

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

Отчет по лабораторной работе № 3
“SQL-DML”
по дисциплине "Базы данных"

Группа: 43501/3
Студент: Кирпиченков П.С.
Преподаватель: Мяснов А.В.

Санкт-Петербург
2016

1. Программа работы

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

2. Теоретические положения

Язык SQL включает две составляющие: язык определения данных DDL и язык манипулирования данными DML. DML включает команды для выборки данных (SELECT), изменения содержимого таблиц (UPDATE), добавления записей (INSERT) и удаления записей (DELETE).

Синтаксис SELECT:

```
SELECT [ALL | DISTINCT] <список_выбора>
FROM <имя_таблицы>, ...
[ WHERE <условие> ]
[ GROUP BY <имя_столбца>,... ]
[ HAVING <условие> ]
[ ORDER BY <имя_столбца> [ASC | DESC],... ]
```

Оператор SELECT позволяет установить набор полей для вывода, таблицу, из которой производится выборка, условие или ограничение на выводимые кортежи, условия группировки и правила сортировки.

Синтаксис INSERT:

```
INSERT INTO <имя_таблицы> [ (<имя_поля>, ...) ] VALUES (<значение>, ...)
```

Вместо непосредственного ввода значений возможно применение оператора SELECT при соблюдении ограничений на область допустимых значений добавляемых атрибутов.

Синтаксис UPDATE:

```
UPDATE <имя_таблицы>
SET <имя_столбца> = <значение>, ...
[WHERE <условие>]
```

Синтаксис DELETE:

```
DELETE FROM <имя_таблицы> [ WHERE <условие> ]
```

3. Ход выполнения

Запросы на выборку всех данных из таблиц:

```
select * from accessory;  
select * from accessory_type;  
select * from caliber;  
select * from country;  
select * from history;  
select * from manufacture;  
select * from mount;  
select * from owner;  
select * from shell;  
select * from shell_type;  
select * from weapon;  
select * from weapon_instance;  
select * from weapon_mount;  
select * from weapon_type;
```

Запрос с применением оператора like:

```
select name  
from weapon  
where name like 'AK%';  
NAME
```

```
=====
```

AK-74

Запрос с применением between:

```
select name, price  
from accessory  
where price between '250' and '350';  
NAME
```

```
===== PRICE
```

M26 MASS	350
----------	-----

Запрос с применением in:

```
select id, diameter, length  
from caliber  
where diameter in ('9', '5.45');
```

```
      ID      DIAMETER      LENGTH  
=====
```

1	5.45	39.00
4	9.00	19.00
5	9.00	18.00

Запрос с применением сортировки по нескольким атрибутам:

```
select id, start_date, end_date  
from history  
order by start_date, end_date;
```

Последняя часть результата:

ID	START_DATE	END_DATE
=====	=====	=====
85478	2016-09-29	2016-10-16
77560	2016-09-29	2016-10-17
73114	2016-09-29	2016-10-18
11714	2016-09-29	2016-10-20
56797	2016-09-29	2016-10-21
42401	2016-09-29	2016-10-22
26354	2016-09-30	2016-10-04
70542	2016-09-30	2016-10-04
41367	2016-09-30	2016-10-05
81710	2016-09-30	2016-10-05
31807	2016-09-30	2016-10-06
16590	2016-09-30	2016-10-07
14927	2016-09-30	2016-10-07
93401	2016-09-30	2016-10-07
97475	2016-09-30	2016-10-09
23268	2016-09-30	2016-10-11
24346	2016-09-30	2016-10-12
72747	2016-09-30	2016-10-19
70714	2016-09-30	2016-10-19
93961	2016-09-30	2016-10-21

ID	START_DATE	END_DATE
=====	=====	=====
40852	2016-09-30	2016-10-22

Использование агрегирующих функций:

```
select MAX(price) as maximum, AVG(price) as average_price, MIN(price) as MINIMUM
from weapon;
```

MAXIMUM	AVERAGE_PRICE	MINIMUM
=====	=====	=====
1000	400	0

Выборка данных из объединенных таблиц:

```
select weapon.name as weapon, shell.name as shell
from weapon
```

```
join caliber on weapon.id_caliber = caliber.id
```

```
join shell on shell.id_caliber = caliber.id;
```

WEAPON	SHELL
=====	=====
AK-74	5.45x39 b
PM	9x18 b
PM	9x18 tr
MP-5	9x19 b

```
select manufacture.name as manufacture, country.name as country
from manufacture join country on manufacture.id_country = country.id;
```

MANUFACTURE	COUNTRY
temp	temp
Kalashnikov Concern (Izhmash)	Russia
BAE Systems	UK
Heckler & Koch	Germany
Fabrique Nationale d'Herstal	Belgium
Colt's Manufacturing Company	USA

Выборка с совокупной характеристикой, ограничением и группировкой:

```
select caliber.diameter as diameter, caliber.length as length, AVG(shell.price) as
average_shell_price
from shell join caliber on shell.id_caliber = caliber.id
group by caliber.diameter, caliber.length
having caliber.diameter > '7';
```

DIAMETER	LENGTH	AVERAGE_SHELL_PRICE
7.62	39.00	100
7.62	51.00	100
9.00	18.00	100
9.00	19.00	100

Вложенный запрос:

```
select shell.name as shell, manufacture.name as manufacture
from shell
join manufacture on shell.id_manufacture = manufacture.id
where manufacture.id_country = (
select id from country where name = 'Belgium'
);
```

SHELL	MANUFACTURE
7.62x51 ap	Fabrique Nationale
d'Herstal	

Запросы на добавление записей с примитивной проверкой на уникальность первичного ключа:

```
insert into country (id, name)
select max(id)+1, 'temp' from country;
insert into accessory_type (id, name)
select max(id)+1, 'temp' from accessory_type;
insert into manufacture (id, name, id_country)
select max(manufacture.id)+1, 'temp', country.id
from country, manufacture
where country.name = 'temp'
group by country.id;
```

```

insert into shell_type (id, name)
    select max(id)+1, 'temp' from shell_type;
insert into caliber (id, name, length, diameter)
    select max(id)+1, 'temp', '0', '0' from caliber;
insert into shell (id, id_caliber, name, id_manufacture, id_shell_type, price)
    select max(shell.id)+1, caliber.id, 'temp', manufacture.id, shell_type.id, '0'
    from shell, caliber, manufacture, shell_type
    where caliber.name='temp' and manufacture.name='temp' and shell_type.name='temp'
    group by caliber.id, manufacture.id, shell_type.id;
insert into weapon_type (id, name)
    select max(id)+1, 'temp' from weapon_type;
insert into mount (id, name)
    select max(id)+1, 'temp' from mount;
insert into accessory (id, name, id_accessory_type, id_mount, id_manufacture, price)
    select max(accessory.id)+1, 'temp', accessory_type.id, mount.id, manufacture.id, '0'
    from accessory, accessory_type, mount, manufacture
    where accessory_type.name='temp' and mount.name='temp' and manufacture.name='temp'
    group by accessory_type.id, mount.id, manufacture.id;
insert into weapon (id, name, id_caliber, id_type, id_manufacture, price)
    select max(weapon.id)+1, 'temp', caliber.id, weapon_type.id, manufacture.id, '0'
    from weapon, caliber, weapon_type, manufacture
    where caliber.name='temp' and weapon_type.name='temp' and manufacture.name='temp'
    group by caliber.id, weapon_type.id, manufacture.id;
insert into weapon_mount(id, id_weapon, id_mount, quantity)
    select max(weapon_mount.id)+1, weapon.id, mount.id, '1'
    from weapon_mount, weapon, mount
    where weapon.name='temp' and mount.name='temp'
    group by weapon.id, mount.id;
insert into owner(id, last_name, first_name, second_name)
    select max(id)+1, 'temp', 'temp', 'temp' from owner;
insert into weapon_instance (id, id_weapon, id_owner)
    select max(weapon_instance.id)+1, weapon.id, owner.id
    from weapon_instance, weapon, owner
    where owner.last_name='temp' and owner.first_name='temp' and owner.second_name='temp'
    and weapon.name='temp'
    group by weapon.id, owner.id;
insert into history (id, id_owner, id_weapon_instance, start_date, end_date)
    select max(history.id)+1, owner.id, weapon_instance.id, '2016-09-30', '2016-09-31'
    from history, owner, weapon_instance
    where owner.last_name='temp' and weapon_instance.code='temp'
    group by owner.id, weapon_instance.id;
insert into history (id, id_owner, id_weapon_instance, start_date, end_date)
    values ('100000', '100000', '100000', '29-SEP-16', '30-SEP-16');

```

Запрос на изменение всех записей с определенным условием:

```

update shell
set description='temporary description', price='300'
where name='temp';

```


Удаление записи по значению совокупной характеристики:

```
delete from shell
where id in
    (select id
     from shell,
     (select max(price) as maximum from shell) m
     where shell.price=m.maximum);
```

Удаление записей, на которые не ссылаются из подчиненной таблицы:

```
delete from country
where id not in
    (select country.id
     from country
     join manufacture on country.id = manufacture.id_country);
```

Индивидуальный запрос: вывести 10 человек, у которых на руках более X единиц оружия калибром более Y.

```
select first 10 owner.last_name, count(w.id_owner)
from owner
join
    (select weapon_instance.id_owner as id_owner
     from caliber
     join weapon on caliber.id = weapon.id_caliber
     join weapon_instance on weapon_instance.id_weapon = weapon.id
     where caliber.diameter > 8) w
on w.id_owner = owner.id
group by owner.last_name, owner.first_name, owner.second_name
having count(w.id_owner)>4;
```

LAST_NAME	COUNT
=====	
=====	
I][h]oQD\$U;y&0Y3}m@0Kve;kZ=YQ07	5
!(tzO*.}dZ1b3PGLAQ_5hjGnS*)?W<%ymK=PL?-itm9a'F#cM;oaE}	5
#6st#ub`^pe"\Qu:``"eUa/O#Q`<Jj+#~)"WE	5
'h;.2\$4<vI{>X6[Gjfo73i2,fMH0w1'90f>!\%ZS'td1>naA{8'U sr"wrGTGEt;V W{Fsn:1 P684>xphz_R d	5
*G\l#D[>=.; o0eZAEzqGb+MA037 QAF	5
*mv1381dtK	6
-%`^8;PVF(GEs w=@a?XS9	5
/sp?xf_&wJW!f	5
0B[Q ZJ_@&-+4&[9yx!yWegA;d;bEG/B-@\$9hIGqm-9Z;yP5roAi+t	5
2mY6d8TtvHo,qM6z}NF;wEV.X UfonUB60HE>H_#wi5vE t^?^g)Te/adF0"sJ)/[o.Pw/ziJg;:CLzOdaobZ\$	5

Индивидуальное задание: удаление неиспользуемых калибров.

```
delete from caliber
where id not in
    (select caliber.id as id
     from accessory join caliber on caliber.id=accessory.id_caliber)
and id not in
```



```

(select caliber.id as id
from shell join caliber on caliber.id=shell.id_caliber)
and id not in
(select caliber.id as id
from weapon join caliber on caliber.id = weapon.id_caliber);

```

Индивидуальное задание: выбор 5 калибров оружия, наиболее часто передаваемого между владельцами.

```

select first 5 caliber.diameter, caliber.length, avg(avg_wpn_period) as
average_caliber_period
from caliber
join weapon on weapon.id_caliber = caliber.id
join
    (select weapon.id as id_weapon, avg(end_date - start_date) as avg_wpn_period
    from weapon
    join weapon_instance on weapon.id = weapon_instance.id_weapon
    join history on weapon_instance.id = history.id_weapon_instance
    group by id_weapon) wid
on wid.id_weapon = weapon.id
group by caliber.length, caliber.diameter
order by average_caliber_period asc;

```

DIAMETER	LENGTH	AVERAGE_CALIBER_PERIOD
0.00	0.00	1
9.00	18.00	3067
9.00	19.00	3075
5.45	39.00	3078

4. Выводы

Запросы позволяют добавлять, извлекать, модифицировать и удалять данные в базе. Возможность создания весьма сложных запросов и использования их в качестве представлений позволяет избежать избыточности при хранении данных – многие производные характеристики можно получить объединяя и преобразуя данные из одной или нескольких таблиц базы. Необходимые для поддержания актуальности базы данных команды добавления, изменения и удаления также могут использовать возможности оператора SELECT для выбора нужных записей. При использовании модифицирующих данные операторов в качестве хранимых процедур достигается ряд преимуществ – не происходит дублирование кода в клиентском приложении, сокращается объем передаваемых данных между клиентом и сервером базы данных, что может быть важно для распределенных систем.