

Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

Отчет по лабораторной работе № 5

«Хранимые процедуры»

по дисциплине «Базы данных»

Работу выполнил:

студент гр. 43501/3

Хуторной Я. В.

Руководитель

Мяснов А. В.

«__» _____ 2015 г

Санкт-Петербург

2015

Цели работы

Познакомить студентов с возможностями реализации более сложной обработки данных на стороне сервера с помощью хранимых процедур.

Программа работы

- Изучить возможности языка PSQL
- Создать две хранимые процедуры в соответствии с **индивидуальным заданием**, полученным у преподавателя
- Выложить скрипт с созданными сущностями в svn
- Продемонстрировать результаты преподавателю

Выполнение работы

1. Был самостоятельно изучен язык PSQ.
 2. В соответствии с индивидуальным заданием были реализованы две хранимые процедуры:
- Вывести динамику продаж треков по заданному автору поквартально в заданный год по сравнению с аналогичным кварталом предыдущего года.

```
set names WIN1251;
connect 'C:\DB\MUSICLIBRARY.FDB' user 'SYSDBA' password 'masterkey';

create procedure HP_1_LAB5 (ARTISTS_NAMEp varchar(255), YEARp integer)
returns(
kv1 integer, kv2 integer, kv3 integer, kv4 integer) /*Динамики продаж
поквартально*/
as
declare variable b1 integer; /*Продажи в первом квартале года YEARp*/
declare variable b2 integer; /*Продажи во втором квартале года YEARp*/
declare variable b3 integer; /*Продажи в третьем квартале года YEARp*/
declare variable b4 integer; /*Продажи в четвертом квартале года YEARp*/

declare variable a1 integer; /*Продажи в первом квартале года YEARp - 1*/
declare variable a2 integer; /*Продажи во втором квартале года YEARp - 1*/
declare variable a3 integer; /*Продажи в третьем квартале года YEARp - 1*/
declare variable a4 integer; /*Продажи в четвертом квартале года YEARp - 1*/

declare variable YEAR2p integer;
begin
YEAR2p = :YEARp - 1;
/*За заданный год ищем значения поквартально проданных дисков*/
```

```
select    count(SALE_TRACK.SALE_TRACK_ID)    from    SALE_TRACK,
TRACK_ARTISTS, ARTISTS where
SALE_TRACK.TRACK_ID = TRACK_ARTISTS.TRACK_ID and
TRACK_ARTISTS.artists_id = ARTISTS.artists_id and
ARTISTS.artists_name = :ARTISTS_NAMEp and
SALE_TRACK.SALE_DATA between :YEARp||"-01-01" and "2015-04-01"
group by ARTISTS.artists_name into :b1;
```

```
select    count(SALE_TRACK.SALE_TRACK_ID)    from    SALE_TRACK,
TRACK_ARTISTS, ARTISTS where
SALE_TRACK.TRACK_ID = TRACK_ARTISTS.TRACK_ID and
TRACK_ARTISTS.artists_id = ARTISTS.artists_id and
ARTISTS.artists_name = :ARTISTS_NAMEp and
SALE_TRACK.SALE_DATA between :YEARp||"-04-01" and "2015-07-01"
group by ARTISTS.artists_name into :b2;
```

```
select    count(SALE_TRACK.SALE_TRACK_ID)    from    SALE_TRACK,
TRACK_ARTISTS, ARTISTS where
SALE_TRACK.TRACK_ID = TRACK_ARTISTS.TRACK_ID and
TRACK_ARTISTS.artists_id = ARTISTS.artists_id and
ARTISTS.artists_name = :ARTISTS_NAMEp and
SALE_TRACK.SALE_DATA between :YEARp||"-07-01" and "2015-10-01"
group by ARTISTS.artists_name into :b3;
```

```
select    count(SALE_TRACK.SALE_TRACK_ID)    from    SALE_TRACK,
TRACK_ARTISTS, ARTISTS where
SALE_TRACK.TRACK_ID = TRACK_ARTISTS.TRACK_ID and
TRACK_ARTISTS.artists_id = ARTISTS.artists_id and
ARTISTS.artists_name = :ARTISTS_NAMEp and
SALE_TRACK.SALE_DATA between :YEARp||"-10-01" and "2016-01-01"
group by ARTISTS.artists_name into :b4;
```

/*За предыдущий год ищем значения поквартально проданных дисков*/

```
select    count(SALE_TRACK.SALE_TRACK_ID)    from    SALE_TRACK,
TRACK_ARTISTS, ARTISTS where
SALE_TRACK.TRACK_ID = TRACK_ARTISTS.TRACK_ID and
TRACK_ARTISTS.artists_id = ARTISTS.artists_id and
ARTISTS.artists_name = :ARTISTS_NAMEp and
SALE_TRACK.SALE_DATA between :YEAR2p||"-01-01" and "2014-04-01"
group by ARTISTS.artists_name into :a1;
```

```
select    count(SALE_TRACK.SALE_TRACK_ID)    from    SALE_TRACK,
TRACK_ARTISTS, ARTISTS where
```

```

SALE_TRACK.TRACK_ID = TRACK_ARTISTS.TRACK_ID and
TRACK_ARTISTS.artists_id = ARTISTS.artists_id and
ARTISTS.artists_name = :ARTISTS_NAMEp and
SALE_TRACK.SALE_DATA between :YEAR2p||"-04-01" and "2014-07-01"
group by ARTISTS.artists_name into :a2;

select    count(SALE_TRACK.SALE_TRACK_ID)    from    SALE_TRACK,
TRACK_ARTISTS, ARTISTS where
SALE_TRACK.TRACK_ID = TRACK_ARTISTS.TRACK_ID and
TRACK_ARTISTS.artists_id = ARTISTS.artists_id and
ARTISTS.artists_name = :ARTISTS_NAMEp and
SALE_TRACK.SALE_DATA between :YEAR2p||"-07-01" and "2014-10-01"
group by ARTISTS.artists_name into :a3;

select    count(SALE_TRACK.SALE_TRACK_ID)    from    SALE_TRACK,
TRACK_ARTISTS, ARTISTS where
SALE_TRACK.TRACK_ID = TRACK_ARTISTS.TRACK_ID and
TRACK_ARTISTS.artists_id = ARTISTS.artists_id and
ARTISTS.artists_name = :ARTISTS_NAMEp and
SALE_TRACK.SALE_DATA between :YEAR2p||"-10-01" and "2015-01-01"
group by ARTISTS.artists_name into :a4;

/*Ищем динамику, т. е. разницу между кварталами*/
kv1 = :b1 - :a1;
kv2 = :b2 - :a2;
kv3 = :b3 - :a3;
kv4 = :b4 - :a4;

end;
commit;

```

The screenshot shows a window titled "Input Parameters [HP_1_LAB5]". Inside, there's a "Parameters" tab and a "SQL" tab. Below these, there's a table with columns: Name, Type, Null, Value, and Description. Two parameters are listed:

Name	Type	Null	Value	Description
ARTISTS_NAMEP	VARCHAR(...)	<input type="checkbox"/>	Steve Morse	
YEARP	INTEGER	<input type="checkbox"/>	2015	

Рис. 1. Параметры процедуры (автор и год)

Результат выполнения процедуры приведен ниже. Из результатов видно, что у автора Steve Morse в 2015 году относительно 2014 года в первом квартале упала продажа треков на 1 трек. Во втором упала на 7 треков. В третьем квартале продажи выросли на 5 треков. В четвертом – упали на 3 трека.

Procedure executing results:

KV1 = -1
KV2 = -7
KV3 = 5
KV4 = -3

- Вывести зависимость изменения популярности композиции с течением времени в зависимости от жанра.

```
set names WIN1251;
connect 'C:\DB\MUSICLIBRARY.FDB' user 'SYSDBA' password 'masterkey';

/* Вывести зависимость изменения популярности композиции с течением
времени в зависимости от жанра.*/

create procedure HP_2_LAB5_11 (Genre_of_song varchar(255), Name_of_song
varchar(255))
returns(Year_ integer, TrackName varchar(255), Genre_ varchar(255), Sales
integer)
as
declare variable i integer default 0;
declare variable yearp integer;
declare variable y integer;

declare variable id_track integer;
begin
select TRACK.TRACK_YEAR , TRACK.TRACK_ID from TRACK
where TRACK.TRACK_NAME = :Name_of_song and
TRACK.TRACK_GENRE = :Genre_of_song into :yearp, :id_track;

y = 2016-yearp;

while (i<y) do
begin
```

```

select TRACK.TRACK_NAME as TRACK_NAME , TRACK.TRACK_GENRE
as GENRE,
count(SALE_TRACK.SALE_TRACK_ID) as NUMBER_OF_SALES
from SALE_TRACK, TRACK
where SALE_TRACK.TRACK_ID = TRACK.TRACK_ID and
TRACK.TRACK_NAME = :Name_of_song and
TRACK.TRACK_GENRE = :Genre_of_song and
SALE_TRACK.TRACK_ID = :id_track and
SALE_TRACK.SALE_DATA between cast((:yearp+:i) as integer)||"-01-01" and
cast((:yearp+1+:i) as integer)||"-01-01"
group by SALE_TRACK.TRACK_ID , TRACK.TRACK_NAME,
TRACK.TRACK_GENRE into :TrackName, :Genre_, :Sales;

Year_ = cast((:yearp+:i) as integer);
suspend;
i = i + 1;
end
end;
commit;

```

Выведем для трека Smoke on the water, исполненного в разных жанрах, зависимость популярности (т. е. количества продаж) по годам с момента выпуска трека.

Name	Type	Null	Value	Desc
GENRE_OF_SONG	VARCHAR(...)	<input type="checkbox"/>	Rock	
NAME_OF_SONG	VARCHAR(...)	<input type="checkbox"/>	Smoke on the water	

Рис. 2. Трек – Smoke on the water в жанре Rock

YEAR_	TRACKNAME	GENRE_	SALES
1 972	Smoke on the water	Rock	13
1 973	Smoke on the water	Rock	14
1 974	Smoke on the water	Rock	7
1 975	Smoke on the water	Rock	6
1 976	Smoke on the water	Rock	12
1 977	Smoke on the water	Rock	12
1 978	Smoke on the water	Rock	11
1 979	Smoke on the water	Rock	10
1 980	Smoke on the water	Rock	10
1 981	Smoke on the water	Rock	11
1 982	Smoke on the water	Rock	8
1 983	Smoke on the water	Rock	13
1 984	Smoke on the water	Rock	14
1 985	Smoke on the water	Rock	14
1 986	Smoke on the water	Rock	10
1 987	Smoke on the water	Rock	10
1 988	Smoke on the water	Rock	14
1 989	Smoke on the water	Rock	13
1 990	Smoke on the water	Rock	8
1 991	Smoke on the water	Rock	21
1 992	Smoke on the water	Rock	11

Рис. 3. Зависимость популярности с течением времени трека Smoke on the water, исполненного в жанре Rock

Input Parameters [HP_2_LAB5_11]

Parameters SQL Create Parameters History Table

Name	Type	Null	Value	Des
GENRE_OF_SONG	VARCHAR(...)	<input type="checkbox"/>	Classic	
NAME_OF_SONG	VARCHAR(...)	<input type="checkbox"/>	Smoke on the water	

Рис. 4. Трек – Smoke on the water в жанре Classic

YEAR_	TRACKNAME	GENRE_	SALES
2 000	Smoke on the water	Classic	44
2 001	Smoke on the water	Classic	33
2 002	Smoke on the water	Classic	33
2 003	Smoke on the water	Classic	22
2 004	Smoke on the water	Classic	34
2 005	Smoke on the water	Classic	25
2 006	Smoke on the water	Classic	39
2 007	Smoke on the water	Classic	42
2 008	Smoke on the water	Classic	28
2 009	Smoke on the water	Classic	25
2 010	Smoke on the water	Classic	33
2 011	Smoke on the water	Classic	27
2 012	Smoke on the water	Classic	29
2 013	Smoke on the water	Classic	30
2 014	Smoke on the water	Classic	37
2 015	Smoke on the water	Classic	34

Рис. 5. Рис. 3. Зависимость популярности с течением времени трека Smoke on the water, исполненного в жанре Classic

Вывод

В результате выполнения работы были изучены хранимые процедуры. Хранимые процедуры позволяют хранить какие-либо запросы в БД и выполнять их на стороне сервера. Это позволяет увеличить скорость работы базы, т. к. серверы обладают большей производительностью. Ряд операций над данными, реализующих общую для всех пользователей логику и не связанных с пользовательским интерфейсом, целесообразно вынести на сервер.

В ХП могут быть входные и выходные параметры, а также внутренние переменные, в них могут быть арифметические операции, операции над символьными данными, содержат логические операторы (IF ... THEN ... ELSE), операторы перехода по условию (SWITCH ... CASE), операторы циклов (FOR, WHILE, UNTIL) и операторы передачи управления в процедуры (CALL, RETURN).