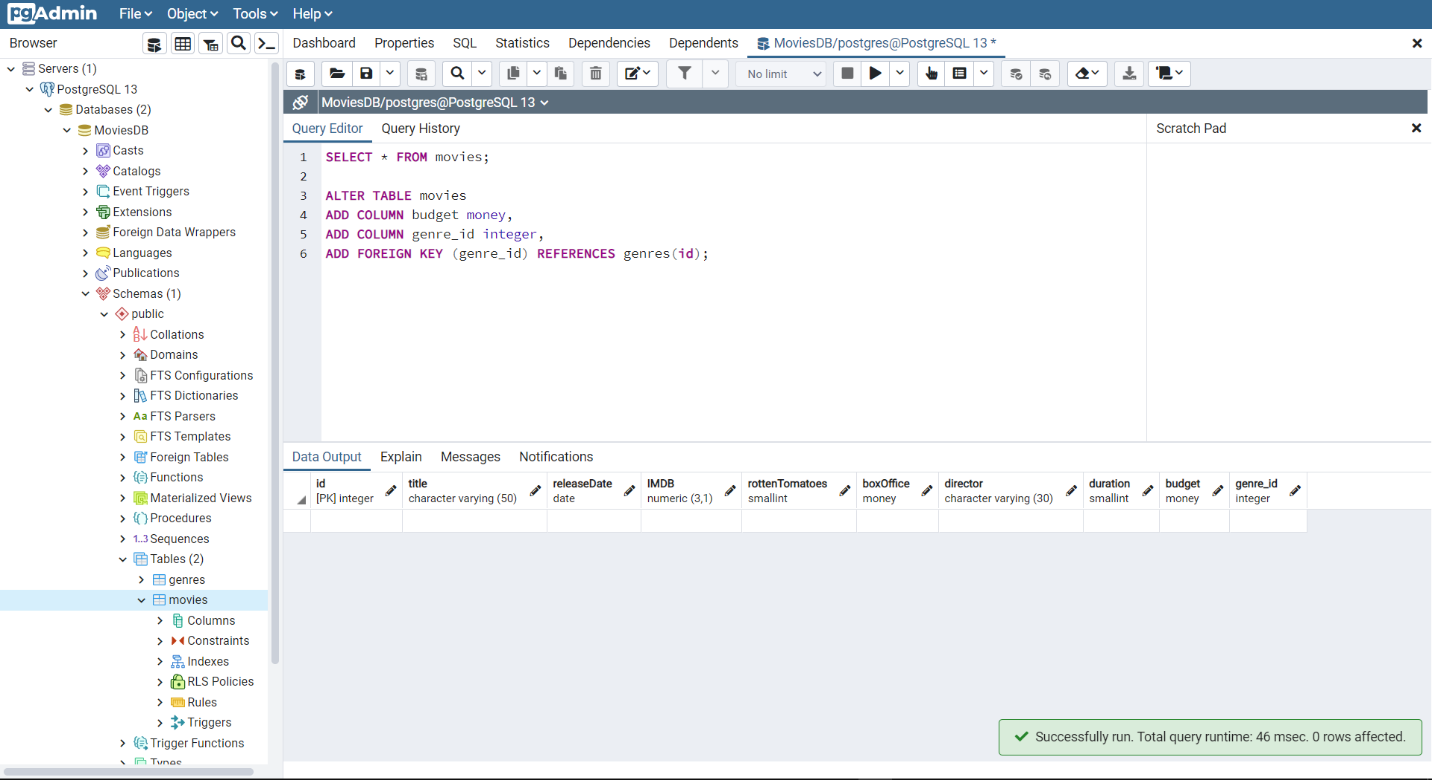
# سوال اول



با دستوری که در عکس آمده است در دیتابیس MoviesDB ، جدول movies ستون جدیدی به نام budgetبا نوع moneyساخته شده. دستور استفاده شده:

ALTER TABLE movies

ADD COLUMN genre\_id integer,

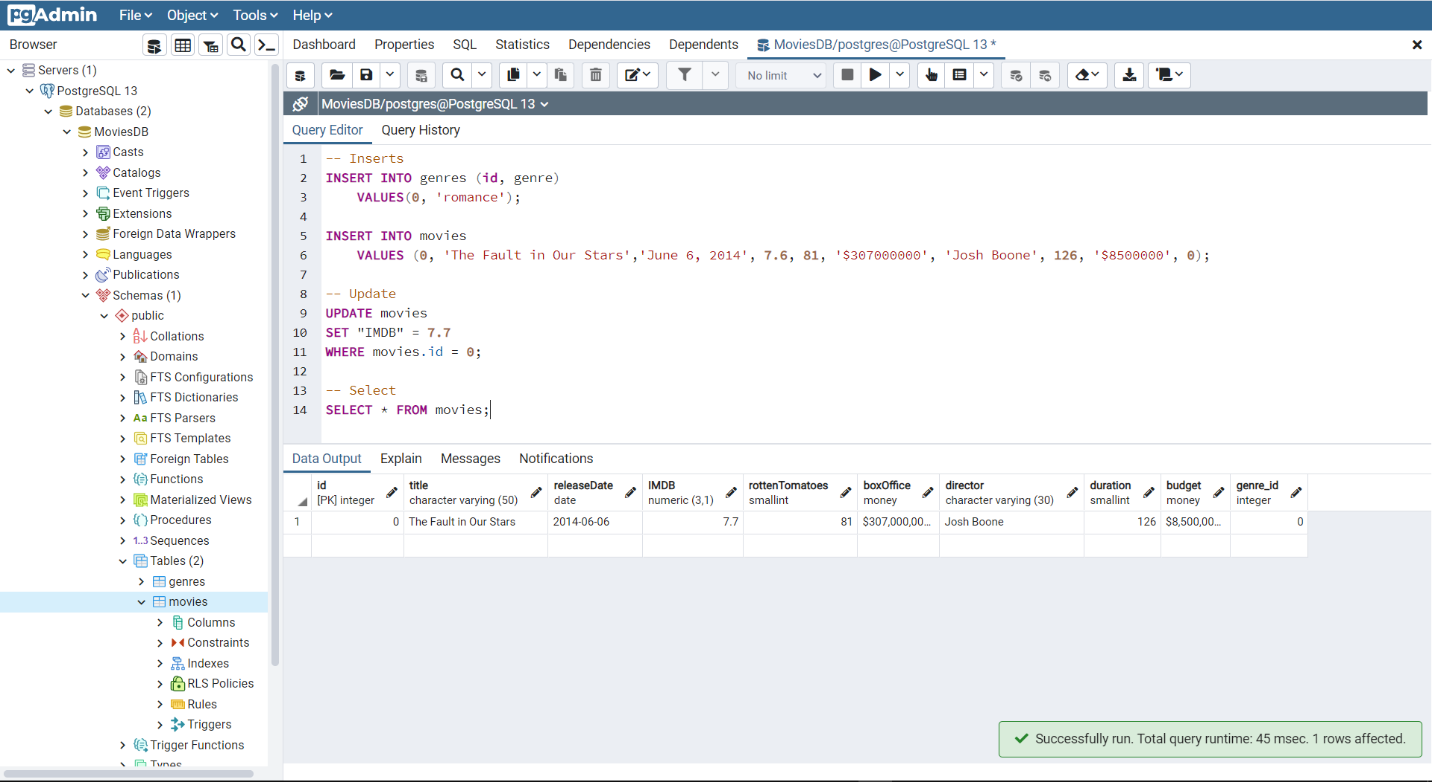
ADD COLUMN budget money;

# سوال دوم

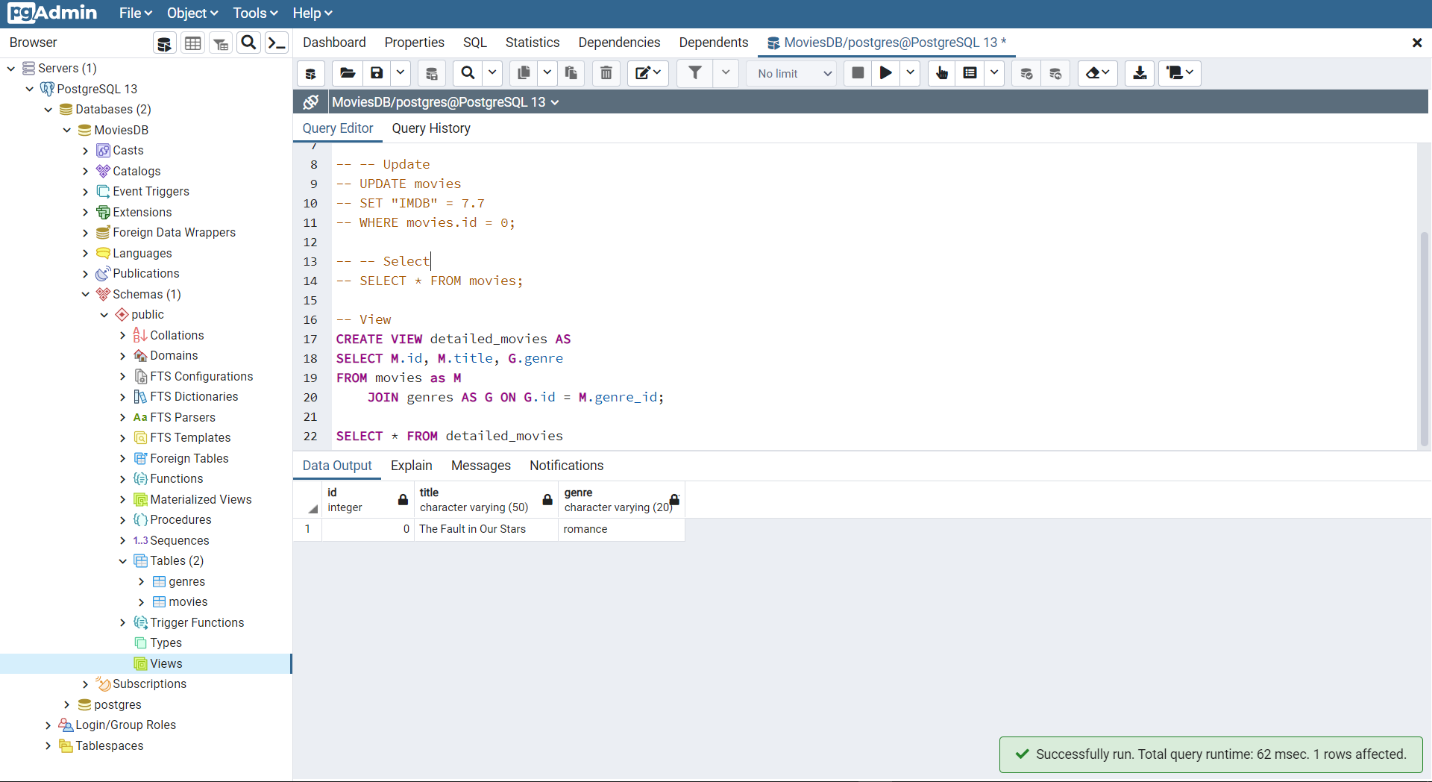
میدانیم برای تعیین کلید اصلی باید مدنظر داشته باشیم که یکتا باشد، همچنین سعی بر آن است که کلید optimal باشد یعنی در صورت امکان از فیلد های موجود استفاده کنیم تا فیلد جدیدی برای PK بسازیم. همچنین برای تعیین کلید خارجی باید در نظر داشته باشیم که هدف در بسیاری از مواقع encode کردن مقادیر یک ستون است (مثلا تکراری نشدن مقادیر رشته ای یک ستون One-to-many) و بهتر است از بهینه ترین data type برای این کلید استفاده شود. همچنین نوع آن با نوع ستون ارجاع شده یکسان باشد. ستونی که کلید خارجی به آن ارجاع داده میشود حتما باید کلید اصلی باشد.

با جدول movies که در سوال اول تغییراتی دادیم کار میکنیم. بهترین کلید اصلی میتواند (title, releaseDate) باشد زیرا title میتواند لزوما یکتا نباشد اما در کنار releaseDate یک زوج مرتب قطعا یکتا ساخته میشود. (البته همانطور که در سوال یک مشاهده میکنید فعلا کلید اصلی یک id است زیرا در طراحی شرطی بر کلید اصلی نبود و طراحی برای جلسه پیشین است). برای کلید خارجی میتوان genre\_id را تعریف کرد که به ستون Id در جدول genres اشاره دارد. (فرض کنیم هر فیلم فقط یک genre دارد!) به طراحی به آن دلیل است که genre ها از نوع رشته هستند و تکرار این ها به طول مستقیم در جدول اصلی طراحی خوبی نیست و حجم بالاتری مصرف میکند.

# سوال سوم



# سوال چهارم



همانطور که در عکس مشاهده میکنید، با دستور

CREATE VIEW detailed\_movies AS

SELECT M.id, M.title, G.genre

FROM movies as M

JOIN genres AS G ON G.id = M.genre\_id;

ویو مورد نظر ساخته شد و با دستور

SELECT \* FROM detailed\_movies

ویو مورد نظر نمایش داده شد.

# سوال چهارم

## عمل گر منطقی

* + AND: برای اشتراک گرفتن چندین شرط where استفاده میشود و این قابلیت را میدهد که در یک query چندین شرط را بطور همزمان برقرار کنیم.
  + BETWEEN: برای تعیین کردن بازه به کار میرود به طوری که میتواند برای عدد، متن یا تاریخ باشد. نکته حائز اهمیت آن است که بازه شروع و پایان هم در محاسبات شامل میشوند یا به عبارتی inclusive است.

## عمل گر های مقایسه ای

* + عمل گر = برای برقراری تساوی بین دو طرف استفاده میشود به طوری که هر دو فیلد از یک جنس (تاریخ، عدد، رشته و ...) باید استفاده شده باشند تا به ارور نخوریم.
  + عمل گر =! دقیقا برعکس عمل گر = عمل کرده و نابرابری را ارزیابی میکند. محدودیت ها و شرایط این دو عمل گر یکی هستند.

## عمل گر های حسابی

* + عمل گر + برای جمع دو یا چند فیلد استفاده میشود به طوری که دیتا تایپ فیلد ها باید یکسان باشد اما در ذات دیتا تایپ محدودیت چندانی نداریم. به طوری که میتوان عدد و تاریخ (و در برخی تایپ ها – بسته به محیط دارد رشته (concat)) ها را جمع کرد.
  + عمل گر \* هم برای ضرب دو یا چند فیلد استفاده میشود. به طوری که برای تایپ های عددی و اعشاری مورد استفاده قرار میگیرد.