Санкт-Петербургский Политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра Компьютерных Систем и Программных Технологий

**ОТЧЁТ О ЛАБОРАТОРНОЙ РАБОТЕ №4**

Дисциплина: **Базы данных**

# Тема: Хранимые процедуры

А.М. Кириллов

А.В. Мяснов

Выполнил студент гр. № 43501/1

Преподаватель

Санкт-Петербург

2016

# Цель работы

# Ознакомиться возможностями реализации более сложной обработки данных на стороне сервера с помощью хранимых процедур.

# 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

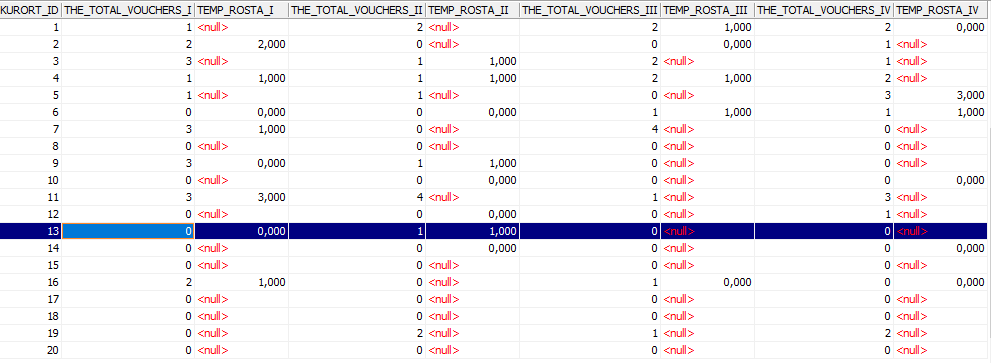
# Ход работы

# Индивидуальное задание:

1. Для каждого курорта вывести для заданного года суммарный объем купленных путевок поквартально и отличие от аналогичного периода годом ранее в %.

|  |
| --- |
| create procedure Kurort\_total (d\_year integer)  returns(  kurort\_id integer,  the\_total\_vouchers\_I integer,  temp\_rosta\_I float,  the\_total\_vouchers\_II integer,  temp\_rosta\_II float,  the\_total\_vouchers\_III integer,  temp\_rosta\_III float,  the\_total\_vouchers\_IV integer,  temp\_rosta\_IV float  ) as  declare variable tourID integer;  declare variable kol\_buy\_this\_year\_I integer;  declare variable kol\_buy\_last\_year\_I integer;  declare variable kol\_buy\_this\_year\_II integer;  declare variable kol\_buy\_last\_year\_II integer;  declare variable kol\_buy\_this\_year\_III integer;  declare variable kol\_buy\_last\_year\_III integer;  declare variable kol\_buy\_this\_year\_IV integer;  declare variable kol\_buy\_last\_year\_IV integer;  declare variable buy\_date date;  begin  for select kurort\_id from kurort order by kurort\_id into :kurort\_id do begin  kol\_buy\_this\_year\_I=0;  kol\_buy\_last\_year\_I=0;  kol\_buy\_this\_year\_II=0;  kol\_buy\_last\_year\_II=0;  kol\_buy\_this\_year\_III=0;  kol\_buy\_last\_year\_III=0;  kol\_buy\_this\_year\_IV=0;  kol\_buy\_last\_year\_IV=0;  for select id from tour where(id\_kurort=:kurort\_id) order by id into :tourID  do  begin  for select start\_time from trip  where (id\_tour=:tourID)and((extract(year from start\_time)=:d\_year)or(extract(year from start\_time)=:d\_year-1))  into :buy\_date do  if (extract(year from buy\_date)=:d\_year) then begin  if ((extract(month from buy\_date)>=1)and(extract(month from buy\_date)<=3))  then  begin kol\_buy\_this\_year\_I=kol\_buy\_this\_year\_I+1;  end  if ((extract(month from buy\_date)>=4)and(extract(month from buy\_date)<=6))  then  begin kol\_buy\_this\_year\_II=kol\_buy\_this\_year\_II+1;  end  if ((extract(month from buy\_date)>=7)and(extract(month from buy\_date)<=9))  then  begin kol\_buy\_this\_year\_III=kol\_buy\_this\_year\_III+1;  end  if ((extract(month from buy\_date)>=10)and(extract(month from buy\_date)<=12)) then  begin kol\_buy\_this\_year\_IV=kol\_buy\_this\_year\_IV+1;  end end else begin  if ((extract(month from buy\_date)>=1)and(extract(month from buy\_date)<=3))  then  begin kol\_buy\_last\_year\_I=kol\_buy\_last\_year\_I+1;  end  if ((extract(month from buy\_date)>=4)and(extract(month from buy\_date)<=6))  then  begin kol\_buy\_last\_year\_II=kol\_buy\_last\_year\_II+1;  end  if ((extract(month from buy\_date)>=7)and(extract(month from buy\_date)<=9))  then  begin kol\_buy\_last\_year\_III=kol\_buy\_last\_year\_III+1;  end  if ((extract(month from buy\_date)>=10)and(extract(month from buy\_date)<=12)) then  begin kol\_buy\_last\_year\_IV=kol\_buy\_last\_year\_IV+1;  end  end  end  the\_total\_vouchers\_I=kol\_buy\_this\_year\_I;  the\_total\_vouchers\_II=kol\_buy\_this\_year\_II;  the\_total\_vouchers\_III=kol\_buy\_this\_year\_III;  the\_total\_vouchers\_IV=kol\_buy\_this\_year\_IV;  if(kol\_buy\_last\_year\_I<>0) then  begin temp\_rosta\_I=kol\_buy\_this\_year\_I/kol\_buy\_last\_year\_I;  end else begin  temp\_rosta\_I=NULL; end  if(kol\_buy\_last\_year\_II<>0) then begin  temp\_rosta\_II=kol\_buy\_this\_year\_II/kol\_buy\_last\_year\_II; end  else begin  temp\_rosta\_II=NULL; end  if(kol\_buy\_last\_year\_III<>0) then begin  temp\_rosta\_III=kol\_buy\_this\_year\_III/kol\_buy\_last\_year\_III; end  else begin  temp\_rosta\_III=NULL; end  if(kol\_buy\_last\_year\_IV<>0) then begin  temp\_rosta\_IV=kol\_buy\_this\_year\_IV/kol\_buy\_last\_year\_IV; end  else begin  temp\_rosta\_IV=NULL;  end  suspend;  end  end |

Результат работы:



**Выводы**

В ходе выполнения данной лабораторной работы мы ознакомились с возможностями реализации обработки данных на стороне сервера с помощью хранимых процедур.

Отличие хранимых процедур от представлений состоит в том, что ХП представляют собой части программы, находящиеся и исполняющиеся непосредственно на сервере базы данных. Они хранятся вместе с базой на сервере в откомпилированном виде и позволяют перенести часто повторяющиеся длительные операции с клиентской машины на SQL-сервер.