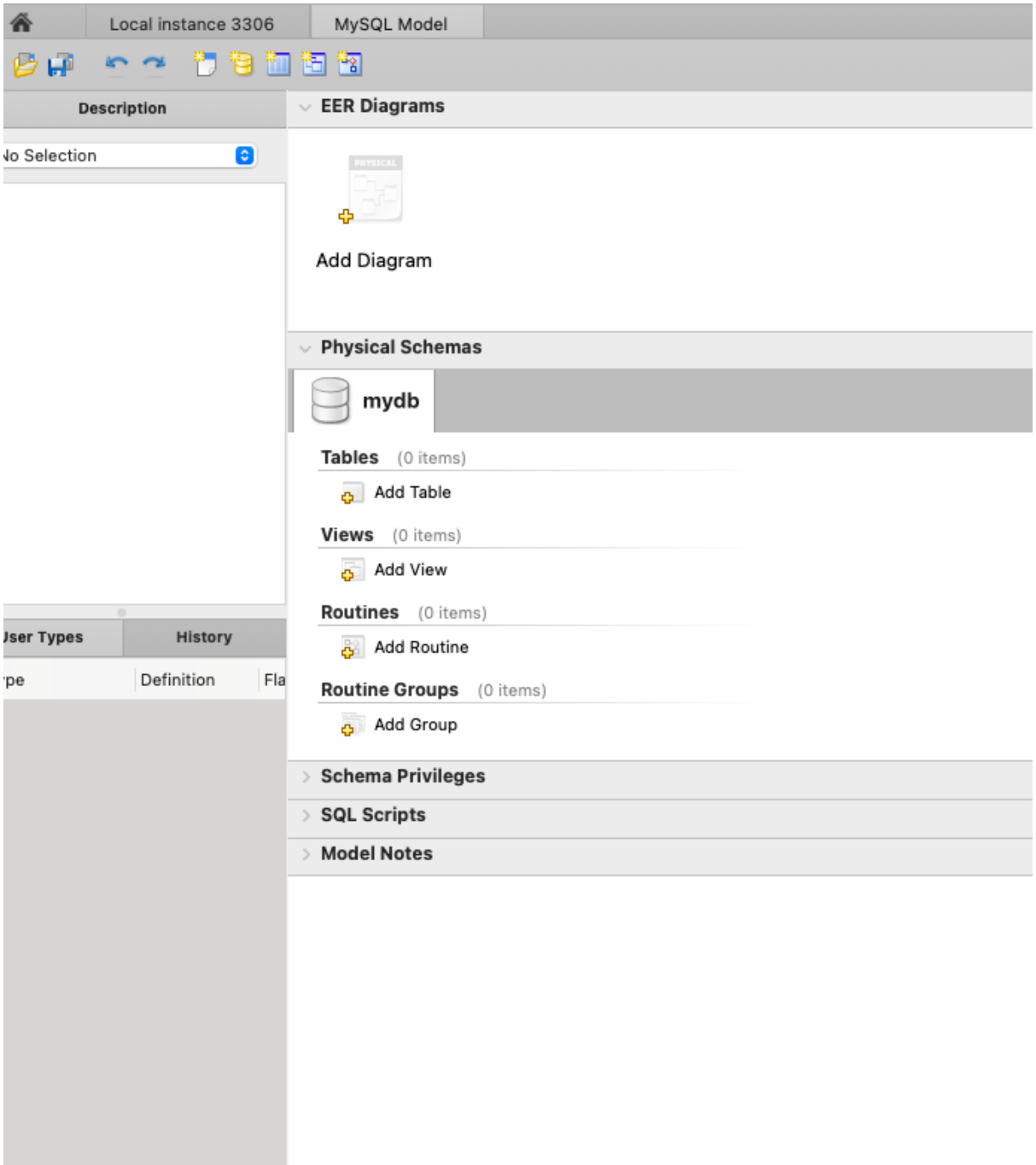
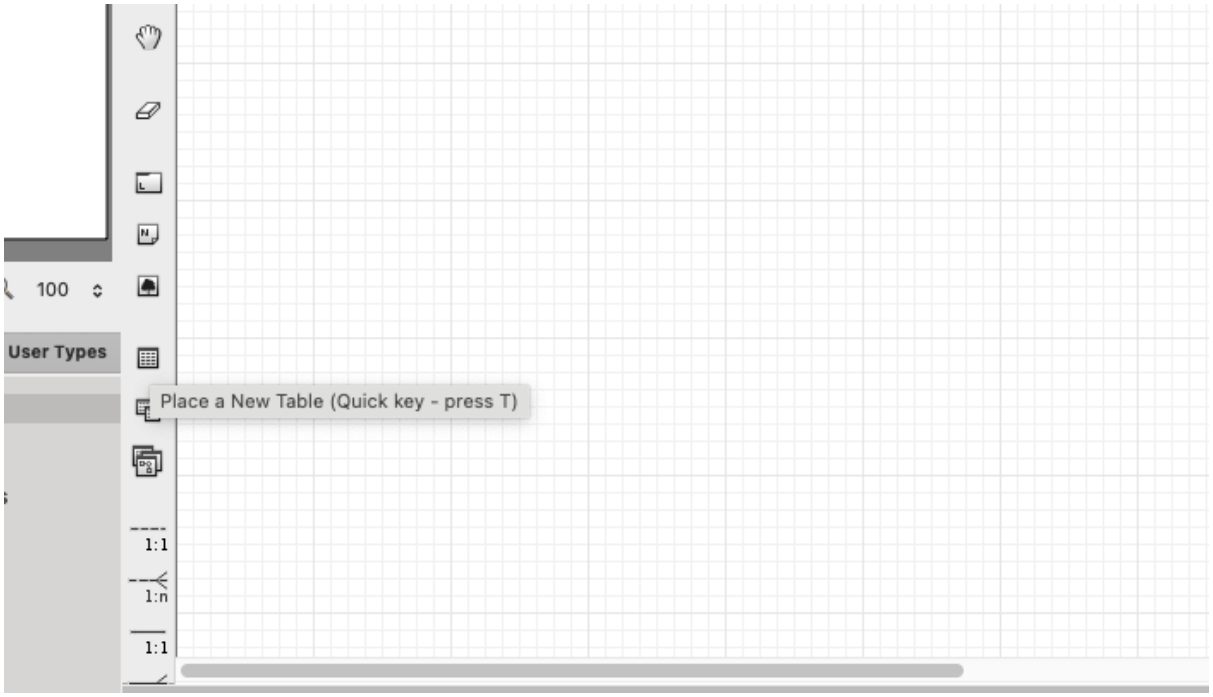
ENTITY RELATIONSHIP DIAGRAM

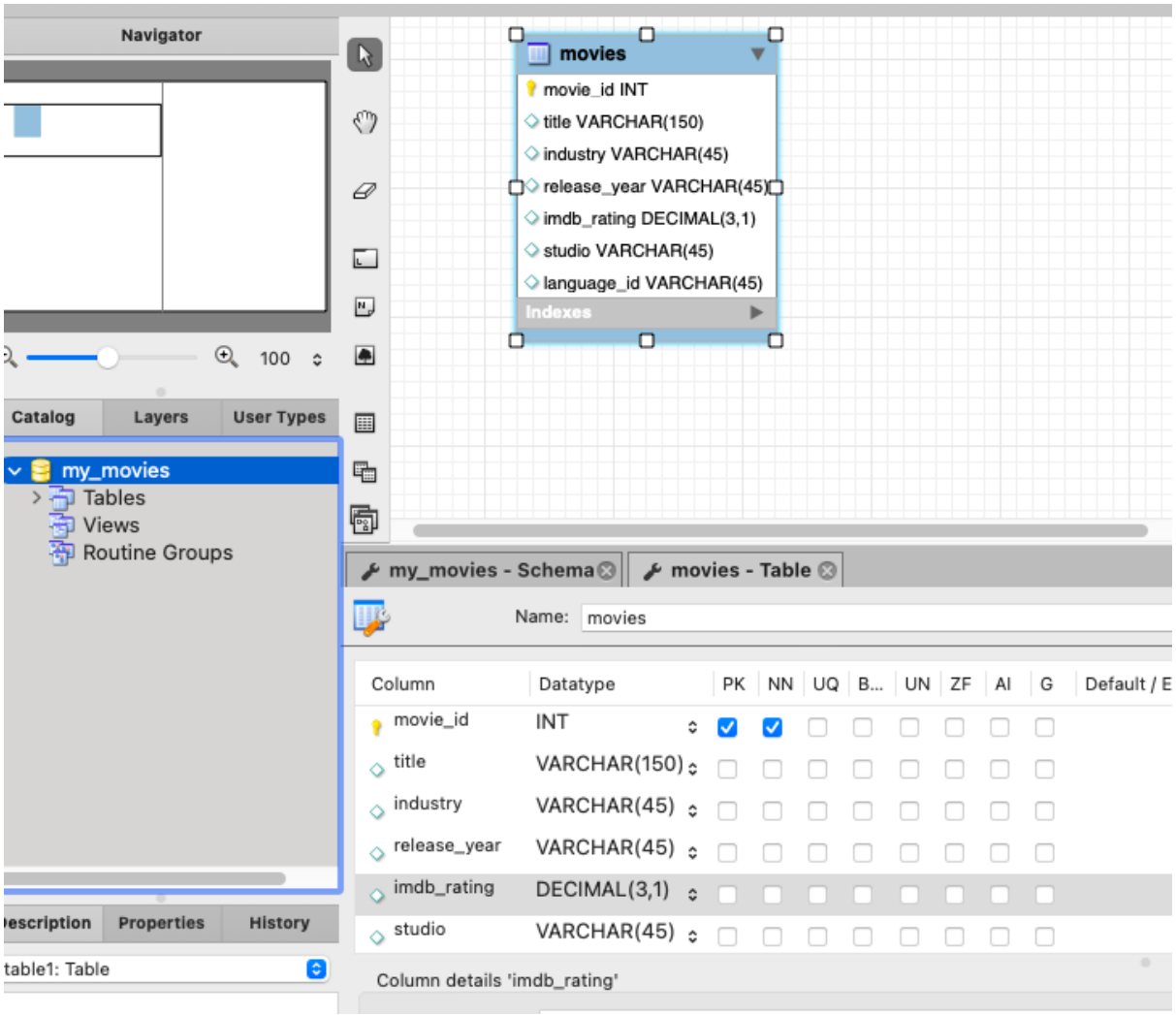
1.SQLWORKBENCH → FILE → New Model



2. Double Click on add Diagram

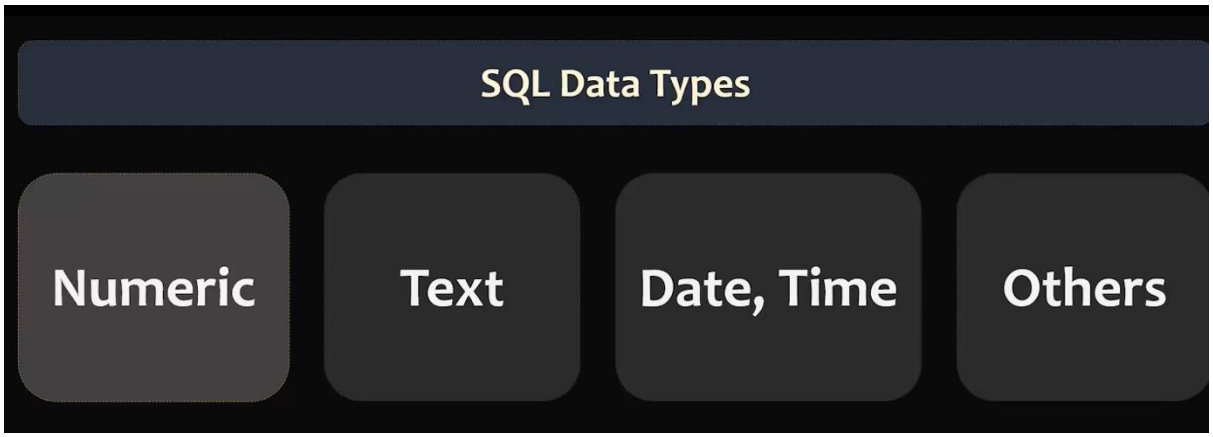


- Double click on place a new table and click the table on sheet.
- Double click over the table. A column structure will appear at the bottom.

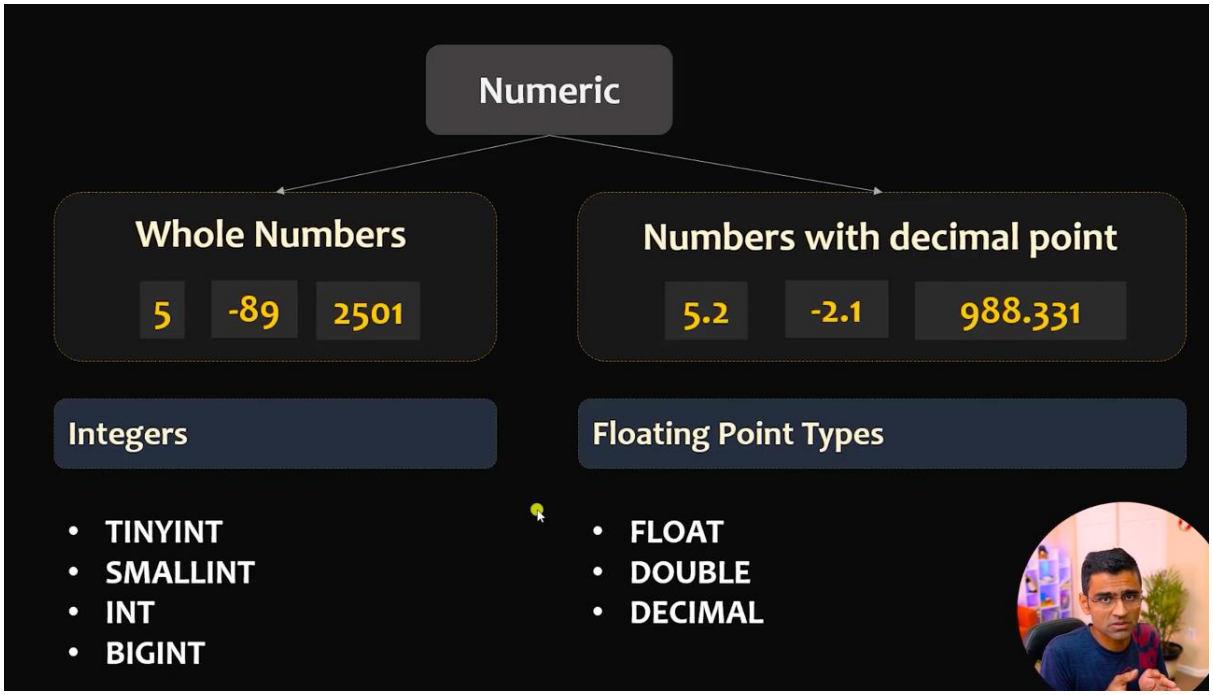


- Double click on mydb(default name and change it to your required name (my_movie)
- Varchar data type is flexible with the memory size of the data
- Release_year will have YEAR() as data type.

SQL DATATYPE

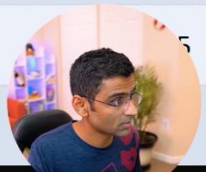


1. NUMERIC DATA TYPE



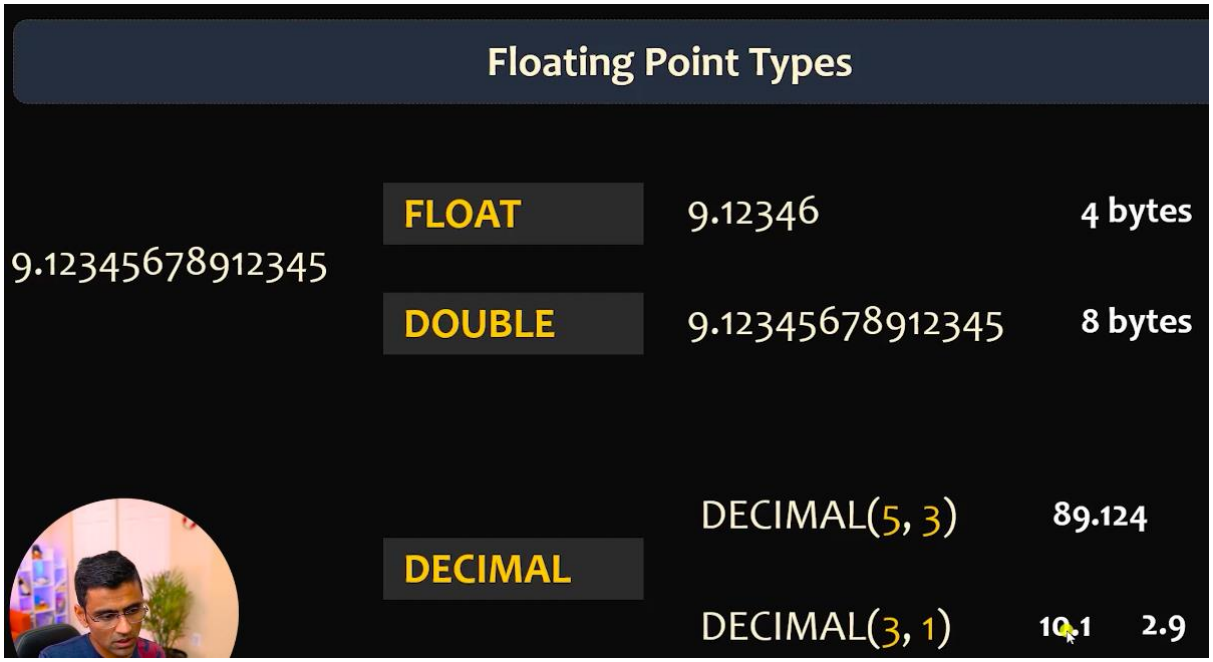
1.1 Whole Number Datatype

		Signed		Unsigned	
Type	Storage (Bytes)	Min	Max	Min	Max
TINYINT	1	-128	127	0	255
SMALLINT	2	-32768	32767	0	65535
MEDIUMINT	3	-8388608	8388607	0	16777215
INT	4	-2147483648	2147483647	0	
BIGINT	8	-2^{63}	$2^{63}-1$	0	

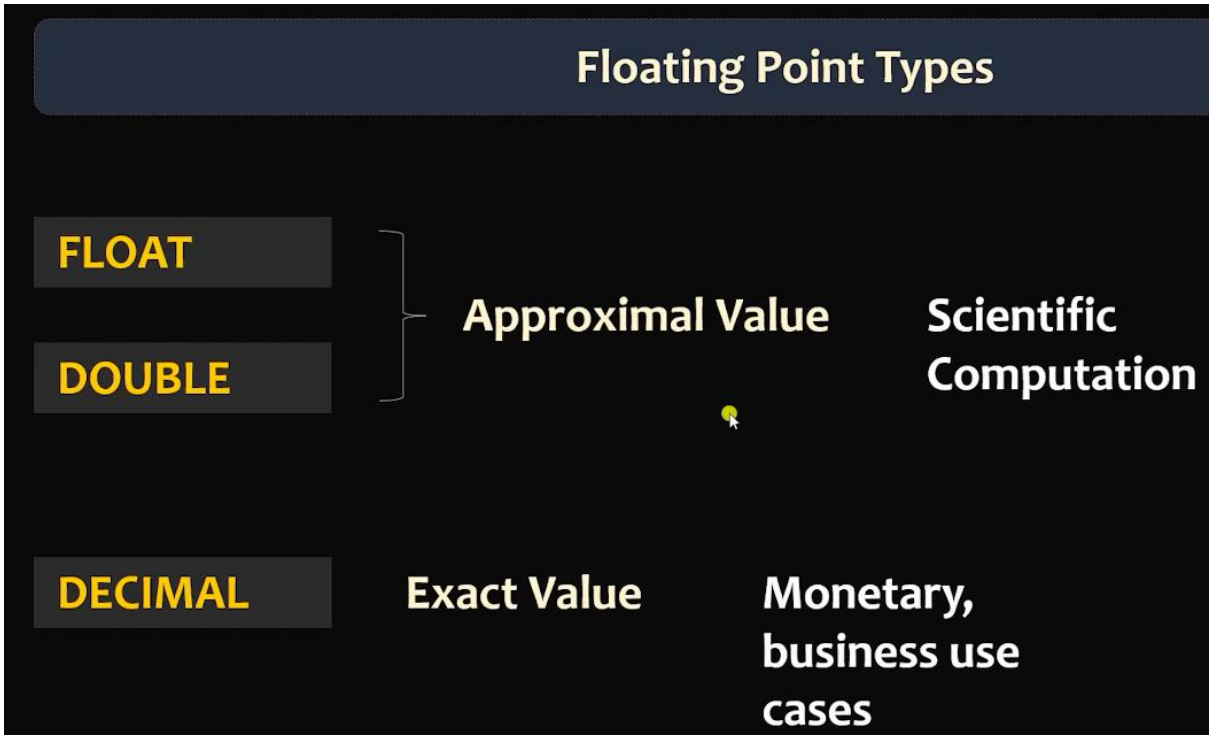


Difference between Signed and Unsigned in MySQL

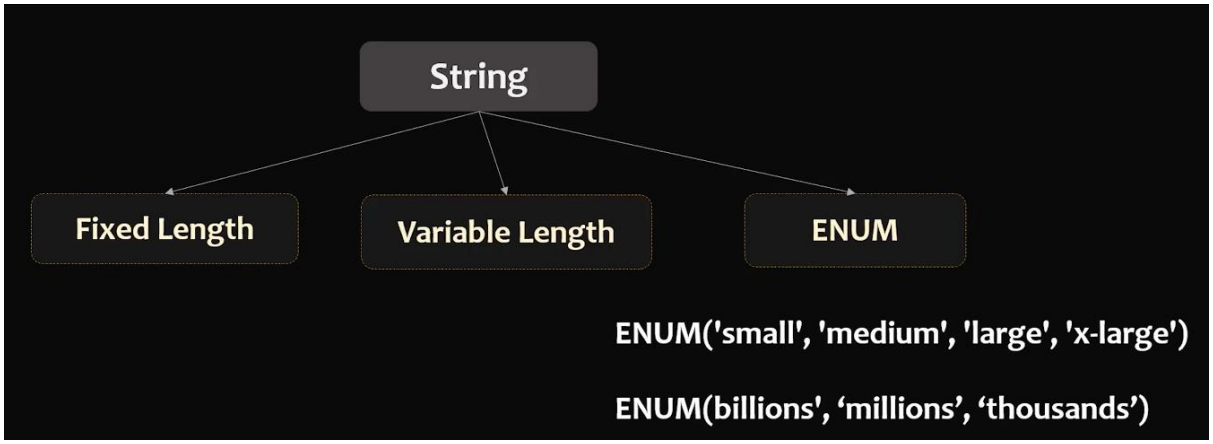
- **Signed value** - Variables with signed numerals can store 0, positive, and negative numbers.
- **Unsigned value** - Variables with unsigned numerals can store only 0 and positive numbers.
- languages can't be more than 255 so we can use TINYINT for it.
- movie_id can be more than 255 so we can't use TINY int for it.



- Float memory is of 4 bytes.(will round off the value onwards)
- Double memory is of 8 bytes.(will store the entire value)



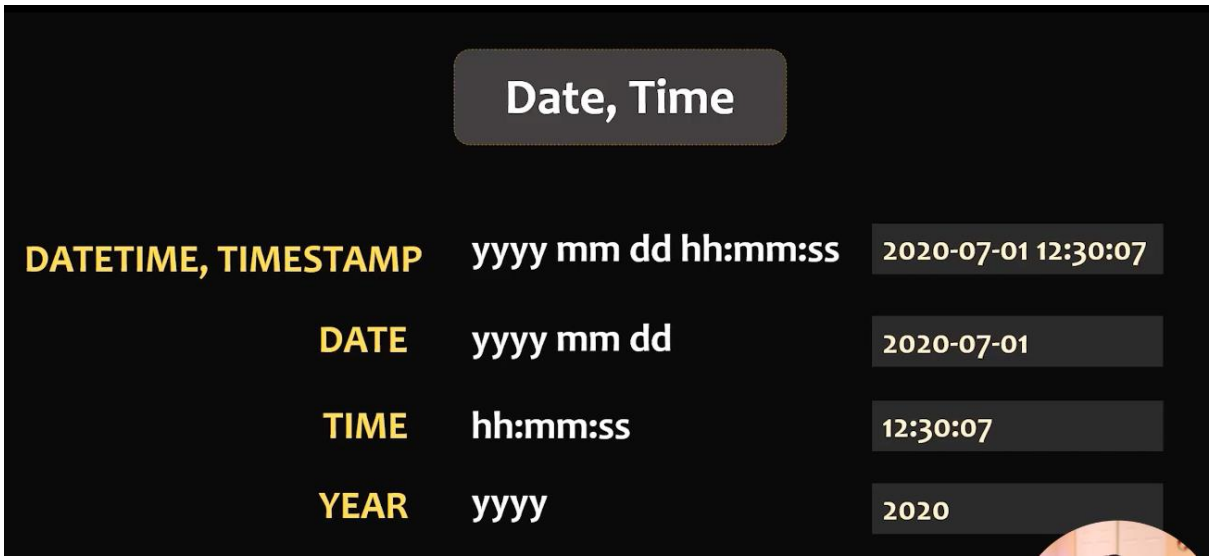
2.STRING DATA TYPE



- Enum is for the values which are predefined as given in the example.
- Blob is a data type that will enable you to store images as binary text in a table.

3.DATE, TIME

- CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP can be used to update timestamp with every entered data.



4.JSON Datatype

- JSON is a popular and efficient data type to store massive amount of data.
- '→' operator is used to extract a JSON object.
- SPATIAL datatype is used to represent geospatial data types like latitude, longitude etc.

```
1 • SELECT *, ST_AsText(location) FROM sakila.address;
```

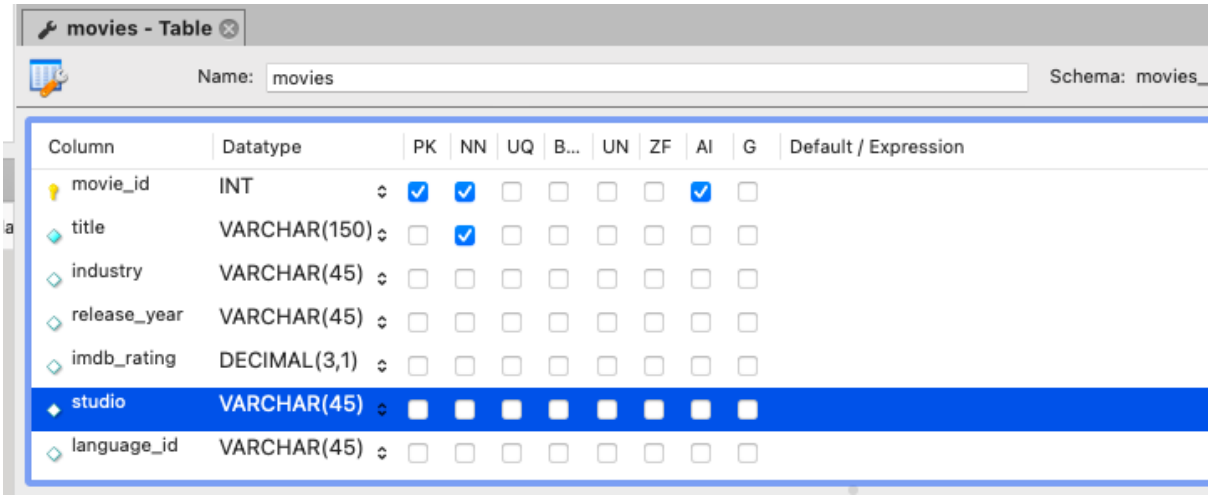
address_id	address	address2	district	city_id	postal_code	phone	location	last_u	ST_AsText(location)
1	47 MySakila Drive	NULL	Alberta	300			BLOB	20...	POINT(-112.8185647 49.6999986)
2	28 MySQL Boulevard	NULL	QLD	576			BLOB	20...	POINT(153.1408538 -27.6333361)
3	23 Workhaven Lane	NULL	Alberta	300		14033...	BLOB	20...	POINT(-112.8185673 49.6999951)
4	1411 Lillydale Drive	NULL	QLD	576		61722...	BLOB	20...	POINT(153.1913094 -27.6333373)
5	1913 Hanoi Way		Nagasaki	463	35200	28303...	BLOB	20...	POINT(129.7227851 33.1591726)
6	1121 Loja Avenue		California	449	17886	83863...	BLOB	20...	POINT(-117.2897714 34.1083428)
7	692 Joliet Street		Attika	38	83579	44847	BLOB	20	POINT(23.7[illegible]2294 37.9794508)

- The above command will display the geometrical location in form of text.

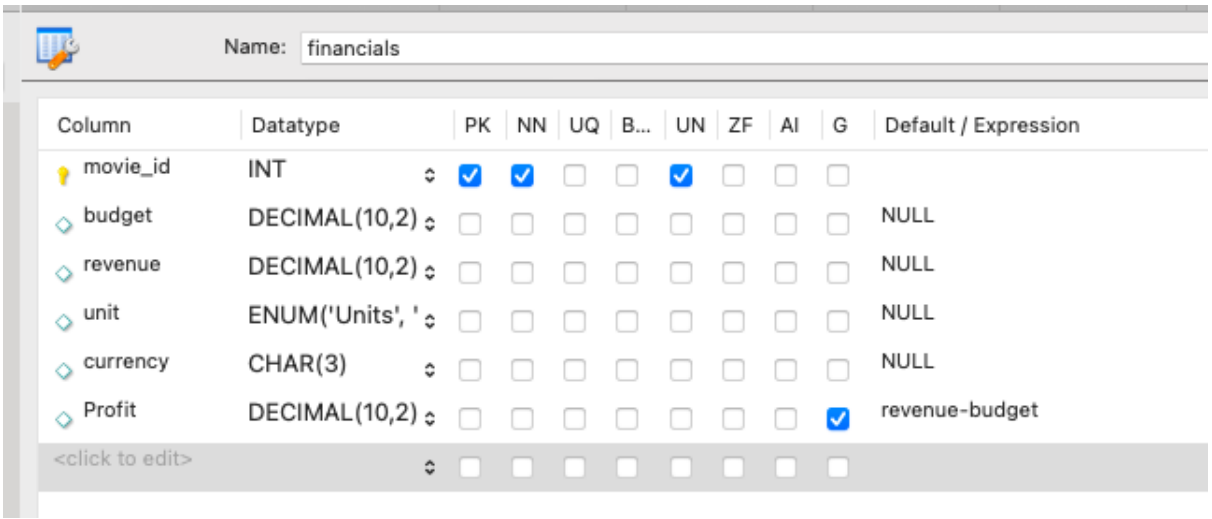
KEYS

- Primary key is unique identifier which cannot have any duplicates.
- Primary key that already exists database is called natural key.
- Primary key that is generated by user artificially is called surrogate key.
- Composite key is a primary key that is generated by combining multiple columns.

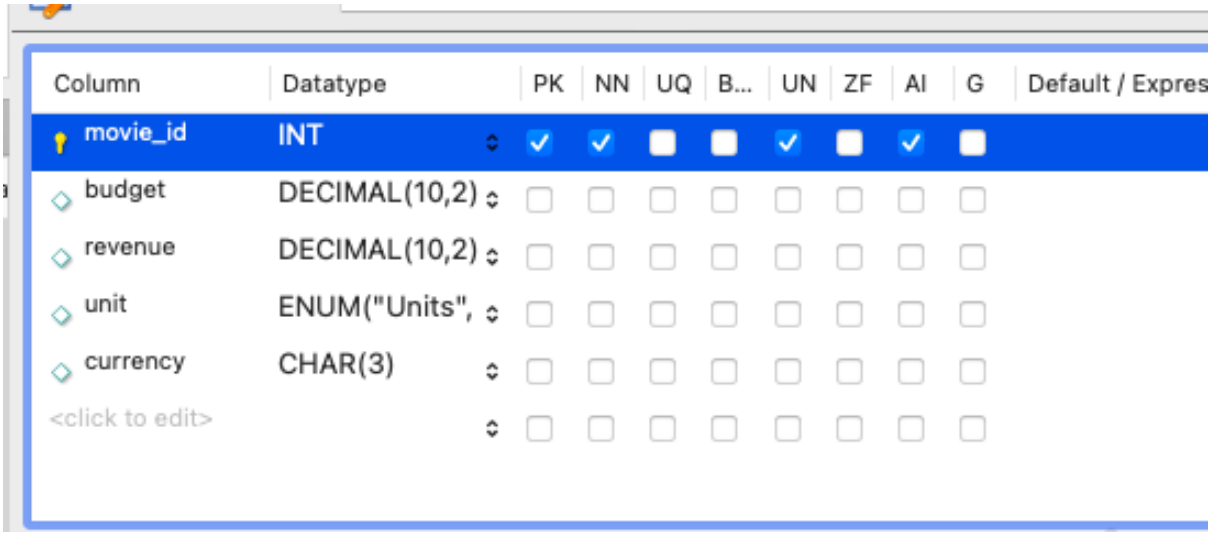
- A natural key is a key that is derived from the data itself, such as a customer ID, a product code, or a date. A surrogate key is a key that is generated artificially, such as a sequential number, a GUID, or a hash



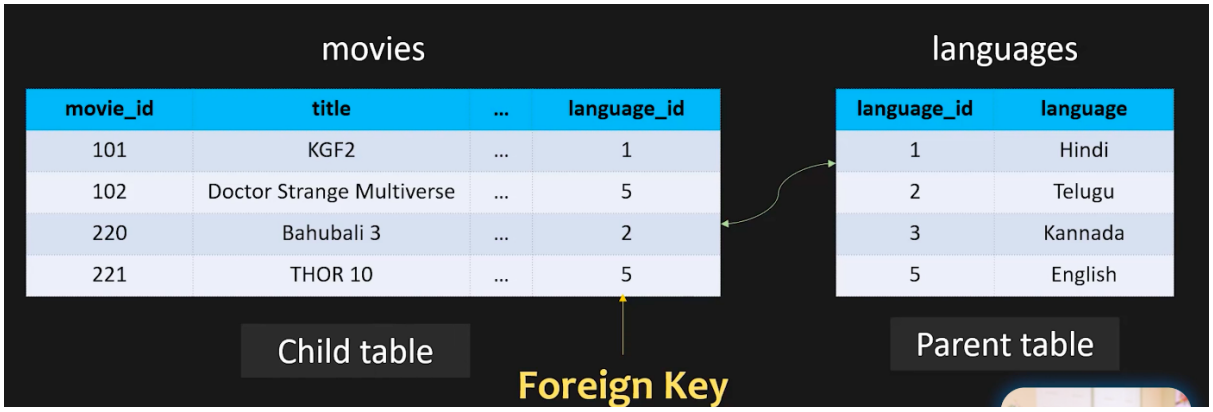
AI - Auto Increment
Will increase the movie id by 1 on addition of every movie id.



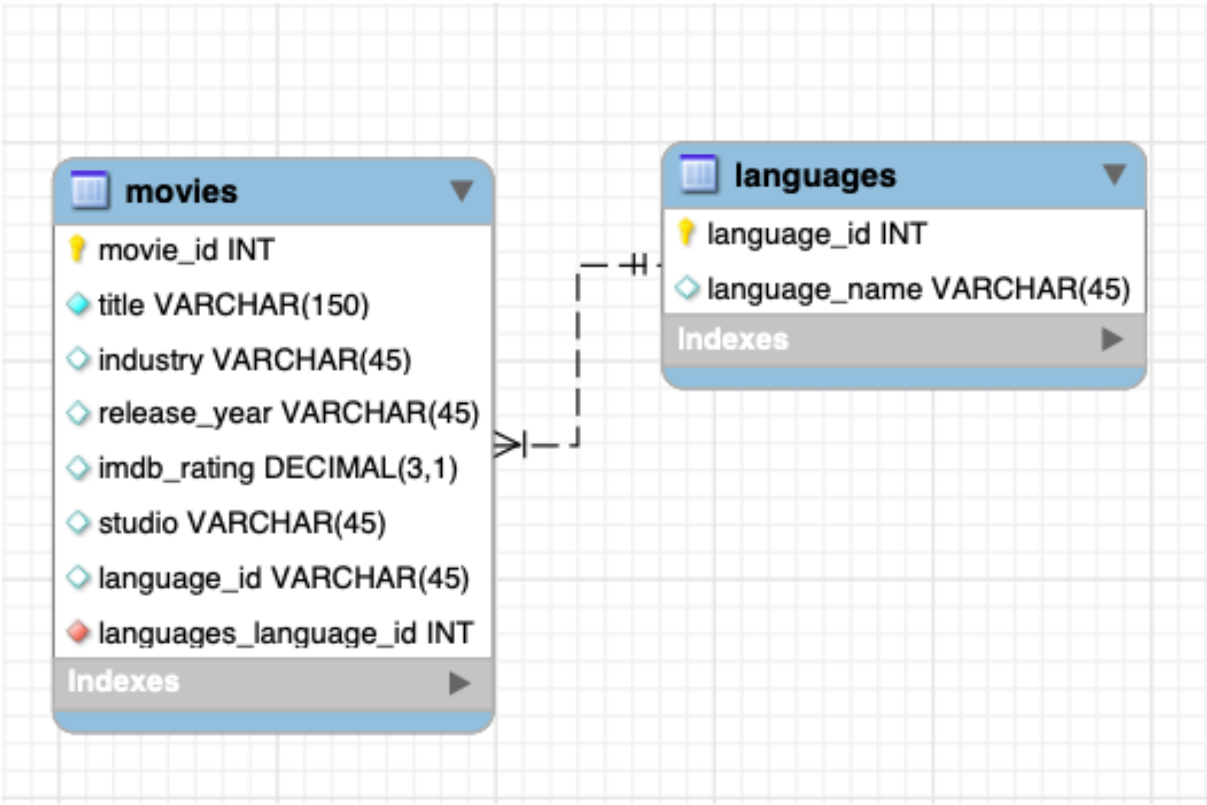
- **Here we have calculated profit column by using default expression (revenue-budget)**



FOREIGN KEY

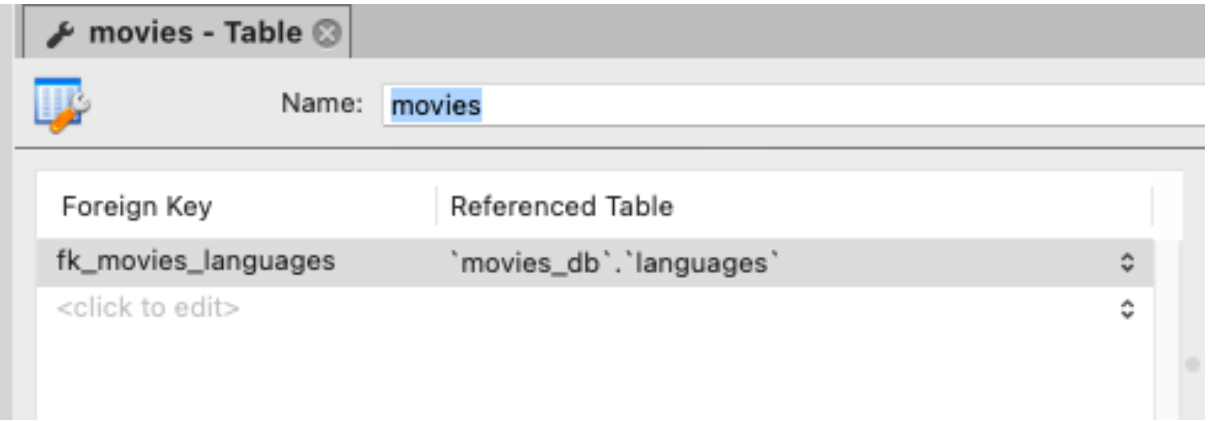


- A FOREIGN KEY is a field (or collection of fields) in one table, that refers to the PRIMARY KEY in another table.
- The table with the foreign key is called the child table, and the table with the primary key is called the referenced or parent table.
- A Foreign key is an attribute that is a Primary key in its parent table but is included as an attribute in the host table.

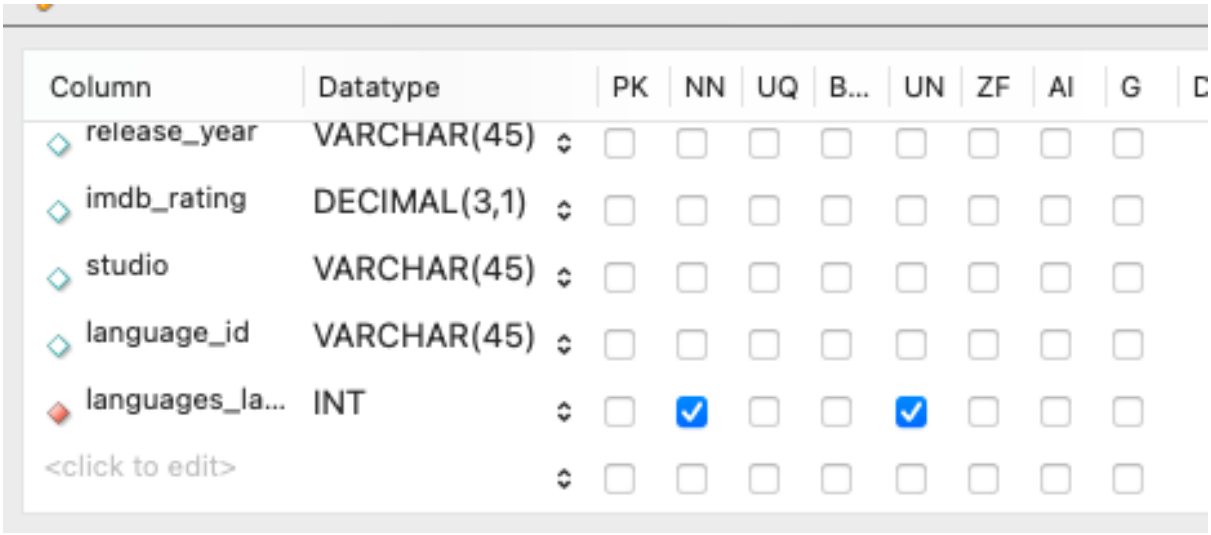


- The relationship between movies and languages is one to many.
- When you are establishing one to many relationship click on child table first and then parent table.
- The key benefit of creating relationship is to prevent having undesirable records in the database.

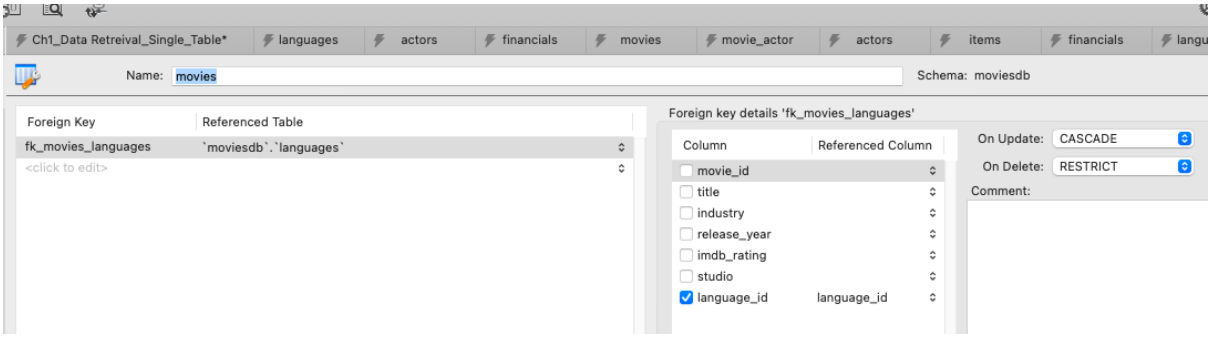
- By creating a relationship you can also update or delete records automatically in the child table based on the action you perform in the parent table.
- There are two types one to many relationships.
- **1. Non-identyfying relationship** - The above table will be non-identifying .
    - The reason is language_id which is foreign key in movies is not part of primary key of movies.
    - Basically language id does not identify unique record in movies table.The unique record in movies table is identified using movie id and language id as no part in it.
- **2. Identifying relationship**



- In foreign key column a foreign key is now registered.

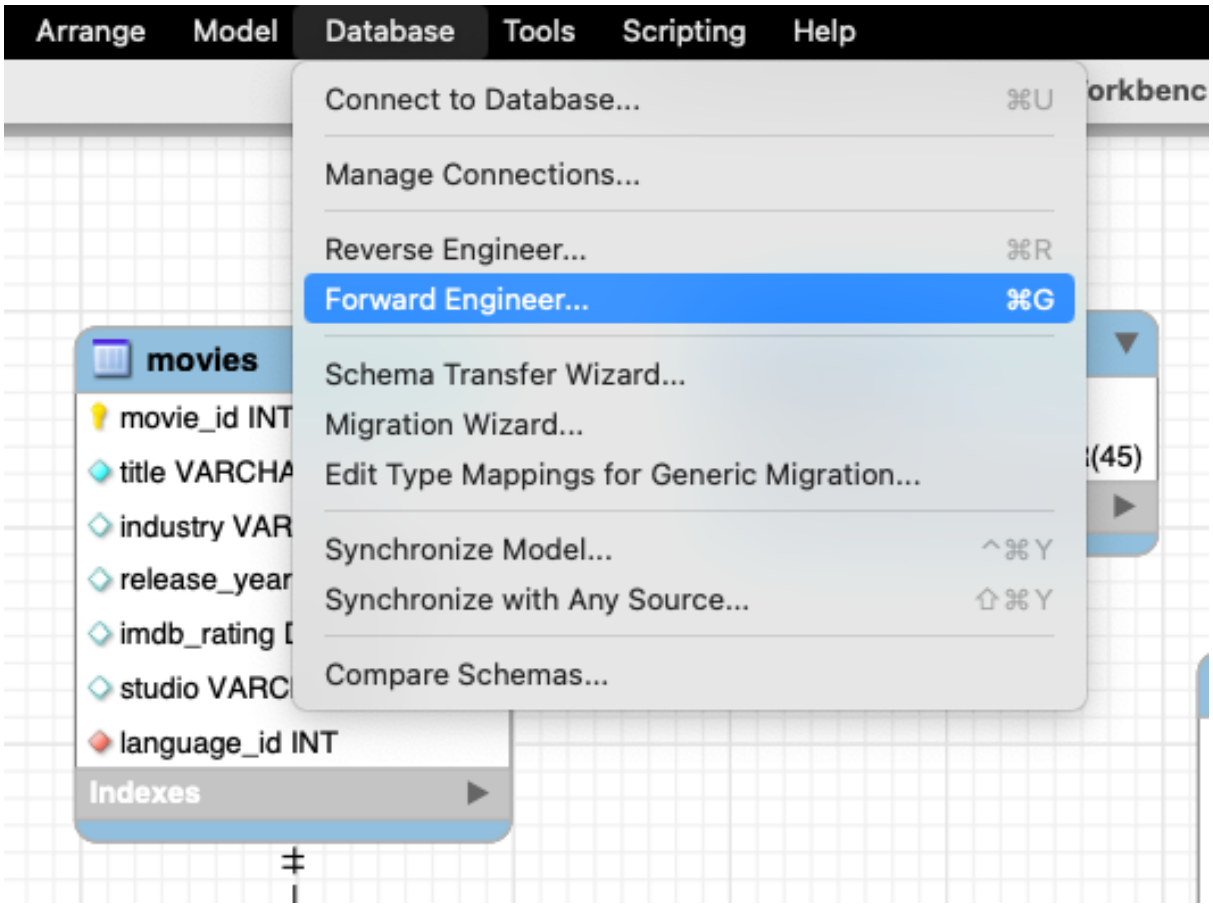


- You can remove the original language_id and rename the new column as language_id.

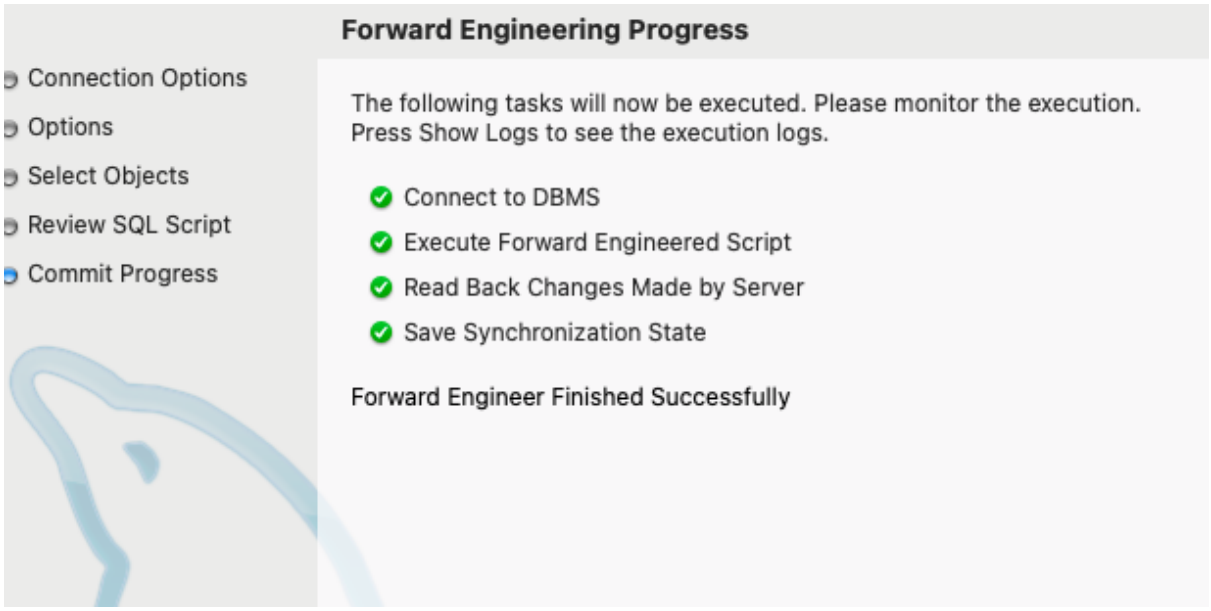


- The foreign key is set to on update : Cascade.
- Update in parent table will update on child table.
- On delete -Restrict will make sure that foreign key is not deleted.

Create a Database From an Entity Relationship Diagram - ERD

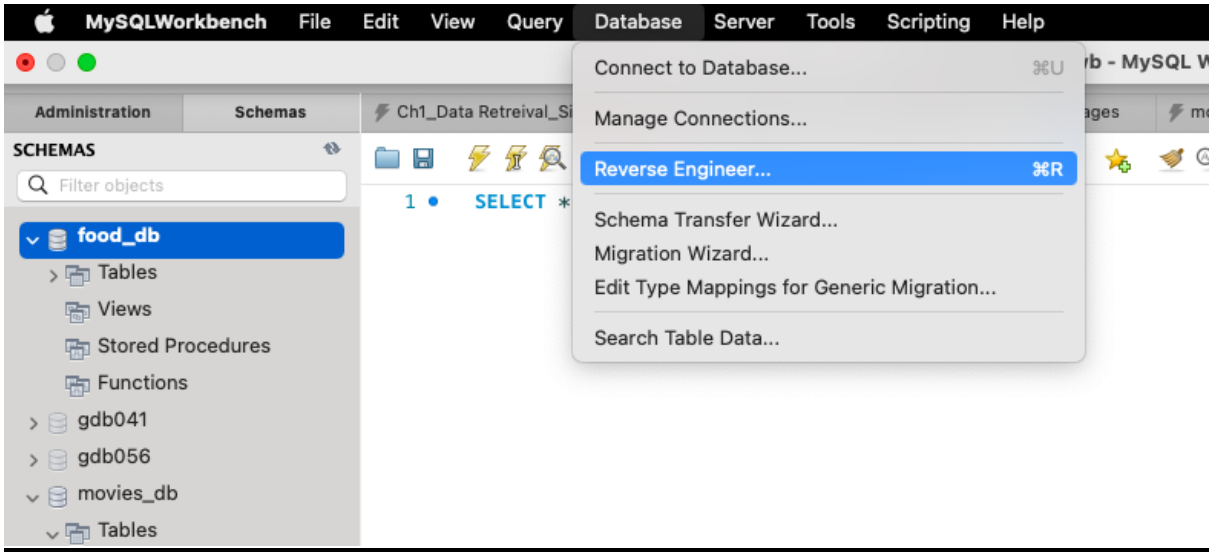


- DATABASE → FORWARD ENGINEER
- Forward engineer is the option to create a database from a data model

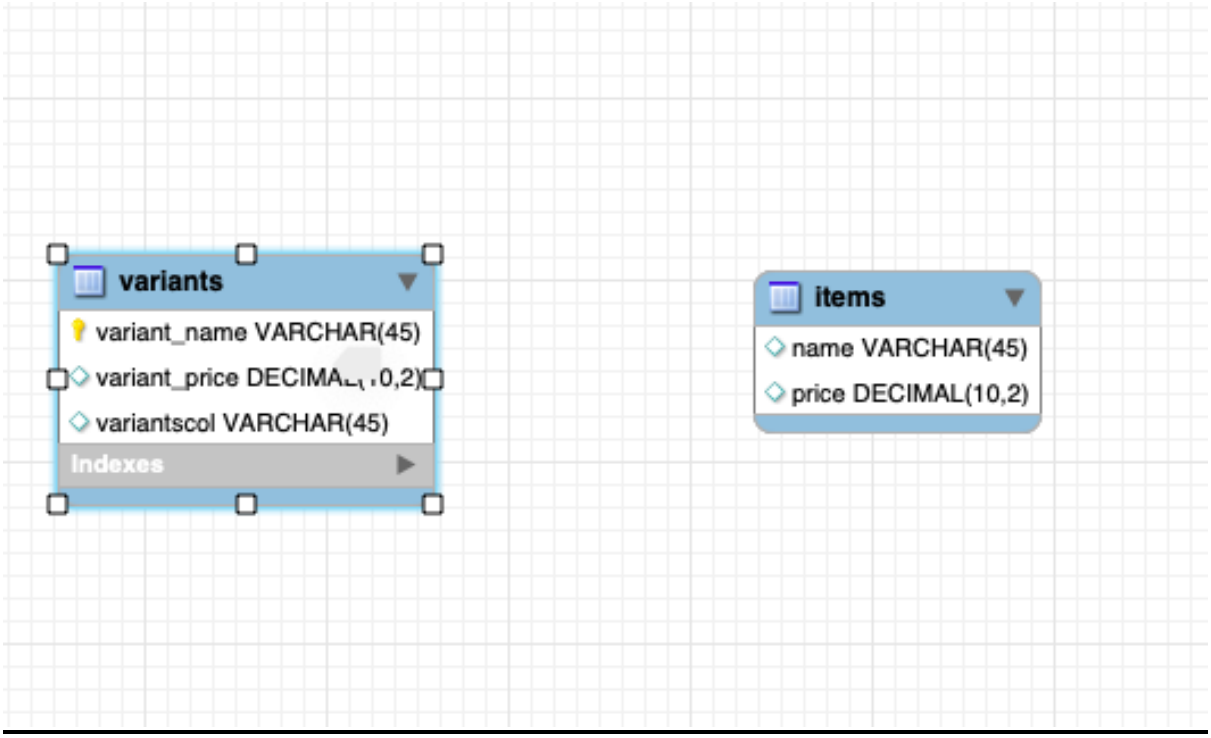


- **Perform all the steps.**

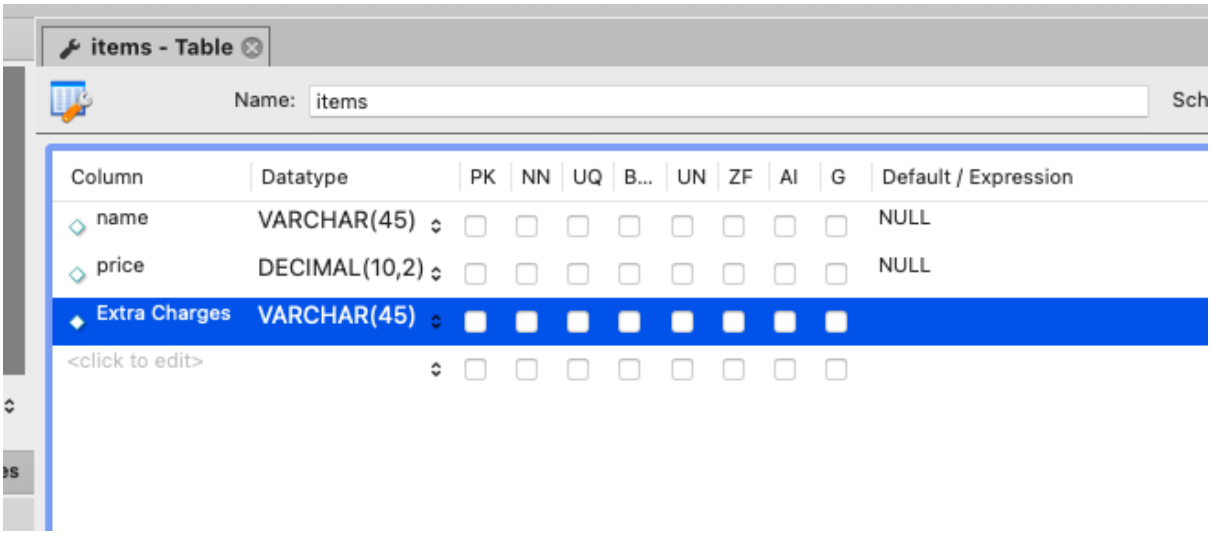
REVERSE ENGINEER



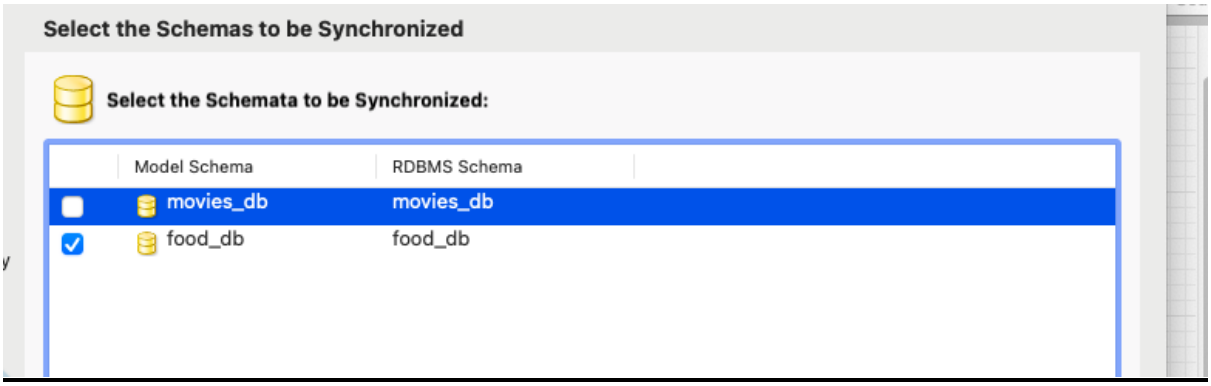
- Choose the table you want to modify.
- It is the option to create/edit a data model from database.

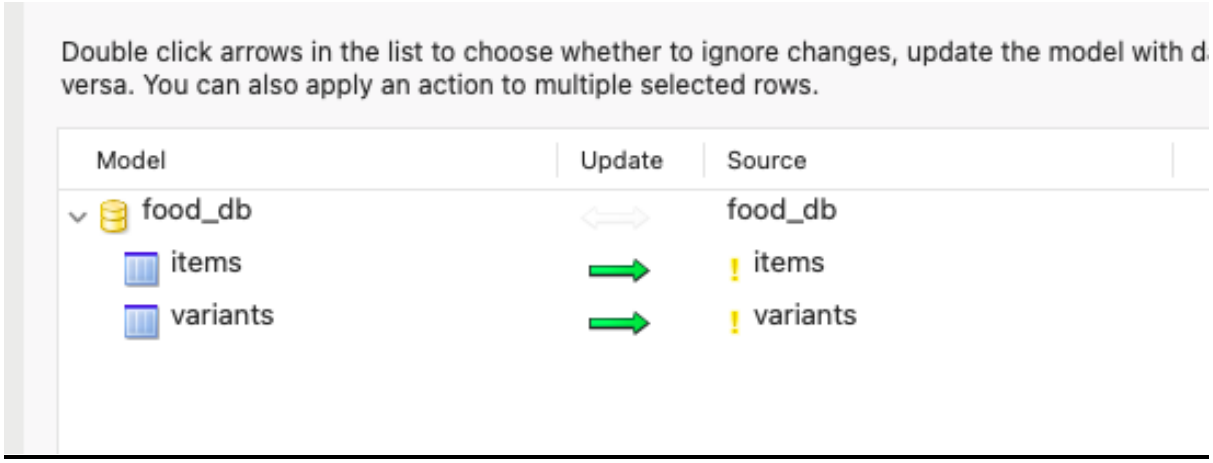


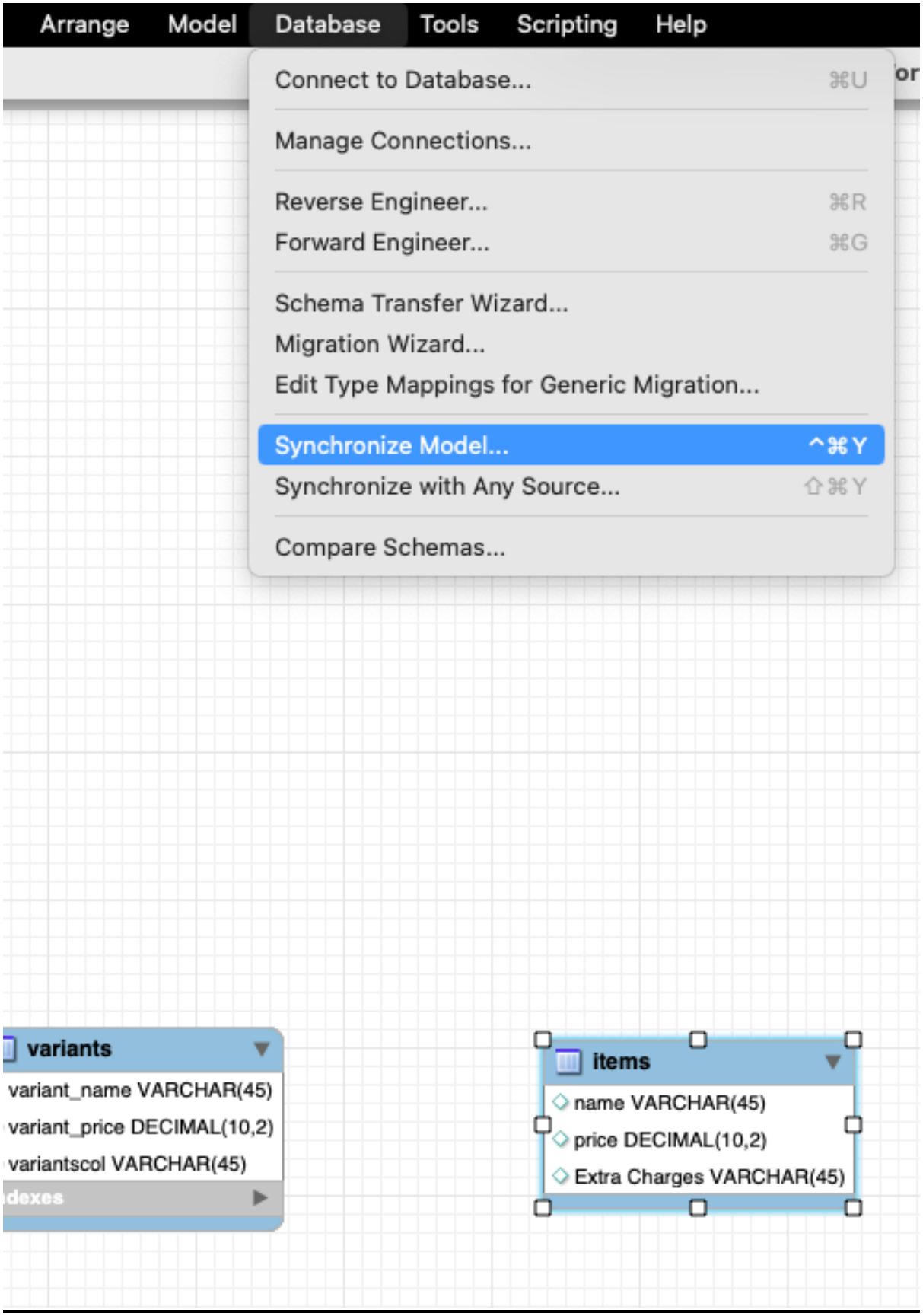
- Lets' say you have modified food data by adding extra charges column in items

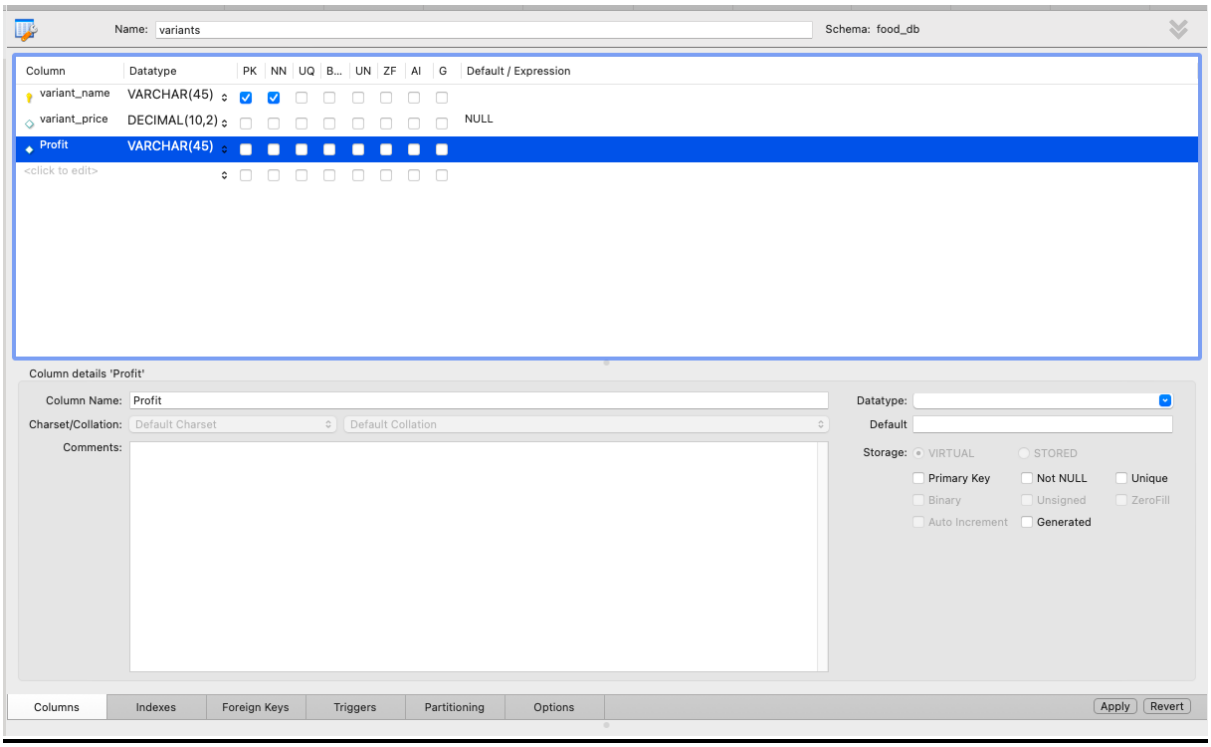


- **<u>Perform all the steps.</u>**

- 
Select the Schemas to be Synchronized
Select the Schemata to be Synchronized:
Model Schema
RDBMS Schema
movies_db
movies_db
food_db
food_db


Double click arrows in the list to choose whether to ignore changes, update the model with d
versa. You can also apply an action to multiple selected rows.
Model
Update
Source
food_db
food_db
items
items
variants
variants

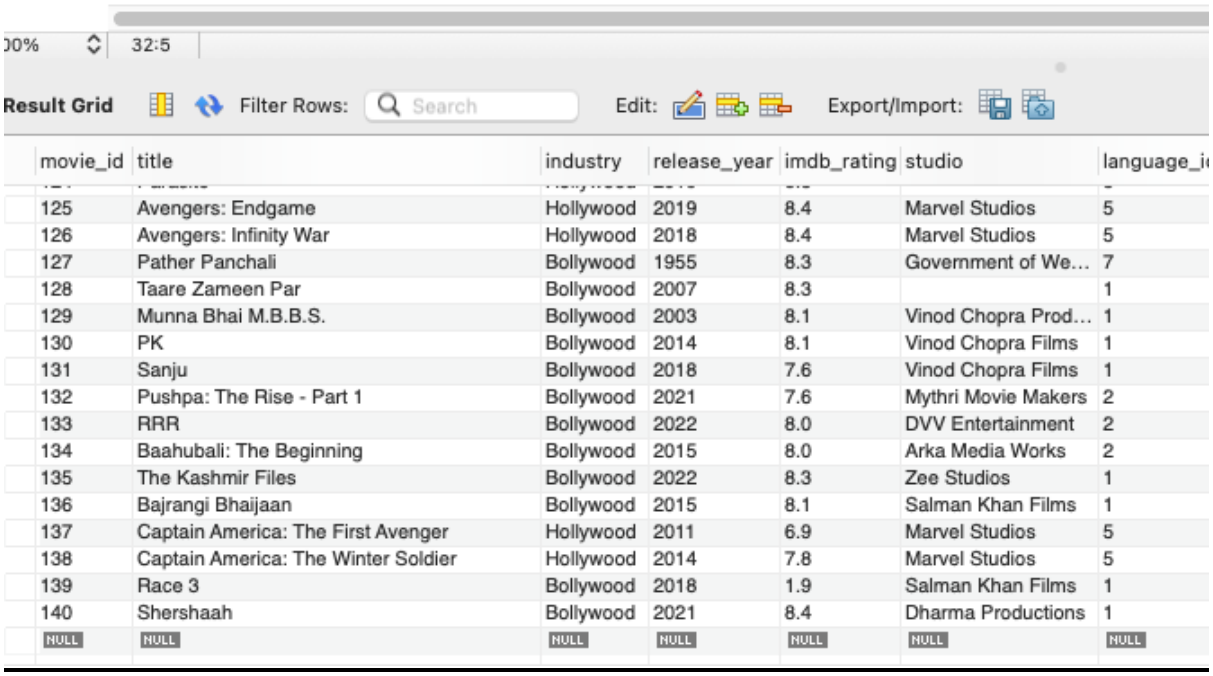

Arrange
Model
Database
Tools
Scripting
Help
Connect to Database... ⌘U
Manage Connections...
Reverse Engineer... ⌘R
Forward Engineer... ⌘G
Schema Transfer Wizard...
Migration Wizard...
Edit Type Mappings for Generic Migration...
Synchronize Model... ^⌘Y
Synchronize with Any Source... ⇧⌘Y
Compare Schemas...
variants
variant_name VARCHAR(45)
variant_price DECIMAL(10,2)
variantscol VARCHAR(45)
items
name VARCHAR(45)
price DECIMAL(10,2)
Extra Charges VARCHAR(45)

- One can also make changes directly in the database which is a popular practice.
- Click on the settings icon besides the table and add new column.
- 

- **INSERTING ROWS IN TABLE**
- **You can insert rows manually by typing below or can use Insert into**

:

```
1 •  INSERT INTO movies
2    VALUES (141,"BAHUBALI","Bollywood",2021,9,"Arka Media Works");
3
4
5 •  SELECT * FROM movies_db.movies;
```



00% 32:5

Result Grid | Filter Rows: Search | Edit: | Export/Import:

movie_id	title	industry	release_year	imdb_rating	studio	language_i
125	Avengers: Endgame	Hollywood	2019	8.4	Marvel Studios	5
126	Avengers: Infinity War	Hollywood	2018	8.4	Marvel Studios	5
127	Pather Panchali	Bollywood	1955	8.3	Government of We...	7
128	Taare Zameen Par	Bollywood	2007	8.3		1
129	Munna Bhai M.B.B.S.	Bollywood	2003	8.1	Vinod Chopra Prod...	1
130	PK	Bollywood	2014	8.1	Vinod Chopra Films	1
131	Sanju	Bollywood	2018	7.6	Vinod Chopra Films	1
132	Pushpa: The Rise - Part 1	Bollywood	2021	7.6	Mythri Movie Makers	2
133	RRR	Bollywood	2022	8.0	DVV Entertainment	2
134	Baahubali: The Beginning	Bollywood	2015	8.0	Arka Media Works	2
135	The Kashmir Files	Bollywood	2022	8.3	Zee Studios	1
136	Bajrangi Bhaijaan	Bollywood	2015	8.1	Salman Khan Films	1
137	Captain America: The First Avenger	Hollywood	2011	6.9	Marvel Studios	5
138	Captain America: The Winter Soldier	Hollywood	2014	7.8	Marvel Studios	5
139	Race 3	Bollywood	2018	1.9	Salman Khan Films	1
140	Shershaah	Bollywood	2021	8.4	Dharma Productions	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL

- Lets' say you don't have the entire row data.

```
•  insert into movies
   (title,industry,release_year)
   values("Bahubali 4","bollywood",2030);

•  SELECT * FROM movies_db.movies;
```

- ---

- You can specify the column names with the data and update it using INSERT

IF YOU DON’t want to specify column you can write Null in the place of missing position

```
INSERT INTO movies

values(DEFAULT,"Bahubali4","Bollywood",2030,NULL,NULL,NULL);
```

Inserting multiple values

```
INSERT INTO movies

values
(DEFAULT,"Bahubali4","Bollywood",2030,NULL,NULL,NULL),
(DEFAULT,"Bahubali 5","Bollywood",2035,NULL,NULL,NULL),
(DEFAULT,"Bahubali 6","Bollywood",2035,NULL,NULL,NULL)
;
```

UPDATE

YOU can update any particular row of information using update function

```
UPDATE movies
SET studio="Warner Bros.",language_id=5
WHERE movie_id=143;
```

- Here we want to update studio and language_id of a particular movie.
- We can use the primary key to do so.

- Let's say you want to update all the Bahubali movies with it's respective production companies.
- You should first filter out the movies using primary key and do the modification.

```
SELECT movie_id from movies where title like "%Bahubali%";

UPDATE movies
SET studio = "Arka Media Works" , language_id=2
WHERE movie_id in (141,142,143,144,145,146);
```

DELETING FROM ANY PARTICULAR COLUMN

```
DELETE from movies
where movie_id = 146;
```

DELETING MULTIPLE COLUMNS

```
DELETE from movies
where movie_id in (144,145,143);
```