Санкт-Петербургский политехнический университет Петра Великого

Институт информационных технологий и управления

Кафедра компьютерных систем и программных технологий

**Отчёт лабораторной работе №3**

**Дисциплина**: Базы данных

**Тема**: SQL-DML

Выполнил студент гр. 43501/1 С.И.Бочкова

(подпись)

Руководитель А.В. Мяснов

(подпись)

“ ” 2015 г.

Санкт - Петербург

2015

1. **Цель работы**

Выполнить различные типы запросов к БД. Изучить команды добавления, редактирования и удаления записей.

1. **Проведение работы**

Выборка всех данных из таблицы выполняется командой

Select \* from <имя таблицы>

Вывод первых 50 человек, родившихся в США и Герамнии:

select first 50 MAN.name,SURNAME from MAN where COUNTRY in (2,3);

Вывод фильмов, продолжительностью от 45 до 90 минут:

select FILM.name,duration from FILM where duration between 45 and 90;

Вывод фильмов, бюджетом меньше 3000000:

select FILM.name,budget from FILM where budget<3000000;

Вывод результата последнего запроса:

select count(FILM.name)from FILM where budget<3000000;

Сортировка по нескольким полям:

select \* from COMPANY order by COMPANY.name asc, year\_of\_foundation desc;

Вывод нескольких совокупных характеристик таблиц:

select sum(money\_of\_prokat),avg(money\_of\_prokat),max(money\_of\_prokat) from COMPANY;

Выборка данных из связанных таблиц

select first 5 MAN.NAME,MAN.SURNAME from MAN,PART,COMPANY

where MAN.MAN\_ID=PART.MAN\_ID and ROLE \_ID=3;

## Запрос, рассчитывающий совокупную характеристику с использованием группировки, наложите ограничение на результат группировки:

select FILM.name,sum(budget) from FILM group by FILM.name having sum(bidget)>1000000;

## Пример использования вложенного запроса:

select MAN.NAME,MAN.SURNAME from MAN,PART where MAN.MAN\_ID=PART.MAN\_ID and

ROLE\_ID=3 and MAN.MAN\_ID in (select first 10 PART.MAN\_ID from

PART where ROLE\_ID=3 group by PART.MAN\_ID order by sum(GANORAR) asc);

Добавление в таблицу выполняется командой insert:

INSERT INTO soundtrack(sound\_id,name,author,film,duration) VALUES(4,’Crazy’,9,5,’03:45’);

С помощью оператора UPDATE изменяем значения нескольких полей у всех записей, отвечающих заданному условию:

select SOUND\_ID,SOUNDTRACK.name,film from SOUNDTRACK;

update SOUNDTRACK set FILM= 2 where FILM =3;

commit;

select SOUND\_ID,SOUNDTRACK.name,film from SOUNDTRACK;

С помощью оператора DELETE удаляем запись, имеющую максимальное (минимальное) значение некоторой совокупной характеристики:

delete from FILM where BUDGET =(select BUDGET from FILM group by BUDGET order by sum(FILM) desc);

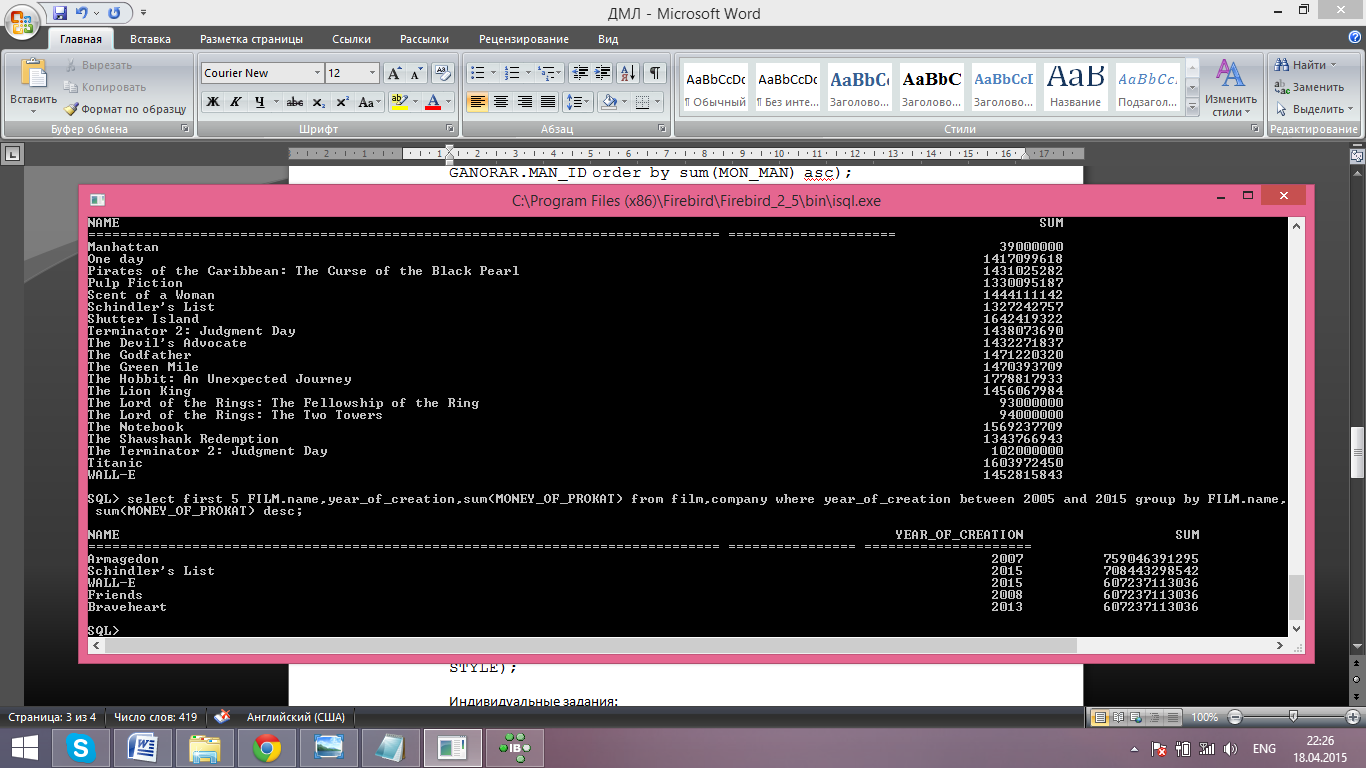
Удалить не задействованные жанры:

delete from GENRE where ID not in (select GENRE\_ID from STYLE);

Индивидуальные задания:

1. Вывести пять фильмов, которые собрали максимальное количество денег в прокате за последние 10 лет.

select first 5 FILM.name,year\_of\_creation,sum(MONEY\_OF\_PROKAT) from film,company where year\_of\_creation between 2005 and 2015 group by FILM.name,FILM.year\_of\_creation order by sum(MONEY\_OF\_PROKAT) desc;

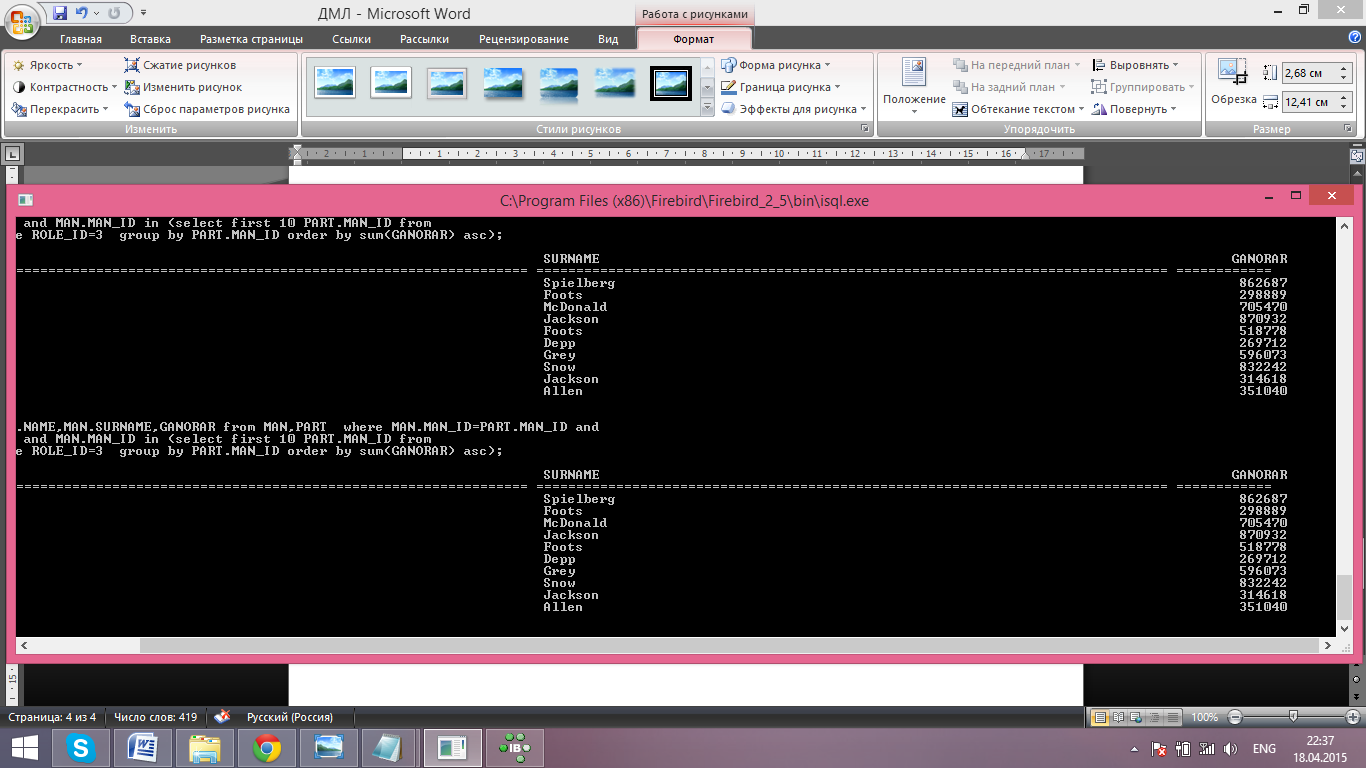
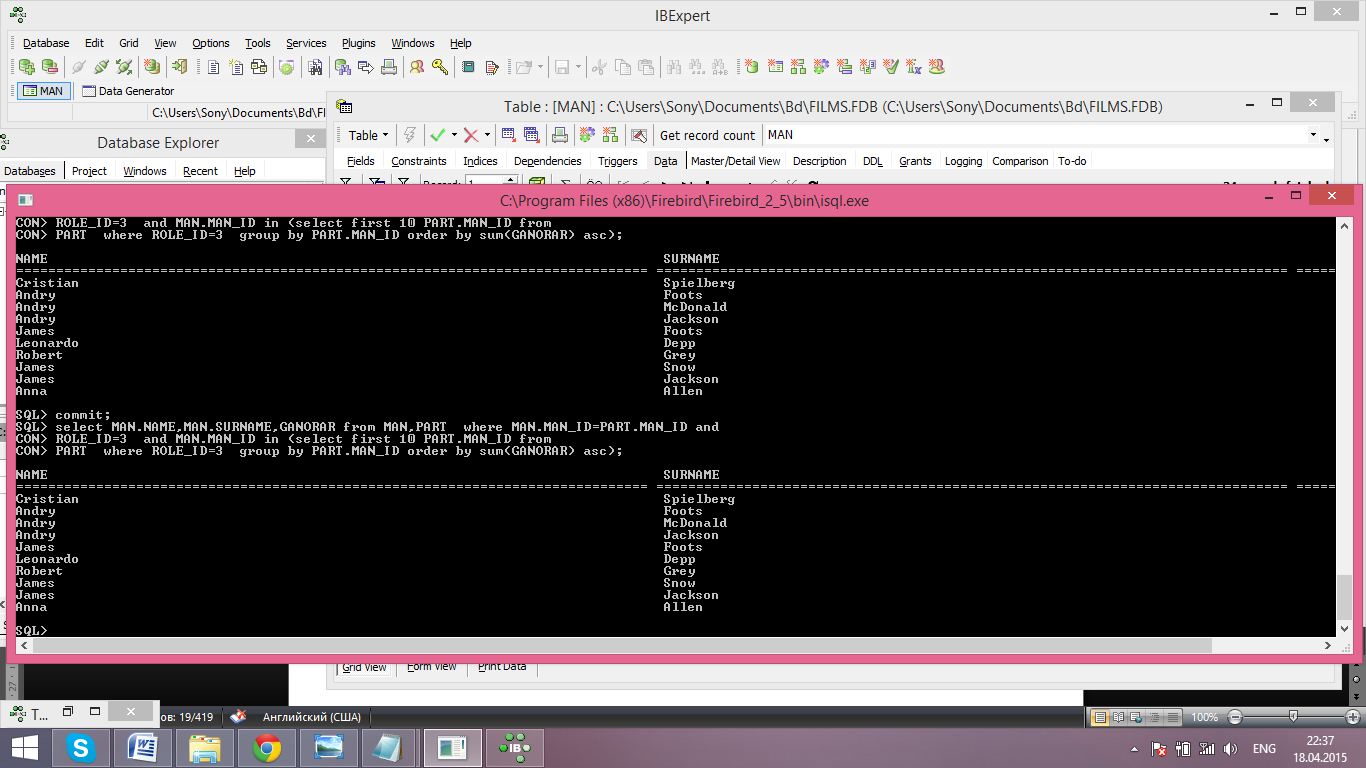


2.Вывести 10 актеров, у которых минимальные суммарные гонорары.

select MAN.NAME,MAN.SURNAME from MAN,PART where MAN.MAN\_ID=PART.MAN\_ID and

ROLE\_ID=3 and MAN.MAN\_ID in (select first 10 PART.MAN\_ID from

PART where ROLE\_ID=3 group by PART.MAN\_ID order by sum(GANORAR) asc);



3.Удалить незадействованные жанры.

delete from GENRE where ID not in (select GENRE\_ID from STYLE);

1. **Вывод**

В данной работе были изучены инструменты по созданию запросов к БД, удалению и модификации записей. Полученные навыки работы с SQL-запросами будут использованы в дальнейшей работе с базой данных.