

Санкт- Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Отчет по лабораторной работе
Курс: «Базы данных»
Тема: «Язык SQL-DML»

Выполнил:
Воробей Е.А. группа 43501/3
Проверил:
Мяснов А.В.

Цель работы

Познакомить студентов с языком создания запросов управления данными SQL-DML.

Программа работы

1. Изучите SQL-DML
2. Выполните все запросы из списка стандартных запросов. Продемонстрируйте результаты преподавателю.
3. Получите у преподавателя и реализуйте SQL-запросы в соответствии с **индивидуальным** заданием. Продемонстрируйте результаты преподавателю.
4. Выполненные запросы SELECT сохраните в БД в виде представлений, запросы INSERT, UPDATE или DELETE -- в виде ХП. Выложите скрипт в Subversion

Список стандартных запросов

- Сделайте выборку всех данных из каждой таблицы
- Сделайте выборку данных из одной таблицы при нескольких условиях, с использованием логических операций, LIKE, BETWEEN, IN (не менее 3-х разных примеров)
- Создайте в запросе вычисляемое поле
- Сделайте выборку всех данных с сортировкой по нескольким полям
- Создайте запрос, вычисляющий несколько совокупных характеристик таблиц
- Сделайте выборку данных из связанных таблиц (не менее двух примеров)
- Создайте запрос, рассчитывающий совокупную характеристику с использованием группировки, наложите ограничение на результат группировки
- Придумайте и реализуйте пример использования вложенного запроса
- С помощью оператора INSERT добавьте в каждую таблицу по одной записи
- С помощью оператора UPDATE измените значения нескольких полей у всех записей, отвечающих заданному условию
- С помощью оператора DELETE удалите запись, имеющую максимальное (минимальное) значение некоторой совокупной характеристики
- С помощью оператора DELETE удалите записи в главной таблице, на которые не ссылается подчиненная таблица (используя вложенный запрос)

Ход выполнения работы

Выполнение запросов из списка стандартных запросов

1. Сделайте выборку всех данных из каждой таблицы
Листинг 1. SELECT *

```
SELECT * FROM CLIENTS;  
SELECT * FROM CONTRACTS;  
SELECT * FROM MEDIUMS;  
SELECT * FROM SERVICES;  
SELECT * FROM SHIPMENT;  
SELECT * FROM TIMETABLE;  
SELECT * FROM CONTRACT_SERVICES;  
SELECT * FROM CONTRACT_SHIPMENTS;
```

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

Листинг 2. SELECT LIKE/BETWEEN/IN

```
SELECT * FROM CLIENTS WHERE NAME LIKE 'Ger%';  
SELECT * FROM SERVICES WHERE QUANTITY BETWEEN 1 AND 10;  
SELECT * FROM SHIPMENT WHERE PRICE IN (100, 500);
```

3. Создайте в запросе вычисляемое поле

Листинг 3. SELECT AS

```
SELECT CONTRACT_ID, CONTRACTS.PRICE, (PRICE*0.97) DISCOUNT FROM CONTRACTS;
```

4. Сделайте выборку всех данных с сортировкой по нескольким полям

Листинг 4. SELECT ORDER

```
SELECT * FROM CONTRACTS ORDER BY CLIENT_ID, PRICE;  
SELECT * FROM TIMETABLE ORDER BY INITIAL_DATE, FINAL_DATE;
```

5. Создайте запрос, вычисляющий несколько совокупных характеристик таблиц

Листинг 5. SELECT MIN MAX

```
SELECT MIN(PRICE), MAX(PRICE) FROM SHIPMENTS;
```

6. Сделайте выборку данных из связанных таблиц (не менее двух примеров)

Листинг 6. SELECT JOIN

```
/* Рекламоносители и расписание занятости для них */  
SELECT MEDIUMS."TYPE" AS MEDIUM, TIMETABLE.INITIAL_DATE, TIMETABLE.FINAL_DATE  
FROM TIMETABLE LEFT JOIN MEDIUMS ON TIMETABLE.MEDIUM_ID = MEDIUMS.MEDIUM_ID  
ORDER BY MEDIUMS."TYPE";
```

/* Клиенты и контракты */

```
SELECT CLIENTS.DATA as CLIENT, CONTRACTS.CONTRACT_ID FROM CONTRACTS LEFT JOIN  
CLIENTS ON CLIENTS.CLIENT_ID = CONTRACTS.CONTRACT_ID ORDER BY CLIENTS.DATA;
```

7. Создайте запрос, рассчитывающий совокупную характеристику с использованием группировки, наложите ограничение на результат группировки

Листинг 7. SELECT COUNT

```
/* Вывод клиентов у которых заключено больше одного контракта с агентством. */  
SELECT CONTRACTS.CLIENT_ID, COUNT(CONTRACTS.CLIENT_ID) AS  
NUMBER_OF_CONTRACTS FROM CONTRACTS  
LEFT JOIN CLIENTS ON CONTRACTS.CLIENT_ID = CLIENTS.CLIENT_ID  
GROUP BY CLIENT_ID HAVING COUNT(CONTRACTS.CLIENT_ID) > 1;
```

8. Придумайте и реализуйте пример использования вложенного запроса

Листинг 8. SELECT(SELECT)

```
/* Расписание занятости рекламоносителя 'Реклама на авто' */  
SELECT TIMETABLE.INITIAL_DATE, TIMETABLE.FINAL_DATE, TIMETABLE.PRICE AS PRICE  
FROM TIMETABLE WHERE MEDIUM_ID = (SELECT MEDIUMS.MEDIUM_ID FROM MEDIUMS  
WHERE MEDIUMS."TYPE" = 'РЕКЛАМА НА АВТО') ORDER BY TIMETABLE.PRICE;
```

9. С помощью оператора INSERT добавьте в каждую таблицу по одной записи

Листинг 9. INSERT

```
INSERT INTO CLIENTS (CLIENT_ID, DATA) VALUES (8, 'Кобеняев Андрей');  
INSERT INTO MEDIUMS (MEDIUM_ID, "TYPE") VALUES (6, 'Воображаемая реклама');  
INSERT INTO TIMETABLE (TIME_ID, MEDIUM_ID, INITIAL_DATE, FINAL_DATE, PRICE)  
VALUES (11, 2, timestamp '11-10-2015', timestamp '11-12-2015', 1100.0);  
INSERT INTO SERVICES (SERVICE_ID, "TYPE", PRICE, QUANTITY)  
VALUES (6, 'Аудиозапись', 1, 1000);  
INSERT INTO CONTRACTS (CONTRACT_ID, CLIENT_ID, TIME_ID, PRICE) VALUES (11, 4, 2, 10);  
INSERT INTO SHIPMENT (SHIPMENT_ID, "TYPE", PRICE)
```

```
VALUES (6, 'Конная повозка', 100000);  
INSERT INTO CONTRACT_SERVICES (CO_SERV_ID, CONTRACT_ID, SERVICE_ID)  
VALUES (8, 4, 2);  
INSERT INTO CONTRACT_SHIPMENTS (CO_SHIP_ID, CO_SERV_ID, SHIPMENT_ID)  
VALUES (6, 4, 1);
```

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

Листинг 10. UPDATE

```
/*Увеличиваем на 50 цену услуги с ID 2 */  
UPDATE SERVICES SET PRICE = PRICE + 50.0 WHERE SERVICES.SERVICE_ID=2;
```

11. С помощью оператора DELETE удалите запись, имеющую максимальное (минимальное) значение некоторой совокупной характеристики

Листинг 11. DELETE

```
DELETE FROM SHIPMENT WHERE SHIPMENT.SHIPMENT_ID =(SELECT  
SHIPMENT.SHIPMENT_ID FROM SHIPMENT WHERE SHIPMENT.PRICE = (SELECT  
MAX(SHIPMENT.PRICE) FROM SHIPMENT));
```

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

Листинг 12. DELETE

```
DELETE FROM CLIENTS WHERE CLIENTS.CLIENT_ID NOT IN (SELECT CONTRACTS.CLIENT_ID  
FROM CONTRACTS);
```

Выполнение индивидуального задания

1. Вывести статистику заполняемости рекламных носителей по месяцам. Заполняемость считать среднее из отношений количества заполненных дней к количеству незаполненных по каждому носителю в заданном месяце.

2. Вывести 5 клиентов, с которыми дольше всего работало рекламное агентство.

```
SELECT FIRST 5 CLIENTS.CLIENT_ID AS CLIENT, CONTRACTS.CONTRACT_ID,  
(DATEDIFF(DAY FROM TIMETABLE.INITIAL_DATE TO TIMETABLE.FINAL_DATE)) AS PERIOD  
FROM CONTRACTS LEFT JOIN CLIENTS ON CLIENTS.CLIENT_ID = CONTRACTS.CLIENT_ID  
LEFT JOIN TIMETABLE ON CONTRACTS.TIME_ID = TIMETABLE.TIME_ID ORDER BY PERIOD DESC;
```

3. Вывести 10 клиентов, которые чаще всего заказывают комплексные услуги (в одном контракте несколько разных способов доставки рекламной информации).

```
SELECT FIRST 10 CONTRACTS.CLIENT_ID, COUNT(CONTRACTS.CLIENT_ID) AS  
NUMBER_OF_CONTRACTS FROM CONTRACTS LEFT JOIN CLIENTS ON CONTRACTS.CLIENT_ID =  
CLIENTS.CLIENT_ID  
GROUP BY CLIENT_ID HAVING COUNT(CONTRACTS.CLIENT_ID)>1 ORDER BY  
CONTRACTS.CLIENT_ID DESC;
```

Вывод

В результате выполнения данной лабораторной работы были изучены основы создания запросов управления SQL-DML. Были изучены основные операции языка SQL_DML: SELECT – выборка из таблицы, INSERT – добавление записи, JOIN – объединение таблиц для выборки из них, WHERE – условие для выборки, ORDER BY– группировка выбранных данных, UPDATE – загрузка новых данных в таблицу, DELETE – удаление данных из таблицы. Научился объединять и группировать записи, делать вложенные запросы. Реализовал SQL-запросы в соответствии с индивидуальным заданием.