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

Отчёт о лабораторной работе

Дисциплина: Базы данных

Тема: SQL-программирование: Хранимые процедуры

Выполнил студент гр. 43501/1 Дроздовский А.А.

Руководитель Мяснов А.В.

Санкт –Петербург

2016

1. Цель работы

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

2. Программа работы

1) Изучить возможности языка PSQL
2) Создать две хранимые процедуры в соответствии с индивидуальным заданием, полученным у преподавателя
3) Выложить скрипт с созданными сущностями в svn
4) Продемонстрировать результаты преподавателю

3. Ход работы

Индивидуальное задание:

1) По заданной студии вывести отчет:
- список 5 лучших (по оценкам критиков и по оценкам пользователей)
- список 5 наиболее кассовых фильмов
- 5 наиболее частых актеров
- 5 самых успешных (по сборам) режиссеров
- усредненный рейтинг студии за 5 последних лет

2) Для заданного фильма спрогнозировать его сборы по имеющимся данным для набора актеров и режиссера.

Текст скрипта для первого задания:

```
--5 лучших фильмов заданной студии
CREATE OR ALTER PROCEDURE Five_best_film(id_studia int)
returns (name_film varchar(30),rate_all float)
as
begin
for
    select first 5 films.name, (avg(Rate_critic.Rate)+avg(Rate_USERK.Rate))/2 as Rate
    from films
    join rate_critic on (rate_critic.id_film=films.id)
    join Rate_USERK on (Rate_USERK.id_film=films.id)
    join Film_Studia on (Film_Studia.id_film=films.id)
    join Studia on (Studia.id=Film_Studia.id_name) where Studia.id=:id_studia
    group by films.name
    order by Rate desc into :name_film, :rate_all
do begin
 suspend;
 end
end^

--5 самых кассовых фильмов заданной студии
CREATE OR ALTER PROCEDURE Five_best_summ(id_studia int)
returns (name_film varchar(30),summ int)
as
begin
for
    select first 5 films.name, sum(collections.collections) as Summ
    from films
    join collections on (collections.id_film=films.id)
```

```
    join Film_Studia on (Film_Studia.id_film=films.id)
    join Studia on (Studia.id=Film_Studia.id_name) where Studia.id=:id_studia AND
collections.id_period=4
    group by films.name
    order by Summ desc into :name_film, :summ
do begin
 suspend;
 end
end^

--5 наиболее частых актеров
CREATE OR ALTER PROCEDURE pop_actor(id_studia int)
returns (name_studia varchar(30),rate_all float)
as
begin
for
    select first 10 worker.name, count(worker.name) as quantity
    from worker
    join film_worker on (film_worker.id_worker=worker.id)
    join Film_studia on (Film_studia.id_film=film_worker.id_film) where
Film_studia.id_name=:id_studia
    AND film_worker.id_prof=:2
    group by worker.name
    order by quantity desc into :name_studia, :rate_all
do begin
 suspend;
 end
end^

--Рейтинг студии
CREATE OR ALTER PROCEDURE Rate_studia(id_studia int, New_year int)
returns (name_studia varchar(30),rate_all float)
as
begin
for
    select first 5 Studia.name, (avg(Rate_critic.Rate)+avg(Rate_USERK.Rate))/2 as Rate
    from studia
    join Film_Studia on (Film_Studia.id_name=Studia.id)
    join rate_critic on (rate_critic.id_film=Film_Studia.id_film)
    join Rate_USERK on (Rate_USERK.id_film=Film_Studia.id_film)
    join films on (films.id=Film_Studia.id_film) where studia.id=:id_studia and :new_year-
films.Years<6
    group by studia.name
    order by Rate desc into :name_studia, :rate_all
do begin
 suspend;
 end
end^
```

Выведем отчет по студии Warner brothers (id = 1):

```
SQL> select * from Five_best_film(1);

NAME_FILM                          RATE_ALL
============================== ==============
CCC                              9.0000000
The Shawshank Redemption         7.7500000
The Green Mile                   4.7500000
Assassins Creed                  3.0000000

SQL> select * from Five_best_summ(1);

NAME_FILM                            SUMM
============================== ============
CCC                                1012000
The Green Mile                      582000
Assassins Creed                     157000
The Shawshank Redemption             26000

SQL> select * from pop_actor(1);

NAME_STUDIA                        RATE_ALL
============================== ==============
Anne Le Ny                       2.0000000
Olivier Nakache                  2.0000000
Michael Fassbender               2.0000000
Omar Sy                          1.0000000
Tim Robbins                      1.0000000

SQL> select * from Five_best_summ_prod(1);

NAME_FILM                            SUMM
============================== ============
Audrey Fleurot                     1012000
Morgan Freeman                      582000
Olivier Nakache                     157000
Tim Robbins                          26000

SQL> select * from Rate_studia(1,2017);

NAME_STUDIA                        RATE_ALL
============================== ==============
Warner brothers                  5.5357141
```

Выведем отчет по студии Marvel studios (id = 2):

```
SQL> select * from Five_best_film(2);

NAME_FILM                          RATE_ALL
============================== ==============
Forrest Gump                     7.6666665
DDD                              5.0000000
EEE                              4.5000000
FFF                              4.0000000
Passengers                       4.0000000

SQL> select * from Five_best_summ(2);

NAME_FILM                            SUMM
============================== ============
DDD                                1105605
EEE                                 889000
Forrest Gump                        613000
Passengers                          309000
FFF                                 255000

SQL> select * from pop_actor(2);
SQL> select * from Five_best_summ_prod(2);

NAME_FILM                            SUMM
============================== ============
Jennifer Lawrence                   613000
JosГcphine de Meaux                 309000

SQL> select * from Rate_studia(2,2017);

NAME_STUDIA                        RATE_ALL
============================== ==============
Marvel studios                   6.7500000
```

Выведем отчет по студии 21st century fox (id=3):

```
SQL> select * from Five_best_film(3);

NAME_FILM                          RATE_ALL
============================== ==============
ЗВВ                             8.0000000
The Shawshank Redemption        7.7500000
Schindlers List                 7.5000000
The Green Mile                  4.7500000

SQL> select * from Five_best_summ(3);

NAME_FILM                            SUMM
============================== ============
Schindlers List                   1023920
The Green Mile                     582000
The Shawshank Redemption            26000
ЗВВ                                 25000

SQL> select * from pop_actor(3);

NAME_STUDIA                        RATE_ALL
============================== ==============
Omar Sy                         1.0000000
Frank Darabont                  1.0000000
FranГ§ois Cluzet                 1.0000000

SQL> select * from Five_best_summ_prod(3);

NAME_FILM                            SUMM
============================== ============
Olivier Nakache                   1023920
Morgan Freeman                     582000
Tim Robbins                         26000

SQL> select * from Rate_studia(3,2017);

NAME_STUDIA                        RATE_ALL
============================== ==============
21st century fox                6.9000001
```

Скрипт для прогноза сбора фильма:

```
--5 работников фильма
CREATE OR ALTER PROCEDURE Ten(id_f int)
returns (id_w int)
as
begin
for
    select first 5 worker.id
    from worker
    join Film_worker on (Film_worker.id_worker=worker.id)
    where Film_worker.id_film=:id_f
    group by worker.id
    into :id_w
do begin
 suspend;
 end
end^

--Нахождение сборов фильма, в которых принимал участие данный актер\продюсер
CREATE OR ALTER PROCEDURE prognoz(id_f int)
returns (summ int)
as
declare variable id_w int;
begin
for select * from Ten(:id_f) into :id_w
do begin
    select  sum(collections.collections)
    from worker, Film_worker, collections
    where  collections.id_period=4 and film_worker.id_worker=:id_w
    and film_worker.id_film!=:id_f and Film_worker.id_worker=worker.id
    and collections.id_film=Film_worker.id_film
    into :summ;
    suspend;
end
end^

--Нахождения среднего сбора фильма - прогноз
```

```
CREATE OR ALTER PROCEDURE prognoz_avg(id_f int)
returns (prognoz int)
as
begin
for select avg(summ) from prognoz(:id_f) into :prognoz
do begin
    suspend;
end

end^
```

Результат работы:

```
SQL> select * from prognoz_avg(3);

     PROGNOZ
============
      157000

SQL> select * from prognoz_avg(2);

     PROGNOZ
============
      157000

SQL> select * from prognoz_avg(1);

     PROGNOZ
============
      264250

SQL> select * from prognoz_avg(4);

     PROGNOZ
============
      771000

SQL> select * from prognoz_avg(5);

     PROGNOZ
============
      393230
```

Прогноз основан на среднем сборе фильмов 5 участников заданного фильма.

4. Выводы

Хранимые процедуры позволяют сохранить часто используемые однотипные операции сложной выборки данных из базы. Особенно это полезно, если операции отличаются только константами, использующимися при наложении условий на данные. Для этих целей существуют параметры, передаваемые в хранимую процедуру.

Кроме того, большие и часто используемые операции имеют преимущество, выполняясь на сервере базы данных, так как это уменьшает количество передаваемых по сети данных.

Также необходимо отметить, что хранимые процедуры позволяют распределить доступ определённым группам пользователей БД к определённым хранимым процедурам.

Но у хранимых процедур есть и недостатки. Необходимо поддерживать актуальность используемых операций при изменении структуры базы данных. Также хранимые процедуры зависят от типа и версии используемой СУБД , то перенос проекта из одной СУБД в другую достаточно сложен.