Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Кафедра інформатики

Звіт з лабораторної роботи №6

По предмету: «Організація баз даних та знань»

Варіант 21

Виконав:

студент группи ІТІНФ-20-1

Самченко С. О.

Харків 2021

Скрипт настройки базы данных:

use Movies1G\_Stud

alter table dbo.genre alter column genre varchar(20) not null

alter table dbo.genre add constraint PK\_Genre primary key(genre)

alter table dbo.movie\_genre alter column movieid int not null

alter table dbo.movie\_genre alter column genre varchar(20) not null

alter table dbo.movie\_genre add constraint FK\_moviegenre foreign key(genre) references dbo.genre(genre)

alter table dbo.movie\_genre add constraint PK\_moviegenre primary key(movieid,genre)

alter table dbo.movie\_genre add constraint PK\_moviegenre2 foreign key(movieid) references dbo.movie(movieid)

delete from dbo.movie\_genre where genre not in (select genre from dbo.genre)

select \*

from dbo.movie\_genre

where genre not in (Select genre from dbo.genre)

select movieid,genre, count(\*)

from dbo.movie\_genre

group by genre,movieId

having count(\*) > 1

alter table dbo.movie alter column movieid int not null

alter table dbo.movie alter column title varchar(150) not null

alter table dbo.movie alter column movie\_year int null

alter table dbo.movie alter column rating\_mean real not null

alter table dbo.movie add constraint PK\_movie primary key(movieid)

alter table dbo.rating alter column movieid int not null

alter table dbo.rating alter column userid int not null

alter table dbo.rating alter column rating real not null

alter table dbo.rating alter column rating\_date datetime not null

select \*

from dbo.rating

where movieId not in (select movieId from dbo.movie)

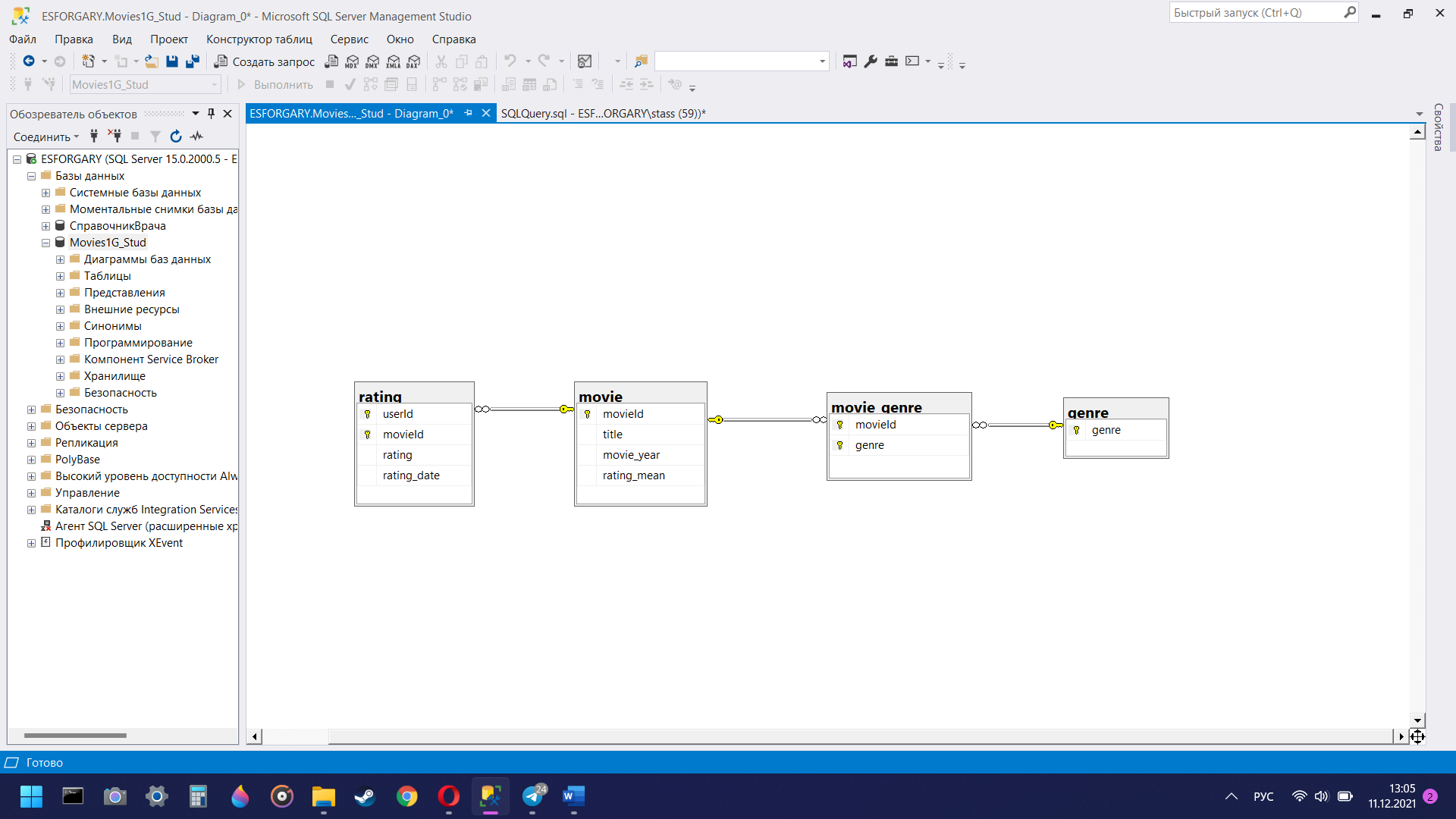
delete from dbo.rating where movieId not in (select movieId from dbo.movie)

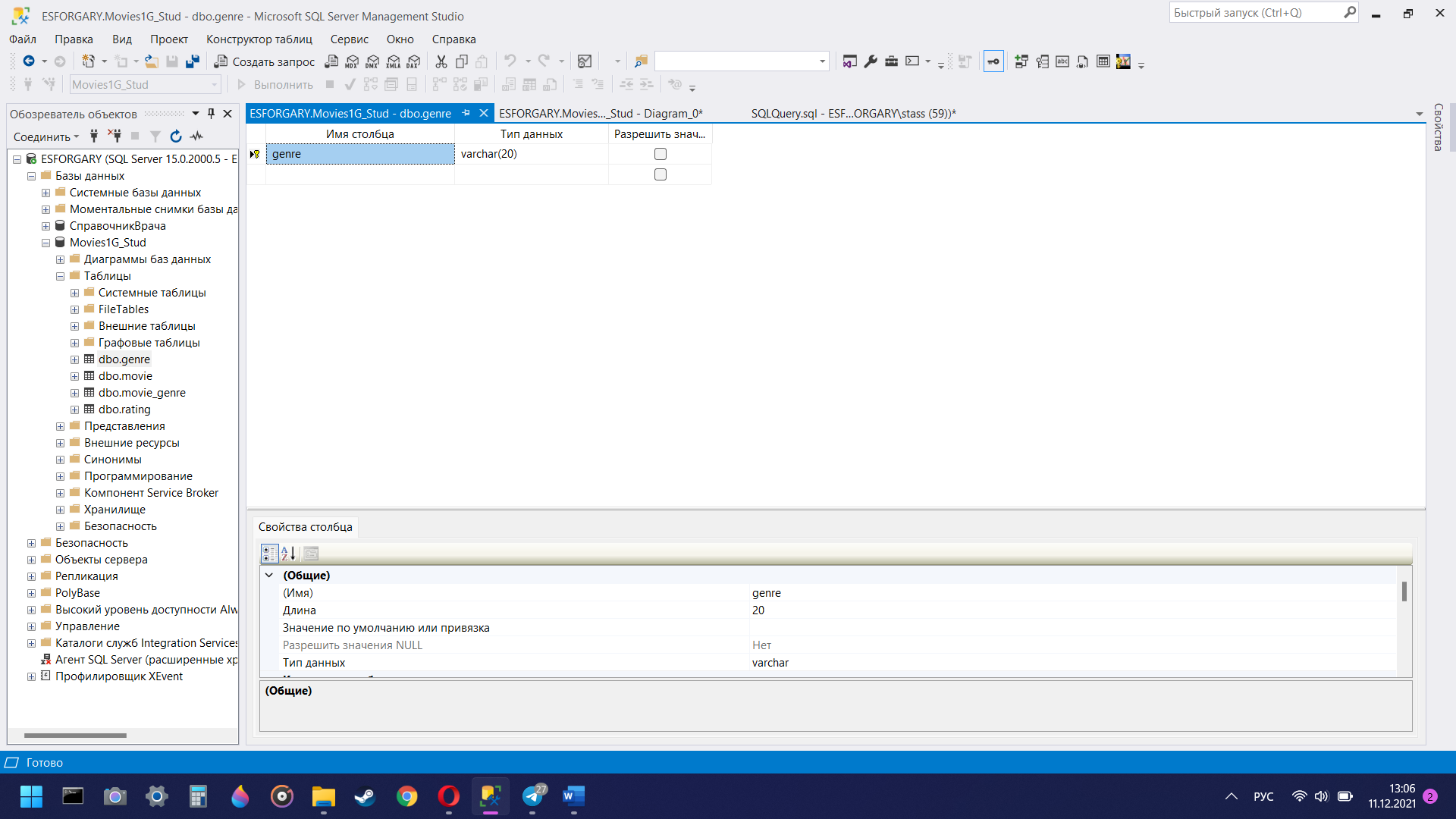
alter table dbo.rating add constraint PK\_rating primary key(userid,movieid)

alter table dbo.rating add constraint fK\_rating foreign key(movieid) references dbo.movie(movieid)

set statistics io on

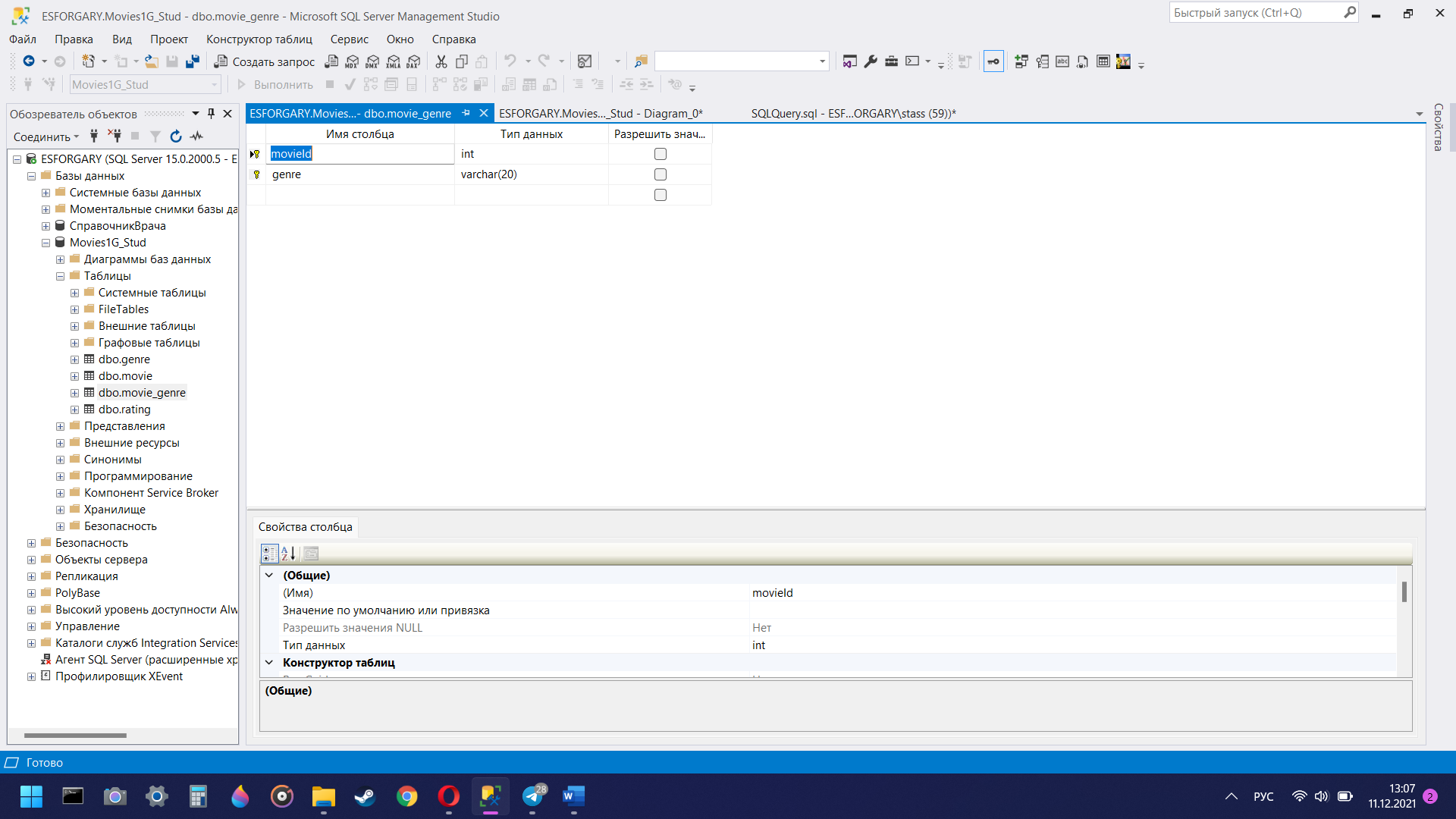
set statistics time on





Изображение выглядит как текст, снимок экрана, компьютер

Автоматически созданное описание



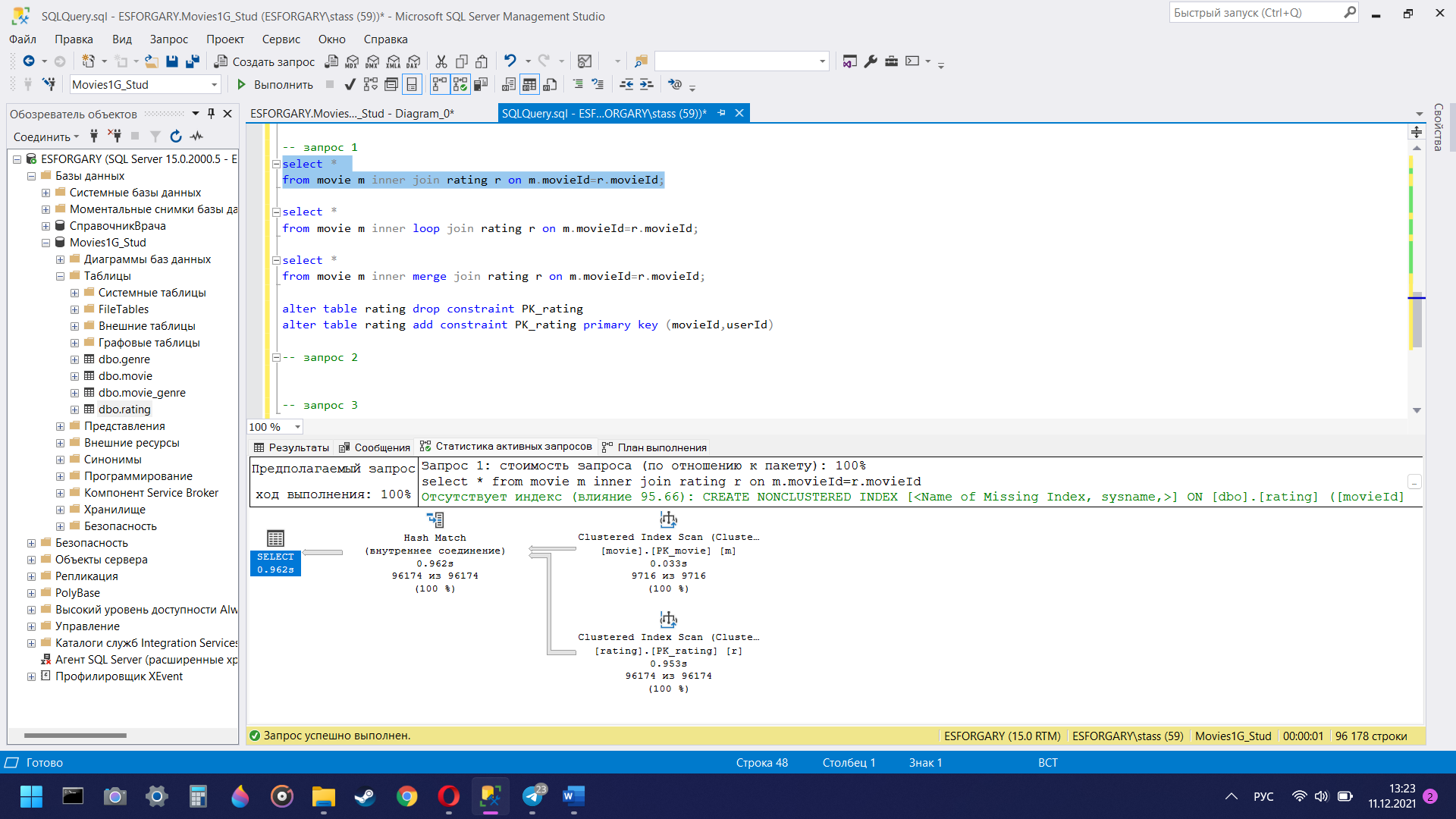
Изображение выглядит как текст

Автоматически созданное описание

**Задание 1.**

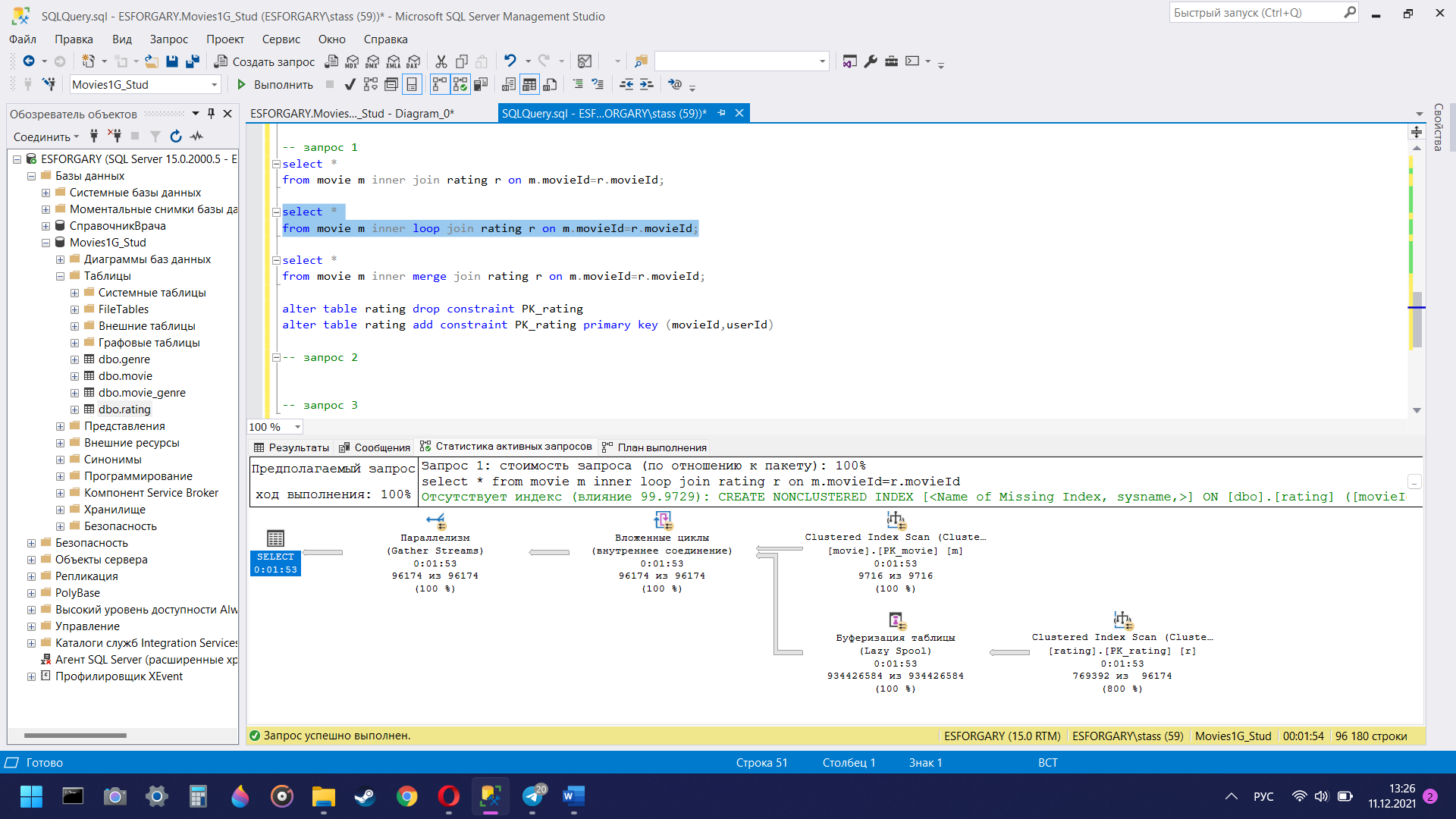
select \*

from movie m inner join rating r on m.movieId=r.movieId;



select \*

from movie m inner loop join rating r on m.movieId=r.movieId;



select \*

from movie m inner merge join rating r on m.movieId=r.movieId;

Изображение выглядит как текст

Автоматически созданное описание

Результаты работы после изменения порядка следования полей в составном первичном ключе:

alter table rating drop constraint PK\_rating

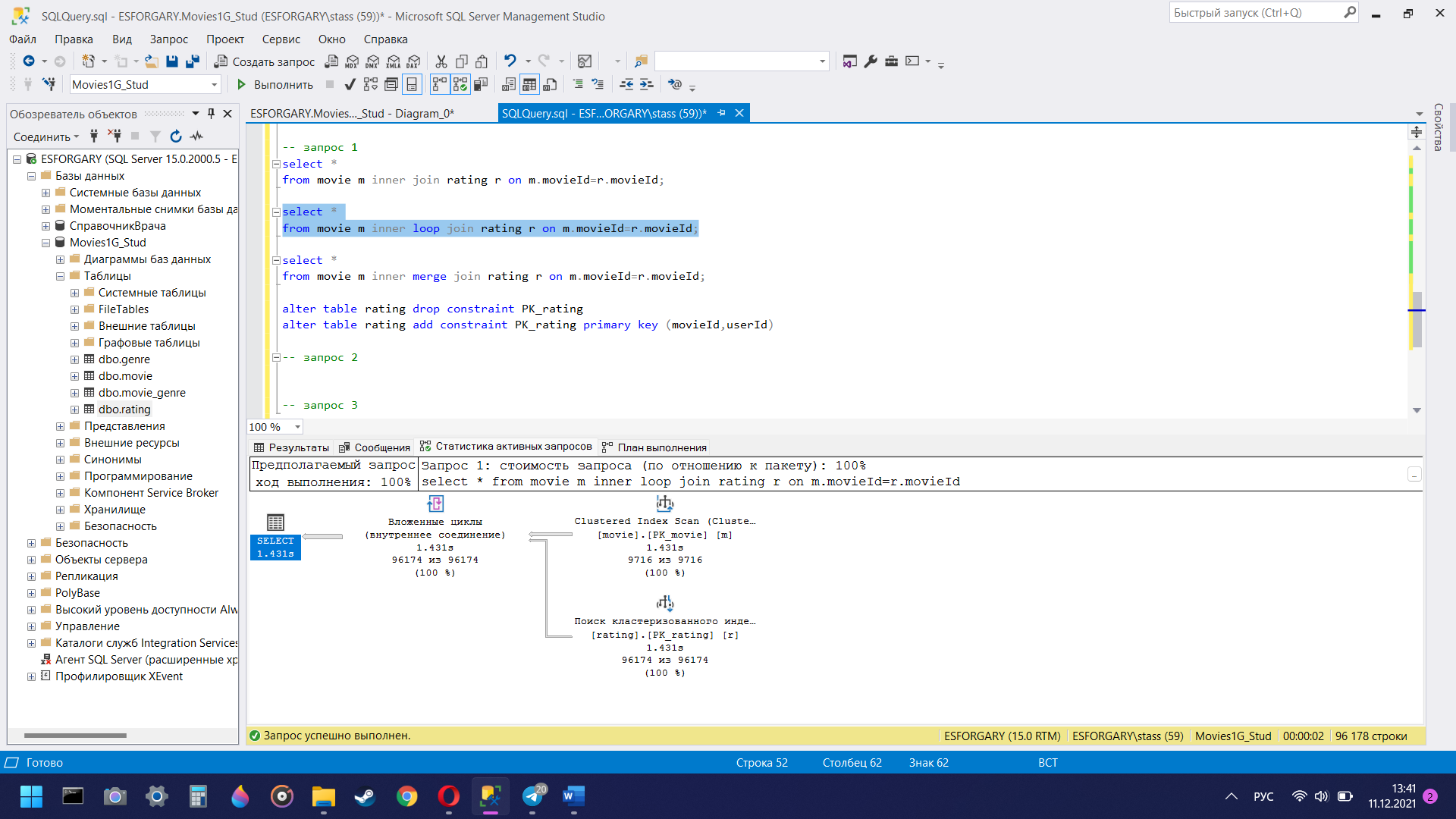
alter table rating add constraint PK\_rating primary key (movieId,userId)

inner join)

Изображение выглядит как текст

Автоматически созданное описание

inner loop join)



inner merge join)

Изображение выглядит как текст

Автоматически созданное описание

**Задание 2.**

2.1)

with TABL\_itsa(genre,num\_movies,num\_marks,avg\_rating) as (

select genre,(select count(m1.movieId) from movie\_genre mg1 inner join movie m1 on mg1.movieId = m1.movieId

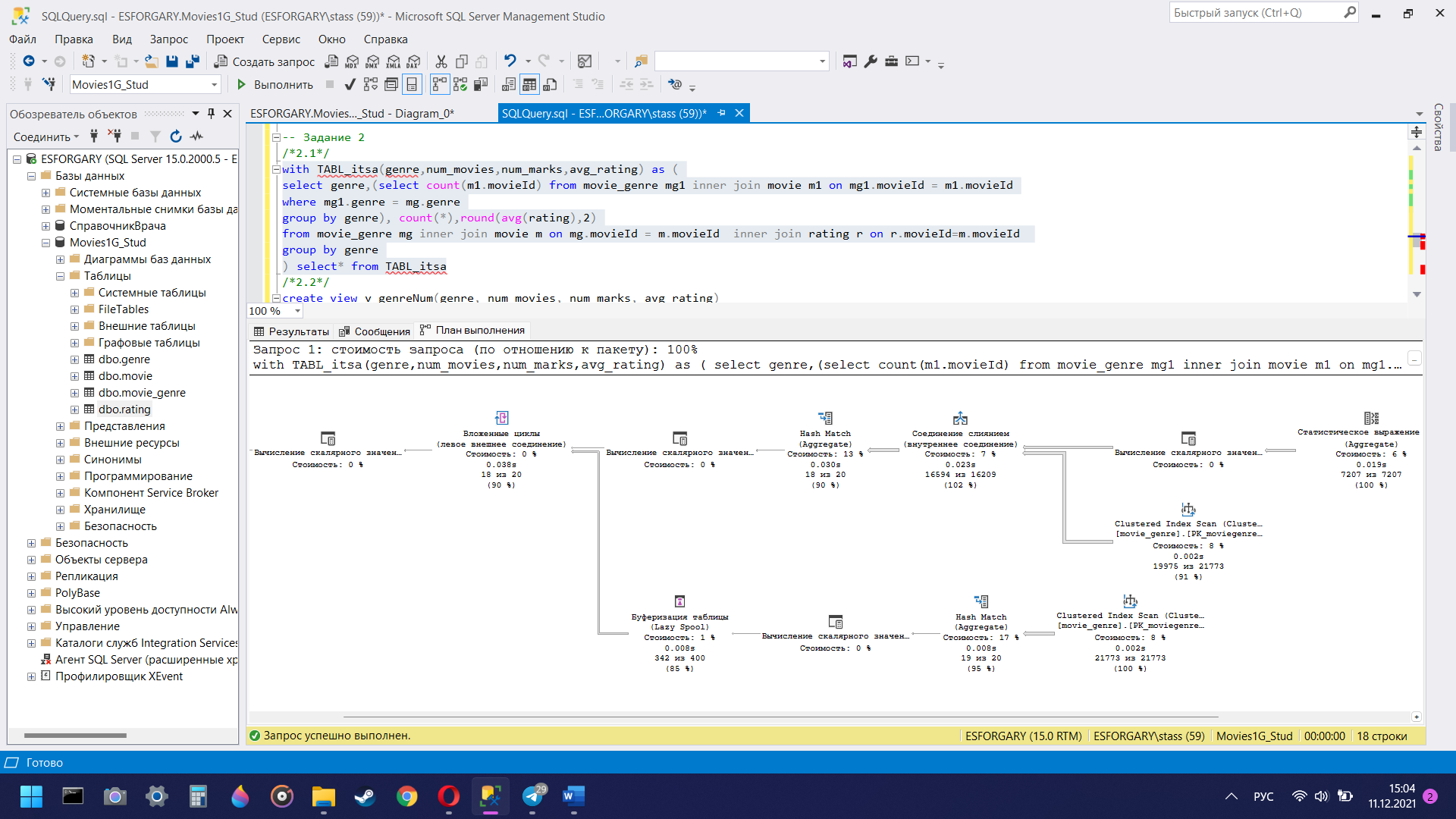
where mg1.genre = mg.genre

group by genre), count(\*),round(avg(rating),2)

from movie\_genre mg inner join movie m on mg.movieId = m.movieId inner join rating r on r.movieId=m.movieId

group by genre

) select\* from TABL\_itsa



2.2)

create view v\_genreNum(genre, num\_movies, num\_marks, avg\_rating) as (

select genre, (select count(m1.movieId) from movie\_genre mg1 inner join movie m1 on mg1.movieId = m1.movieId

where mg1.genre = mg.genre

group by genre), count(\*),round(avg(rating),2)

from movie\_genre mg inner join movie m on mg.movieId = m.movieId inner join rating r on r.movieId=m.movieId

group by genre

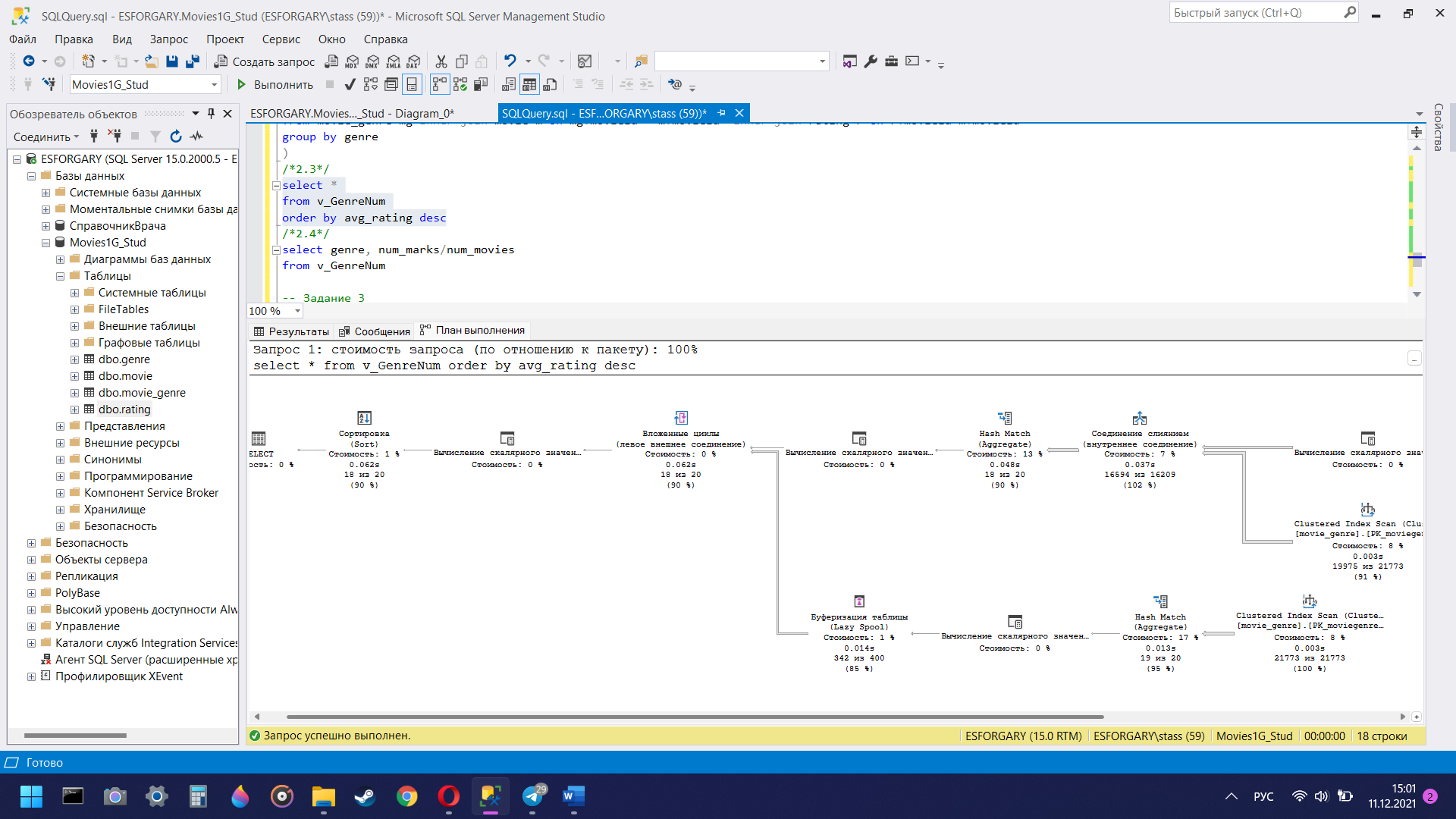
)

2.3)

select \*

from v\_GenreNum

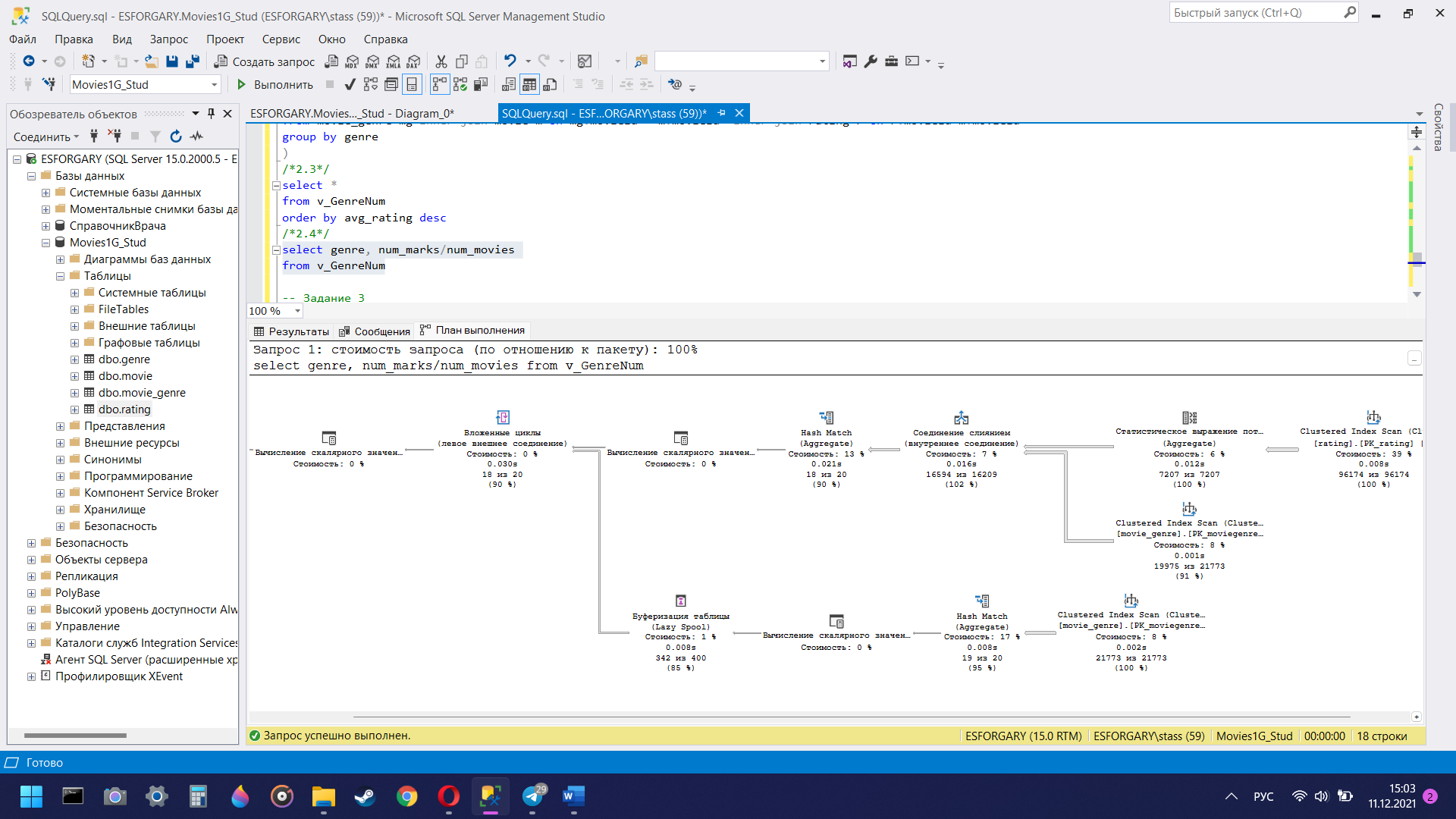
order by avg\_rating desc



2.4)

select genre, num\_marks/num\_movies

from v\_GenreNum



**Задание 3.**

with Movie\_3(movieID,title,rating\_old,rating\_new,dif)as

(select m.movieId, title,rating\_mean,rating,round((abs(rating\_mean-rating)),2)

from movie m inner join rating r on r.movieId = m.movieId)

select \* from Movie\_3

