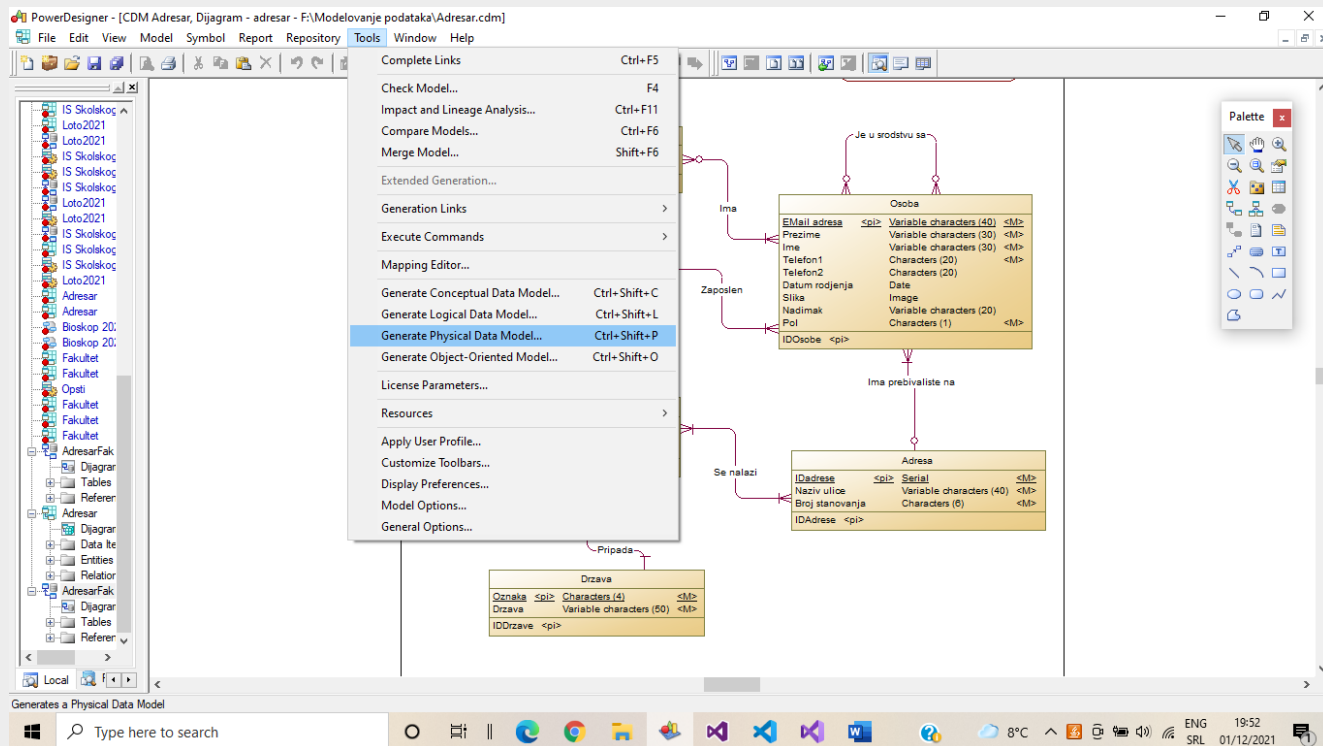
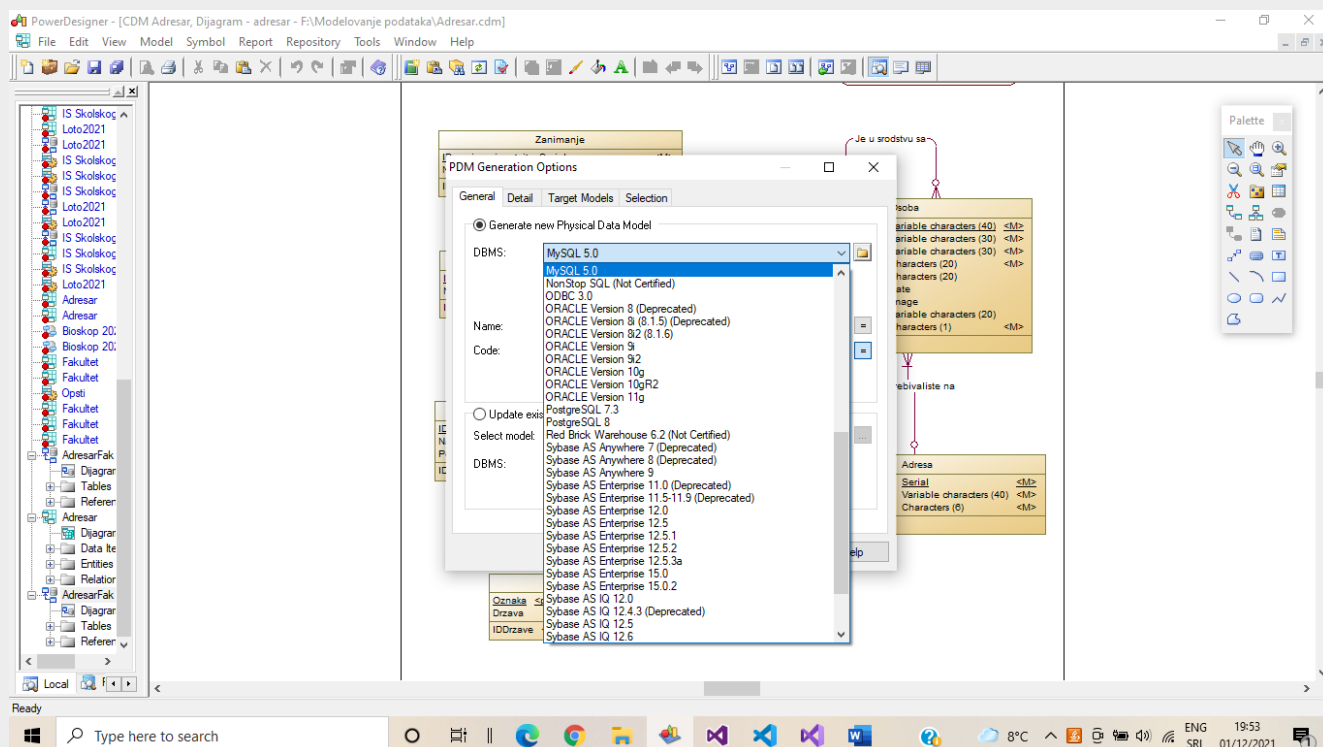


# TUTORIJAL ZA RAD SA SYBASE POWER DESIGNER CASE ALATOM (PDM)

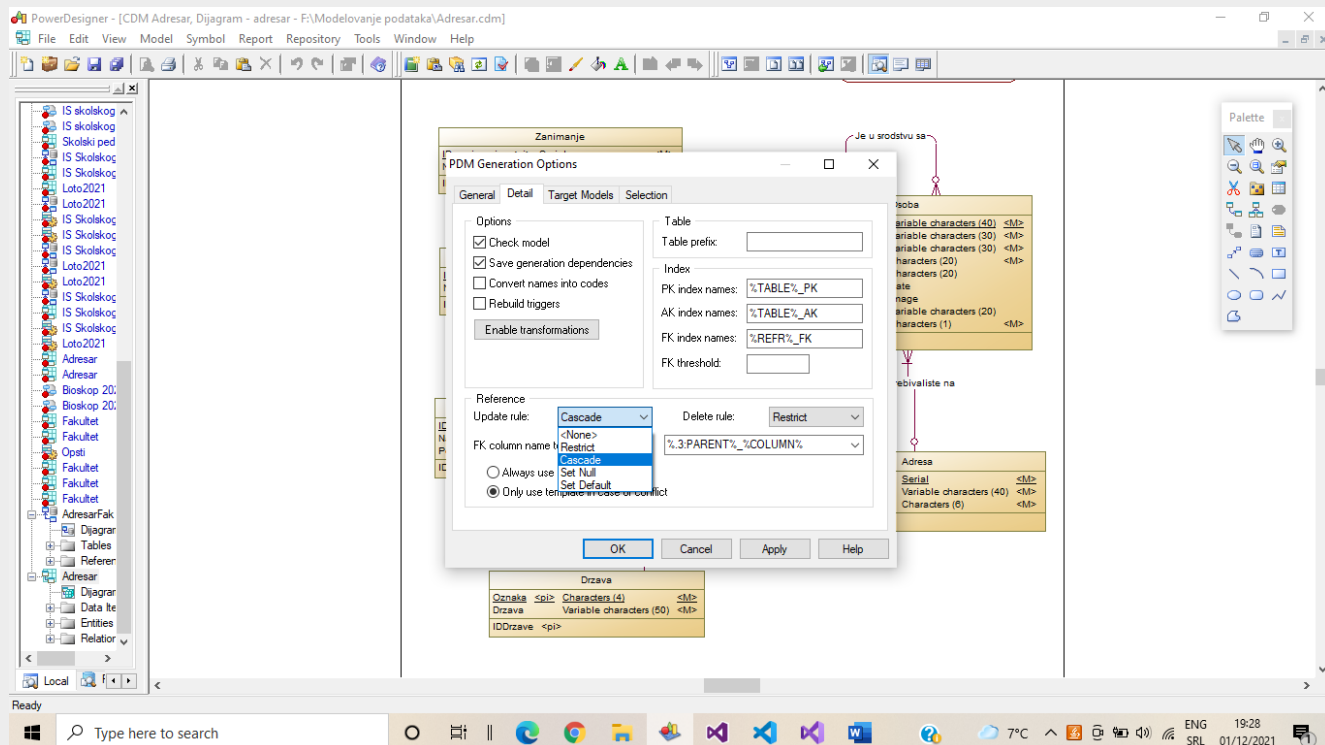
Kreiranje fizičkog modela podataka (Physical Data Model) na osnovu postojećeg CDM modela se vrši preko opcije glavnog menija CASE alata: **Tools-Generate Physical Data Model**. Alternativno se može prvo uraditi Relacioni model podataka nezavistan od implementacije u konkretnom sistemu za upravljanje bazama podataka (SUBP) (Logical Data Model), pa tek tada Physical Data Model, koji sadrži i uključuje karakteristike konkretnog SUBP-a.



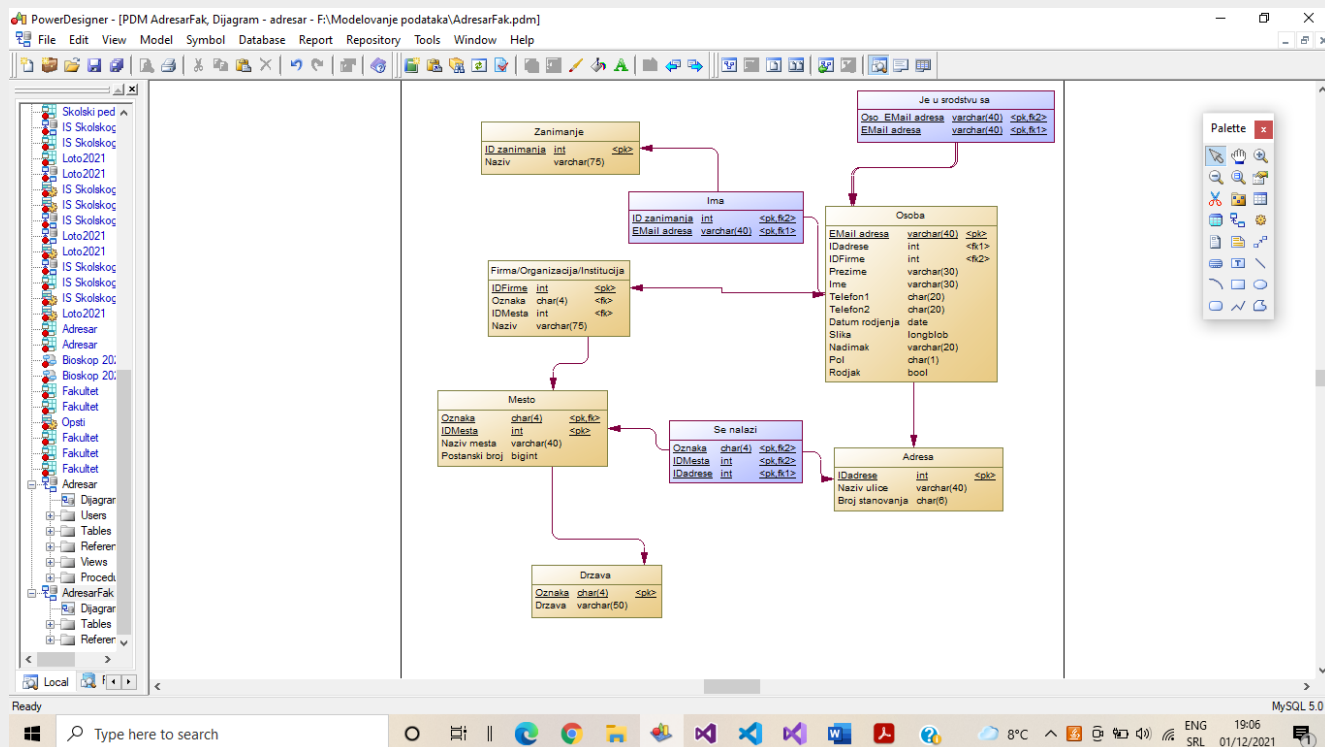
Kreiranje fizičkog modela podataka, drugi korak, jeste provera ili upis novog naziva modela (**Name**), pa izbor odgovarajućeg DBMS-a (SUBP-a) u kom se želi implementirati model.



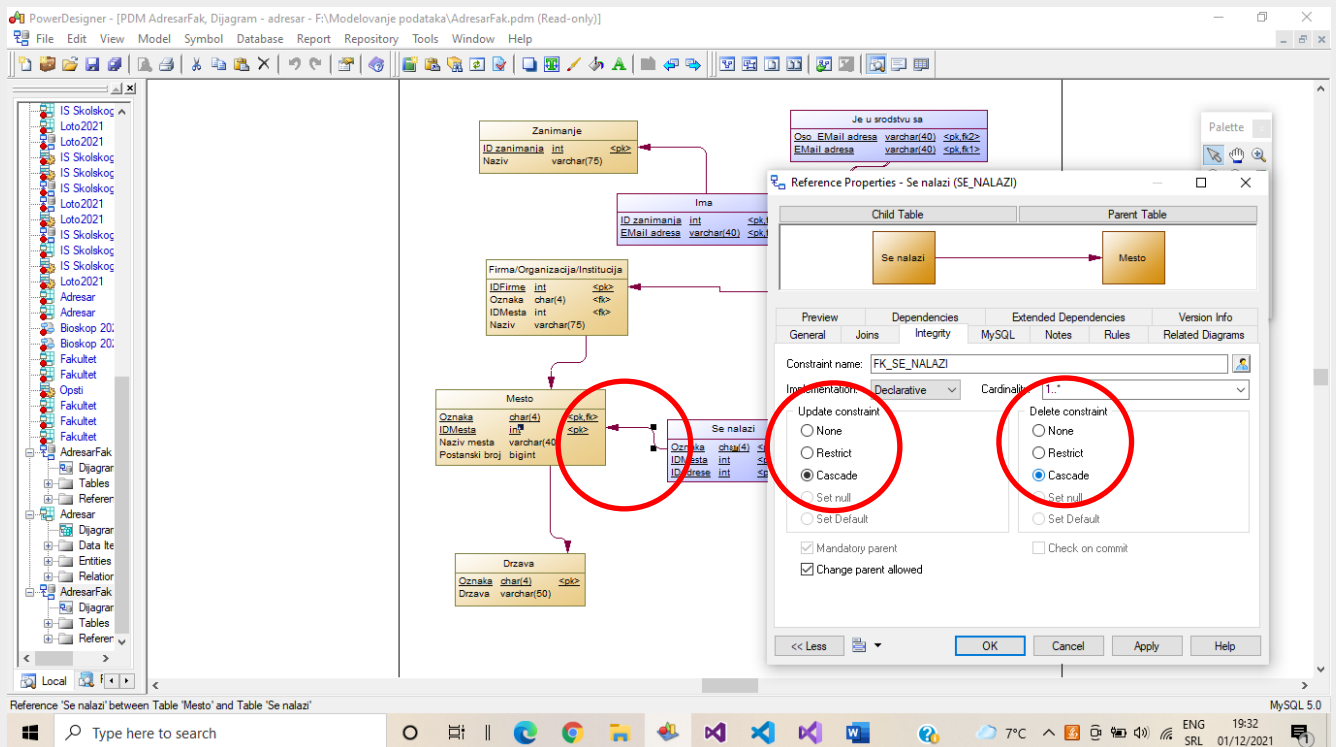
U kartici **Detail** se određuju načini imenovanja objekata u fizičkom modelu i određuju pravila za očuvanje referencijalnog integriteta, u listama grupe **Reference**: za izmenu podataka u međusobno povezanim tabelama (**Update rule**) kaskadno (**Cascade**) ili restriktivno (**Restrict**), kao i za brisanje podataka u međusobno povezanim tabelama (**Delete rule**) kaskadno ili restriktivno. Najčešća kombinacija koja odgovara većini situacija iz realnog sveta jesta: kaskadna izmena podataka i restriktivno brisanje.



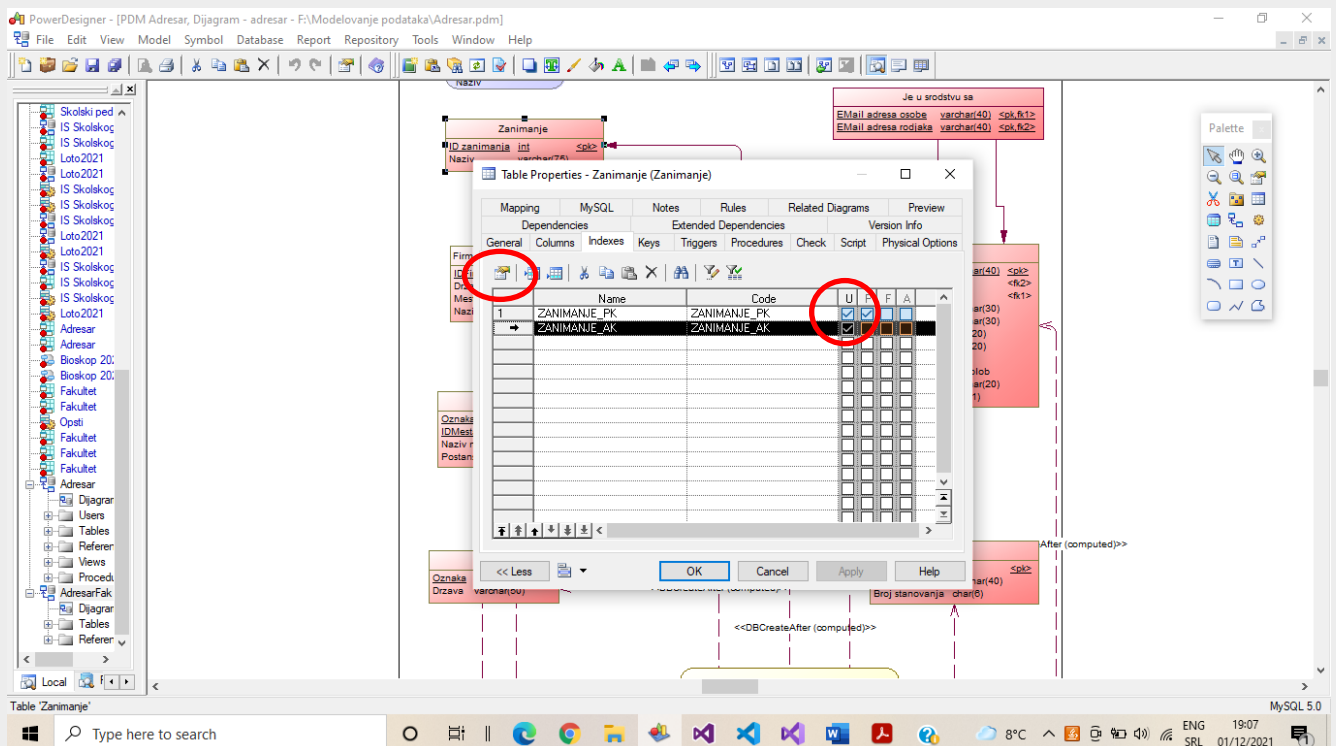
Kreirani fizički model podataka prikazan je na sledećoj slici:



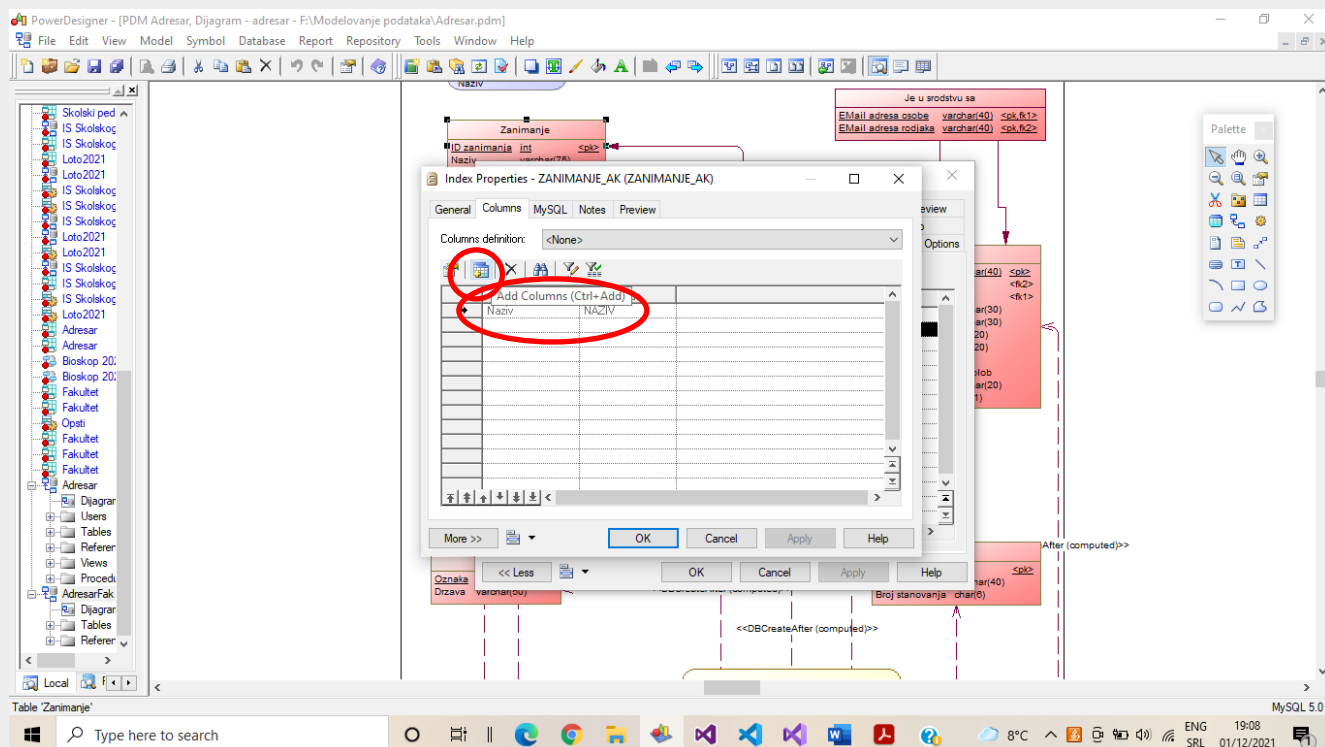
Ukoliko se žele postaviti drugačija pravila za očuvanje referencijalnog integriteta, mogu se promeniti osobine neke konkretne veze između tabela (**Reference - Properties**). Nakon otvora prozora sa osobinama veze, na kartici **Integrity** se može izabrati i neka druga kombinacija pravila od inicijalno postavljene prilikom prevođenja CDM modela u PDM: npr. kaskadna izmena i kaskadno brisanje podataka.



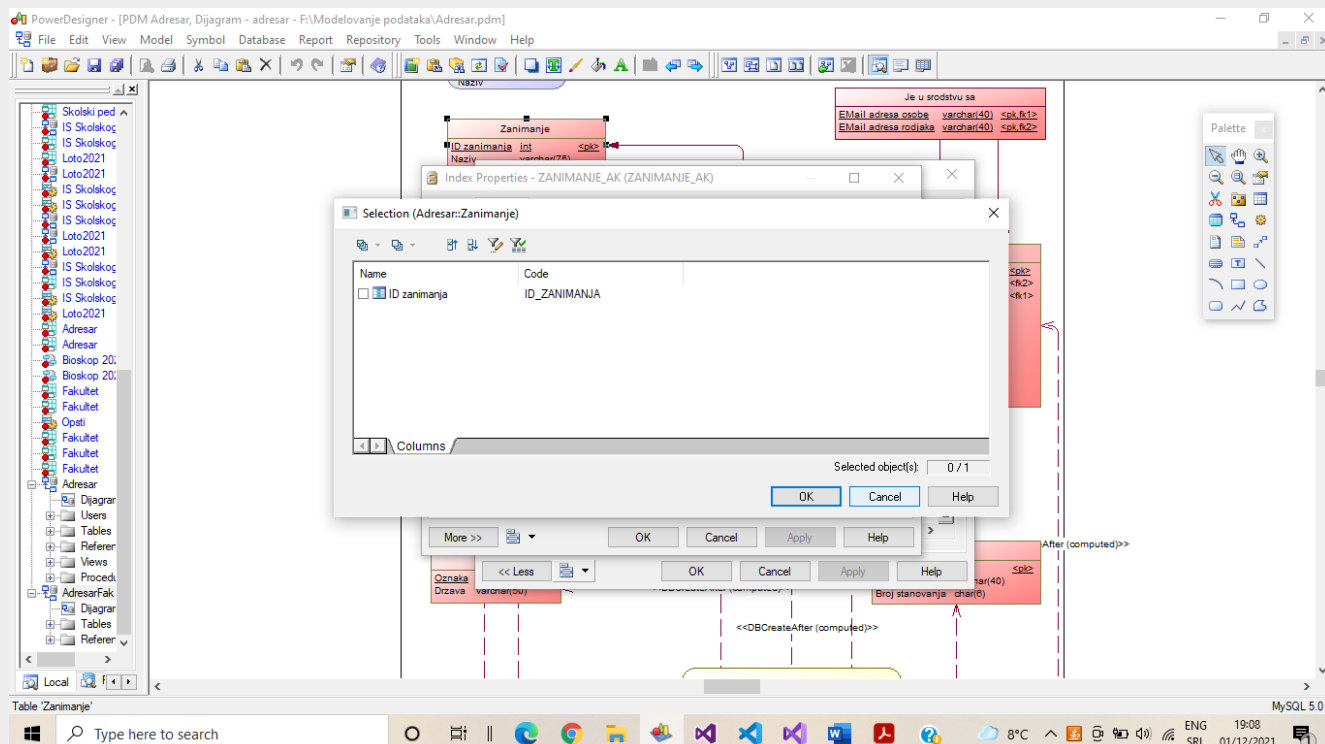
**Definisanje alternativnih ključeva** u PDM modelu, u slučaju da se želi sprečiti redunansa podataka (npr. za tabelu Zanimanje)... otvoriti prozor za definisanje osobina table (**Properties prozor**), upisati naziv alternativnog ključa, izabrati osobinu **Unique=True**, pa se zatim primenjuje Properties alatka sa palete alata (prva sa leve strane).



Otvara se dodatni prozor za dodavanje kolona iz table koje će činiti alternativni ključ, izabrati opciju **Add Columns...**

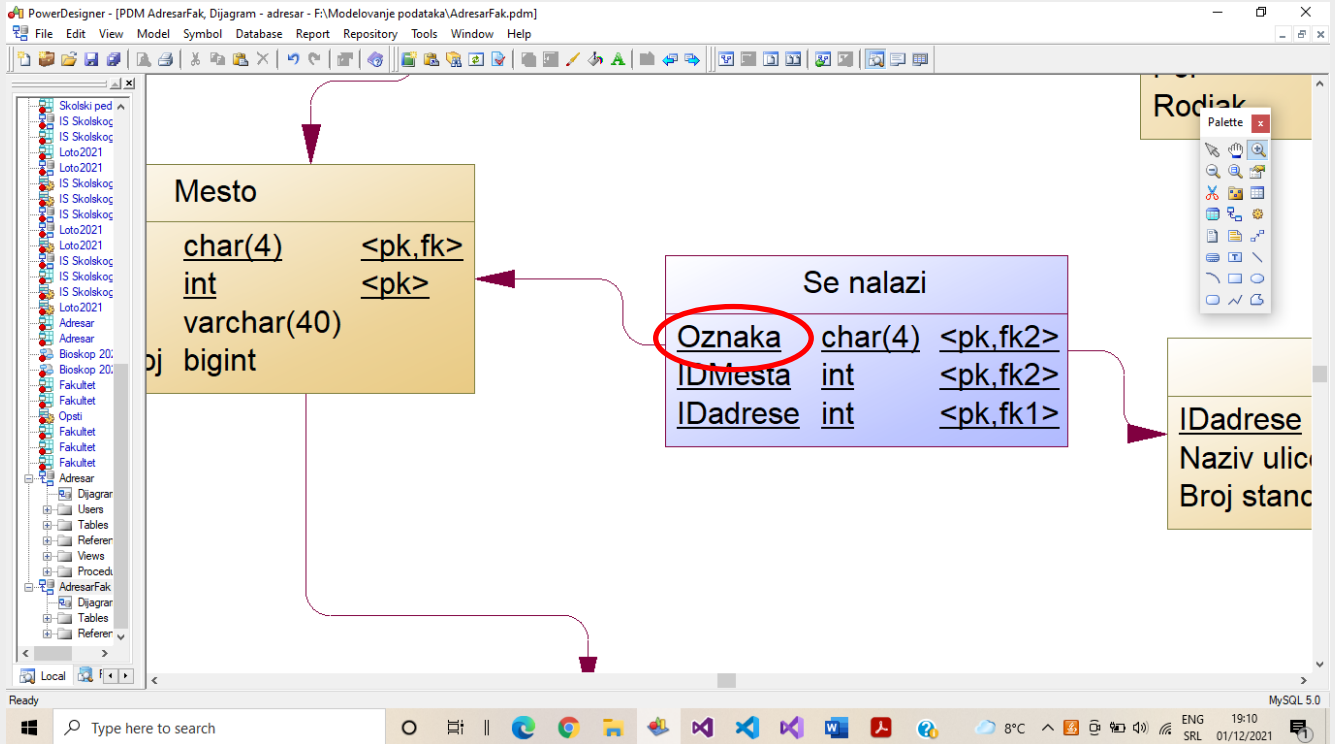


... izabrati naziv jedne ili više kolona iz table koje će činiti alternativni ključ. Kriterijumi za izbor: jedinstvenost i minimalnost. U primeru je izabran naziv zanimanja (prethodna slika).

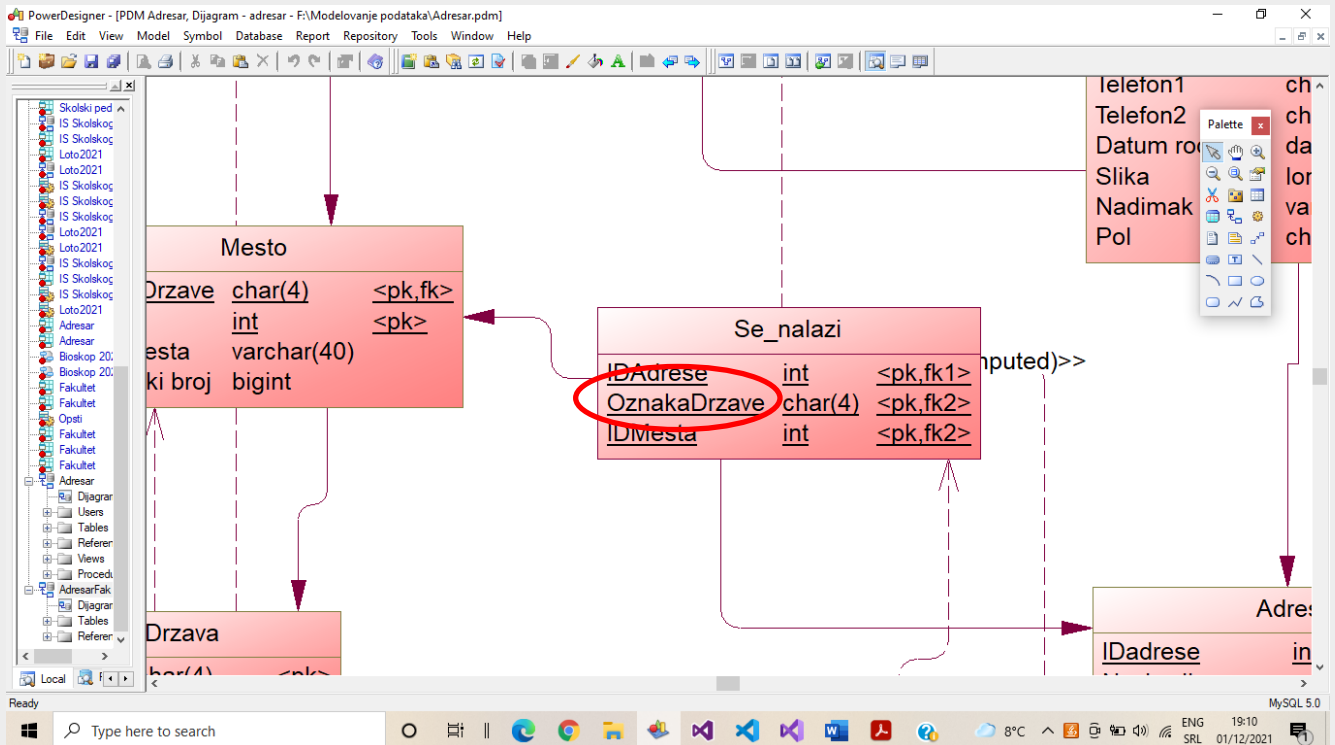


Formiranje ovakvog alternativnog ključa za tabelu zanimanja znači da se neće moći kreirati dva zanimanja sa istim nazivom.

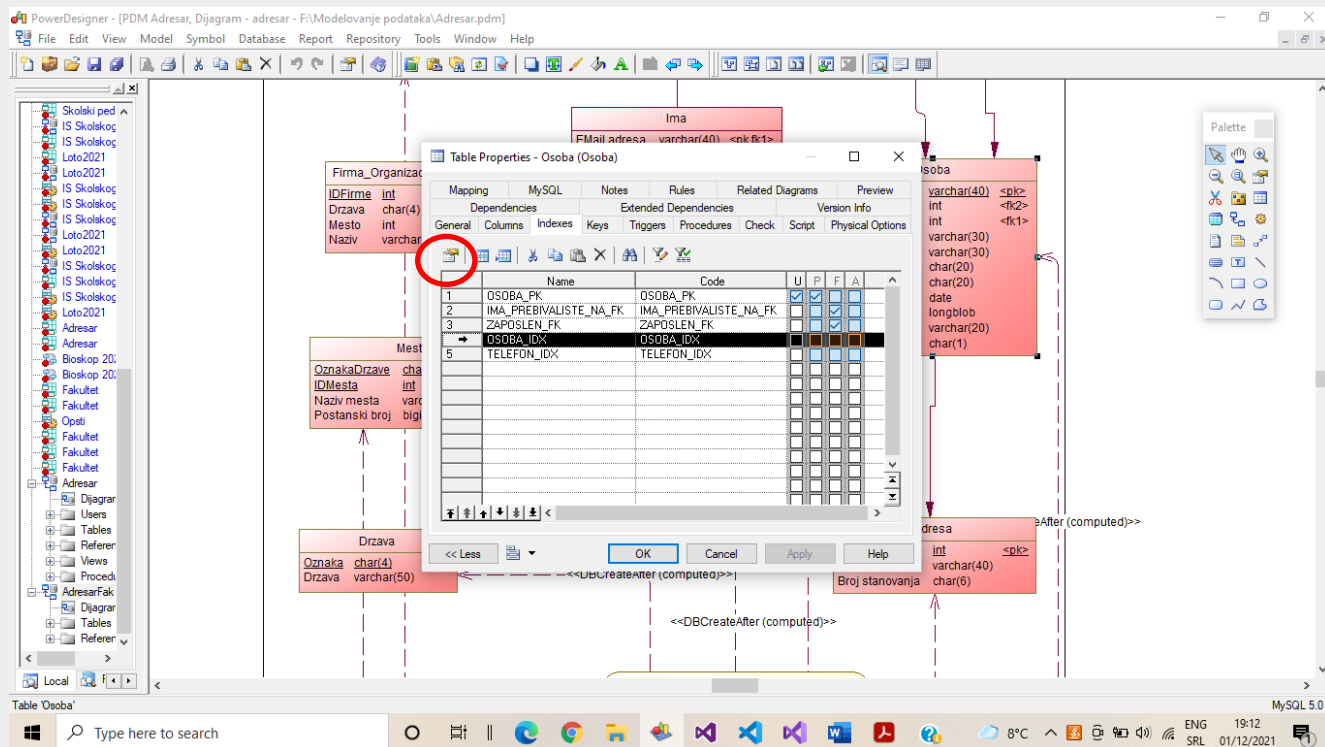
U PDM modelu je moguće, a nekada i potrebno, izvršiti preimenovanje kolona ukoliko naziv nije odgovarajući (npr. kod stranih ključeva ili kod većeg broja veza između tabela). Otvoriti osobine veze između tabela i pogledati prirodu odnosa tabela u naslovnoj liniji...



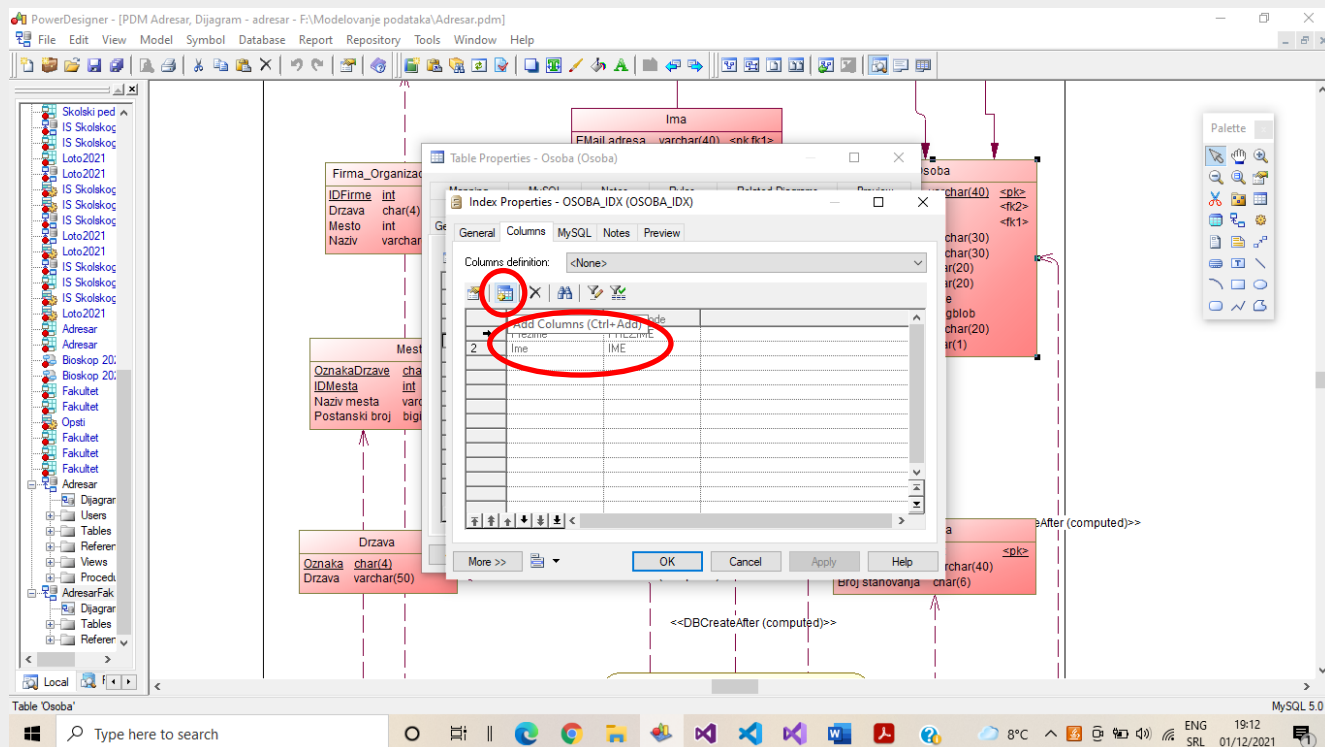
... što može uticati na naziv. Promena naziva kolona na primeru table Se\_nalazi, kod stranog ključa: Oznaka u OznakaDrzave.



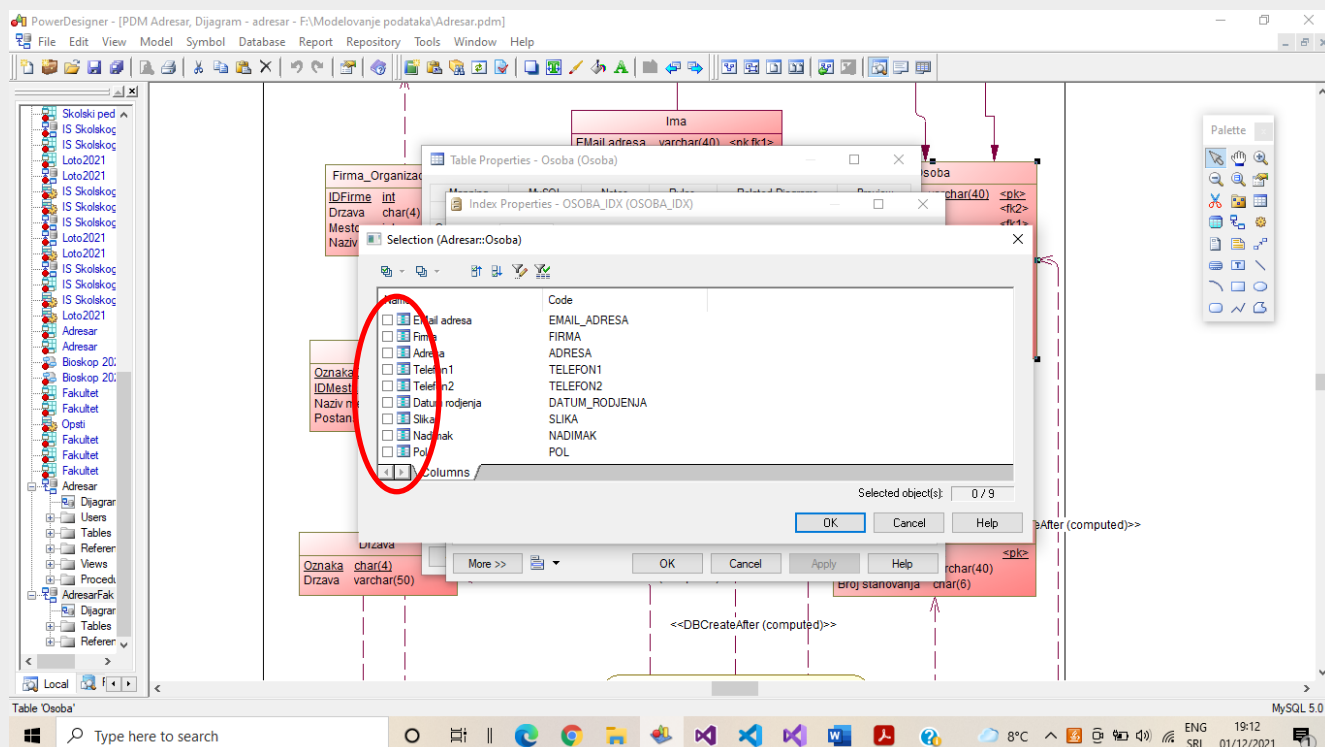
**Dodavanje indeksa** koji utiču na brzinu pretrage podataka, izdvajanja podataka SQL upitima, se vrši u okviru tabela. Otvara se prozor sa osobinama table (**Table Properties**), na kartici za definisanje indeksa (**Indexes**), gde već postoje indeksi za ključna obeležja, tj. kolone table, se dodaje novi indeks. Upisuje se njegov naziv, ne sme se izabrati osobina **Unique** za jedinstvenu vrednost...



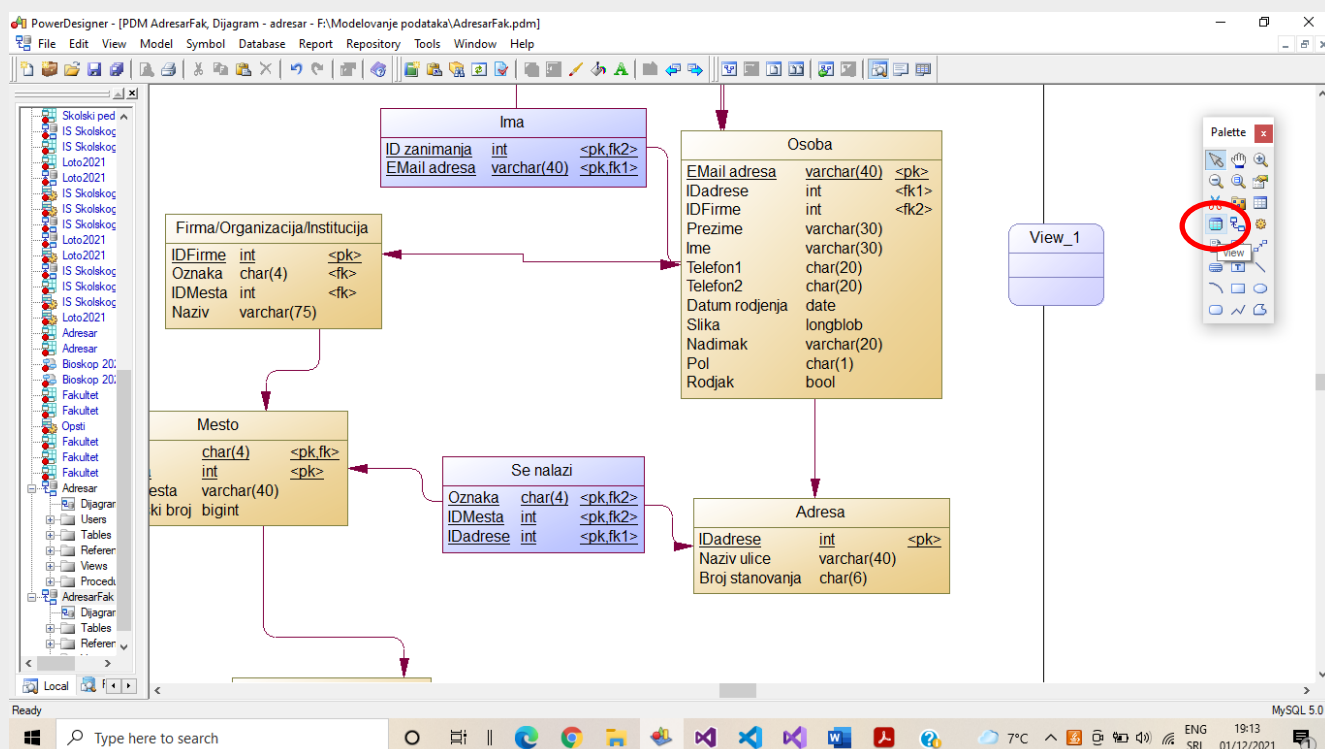
... bira se taster sa palete alata – **Add Columns**, koji inicira izbor kolona table za indeks...



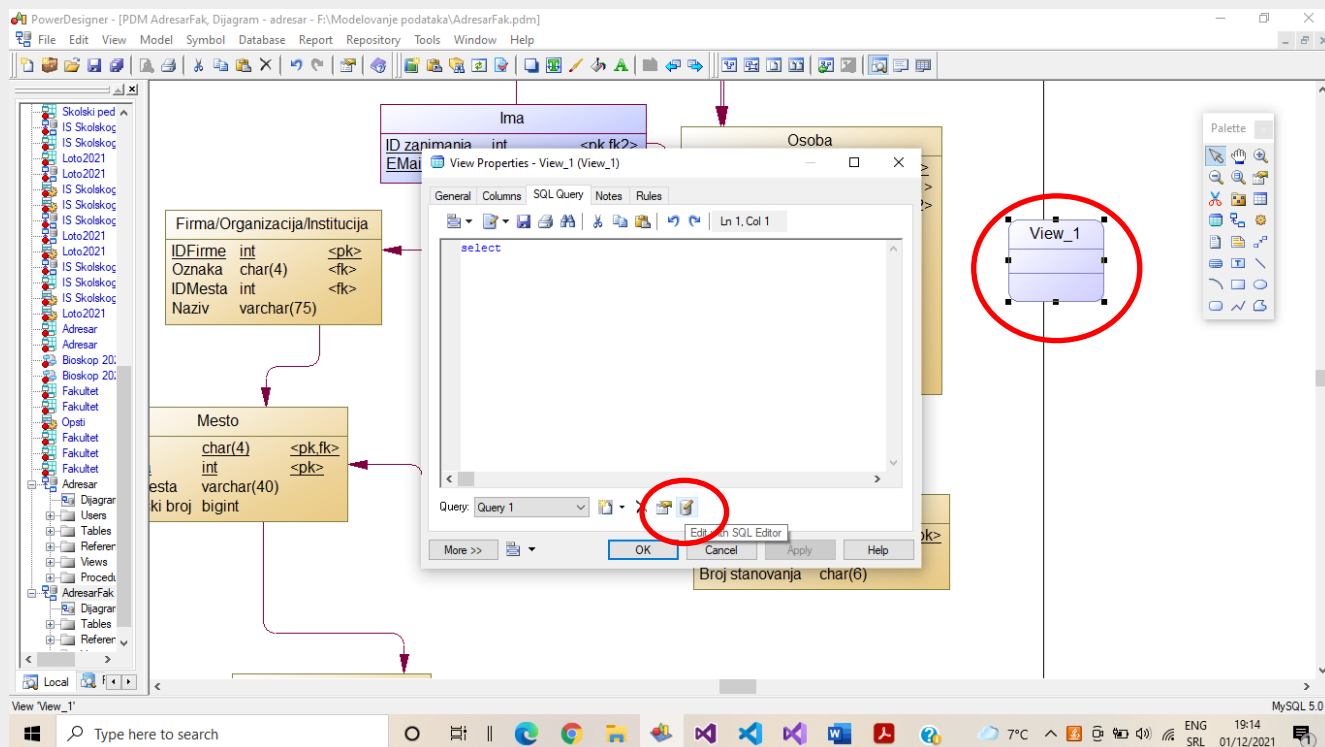
... zatim treba izabrati naziv kolone(a) iz table za koju se kreira indeks (sledeća slika). U primeru je izabrano ime i prezime osobe (prethodna slika).



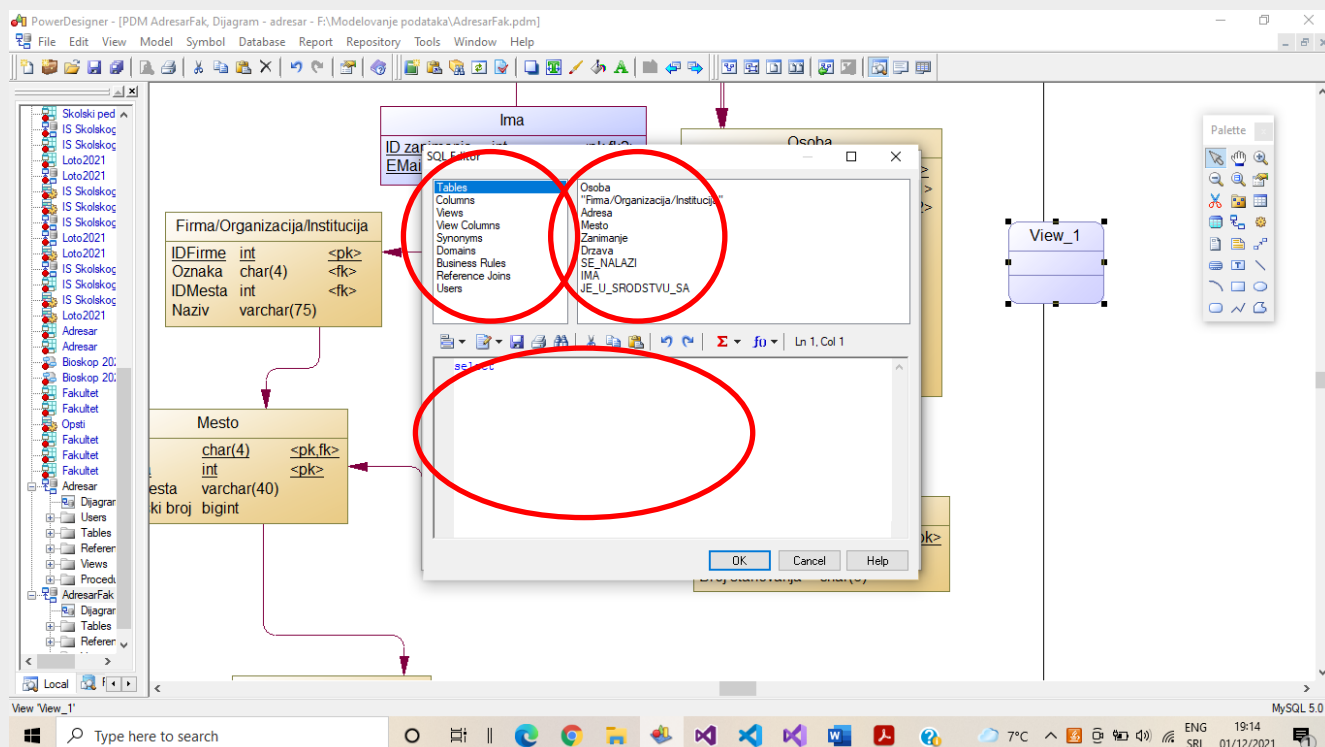
**Dodavanje pogleda** (SQL upita koji će se snimiti u implementiranu bazu podataka) u PDM model se radi tako što se, sa palete alata, izabere alat **View**, pa se u dijagramu kreira prazan pogled (View\_1 na slici).



Potrebno je pristupiti editoru za pisanje SQL naredbi. Pristupa se osobinama pogleda (**View Properties**) na isti način kao i za sve ostale osobine drugih objekata u CASE alatu (npr. 2x levi klik na pogled).



Na prvoj kartici – **General** se unosi ime pogleda (polje **Name**). Na drugoj se mogu izabrati kolone čije će vrednosti pogled prikazivati, a prethodna slika prikazuje karticu **SQL Query** za unos upita. Pisanje teksta upita može da se ubrza uz pomoć SQL editora, koji se pokreće pritiskom na taster **Edit with SQL Editor**, sa male palete alata u donjem delu prozora. Otvara se editor za brže formiranje upita.



U listi levo gore se može izabrati vrsta objekta koja se koristi u upitu, a zatim u listi gore desno se bira naziv tabele, kolone, spoja itd. Upit se formira u donjem prozoru, koji kosirnik može još i ručno menjati, te ukucati i dodati neki deo upita, poput uslova npr.



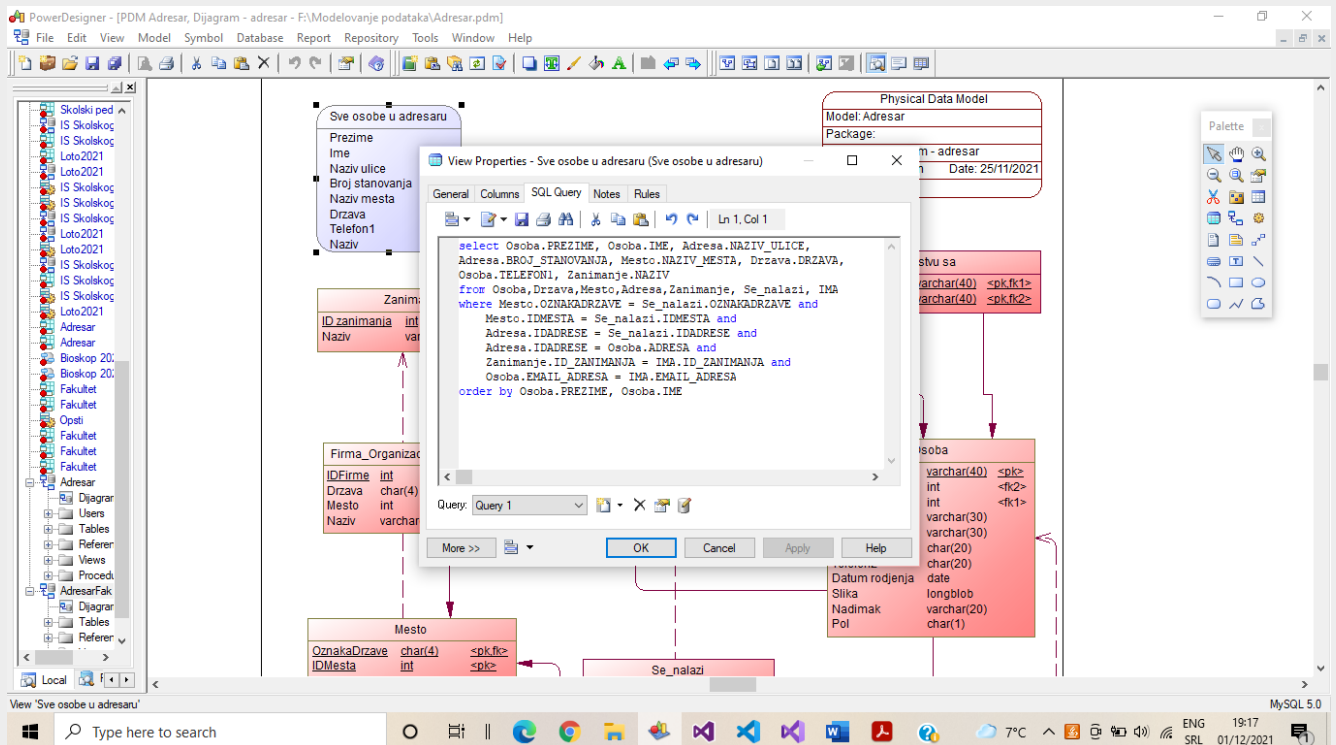
Opšti oblik jednostavnog SQL upita za izdvajanje podataka SELECT naredbom:

SELECT nazivi kolona FROM nazivi tabela

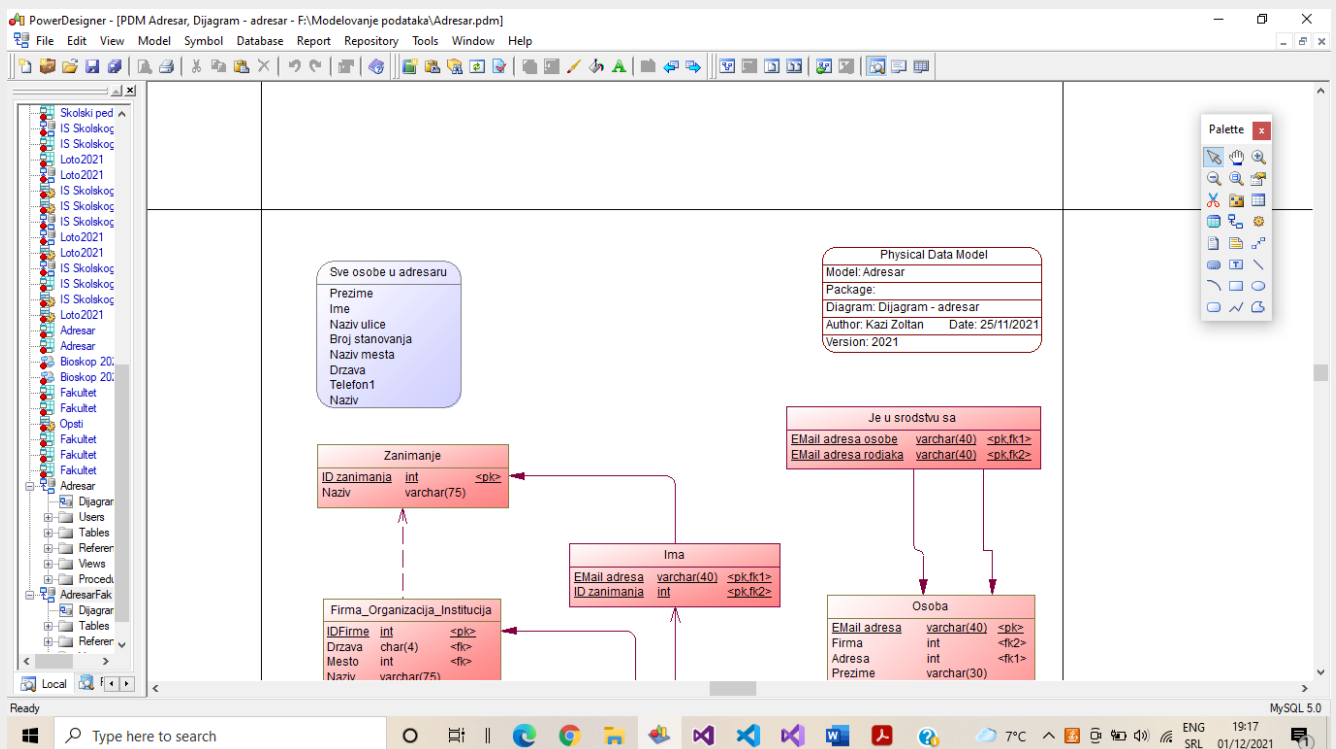
WHERE tabela1.kolonaPK=tabela2.FK AND tabela2.kolonaPK=tabela3.FK ...

ORDER BY nazivi kolona ASC/DESC

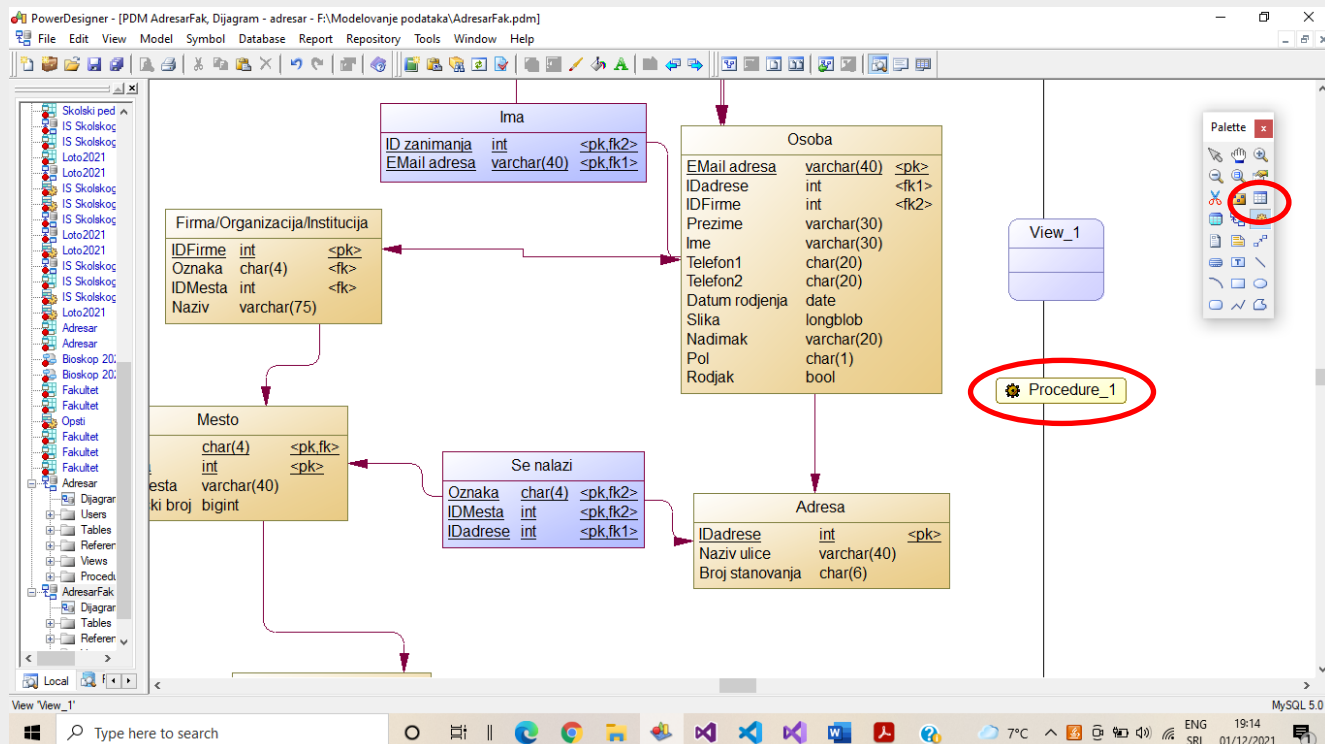
Prikaz gotovog teksta upita za izdvajanje podataka o osobama upisanim u adresar/imenik:



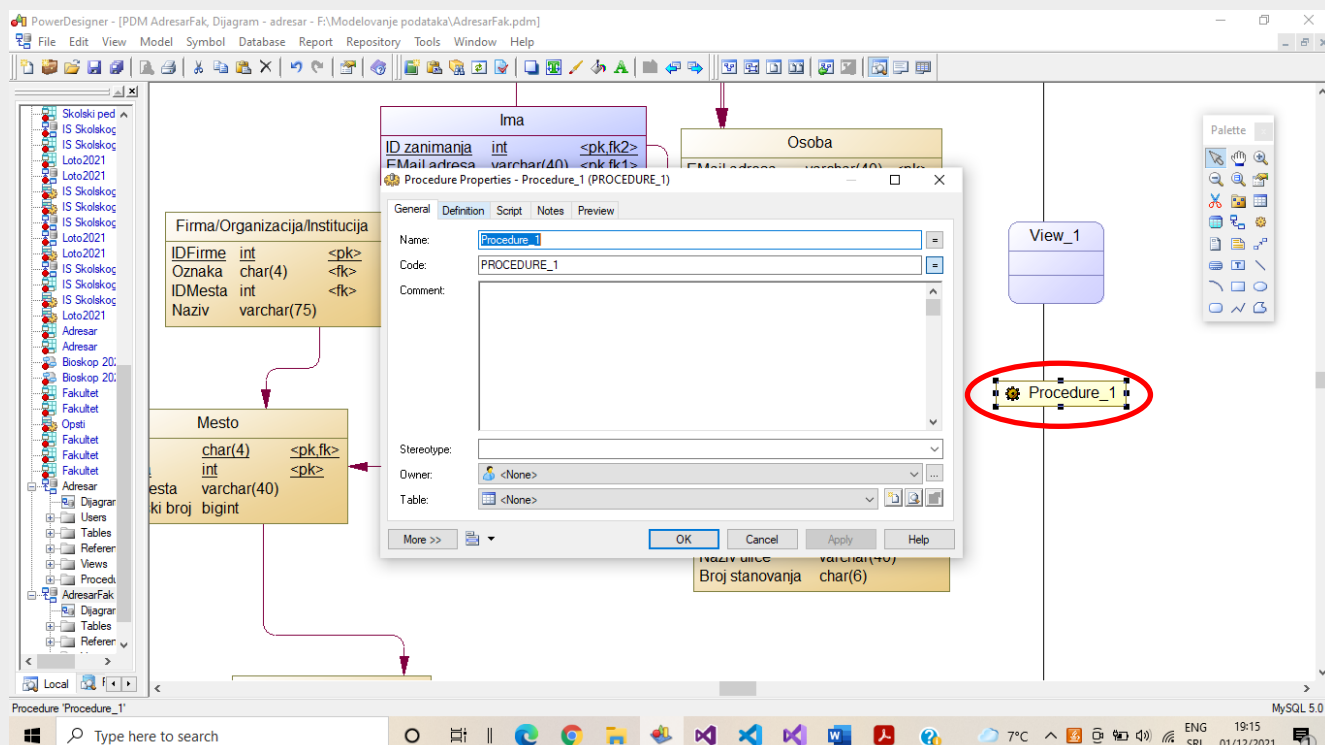
Novoformirani pogled (Sve osobe u adresaru) je prikazan na sledećoj slici:



**Dodavanje uskladištenih procedura** (SQL upita, sa parametrima najčešće, koji će se snimiti u implementiranu bazu podataka poput pogleda) u PDM model se radi tako što se, sa palete alata, izabere alat **Procedure**, pa se u dijagramu kreira prazna procedura (Procedure\_1 na slici).

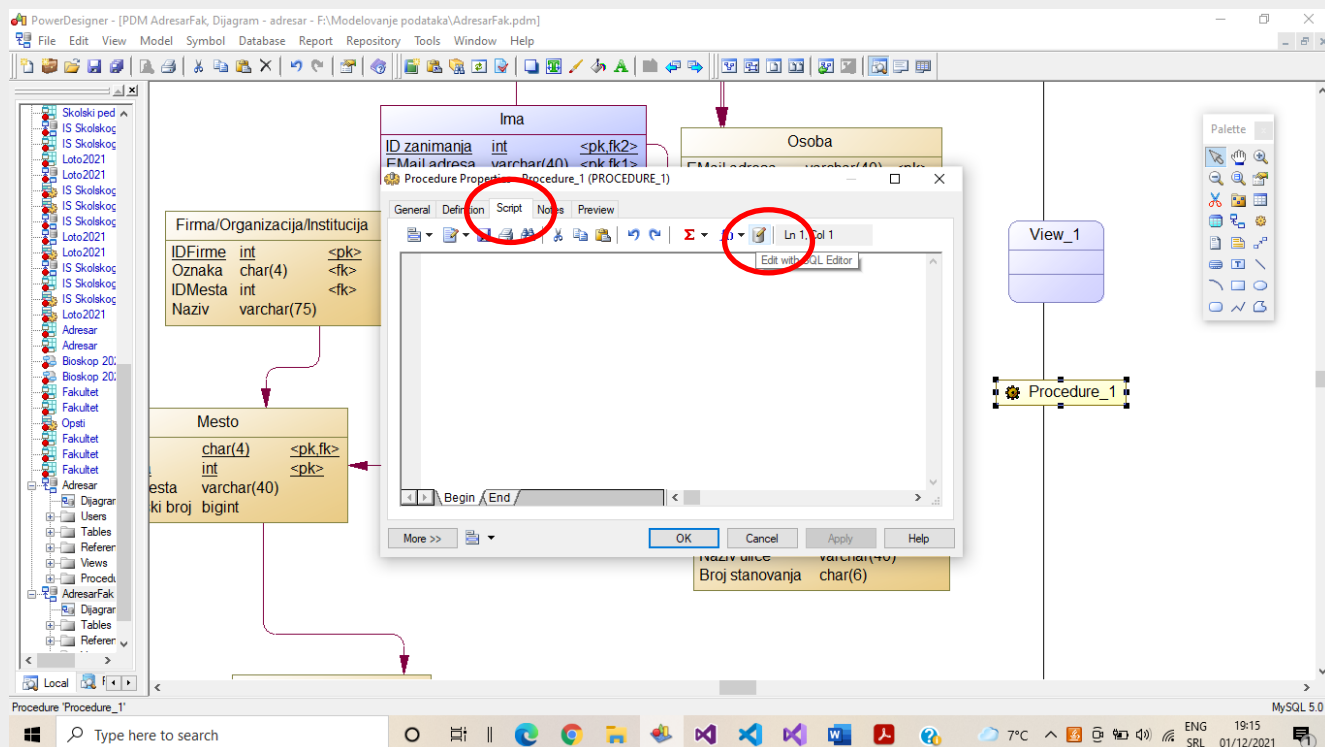


I u ovom slučaju je potrebno je pristupiti editoru za pisanje SQL naredbi. Otvara se prozor sa osobinama pogleda (**Procedure Properties**) na isti način kao što je već opisano.

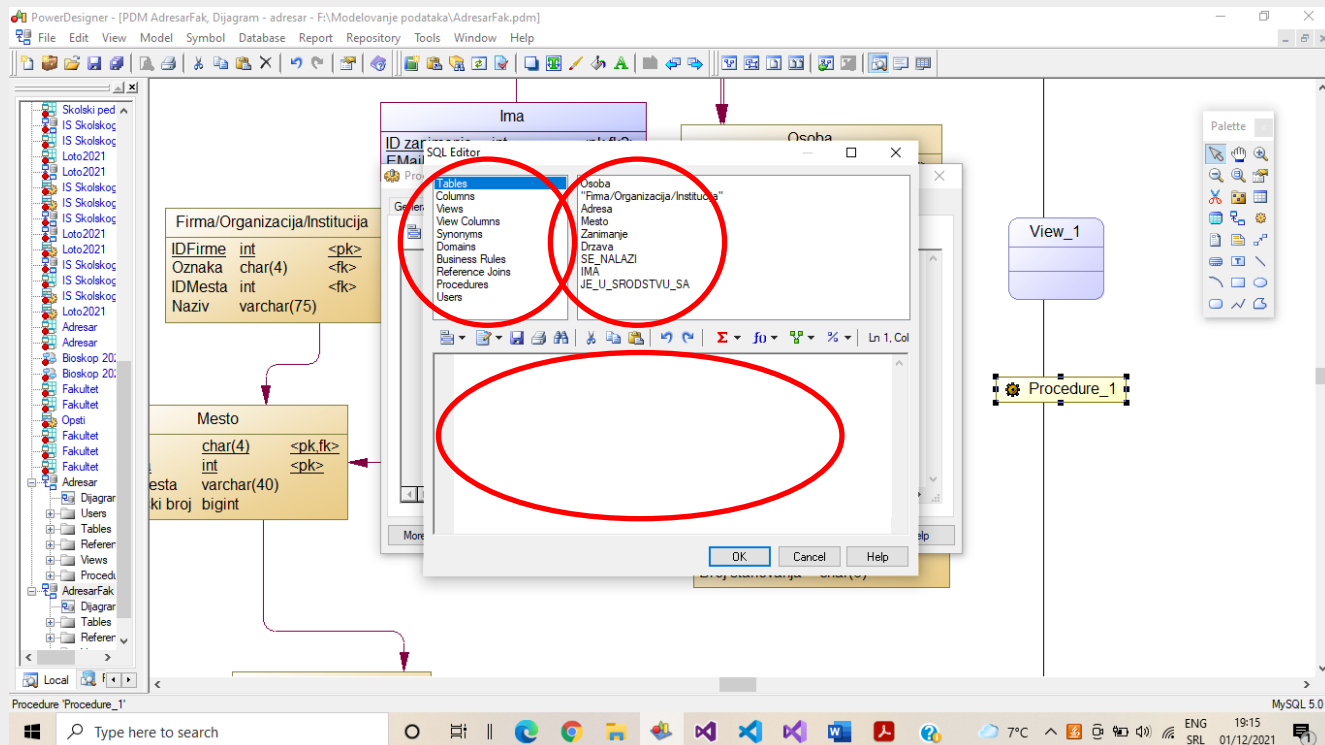


U kartici **General** se upisuje naziv procedure.

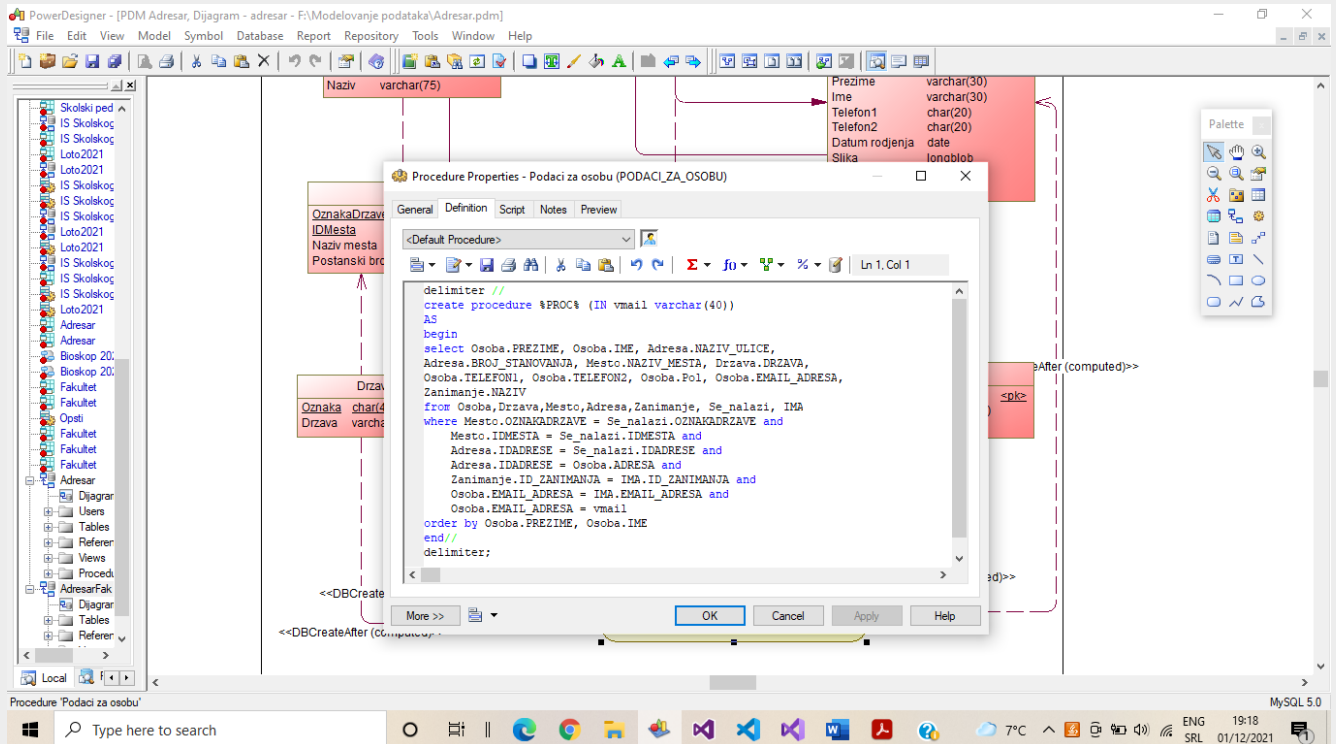
U okviru treće kartice – Script se ili unosi tekst SQL skripta sa jednim ili više upita, ili se pak može otvoriti editor za formiranje SQL upita (poslednji taster na paleti alata).



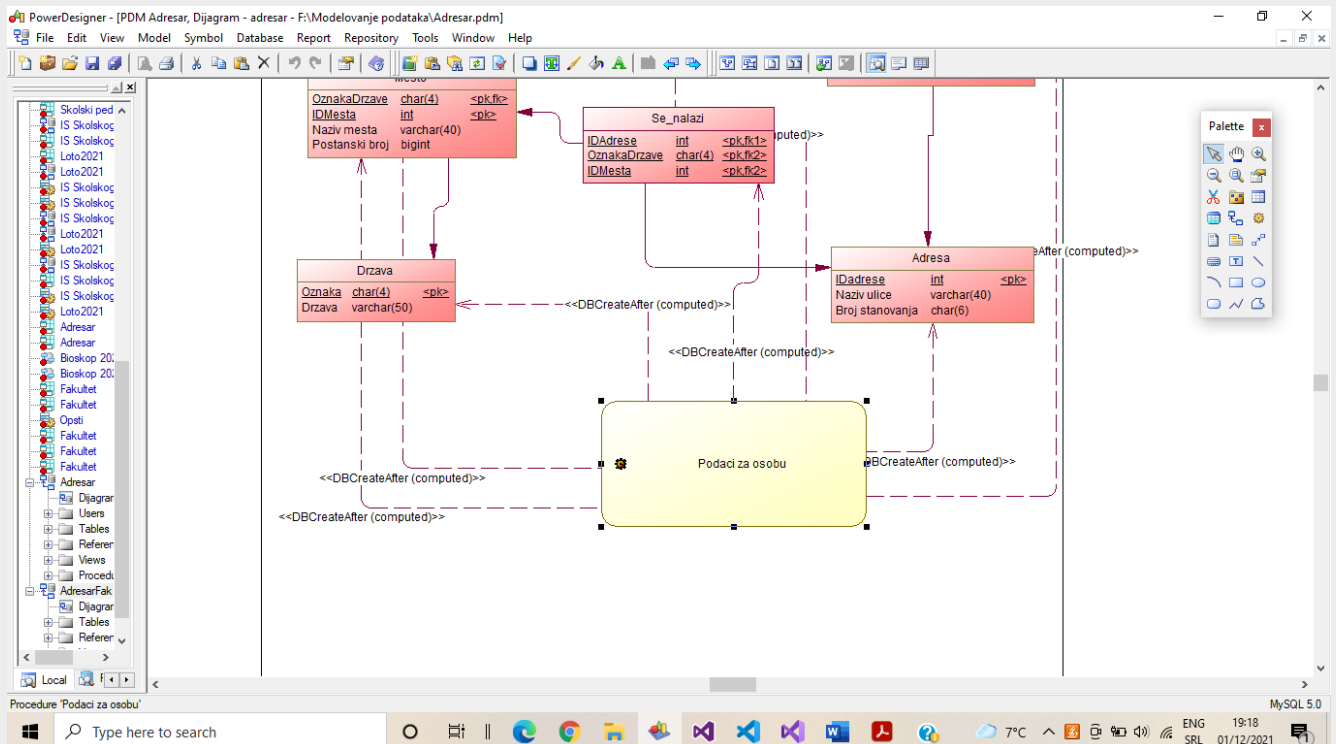
U listi levo gore se može izabrati vrsta objekta koja se koristi u upitu, a zatim u listi gore desno se bira naziv tabele, kolone, spoja itd. Upit se formira u donjem prozoru, koji kosirnik može još i ručno menjati, te ukucati i dodati neki deo upita.



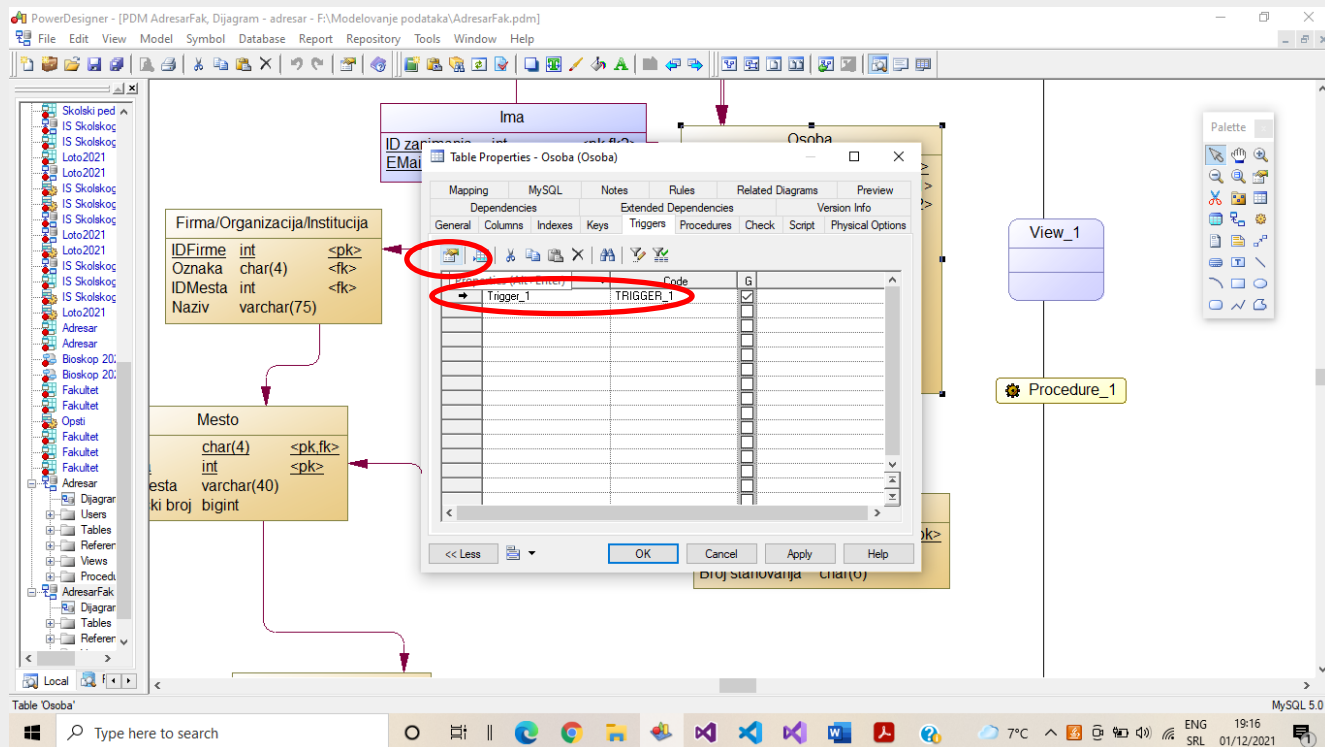
Prikaz gotovog teksta upita u okviru procedure sa ulaznim parametrom za izdvajanje podataka o jednoj osobi iz adresara/imenika (na slici je prikazan MySQL dijalekt SQL skripta):



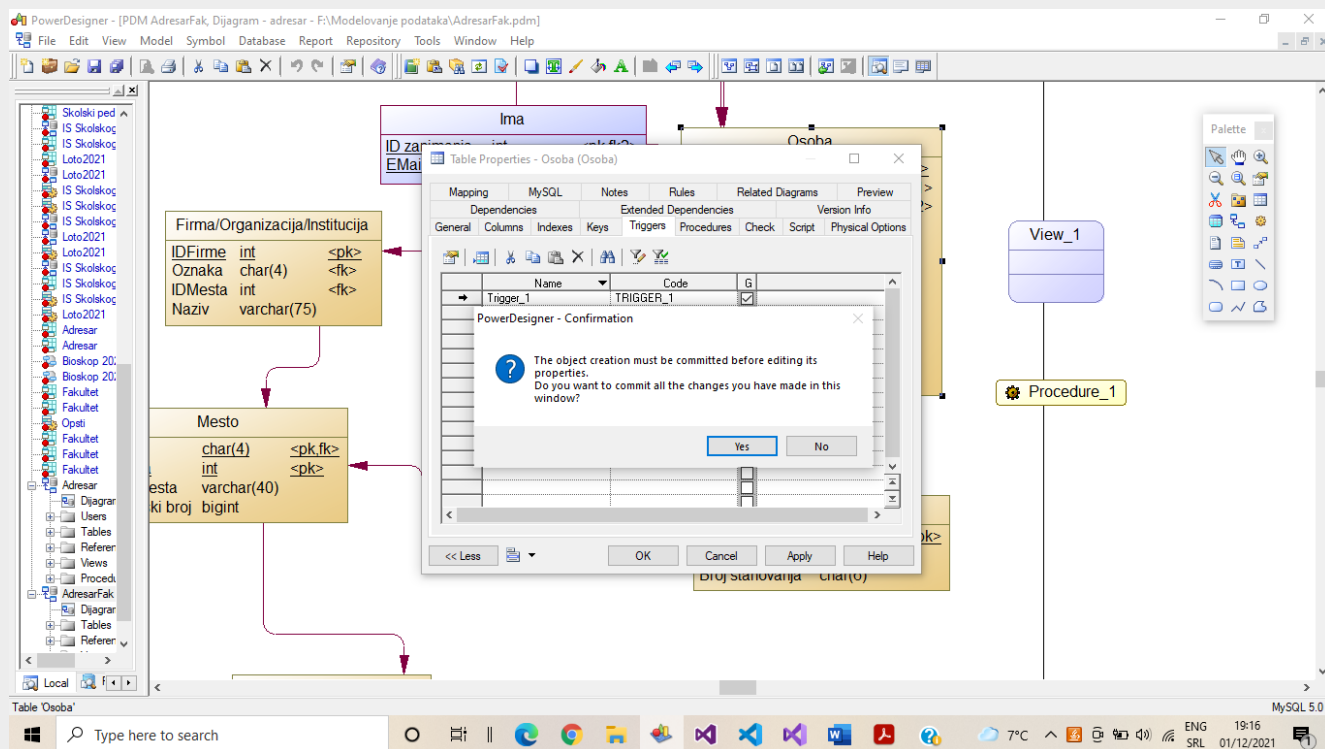
Uskladištena procedura (Podaci za osobu) je prikazana na sledećoj slici:



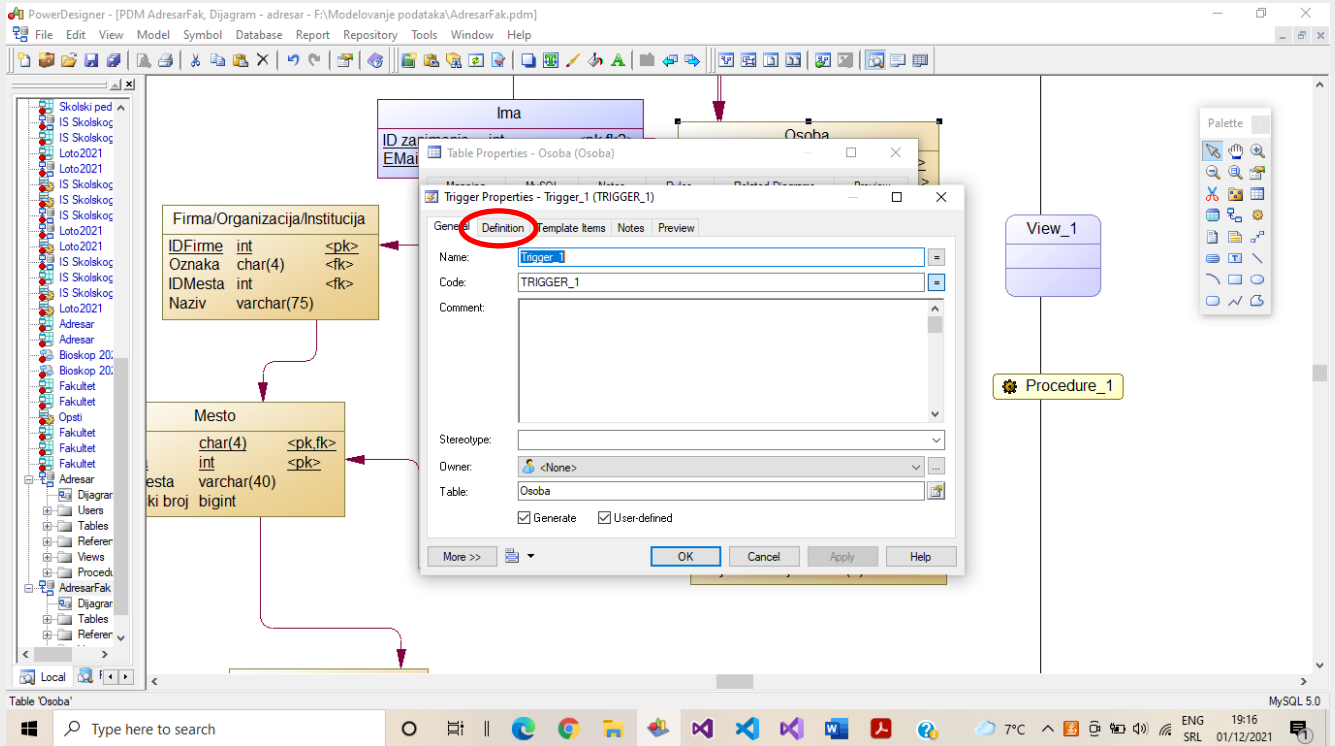
**Kreiranje okidača** se malo razlikuje od toga kako se kreiraju pogledi i procedure. Okidač je, takođe, SQL skript koji je snimljen i sačuvan u bazi podataka, ali se vezuje i sastavni je deo tabele. Znači da ne postoji posebna alatka za kreiranje okidača, već se pristupa osobinama tabele (**Table Properties**), pa se aktivira kartica **Triggers** (napomena: samo u onim SUBP-ima koji podržavaju rad sa okidačima). Upisuje se naziv okidača u tabelu (**Name**, Code se automatski generiše).



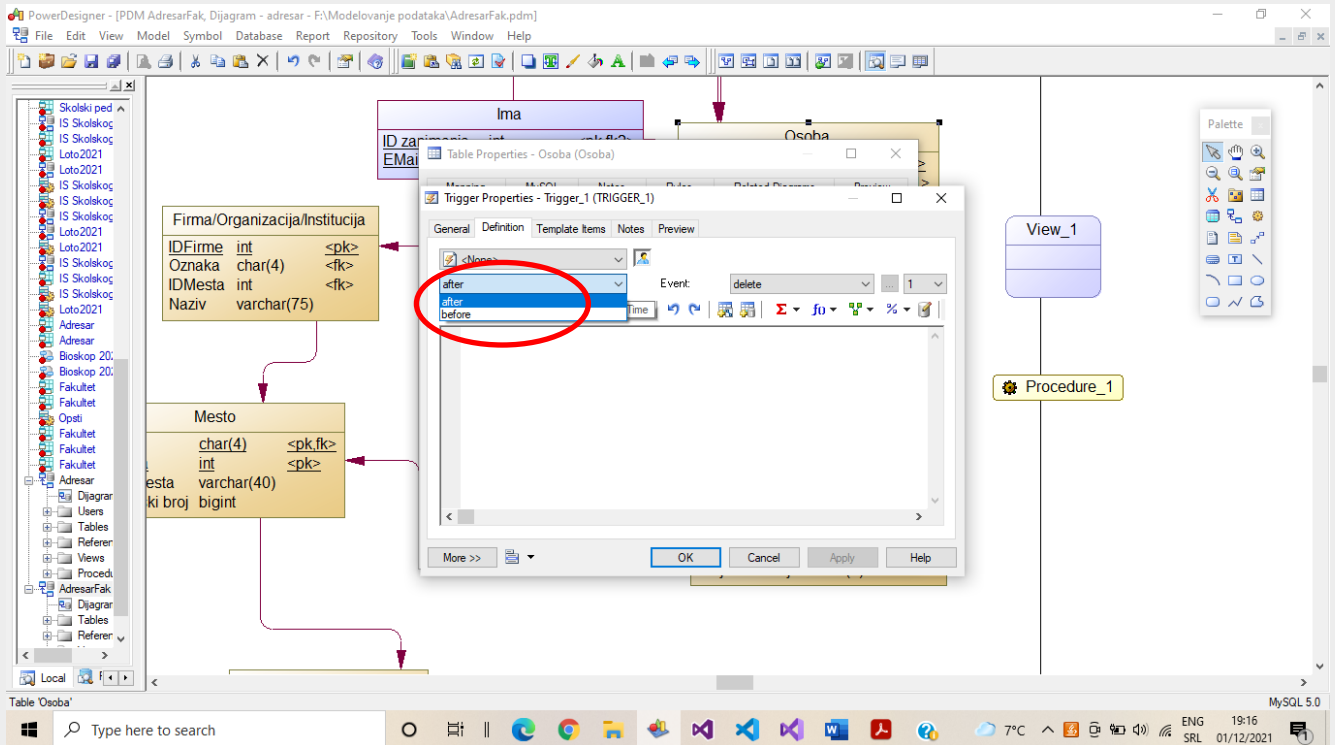
Zatim se otvara prozor za definisanje ostalih osobina okidača (**Properties**). Sledi dijalog za prozor za pitanjem za potvrdu izmena u tabeli (**Yes**)...



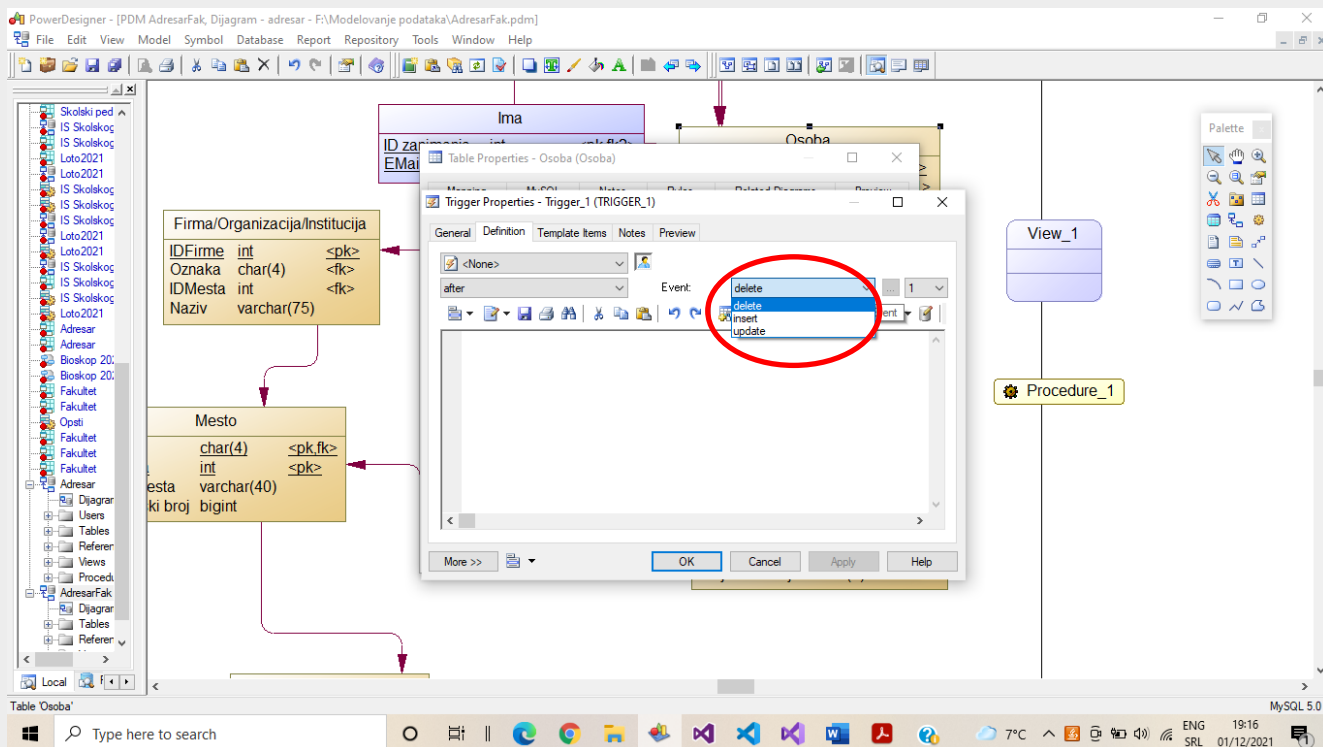
... u polje **Name** se upisuje ime okidača, a na kartici **Definition** se vrši formiranje tela okidača.



Sledi izbor operacije i događaja za koji će se okidač automatski aktivirati i izvršiti. U listi levo bira se **After** (posle) ili **Before** (pre)...

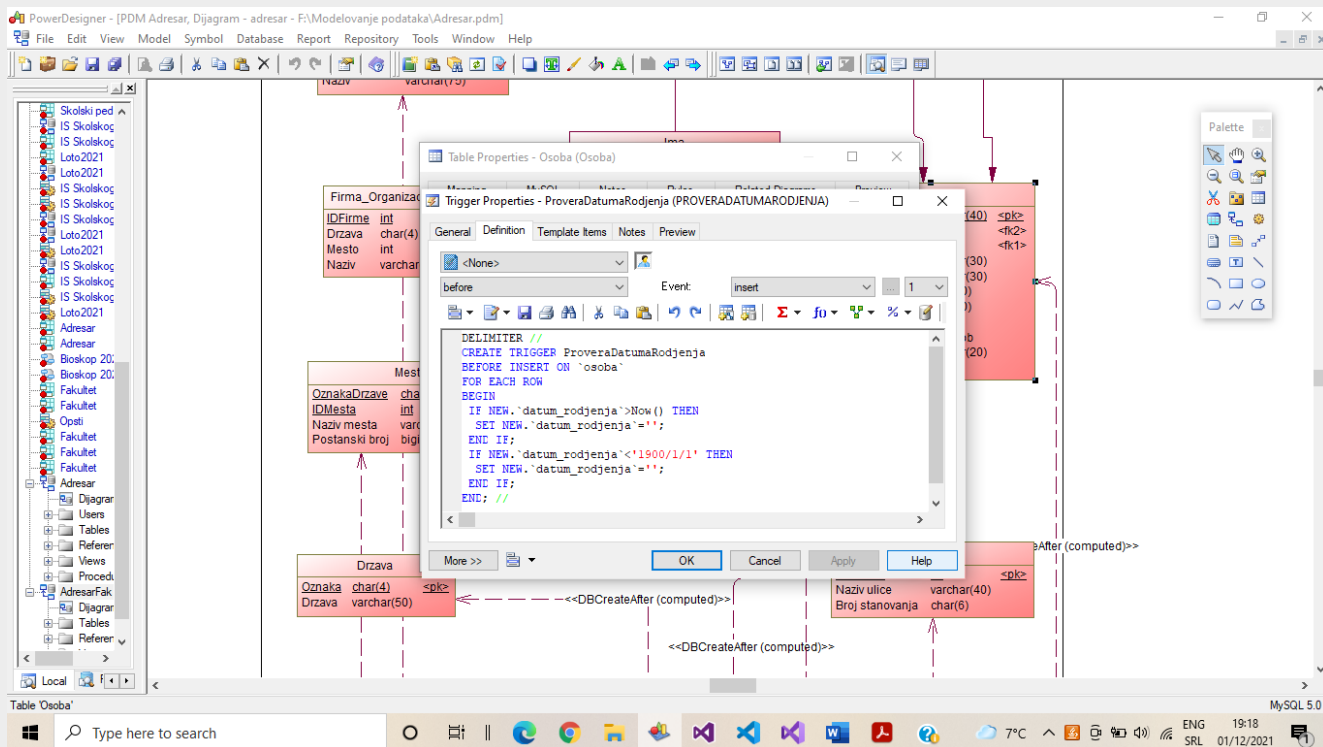


... a u listi desno (Event) operacija: unos (Insert), izmena (Update) ili brisanje (Delete) podataka u tabeli.



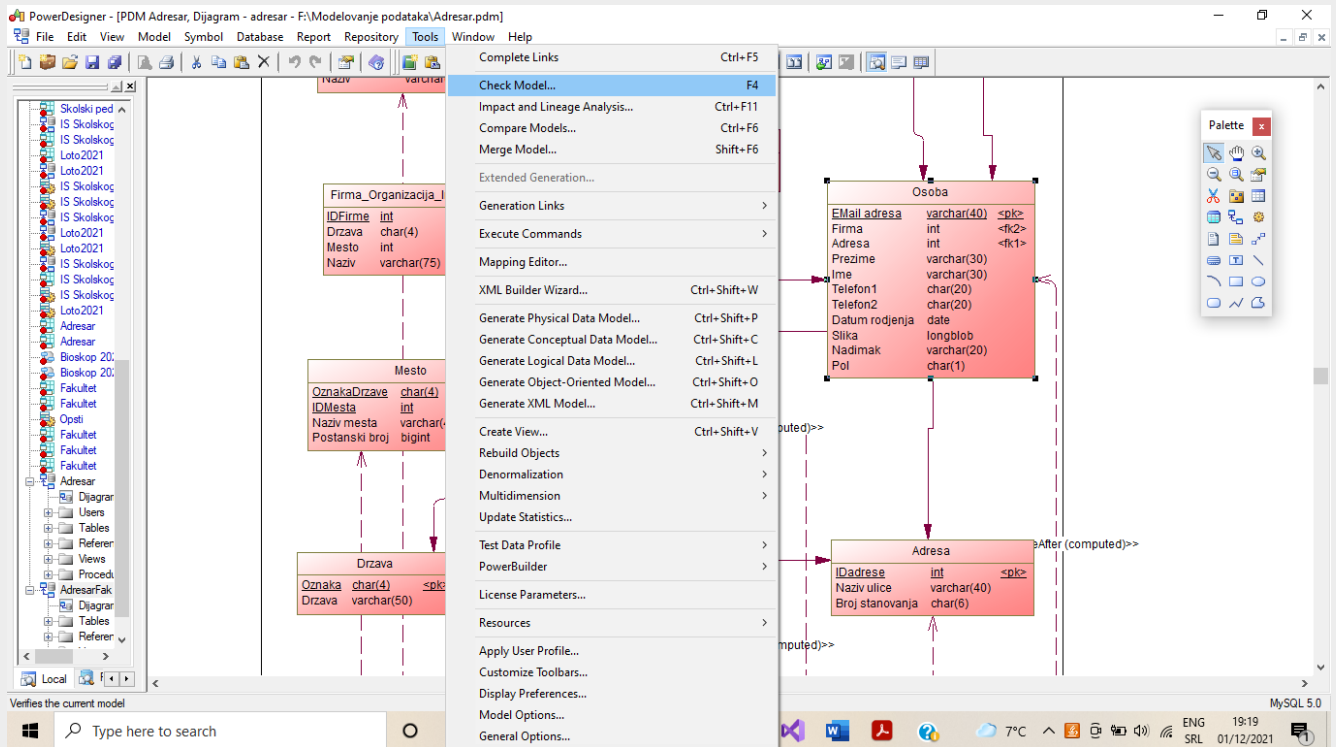
**Napomena:** sadržaj ovih lista se razlikuje u zavisnosti od toga koji SUBP je izabran na početku kreiranja modela. Ne podržavaju svi softveri za rad sa bazama iste dgadžaje.

Primer gotovog okidača za proveru unosa datuma rođenja osobe koji ne može biti veći od tekućeg datuma ni manji od npr. 1.1.1900. godine. U tim slučajevima, okidač postavlja NULL (ništa) kao vrednost kolone.

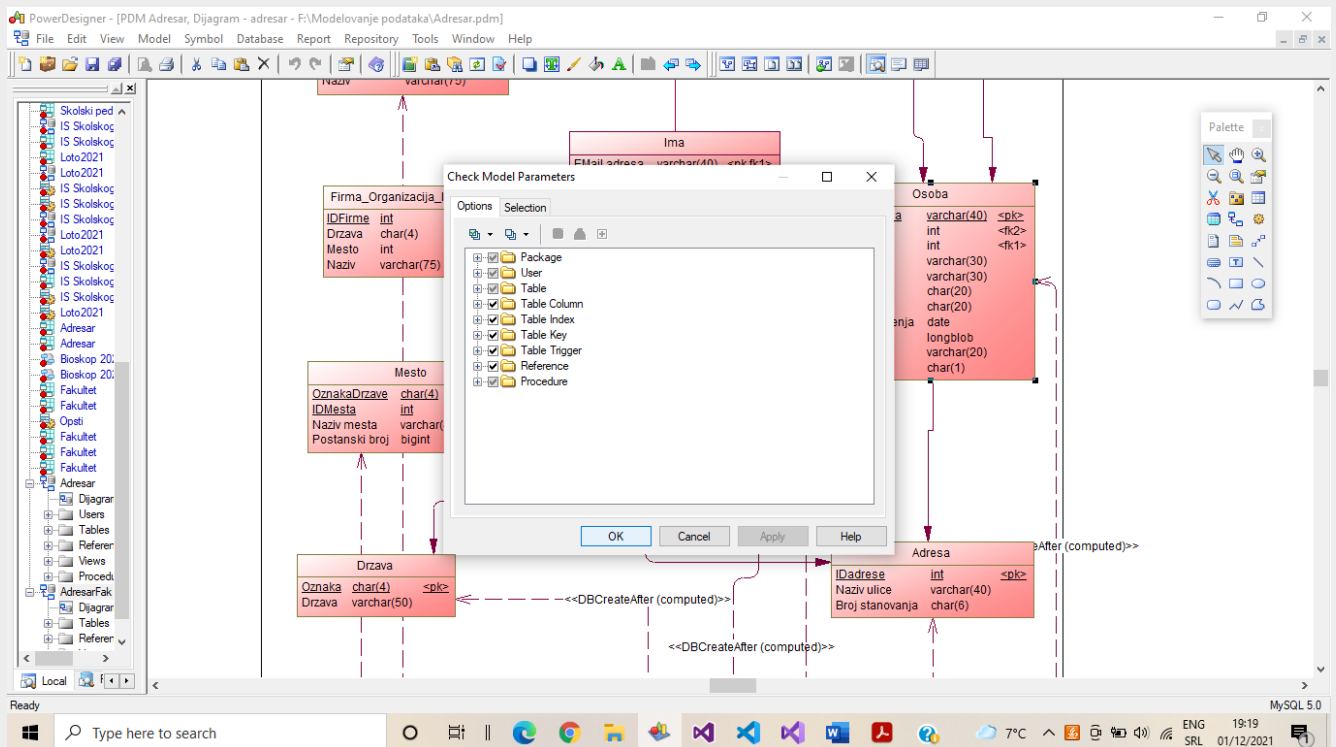


Ekstenzija datoteke za PDM model jeste .pdm.

Provera ispravnosti modela sa aspekta korektnog crtanja dijagram i formiranja osobina, bez mogućnosti provere kvaliteta modela i njegove semantike, se vrši preko opcije: **Tools – Check Model**.



Izbor parametara provere ispravnosti modela:





## Prikaz rezultata provere ispravnosti modela (u primeru model nema grešaka):

The screenshot shows the PowerDesigner interface with a data model diagram. The diagram includes tables: **Mesto** (OznakaDrzave, IDMesta, Naziv mesta, Postanski broj), **Se\_nalazi** (IDAdrese, OznakaDrzave, IDMesta), **Osoba** (EMail adresa, Firma, Adresa, Prezime, Ime, Telefon1, Telefon2, Datum rođenja, Slika, Nadimak, Pol), and **Adresa** (EMail adresa, Firma, Adresa, Prezime, Ime, Telefon1, Telefon2, Datum rođenja, Slika, Nadimak, Pol). Relationships are shown with lines connecting attributes to other tables. A 'Check Model' dialog box is open, displaying a list of checks and their results. The status at the bottom indicates '0 error(s), 4 warning(s). The Physical Data Model is correct, no errors were found.'

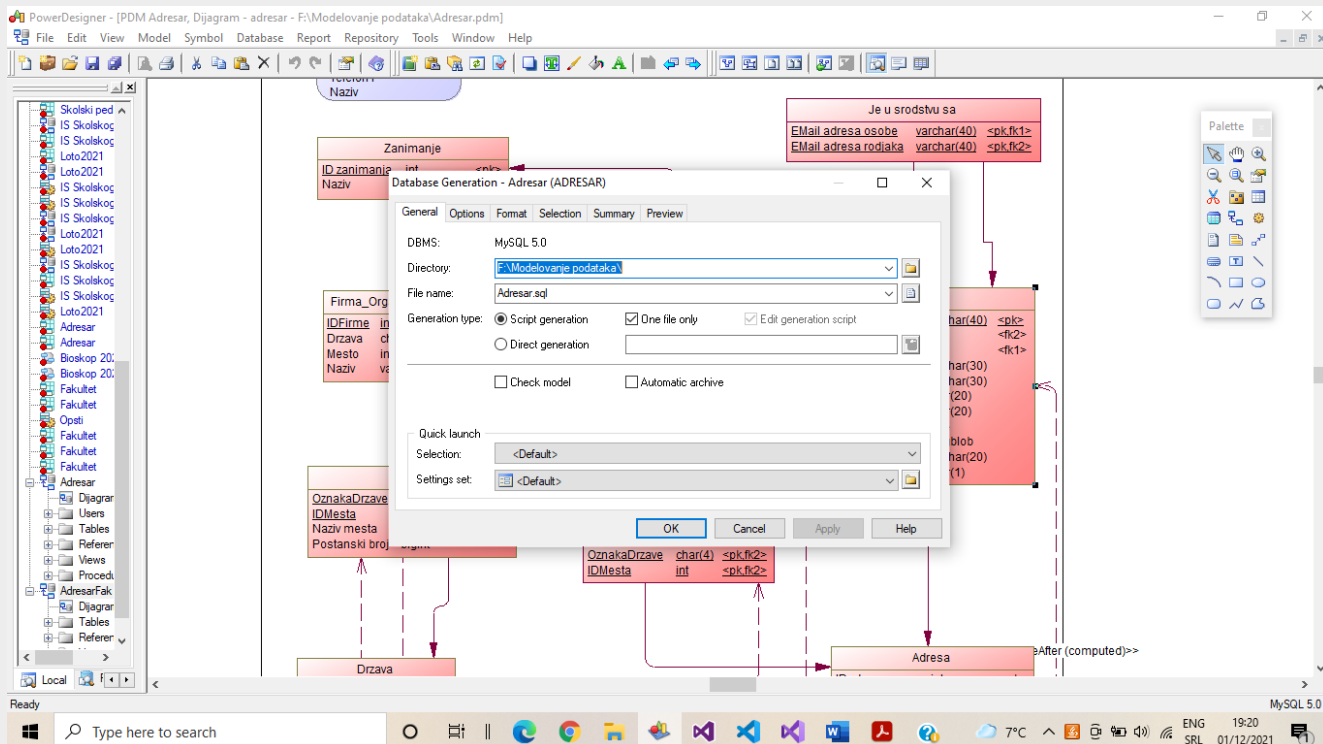
Category	Check	Object	Location
Table Index	Index inclusion	Index 'Ima.IMA_...	<Model>::Ima...
Table Index	Index inclusion	Index 'Je u srods...	<Model>::Je u sr...
Table Index	Index inclusion	Index 'Mesto.ME...	<Model>::Mesto...
Table Index	Index inclusion	Index 'Se_nalazi...	<Model>::Se_na...

Implementacija PDM modela u konkretnom SUBP-u se može raditi na dva načina: 1. direktno preko ODBC mehanizma (povezivanjem sa praznom bazom kreiranom u SUBP-u) ili 2. kreiranjem SQL skripta sa DDL naredbama koje se moraju izvršiti u SQL editoru u softveru za administraciju relacione baze podataka.

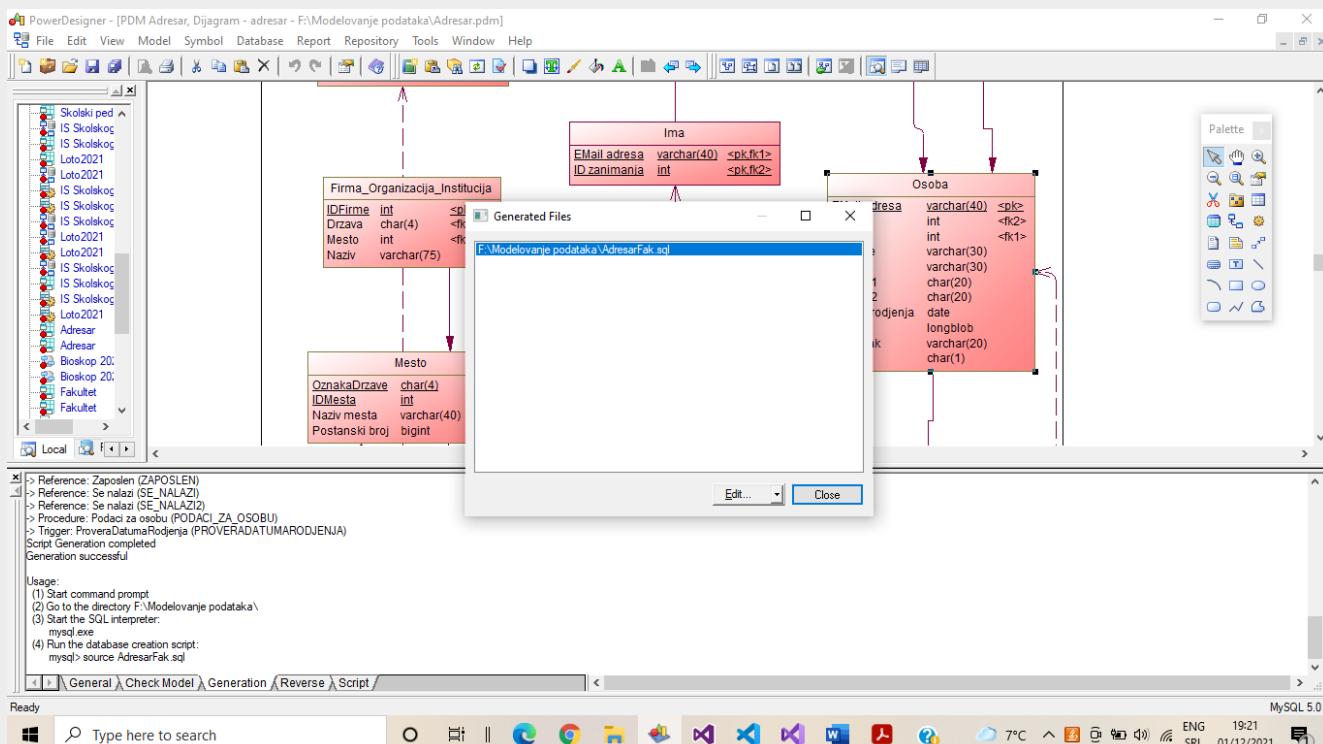
The screenshot shows the PowerDesigner interface with the 'Database' menu open. The 'Generate Database...' option is selected. The data model diagram is visible in the background, showing the same tables and relationships as in the previous screenshot. The status bar at the bottom indicates 'Generates a database'.

Sa glavnog menija CASE alata bira se meni **Database**, pa stavka **Generate Database...**

... bira se putanja lokacije gde će se kreirati datoteka sa SQL DDL skriptom (**Directory**), ime datoteke (**File name**), način generisanja (**Script generation**), pa pritiska taster OK.



Generiše se tekstualna datoteka sa ekstenzijom .sql, koja se može zatvoriti ili odmah otvoriti u tekst procesoru (**Edit**).



## Prikaz generisanog SQL DDL skripta za MySQL SUBP:

```
AdresarFak - Notepad
File Edit Format View Help

drop index TELEFON_IDX on Osoba;

drop index OSOBA_IDX on Osoba;

drop table if exists Osoba;

drop table if exists Se_nalazi;

drop index ZANIMANJE_AK on Zanimanje;

drop table if exists Zanimanje;

/*=====*/
/* User: VLASNIK_ADRESARA */
/*=====*/
create user VLASNIK_ADRESARA identified by 'adresar';

/*=====*/
/* Table: Adresa */
/*=====*/
create table Adresa
(
  IDADRESE          int not null auto_increment,
  NAZIV_ULICE       varchar(40) not null,
  BROJ_STANOVANJA   char(6) not null,
  primary key (IDADRESE)
);

/*=====*/
/* Index: ADRESA_AK */
/*=====*/
create unique index ADRESA_AK on Adresa
(
  NAZIV_ULICE,
  BROJ_STANOVANJA
);
```

NAPOMENA: Naredbe koje se nalaze u ovoj datoteci zavise od toga šta je sve projektant osmislio u modelu, ali tu svakako moraju biti komande: **Create Table, Create Index, Create Constraint, Ater Table** i sl. **Drop** naredbe za brisanje objekata koji još ne postoje u bazi podataka, u prvom izvršavanju skripta se ne moraju izvršiti (ali samo prilikom prvobitnog kreiranja elemenata i objekata baze).