**Санкт-Петербургский Политехнический Университет Петра Великого**

**Институт Информационных Технологий и Управления**

**Кафедра Компьютерных Систем и Программных Технологий**

**ОТЧЕТ**

**по лабораторной работе №4**

# « Язык SQL-DML»

**Студент гр.43501/1: Данг Хань**

**Преподаватель: Мяснов А.В.**

**Санкт-Петербург**

**2015**

1. **Цель работы:**

Ознакомление с основами проектирования схемы БД и способами нормализации отношений.

1. **Программа работы:**

**а)** Сделать выборку всех данных из каждой таблицы.

**б)** Сделать выборку данных из одной таблицы при нескольких условиях, с использованием логических операций, LIKE, BETWEEN, IN (не менее 3-х разных примеров).

**в)** Создать в запросе вычисляемое поле.

**г)** Сделать выборку всех данных с сортировкой по нескольким полям

**д)** Создать запрос, вычисляющий несколько совокупных характеристик таблиц.

**е)** Сделать выборку данных из связанных таблиц (не менее двух примеров).

**ё)** Создать запрос, рассчитывающий совокупную характеристику с использованием группировки, наложите ограничение на результат группировки.

**ж)** Придумать и реализовать пример использования вложенного запроса

**з)** С помощью оператора INSERT добавить в каждую таблицу по одной записи.

**и)** С помощью оператора UPDATE измените значения нескольких полей у всех записей, отвечающих заданному условию.

**к)** С помощью оператора DELETE удалить запись, имеющую максимальное (минимальное) значение некоторой совокупной характеристики.

**л)** С помощью оператора DELETE удалить записи в главной таблице, на которые не ссылается подчиненная таблица (используя вложенный запрос).

**м)** Выполнить серию индивидуальных заданий.

1. **Выполнение работы:**

**а)** **Произведем выборку всех данных из каждой таблицы с помощью опертора выбора SELECT**:

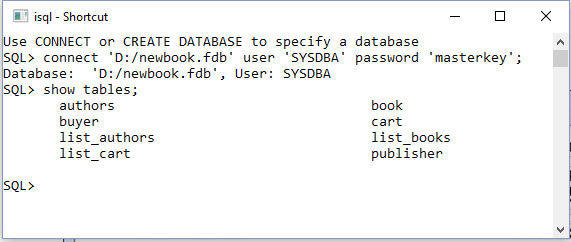


Рис.3.1. Список всех имющихся таблиц.

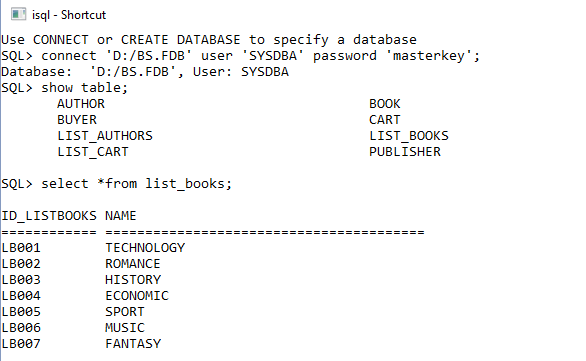


Рис.3.1. Содержимое таблицы list\_books.

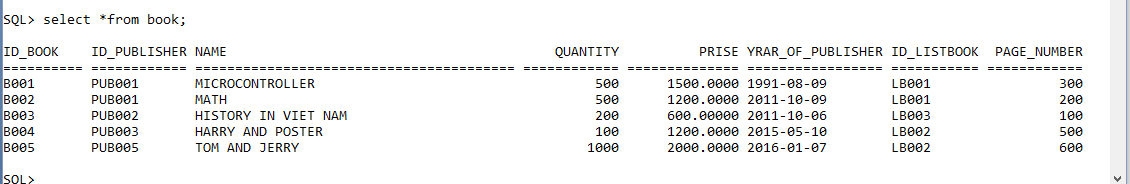


Рис.3.2. Содержимое таблицы book.

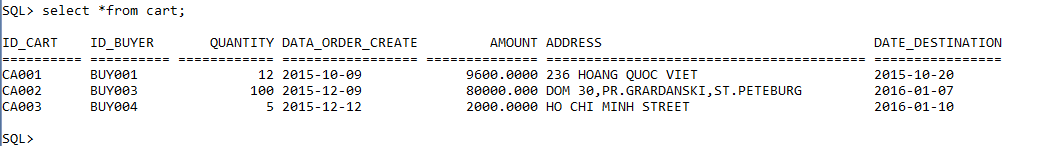


Рис.3.3. Содержимое таблицы cart.

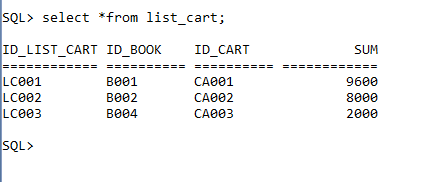


Рис.3.4. Содержимое таблицы list\_cart

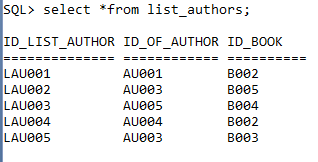


Рис.3.5. Содержимое таблицы List\_authors.

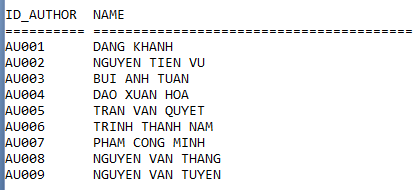
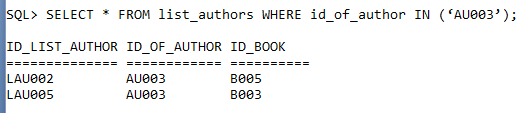


Рис.3.5. Содержимое таблицы author.

**б) Сделать выборку данных из одной таблицы при нескольких условиях, с использованием логических операций LIKE, BETWEEN, IN (не менее 3-х разных примеров):**

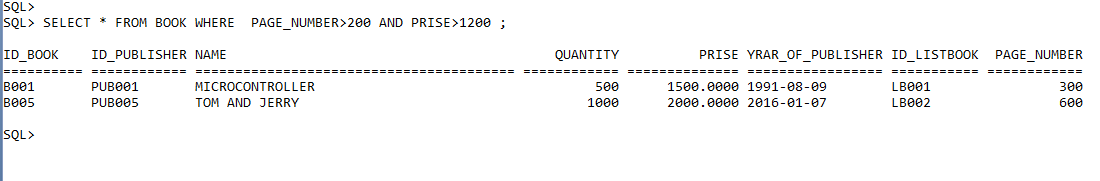
**Пример 1.** Выведем все записи терапевтов:

SELECT \* FROM list\_authors WHERE id\_of\_author IN (‘AU003’);



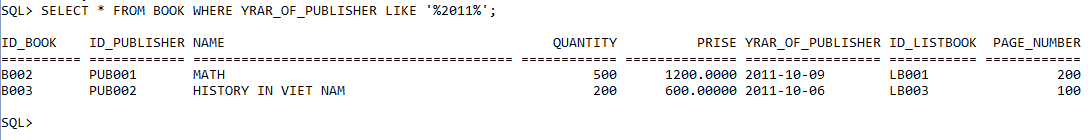
**Пример 2.** Выведем все записи хирургов, заключение о присвоении группы которых, все кроме группы «А»:

SELECT \* FROM BOOK WHERE PAGE\_NUMBER>200 AND PRISE>1200 ;



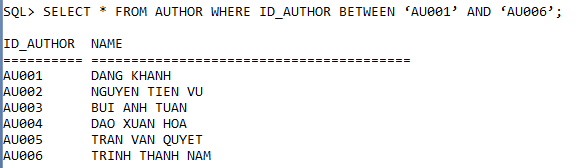
**Пример 3.** Выведем заметки всех КНИГ, в которых содержаться число 2011:

SELECT \* FROM BOOK WHERE YRAR\_OF\_PUBLISHER LIKE '%2011%';



**Пример 4.** Выведем из 6-ти первых в списке всех АВТОРОВ, только записи окулистов:

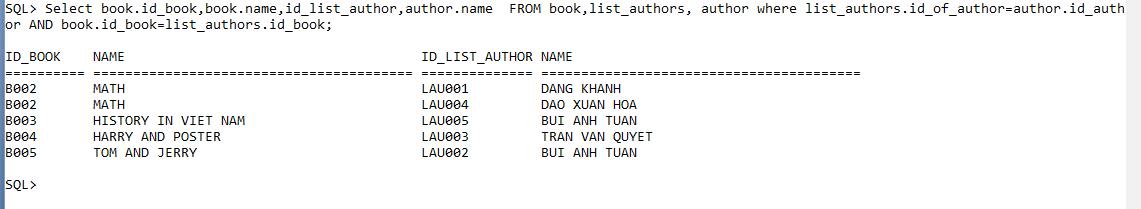
SELECT \* FROM AUTHOR WHERE ID\_AUTHOR BETWEEN ‘AU001’ AND ‘AU006’;



**в)** **Создать в запросе вычисляемое поле:**

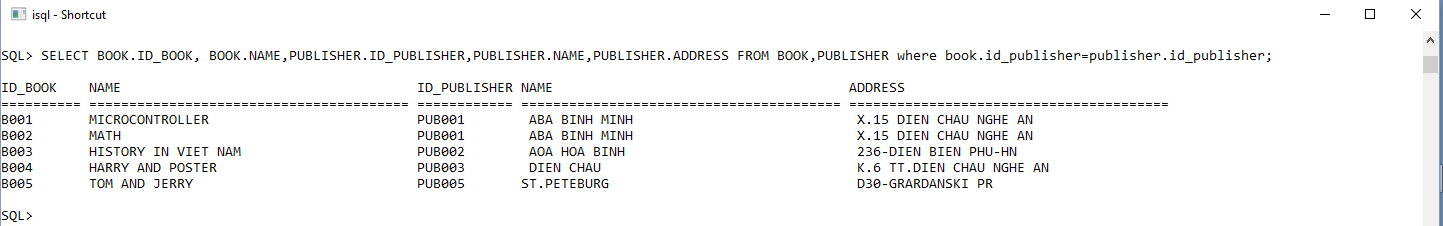
*выборка идентификатора и названия книги с именем её автора*

**Select book.id\_book,book.name,id\_list\_author, author.name FROM book, list\_authors,author where list\_authors.id\_of\_author=author.id\_author AND book.id\_book=list\_authors.*id\_book;***



*название книги и названия издательста, где она выпущена*

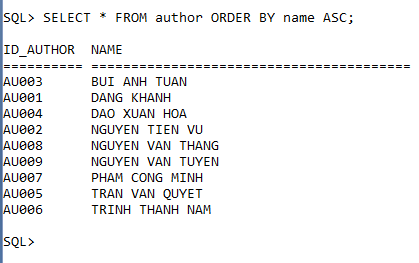
-SQL>SELECT BOOK.ID\_BOOK, BOOK.NAME,PUBLISHER.ID\_PUBLISHER,PUBLISHER.NAME,PUBLISHER.ADDRESS FROM BOOK,PUBLISHER where book.id\_publisher=publisher.id\_publisher;



**г)** **Сделать выборку всех данных с сортировкой по нескольким полям:**

Отсортируем по имени фамили и отчеству список всех авторов числящихся в БД.

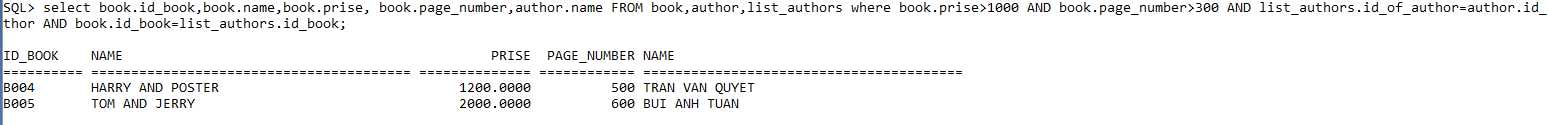
SELECT \* FROM author ORDER BY name ASC;



**д)** **Создать запрос, вычисляющий несколько совокупных характеристик таблиц:**

выьирать книги c имени и их авторов, которые страница >300 и цена >1000

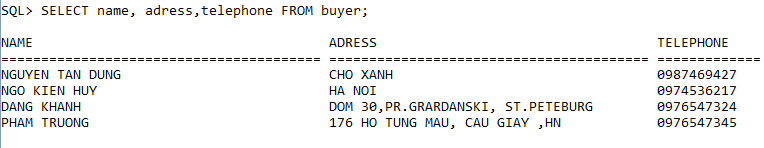
select book.id\_book,book.name,book.prise, book.page\_number,author.name FROM book,author where book.prise>1000 AND book.page\_number>300 AND list\_authors.id\_of\_author=author.id\_author AND book.id\_book=list\_authors.id\_book;



**е)** **Сделать выборку данных из связанных таблиц (не менее двух примеров):**

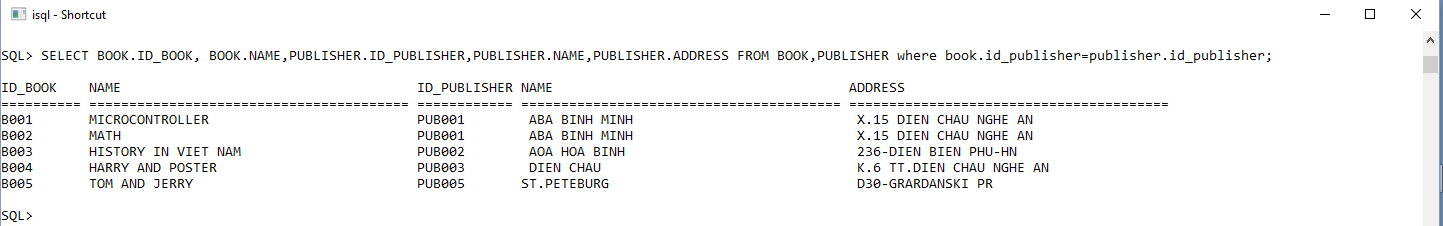
**Пример 1.** Выведем ФИО и телефонные номера:

SELECT name, adress,telephone \*FROM buyer;



**Пример 2.** *название книги и названия издательста, где она выпущена*

-SQL>SELECT BOOK.ID\_BOOK, BOOK.NAME,PUBLISHER.ID\_PUBLISHER,PUBLISHER.NAME,PUBLISHER.ADDRESS FROM BOOK,PUBLISHER where book.id\_publisher=publisher.id\_publisher;

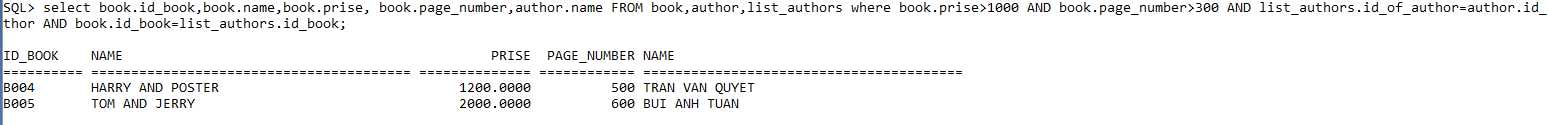


**ё)** **Создать запрос, рассчитывающий совокупную характеристику с использованием группировки, наложите ограничение на результат группировки:**

Для использования группировки будем использовать GROUP BY, а для наложения ограничения на результат вместо WHERE будем использовать оператор HAVING. Допустим, нам требуется узнать сколько записей имеется у всех врачей в БД военкомата:

выьирать книги c имени и их авторов, которые страница >300 и цена >1000

select book.id\_book,book.name,book.prise, book.page\_number,author.name FROM book,author where book.prise>1000 AND book.page\_number>300 AND list\_authors.id\_of\_author=author.id\_author AND book.id\_book=list\_authors.id\_book;

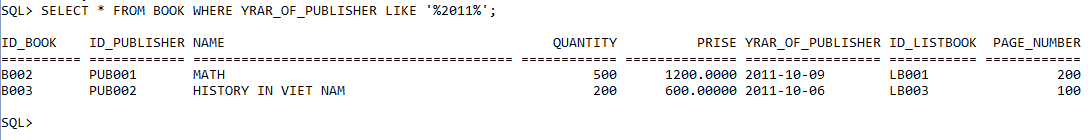


**ж)** **Придумать и реализовать пример использования вложенного запроса:**

Для реализации вложенного запроса используем логическую операцию IN. К сожалению, FireBird не понимает русских символов в логической операции LIKE, пример будем реализовывасть без использования запросов с русскими символами.

Выведем заметки всех КНИГ изатели с 2011 года:

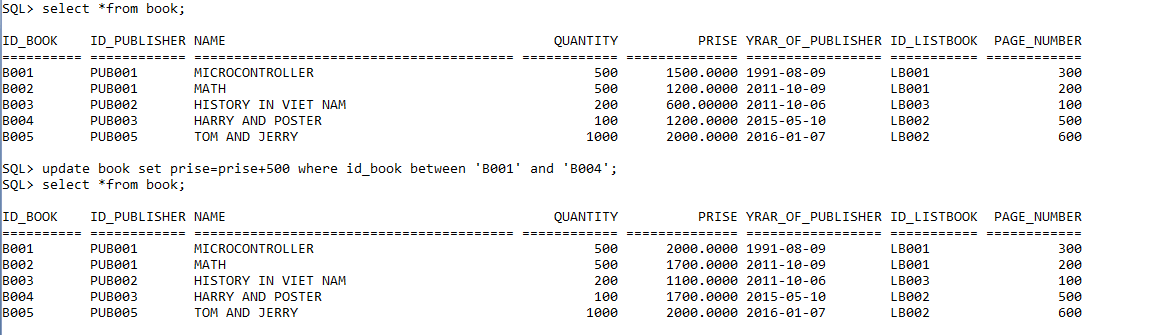
SELECT \* FROM BOOK WHERE YRAR\_OF\_PUBLISHER LIKE '%2011%';



**з) С помощью оператора INSERT добавим в каждую таблицу по одной записи:**

Данная операция многократно делается в скрипте при создании БД.

**и) С помощью оператора UPDATE изменим значения нескольких полей у всех записей, отвечающих заданному условию:**

****

В данном примере, с помощью оператора UPDATE мы повысили цену книги на 500 рублей:

UPDATE book SET prise = prise + 500 WHERE id\_book between ‘B001’ AND ‘B004’ ;

**к) С помощью оператора UPDATE изменим значения нескольких полей у всех записей, отвечающих заданному условию:**

Удалим должность, которая имеет самую наивысшую зарплату. Для этого сделаем вложенный запрос к условию удаления. Просортируем по убыванию и 1 запись вложенного запроса передадим оператору DELETE:

DELETE FROM BOOK WHERE prise = (select max(prise) from book);

**л) С помощью оператора DELETE удалим записи в главной таблице, на которые не ссылается подчиненная таблица (используя вложенный запрос):**

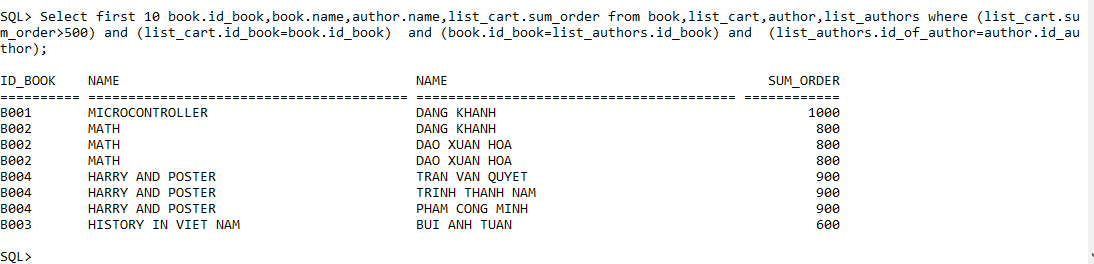
**select id\_book from books where id\_book not in (select id\_book from reviews group by id\_book)** – *удалениязаписейизтаблицыкниг, накоторыененаписанорецензий*

**2.Индивидуальное задание**

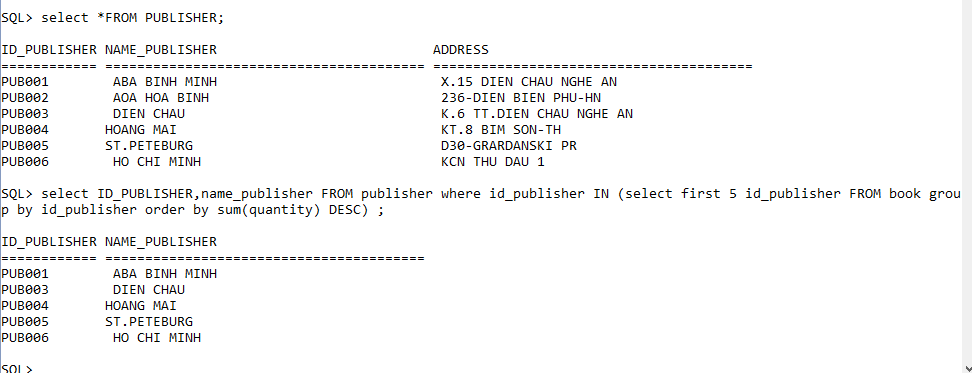
2.1. Вывести 10 авторов, книги которых лучше всего продавались за заданный  
 период.

Select first 10 book.id\_book,book.name,author.name,list\_cart.sum\_order from book,list\_cart,author,list\_authors where (list\_cart.sum\_order>500) and (list\_cart.id\_book=book.id\_book) and (book.id\_book=list\_authors.id\_book) and (list\_authors.id\_of\_author=author.id\_author);

Выбирал 10 книги и их авторов , которые количество заказ >500 (или продавались период >500)

  
  2.2. Вывести 5 издательств, книги которых были заказаны на максимальную  
 сумму.

Select id\_publisher, name\_publisher FROM publisher where id\_publisher IN (select first 5 id\_publisher FROM book group by id\_publisher ORDER BY SUM (quantity) DESC);



3.Вывод

В ходе работы были исследованы возможности языка DML. Он предоставляет гибкую структуру для осуществления выборок, соединения, модификации и удаления записей таблиц. Возможность производить выборку из таблицы, которая является результатом слияния других таблиц, сортировка и группировка по нескольким полям, позволяют в одном запросе реализовать сложную выборку из нескольких таблиц. Часто используемые запросы можно сохранить на сервере при помощи представлений. А операторы DELETEи UPDATEс условиями предоставляют мощный инструмент для удаления и модификации данных, неудовлетворяющих заданным показателям.

https://ssl.gstatic.com/ui/v1/icons/mail/images/cleardot.gif