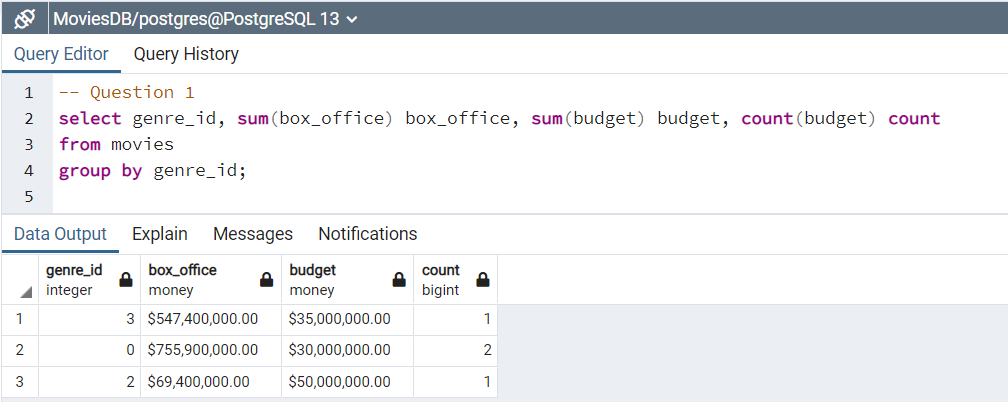
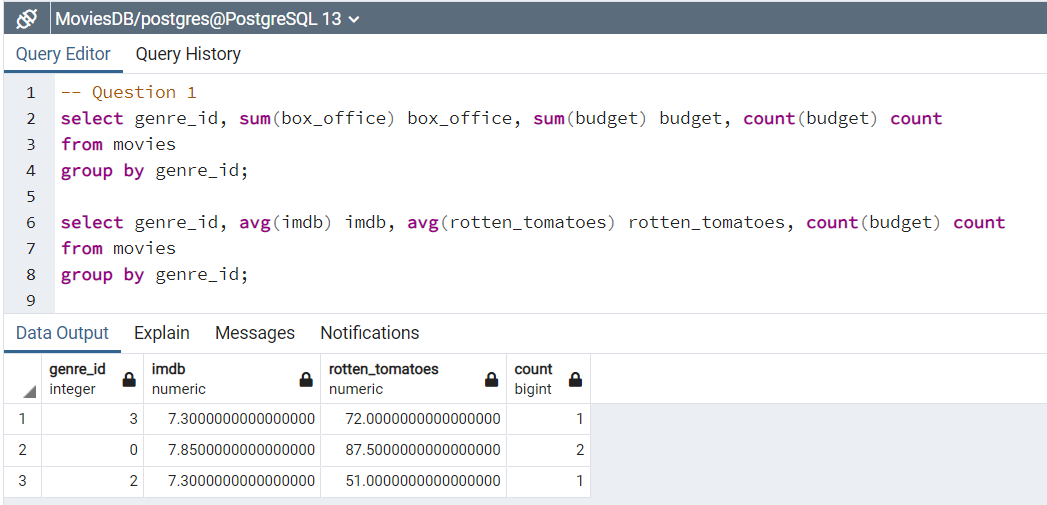
# سوال اول

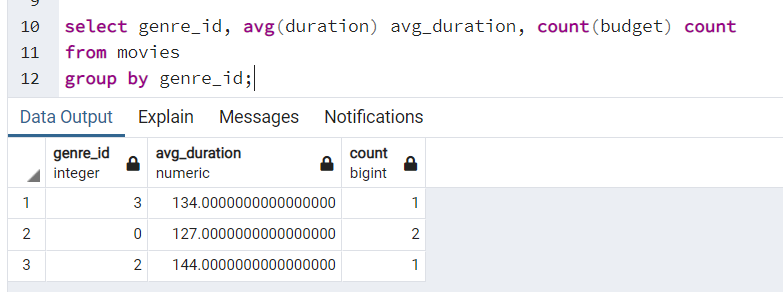
به وسیله این دستور میتوان فیلم های موجود را بر اساس ژانر، دسته بندی کرد به طوری که مجموع box\_office و budget کلی این فیلم ها را خروجی گرفت. همانطور که مشاهده میکنید برای ژانر دوم (romance) دو فیلم وجود دارد:



در دستور بعدی میانگین امتیاز imdb و rotten\_tomatoes هر ژانر آورده شده که مانند دستور قبل بر اساس شماره ژانر گروه بندی شده اند:

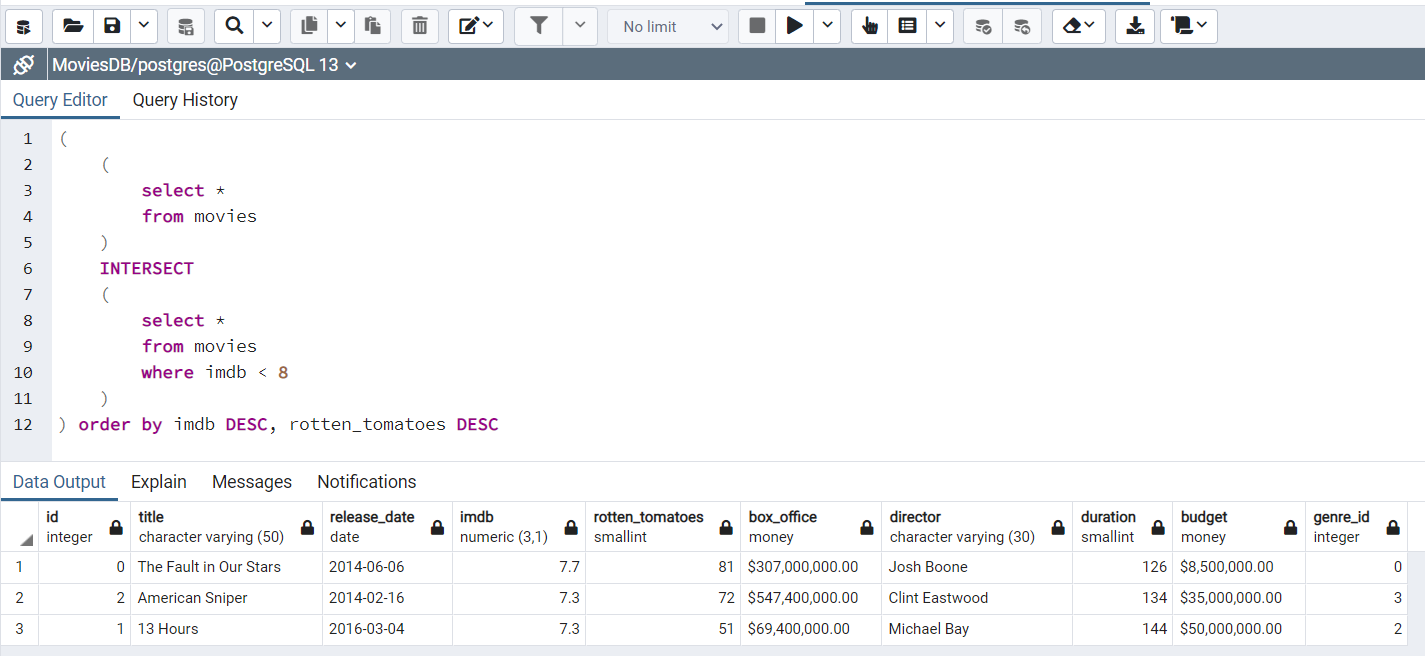


در آخرین دستور میانگین مدت زمان هر ژانر خروجی داده شده. به طوری که ژانر شماره 2 که action است بیشتر مدت زمان فیلم را دارد:



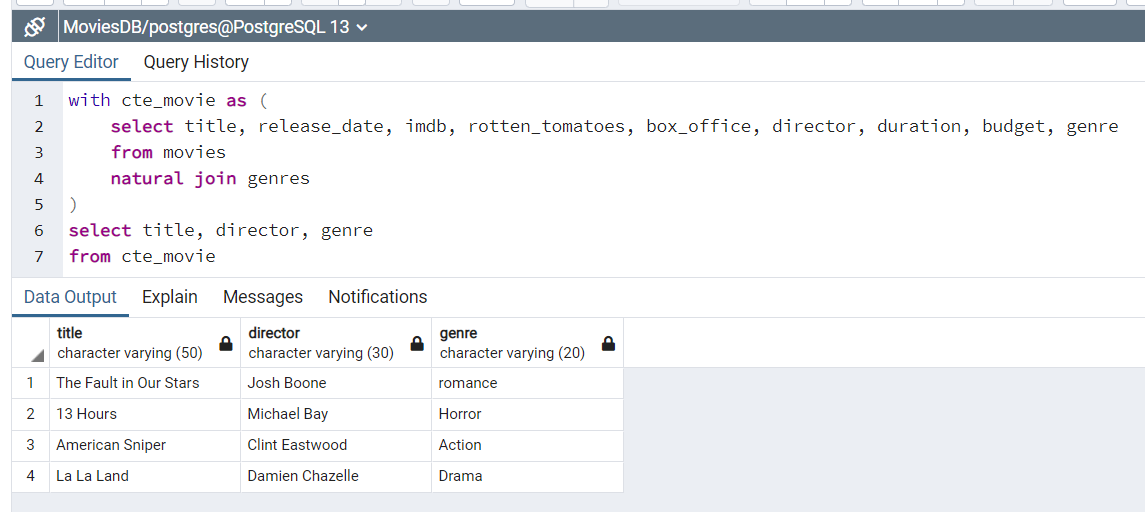
# سوال دوم

در این سوال از دستور intersect برای اشتراک گیری بین دو select استفاده شد و خروجی به ترتیب نزولی دو ستون عددی نمایش داده شده است:



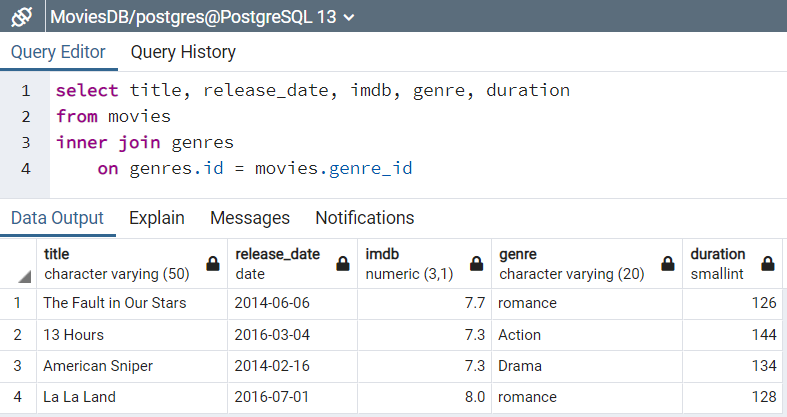
# سوال سوم

ابتدا از جدول genres ژانر های مربوطه را با جدول movies یکی کرده و در دستور دوم 3 ستون خاص را میخوانیم:

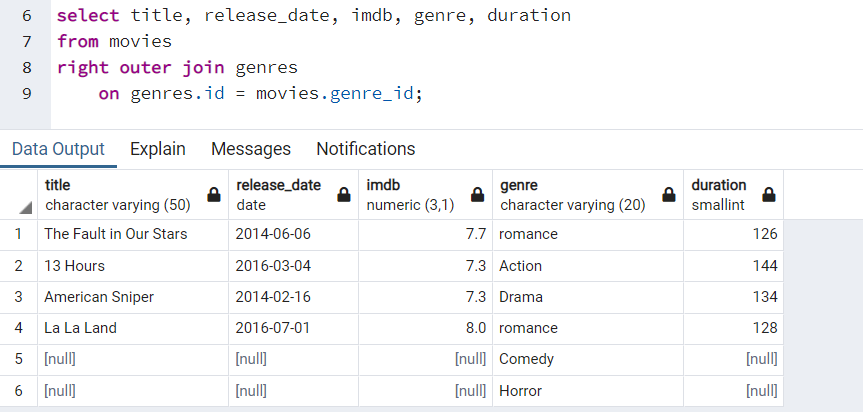


# سوال چهارم

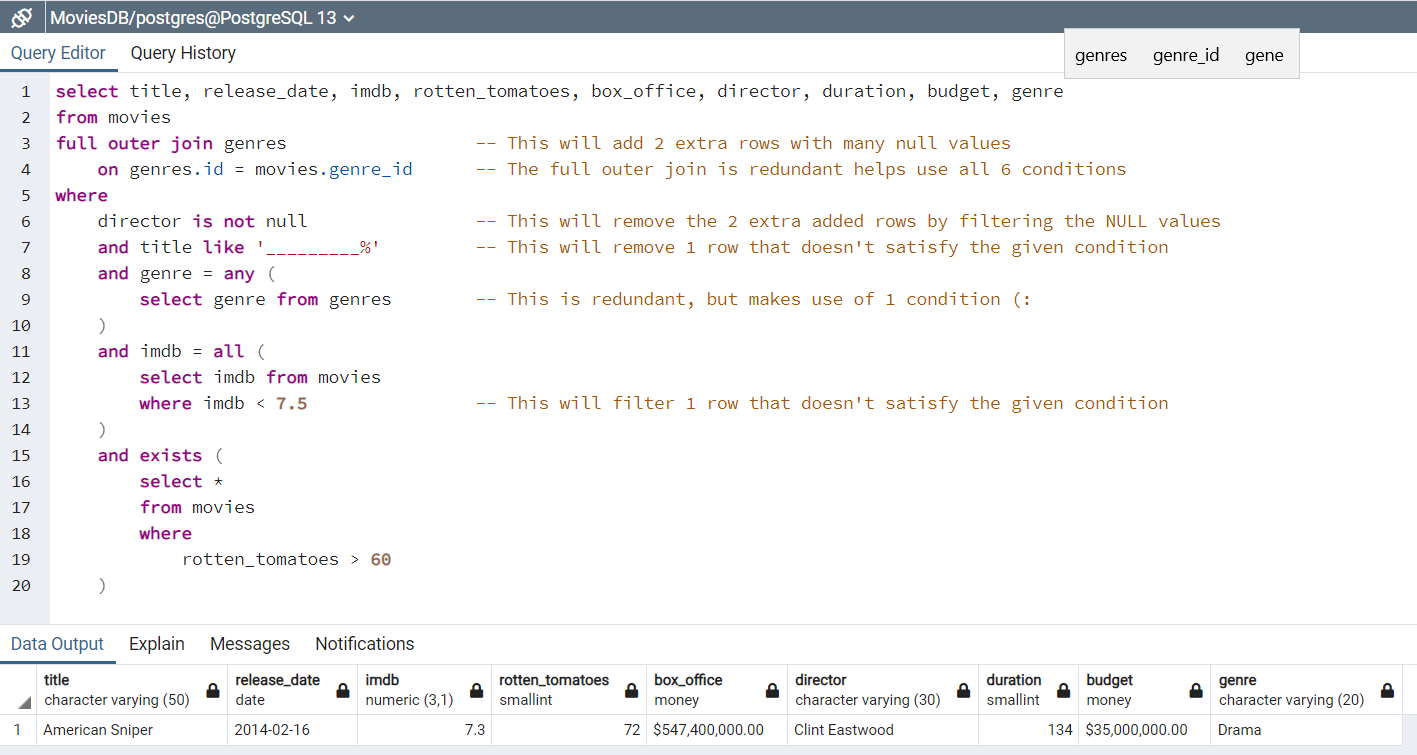
در این دستور با استفاده از inner join مشخصات تمام فیلم هایی که ژانر دارند و ژانر آنها در جدول genres وجود دارد چاپ میشود:



در این دستور با استفاده از right outer join تمام ژانر ها - جدا از اینکه آیا فیلمی داریم که این ژانر را داشته باشند یا خیر – چاپ شده اند. برای دو ژانر comedy و horror چون فیلمی نداریم پس ستون های نظیر مقدار null دارند:



# سوال پنجم



برای این سوال:

* از دستور full outer join استفاده شد تا دو ردیف null برگردانده شود که در دستور بعدی استفاده خواهد شد.
* از دستور is not null برای حذف کردن دو ردیف اضافه شده توسط دستور قبل استفاده شد.
* از دستور like '\_\_\_\_\_\_\_\_\_%' استفاده شد تا یک ردیف را حذف کنیم که در شرط صدق نمیکرد.
* برای استفاده از دستور any چک کردیم تا تمام ژانر های فیلم ها در جدول genres موجود باشند.
* برای دستور all تمام فیلم هایی را برگرداندیم که imdb کمتر از 7.5 داشته باشند (یک ردیف حذف شد)
* برای دستور exists تمام فیلم هایی را برگرداندیم که rotten tomatoes بیشتر از 60 دارند.
* برای مقایسه ها از دستور > استفاده شد.