

**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:

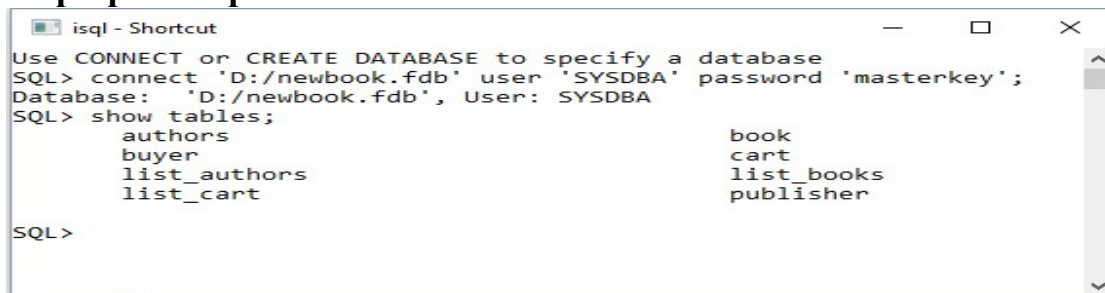
Ознакомление с основами проектирования схемы БД и способами нормализации отношений.

### 2. 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 изменить 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.

### 3. Выполнение работы:

- а) Произведем выборку всех данных из каждой таблицы с помощью оператора выбора SELECT:



```
isql - Shortcut
Use CONNECT or CREATE DATABASE to specify a database
SQL> connect 'D:/newbook.fdb' user 'SYSDBA' password 'masterkey';
Database: 'D:/newbook.fdb', User: SYSDBA
SQL> show tables;
      authors
      buyer
      list_authors
      list_cart
      book
      cart
      list_books
      publisher
SQL>
```

Рис.3.1. Список всех имеющихся таблиц.

```

isql - Shortcut
Use CONNECT or CREATE DATABASE to specify a database
SQL> connect 'D:/BS.FDB' user 'SYSDBA' password 'masterkey';
Database: 'D:/BS.FDB', User: SYSDBA
SQL> show table;
          AUTHOR          BOOK
          BUYER           CART
          LIST_AUTHORS    LIST_BOOKS
          LIST_CART       PUBLISHER

SQL> select *from list_books;

ID_LISTBOOKS NAME
=====
LB001        TECHNOLOGY
LB002        ROMANCE
LB003        HISTORY
LB004        ECONOMIC
LB005        SPORT
LB006        MUSIC
LB007        FANTASY

```

Рис.3.1. Содержимое таблицы list\_books.

```

SQL> select *from book;

ID_BOOK  ID_PUBLISHER NAME          QUANTITY  PRISE  YRAR_OF_PUBLISHER ID_LISTBOOK  PAGE_NUMBER
=====
B001     PUB001     MICROCONTROLLER      500      1500.0000 1991-08-09   LB001        300
B002     PUB001     MATH                  500      1200.0000 2011-10-09   LB001        200
B003     PUB002     HISTORY IN VIET NAM   200      600.00000 2011-10-06   LB003        100
B004     PUB003     HARRY AND POSTER     100      1200.0000 2015-05-10   LB002        500
B005     PUB005     TOM AND JERRY        1000     2000.0000 2016-01-07   LB002        600
SQL>

```

Рис.3.2. Содержимое таблицы book.

```

SQL> select *from cart;

ID_CART  ID_BUYER  QUANTITY DATA_ORDER_CREATE  AMOUNT  ADDRESS          DATE_DESTINATION
=====
CA001    BUY001    12 2015-10-09      9600.0000 236 HOANG QUOC VIET 2015-10-20
CA002    BUY003    100 2015-12-09      80000.000 DOM 30,PR.GRARDANSKI,ST.PETEBURG 2016-01-07
CA003    BUY004    5 2015-12-12      2000.0000 HO CHI MINH STREET 2016-01-10
SQL>

```

Рис.3.3. Содержимое таблицы cart.

```

SQL> select *from list_cart;

ID_LIST_CART ID_BOOK  ID_CART  SUM
=====
LC001        B001     CA001    9600
LC002        B002     CA002    8000
LC003        B004     CA003    2000

SQL>

```

Рис.3.4. Содержимое таблицы list\_cart

```
SQL> select *from list_authors;
```

ID_LIST_AUTHOR	ID_OF_AUTHOR	ID_BOOK
LAU001	AU001	B002
LAU002	AU003	B005
LAU003	AU005	B004
LAU004	AU004	B002
LAU005	AU003	B003

Рис.3.5. Содержимое таблицы List\_authors.

ID_AUTHOR	NAME
AU001	DANG KHANH
AU002	NGUYEN TIEN VU
AU003	BUI ANH TUAN
AU004	DAO XUAN HOA
AU005	TRAN VAN QUYET
AU006	TRINH THANH NAM
AU007	PHAM CONG MINH
AU008	NGUYEN VAN THANG
AU009	NGUYEN VAN TUYEN

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');
```

```
SQL> SELECT * FROM list_authors WHERE id_of_author IN ('AU003');
```

ID_LIST_AUTHOR	ID_OF_AUTHOR	ID_BOOK
LAU002	AU003	B005
LAU005	AU003	B003

**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 ;
```

```
SQL>
SQL> SELECT * FROM BOOK WHERE PAGE_NUMBER>200 AND PRISE>1200 ;
```

ID_BOOK	ID_PUBLISHER	NAME	QUANTITY	PRISE	YRAR_OF_PUBLISHER	ID_LISTBOOK	PAGE_NUMBER
B001	PUB001	MICROCONTROLLER	500	1500.0000	1991-08-09	LB001	300
B005	PUB005	TOM AND JERRY	1000	2000.0000	2016-01-07	LB002	600

```
SQL>
```

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

```
SELECT * FROM BOOK WHERE YRAR_OF_PUBLISHER LIKE '%2011%';
```

```
SQL> SELECT * FROM BOOK WHERE YRAR_OF_PUBLISHER LIKE '%2011%';
```

ID_BOOK	ID_PUBLISHER	NAME	QUANTITY	PRISE	YRAR_OF_PUBLISHER	ID_LISTBOOK	PAGE_NUMBER
B002	PUB001	MATH	500	1200.0000	2011-10-09	LB001	200
B003	PUB002	HISTORY IN VIET NAM	200	600.00000	2011-10-06	LB003	100

```
SQL>
```

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

```
SELECT * FROM AUTHOR WHERE ID_AUTHOR BETWEEN 'AU001' AND 'AU006';
```

```
SQL> SELECT * FROM AUTHOR WHERE ID_AUTHOR BETWEEN 'AU001' AND 'AU006';
```

ID_AUTHOR	NAME
AU001	DANG KHANH
AU002	NGUYEN TIEN VU
AU003	BUI ANH TUAN
AU004	DAO XUAN HOA
AU005	TRAN VAN QUYET
AU006	TRINH THANH NAM

**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;
```

```
SQL> Select book.id_book,book.name,id_list_author,author.name FROM book,list_authors, author where list_authors.id_of_author=author.id_auth
or AND book.id_book=list_authors.id_book;
```

ID_BOOK	NAME	ID_LIST_AUTHOR	NAME
B002	MATH	LAU001	DANG KHANH
B002	MATH	LAU004	DAO XUAN HOA
B003	HISTORY IN VIET NAM	LAU005	BUI ANH TUAN
B004	HARRY AND POSTER	LAU003	TRAN VAN QUYET
B005	TOM AND JERRY	LAU002	BUI ANH TUAN

```
SQL>
```

*название книги и названия издательства, где она выпущена*

```
-SQL>SELECT BOOK.ID_BOOK,
BOOK.NAME,PUBLISHER.ID_PUBLISHER,PUBLISHER.NAME,PUBLIS
HER.ADDRESS FROM BOOK,PUBLISHER where
book.id_publisher=publisher.id_publisher;
```

```
isql - Shortcut
```

```
SQL> SELECT BOOK.ID_BOOK, BOOK.NAME,PUBLISHER.ID_PUBLISHER,PUBLISHER.NAME,PUBLISHER.ADDRESS FROM BOOK,PUBLISHER where book.id_publisher=publisher.id_publisher;
```

ID_BOOK	NAME	ID_PUBLISHER	NAME	ADDRESS
B001	MICROCONTROLLER	PUB001	ABA BINH MINH	X.15 DIEN CHAU NGHE AN
B002	MATH	PUB001	ABA BINH MINH	X.15 DIEN CHAU NGHE AN
B003	HISTORY IN VIET NAM	PUB002	AOA HOA BINH	236-DIEN BIEN PHU-HN
B004	HARRY AND POSTER	PUB003	DIEN CHAU	K.6 TT.DIEN CHAU NGHE AN
B005	TOM AND JERRY	PUB005	ST.PETEBURG	D30-GRADANSKI PR

```
SQL>
```

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

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

```
SELECT * FROM author ORDER BY name ASC;
```

```
SQL> SELECT * FROM author ORDER BY name ASC;
```

ID_AUTHOR	NAME
AU003	BUI ANH TUAN
AU001	DANG KHANH
AU004	DAO XUAN HOA
AU002	NGUYEN TIEN VU
AU008	NGUYEN VAN THANG
AU009	NGUYEN VAN TUYEN
AU007	PHAM CONG MINH
AU005	TRAN VAN QUYET
AU006	TRINH THANH NAM

```
SQL>
```



д) Создать запрос, вычисляющий несколько совокупных характеристик таблиц:

выбирать книги с 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;

```
SQL> select book.id_book,book.name,book.prise, book.page_number,author.name FROM book,author,list_authors where book.prise>1000 AND book.page_number>300 AND list_authors.id_of_author=author.id_
thor AND book.id_book=list_authors.id_book;
```

ID_BOOK	NAME	PRICE	PAGE_NUMBER	NAME
B004	HARRY AND POSTER	1200.0000	500	TRAN VAN QUYET
B005	TOM AND JERRY	2000.0000	600	BUI ANH TUAN

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

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

SELECT name, adress,telephone \*FROM buyer;

```
SQL> SELECT name, adress,telephone FROM buyer;
```

NAME	ADRESS	TELEPHONE
NGUYEN TAN DUNG	CHO XANH	0987469427
NGO KIEN HUY	HA NOI	0974536217
DANG KHANH	DOM 30,PR.GRARDANSKI, ST.PETEBURG	0976547324
PHAM TRUONG	176 HO TUNG MAU, CAU GIAY ,HN	0976547345

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

```
-SQL>SELECT BOOK.ID_BOOK,
BOOK.NAME,PUBLISHER.ID_PUBLISHER,PUBLISHER.NAME,PUBLIS
HER.ADDRESS FROM BOOK,PUBLISHER where
book.id_publisher=publisher.id_publisher;
```

```

isql - Shortcut
SQL> SELECT BOOK.ID_BOOK, BOOK.NAME,PUBLISHER.ID_PUBLISHER,PUBLISHER.NAME,PUBLISHER.ADDRESS FROM BOOK,PUBLISHER where book.id_publisher=publisher.id_publisher;

ID_BOOK  NAME                                ID_PUBLISHER  NAME                                ADDRESS
=====
B001     MICROCONTROLLER                    PUB001        ABA BINH MINH                      X.15 DIEN CHAU NGHE AN
B002     MATH                               PUB001        ABA BINH MINH                      X.15 DIEN CHAU NGHE AN
B003     HISTORY IN VIET NAM                PUB002        AOA HOA BINH                      236-DIEN BIEN PHU-HN
B004     HARRY AND POSTER                   PUB003        DIEN CHAU                          K.6 TT.DIEN CHAU NGHE AN
B005     TOM AND JERRY                      PUB005        ST.PETEBURG                        D30-GRARDANSKI PR

SQL>

```

**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:

выбирать книги с имени и их авторов, которые страница >300 и цена >1000

```

select book.id_book,book.name,book.prixe, book.page_number,author.name
FROM book,author where book.prixe>1000 AND book.page_number>300 AND
list_authors.id_of_author=author.id_author AND
book.id_book=list_authors.id_book;

```

```

SQL> select book.id_book,book.name,book.prixe, book.page_number,author.name FROM book,author,list_authors where book.prixe>1000 AND book.page_number>300 AND list_authors.id_of_author=author.id_
thor AND book.id_book=list_authors.id_book;

```

ID_BOOK	NAME	PRISE	PAGE_NUMBER	NAME
B004	HARRY AND POSTER	1200.0000	500	TRAN VAN QUYET
B005	TOM AND JERRY	2000.0000	600	BUI ANH TUAN

**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 реализовывать без использования запросов с русскими символами.

Выведем заметки всех КНИГ изатели с 2011 года:

```

SELECT * FROM BOOK WHERE YRAR_OF_PUBLISHER LIKE
'%2011%';

```



```
SQL> SELECT * FROM BOOK WHERE YRAR_OF_PUBLISHER LIKE '%2011%';
```

ID_BOOK	ID_PUBLISHER	NAME	QUANTITY	PRISE	YRAR_OF_PUBLISHER	ID_LISTBOOK	PAGE_NUMBER
B002	PUB001	MATH	500	1200.0000	2011-10-09	LB001	200
B003	PUB002	HISTORY IN VIET NAM	200	600.00000	2011-10-06	LB003	100

```
SQL>
```

**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:**

```
SQL> select *from book;
```

ID_BOOK	ID_PUBLISHER	NAME	QUANTITY	PRISE	YRAR_OF_PUBLISHER	ID_LISTBOOK	PAGE_NUMBER
B001	PUB001	MICROCONTROLLER	500	1500.0000	1991-08-09	LB001	300
B002	PUB001	MATH	500	1200.0000	2011-10-09	LB001	200
B003	PUB002	HISTORY IN VIET NAM	200	600.00000	2011-10-06	LB003	100
B004	PUB003	HARRY AND POSTER	100	1200.0000	2015-05-10	LB002	500
B005	PUB005	TOM AND JERRY	1000	2000.0000	2016-01-07	LB002	600

```
SQL> update book set prise=prise+500 where id_book between 'B001' and 'B004';
```

```
SQL> select *from book;
```

ID_BOOK	ID_PUBLISHER	NAME	QUANTITY	PRISE	YRAR_OF_PUBLISHER	ID_LISTBOOK	PAGE_NUMBER
B001	PUB001	MICROCONTROLLER	500	2000.0000	1991-08-09	LB001	300
B002	PUB001	MATH	500	1700.0000	2011-10-09	LB001	200
B003	PUB002	HISTORY IN VIET NAM	200	1100.0000	2011-10-06	LB003	100
B004	PUB003	HARRY AND POSTER	100	1700.0000	2015-05-10	LB002	500
B005	PUB005	TOM AND JERRY	1000	2000.0000	2016-01-07	LB002	600

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)** – удаления записей из таблицы книг, на которые не написаны отзывы

## 2. Индивидуальное задание

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

Для того, выбрал 10 идентификаторов (id\_list\_author) из таблицы список авторов (list\_authors), которые есть максимум заказов, за заданный период например between '10.10.2015' and '30.12.2015'

SQL> select first 10 id\_list\_author, sum(sum\_order) from list\_cart, list\_authors, cart where list\_authors.id\_book = list\_cart.id\_book and (list\_cart.id\_cart = cart.id\_cart and cart.date\_dostavki between '10.10.2015' and '30.12.2015') group by id\_list\_author order by sum(sum\_order) DESC;

```
isql - Shortcut
SQL> select * from list_cart;
```

ID_LIST_CART	ID_BOOK	ID_CART	SUM_ORDER
LC001	B001	CA001	1000
LC002	B002	CA002	800
LC003	B004	CA003	900
LC005	B005	CA002	400
LC006	B013	CA004	750
LC007	B009	CA007	850
LC004	B003	CA003	600
LC008	B012	CA010	860

```
SQL> select * from cart;
```

ID_CART	ID_BUYER	KOLICHESTVO_DOSTAVKI	DATE_DOSTAVKI	AMOUNT	DATE_RECEIVED
CA001	BUY001	12	2015-10-09	9600.0000	2015-10-20
CA002	BUY003	100	2015-12-09	80000.000	2016-01-07
CA003	BUY004	5	2015-12-12	2000.0000	2016-01-10
CA004	BUY002	40	2015-09-12	100.00000	2015-09-20
CA005	BUY001	60	2015-11-20	400.00000	2015-12-30
CA006	BUY005	100	2015-12-09	150.00000	2016-01-10
CA007	BUY006	150	2015-12-15	6000.0000	2015-12-30
CA008	BUY008	200	2015-11-20	4000.0000	2015-12-22
CA009	BUY006	350	2015-12-18	4600.0000	2016-01-01
CA010	BUY007	400	2015-10-10	8200.0000	2015-12-11

```
SQL> select first 10 id_list_author, sum(sum_order) from list_cart, list_authors, cart where list_authors.id_book = list_cart.id_book and (list_cart.id_cart = cart.id_cart and cart.date_dostavki between '10.10.2015' and '30.12.2015') group by id_list_author order by sum(sum_order) DESC;
```

ID_LIST_AUTHOR	SUM
LAU012	900
LAU003	900
LAU013	900
LAU010	850
LAU001	800
LAU004	800
LAU005	600
LAU002	400
LAU009	400

```
SQL>
```

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

В данном задании необходимо сгруппировать продажи книг по издательствам и просуммировать стоимость проданных книг, результирующую выборку необходимо отсортировать по убыванию и выбрать 5 верхних строк.

SQL> select first 5 publisher.id\_publisher,sum(sum\_order) from publisher,list\_cart,book where (list\_cart.id\_book=book.id\_book) and (book.id\_publisher=publisher.id\_publisher) group by publisher.id\_publisher order by sum(sum\_order) DESC;

isql - Shortcut

ID_PUBLISHER	NAME_PUBLISHER	ADDRESS
PUB001	ABA BINH MINH	X.15 DIEN CHAU NGHE AN
PUB002	AOA HOA BINH	236-DIEN BIEN PHU-HN
PUB003	DIEN CHAU	K.6 TT.DIEN CHAU NGHE AN
PUB004	HOANG MAI	KT.8 BIM SON-TH
PUB005	ST.PETEBURG	D30-GRARDANSKI PR
PUB006	HO CHI MINH	KCN THU DAU 1

SQL> select \*from book;

ID_BOOK	ID_PUBLISHER	NAME_BOOK	KOLICHESTVO	PRICE	YEAR_OF_PUBLISHER	ID_LISTBOOK	PAGE_NUMBER	RATING_BOOK
B001	PUB001	MICROCONTROLLER	500	1500.0000	1991-08-09	LB001	300	10
B002	PUB001	MATH	500	1200.0000	2011-10-09	LB001	200	8
B003	PUB002	HISTORY IN VIET NAM	200	600.00000	2011-10-06	LB003	100	7
B004	PUB003	HARRY AND POSTER	100	1200.0000	2015-05-10	LB002	500	9
B005	PUB005	TOM AND JERRY	1000	2000.0000	2016-01-07	LB002	600	6
B006	PUB005	LANGUAGE	1000	800.00000	2015-10-10	LB010	500	5
B007	PUB005	ST.PETEBURG CITY	800	400.00000	2014-12-12	LB008	300	7
B008	PUB003	TAT DEN	600	1200.0000	2009-09-18	LB008	800	8
B009	PUB003	DE MEN PHIEU LUU KY	1500	1300.0000	2007-12-20	LB002	700	2
B010	PUB004	WE ARE ONE	200	400.00000	2011-04-09	LB002	300	4
B011	PUB006	EVERY DAY I LOVE U	350	600.00000	2014-02-25	LB006	20	7
B012	PUB006	GIAI TOAN VAT LY 10	900	600.00000	2012-12-22	LB001	260	9
B013	PUB004	ECONOMIC RUSSIA	1500	320.00000	2010-04-20	LB004	90	3

SQL> select \*from list\_cart;

ID_LIST_CART	ID_BOOK	ID_CART	SUM_ORDER
LC001	B001	CA001	1000
LC002	B002	CA002	800
LC003	B004	CA003	900
LC005	B005	CA002	400
LC006	B013	CA004	750
LC007	B009	CA007	850
LC004	B003	CA003	600
LC008	B012	CA010	860

SQL> select first 5 publisher.id\_publisher,sum(sum\_order) from publisher,list\_cart,book where (list\_cart.id\_book=book.id\_book) and (book.id\_publisher=publisher.id\_publisher) group by publisher.id\_publisher order by sum(sum\_order) DESC;

ID_PUBLISHER	SUM
PUB001	1800
PUB003	1750
PUB006	860
PUB004	750
PUB002	600

2.3 • Вывести процент заказов, которые используют доставку.

kolichestvo\_dostavki- количество использовал доставку

sum\_order- общий заказ

SQL> select (sum(kolichestvo\_dostavki)\*100/sum(sum\_order)) from cart,list\_cart ;

```
SQL> select (sum(kolichestvo_dostavki)*100/sum(sum_order)) from cart,list_cart ;
```

	DIVIDE
=====	
	18

### 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.