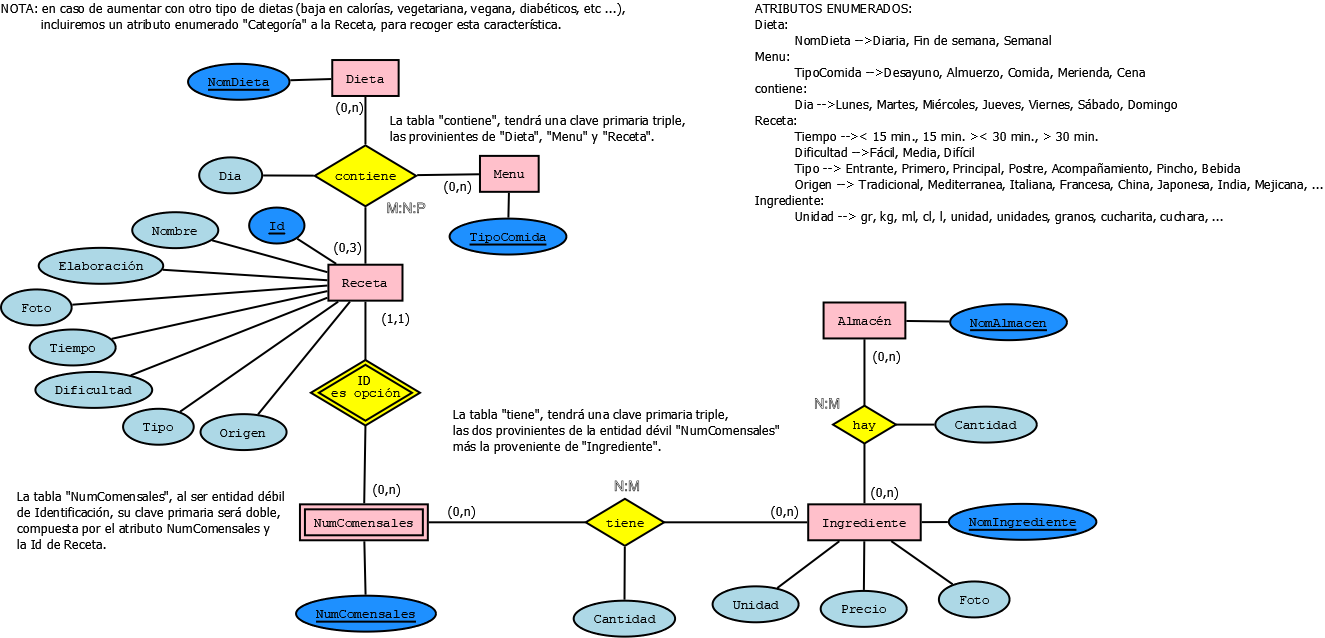
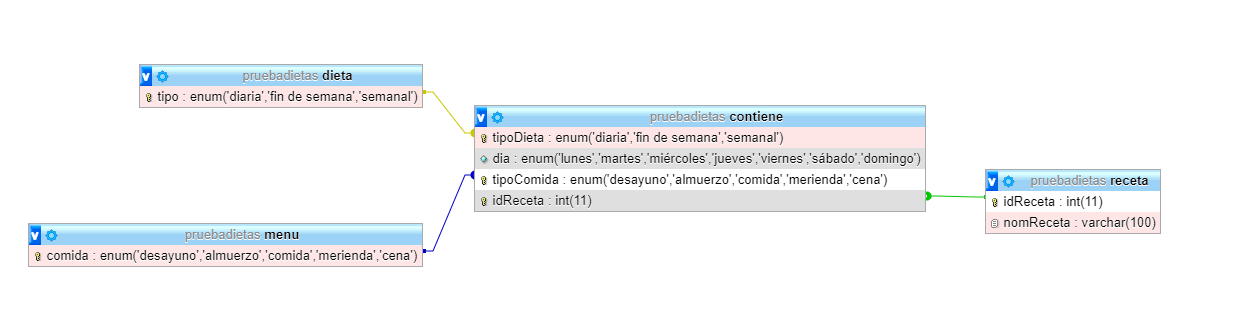
AMPLIACIÓN DE LA BBDD PARA GUARDAR DISTINTAS DIETAS



Realizo prueba en MySQL para comprobar funcionamiento (en Receta sólo uso Id y Nombre):

El diseño



El Script

create database pruebaDietas collate utf8mb4\_spanish\_ci;

use pruebaDietas;

create table RECETA

(idReceta int AUTO\_INCREMENT primary key,

nomReceta varchar(100) not null);

create table MENU

(comida enum('desayuno','almuerzo','comida','merienda','cena') primary key);

create table DIETA

(tipo enum('diaria','fin de semana','semanal') primary key);

create table CONTIENE

(tipoDieta enum('diaria','fin de semana','semanal'),

dia enum('lunes','martes','miércoles','jueves','viernes','sábado','domingo'),

tipoComida enum('desayuno','almuerzo','comida','merienda','cena'),

idReceta int,

constraint fk\_idReceta foreign key(idReceta) references RECETA(idReceta) on delete cascade on update cascade,

constraint fk\_tipoComida foreign key(tipoComida) references MENU(comida) on delete cascade on update cascade,

constraint fk\_tipoDieta foreign key(tipoDieta) references DIETA(tipo) on delete cascade on update cascade,

primary key(idReceta,tipoComida,tipoDieta));

insert into RECETA values (1,'Café con leche');

insert into RECETA values (2,'Pincho de tortilla');

insert into RECETA values (3,'Arroz a la cubana');

insert into RECETA values (4,'Filete de ternera con patatas');

insert into RECETA values (5,'Arroz con leche');

insert into RECETA values (6,'Té con pastas');

insert into RECETA values (7,'Tortilla francesa');

insert into RECETA values (8,'Yogur natural');

insert into MENU values ('desayuno');

insert into MENU values ('almuerzo');

insert into MENU values ('comida');

insert into MENU values ('merienda');

insert into MENU values ('cena');

insert into DIETA values ('diaria');

insert into DIETA values ('fin de semana');

insert into DIETA values ('semanal');

insert into contiene values ('diaria','lunes','desayuno',1);

insert into contiene values ('diaria','lunes','almuerzo',2);

insert into contiene values ('diaria','lunes','comida',3);

insert into contiene values ('diaria','lunes','comida',4);

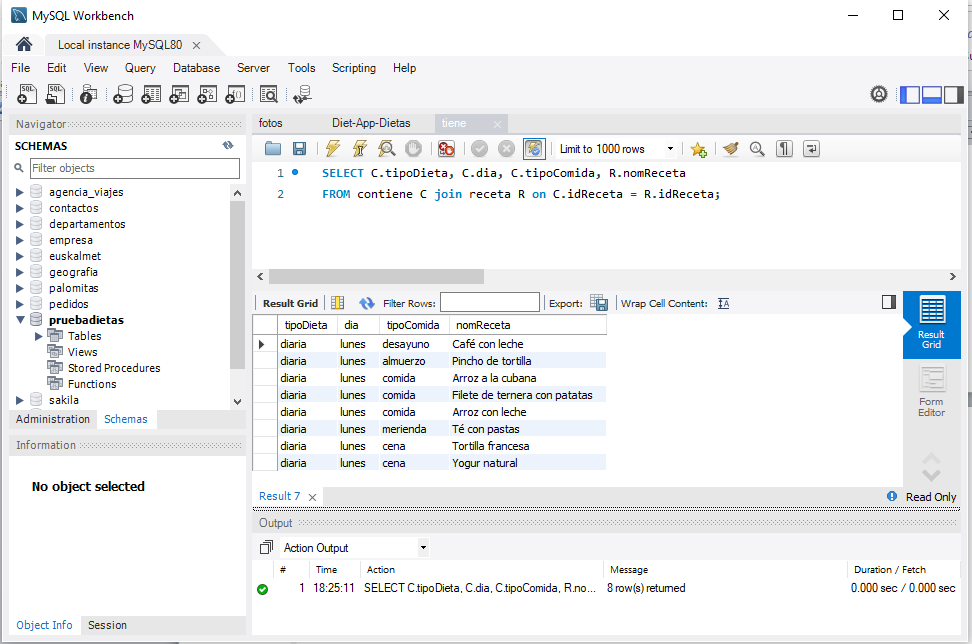
insert into contiene values ('diaria','lunes','comida',5);

insert into contiene values ('diaria','lunes','merienda',6);

insert into contiene values ('diaria','lunes','cena',7);

insert into contiene values ('diaria','lunes','cena',8);

Ejecución de consulta



**NOTA:** Los tipos de datos ENUM no existen en SQLite, pero se soluciona utilizando la restricción CHECK, como se explica en el siguiente enlace:

[How to create ENUM type in SQLite? - Stack Overflow](https://stackoverflow.com/questions/5299267/how-to-create-enum-type-in-sqlite)

SQLite way is to use a [CHECK constraint](https://www.sqlite.org/lang_createtable.html#ckconst).

Some examples:

CREATE TABLE prices (

id INTEGER PRIMARY KEY,

pName TEXT CHECK( LENGTH(pName) <= 100 ) NOT NULL DEFAULT '',

pType TEXT CHECK( pType IN ('M','R','H') ) NOT NULL DEFAULT 'M',

pField TEXT CHECK( LENGTH(pField) <= 50 ) NULL DEFAULT NULL,

pFieldExt TEXT CHECK( LENGTH(pFieldExt) <= 50 ) NULL DEFAULT NULL,

cmp\_id INTEGER NOT NULL DEFAULT '0'

)

This will limit the pType column to just the values M, R, and H, just like enum("M", "R", "H") would do in some other SQL engines.

[edited Mar 14 '19 at 22:38](https://stackoverflow.com/posts/17203007/revisions) [[https://www.gravatar.com/avatar/3d53595cb2a89131fb3be7b55b3a9ef0?s=32&d=identicon&r=PG](https://stackoverflow.com/users/660921/martin-tournoij)](https://stackoverflow.com/users/660921/martin-tournoij)

[Martin Tournoij](https://stackoverflow.com/users/660921/martin-tournoij)

**23.3k**2323 gold badges8989 silver badges122122 bronze badges