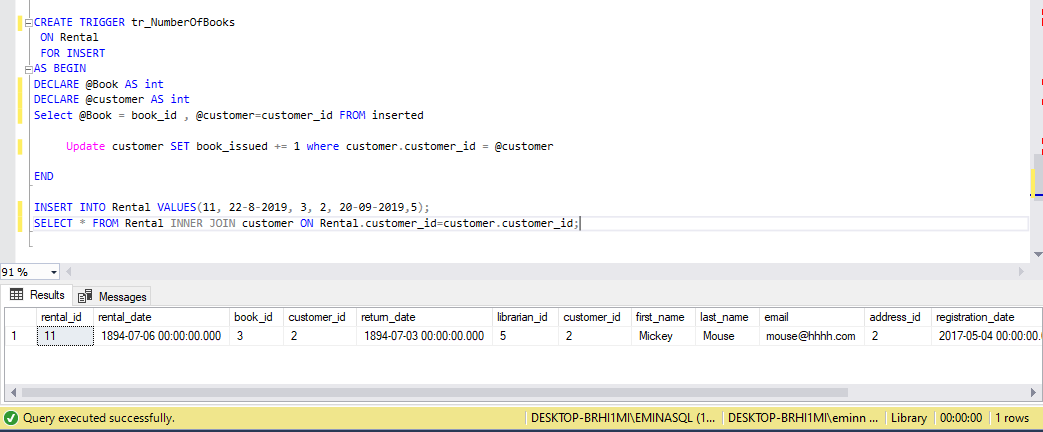
Slijedi okidač koji bi provjeravao da li je korisnik koji pokušava da se registruje pravilno unio password :



Napraviću i okidač koji će provjeravati da li postoji na stanju knjiga koju korisnik želi, u slučaju online provjere :

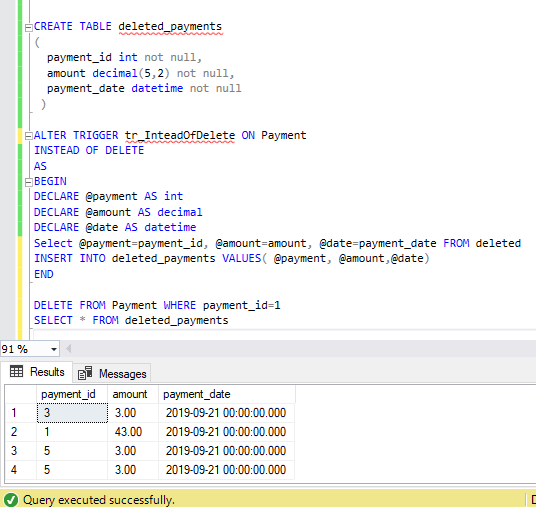


Sljedeći okidač nam služi da pri novom unosu u tabelu rental promijenimo i broj iznajmljenih knjiga u tabeli customer :



Kao što sam očekivala, broj iznajmljenih knjiga za korisnika pod ID brojem 2 se promijenio.

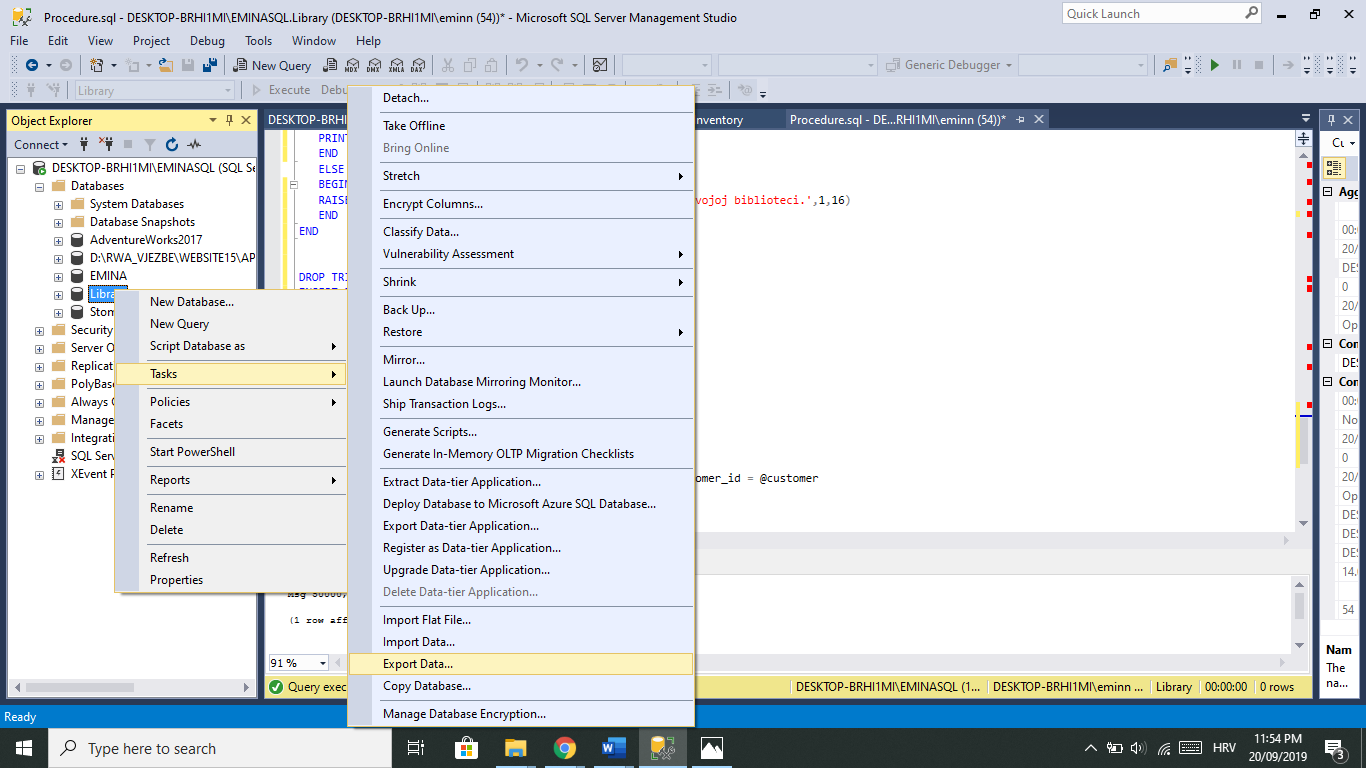
Posljednji okidač će sačuvati obrisane podatke u posebnoj tabeli. Prvo sam kreirala novu tabelu deleted\_payments, zatim napravila trigger koji će izbrisane podatke iz dabele Payment vratiti u ovu novu tabelu :



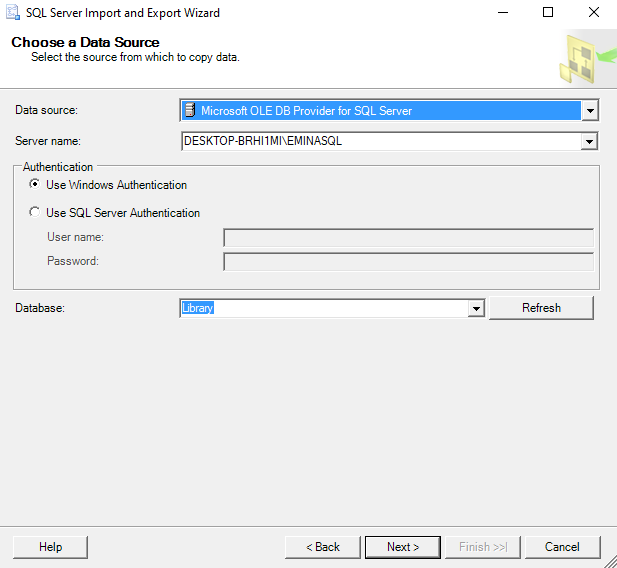
# 6.Export Podataka u Microsoft Access

Posljednji korak koji trebam izvršiti je export podataka iz Microsoft SQL Management Studia u Microsoft Access bazu podataka.

Desni klik na bazu podataka, i otvara se meni na kojem odabiremo „Tasks“ i „Export Data...“

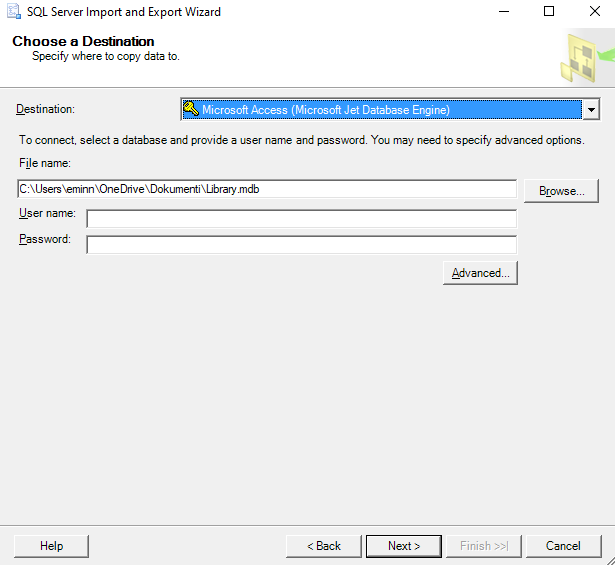


Nakon toga pojavlljuje se SQL Server Import and Expert Wizzard gdje biramo “Microsoft OLE DB Provider for SQL Server“ kao Data source. Server name postavimo je ime našeg servera. U dijelu „Authentification“ odabiremo da li je Windows ili SQL Server autentifikaicja i u dijelu Database odabiremo našu željenu bazu podataka iz SQL Servera.



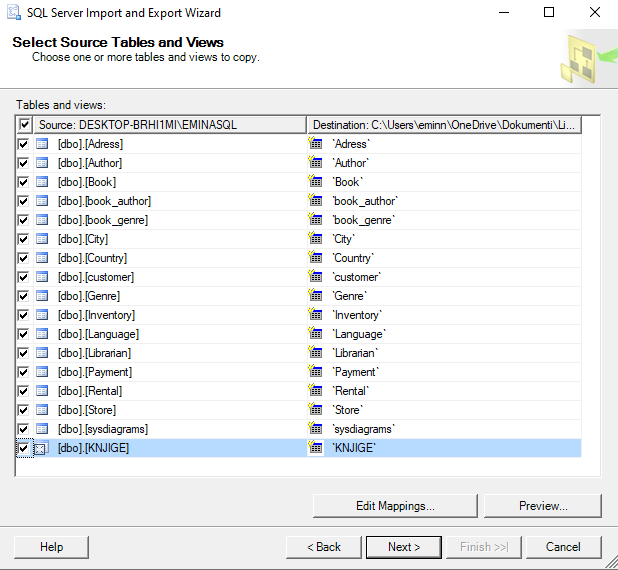
Zatim određujemo odredišnu destinaciju i postavljamo je na „Microsoft Access

(Microsoft Access Database Engine)“. Klikom na dugme „Browse...“ otvara se prozor gdje tražimo kreiranu Access bazu u koju želimo eksportovati podatke. Nakon sto to završimo kliknemo na „NEXT>“.



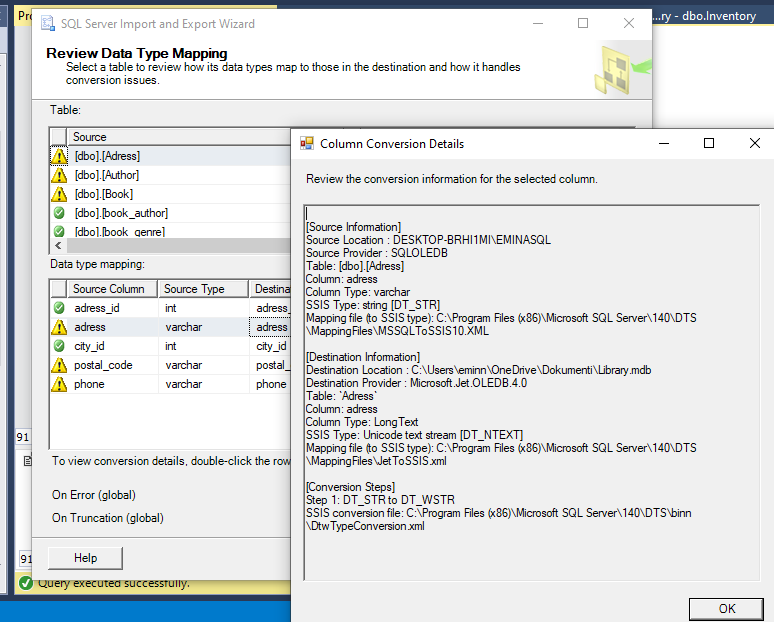
Sada se otvara mogućnost da kopiramo podatke iz tabela ili da napisemo query kojim ćemo definisati prenos podataka. Odabiremo opciju da kopiramo i „NEXT“.

Poslije tog nam se otvara prozor, gdje selektujemo tabele, i poglede koje želimo exportovati :



„Next“ i otvara nam se prozor sa informacijama o podacima i načinu exporta.

Ako kliknemo dva puta na tabelu, vidjećemo sve informacije konverzije i prenosa. :



Nakon exportovanja podataka podataka iz Microsoft SQL Management Studi-a u Microsoft Access bazu podataka izgled exportovanih baza je prikazan sljedećom slikom :

