Санкт-Петербургский политехнический университет им. Петра Великого

Кафедра компьютерных систем и программных технологий

**Отчёт по лабораторной работе №5**

**Дисциплина**: Базы данных

**SQL-программирование: Хранимые процедуры**

Выполнил студент гр. 43501/3 Анисимов А.А.

Преподаватель: Мяснов А. В.

Санкт-Петербург

2016

**Цели работы**

Знакомство с 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

**Индивидуальное задание:**

Реализовать следующие хранимые процедуры:

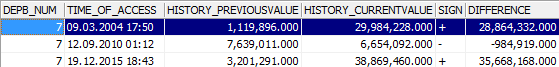
1. Отобразить статистику по клиентам: на какую сумму в среднем за год увеличивается суммарная стоимость ценностей данного клиента.
2. Реализовать процедуру проверки и внесения данных о доступе.

**Выполнение работы:**

1. Реализовать процедуру пересчета стоимости содержимого ячейки по истории.

|  |
| --- |
| SET TERM ^ ;  create or alter procedure TAKE\_THE\_NUMBER\_BY\_HISTORY (  WHICH\_BOX integer)  returns (  DEPB\_NUM integer,  TIME\_OF\_ACCESS timestamp,  HISTORY\_PREVIOUSVALUE float,  HISTORY\_CURRENTVALUE float,  "SIGN" varchar(1),  DIFFERENCE float)  as  begin  /\* Procedure Text \*/  for SELECT  deposit\_box.depb\_num,  box\_history.time\_of\_access,  box\_history.history\_previousvalue,  box\_history.history\_currentvalue,  CASE  WHEN box\_history.history\_previousvalue > box\_history.history\_currentvalue THEN '-'  WHEN box\_history.history\_previousvalue < box\_history.history\_currentvalue THEN '+'  ELSE '='  END,  history\_currentvalue - history\_previousvalue AS difference  FROM deposit\_box, box\_history  WHERE deposit\_box.depb\_num = :which\_box AND deposit\_box.depb\_id = box\_history.depb\_id  ORDER BY box\_history.time\_of\_access  into :depb\_num, :time\_of\_access, :history\_previousvalue, :history\_currentvalue, :sign, :difference  do begin  suspend;  end  end^  SET TERM ; ^ |

Результаты:



Производительность:

Prepare time = 31ms

Execute time = 31ms

Avg fetch time = 10.33 ms

Current memory = 10,204,992

Max memory = 10,282,528

Memory buffers = 2,048

Reads from disk to cache = 3

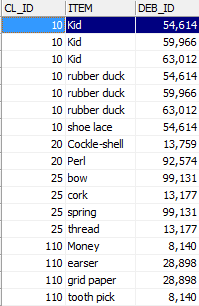
Writes from cache to disk = 0

Fetches from cache = 14

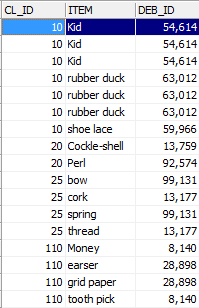
1. Реализовать процедуру, позволяющую перераспределить однотипные объекты в ячейки одного клиента.

|  |
| --- |
| SET TERM ^ ;  create or alter procedure NEW\_PROCEDURE  returns (  CL\_ID integer,  ITEM varchar(128),  DEB\_ID integer)  as  declare variable TMP\_CLIENT\_ID integer;  declare variable TMP\_ITEM varchar(128);  declare variable TMP\_DEB\_ID integer;  declare variable TMP\_NEXT\_DEB\_ID integer;  declare variable TMP\_CURRENT\_CL\_ID integer;  begin  /\* Procedure Text \*/  for  SELECT FIRST 4 cntr.client\_id  FROM contract AS cntr, deposit\_box  JOIN contract ON contract.depb\_id = deposit\_box.depb\_id  WHERE cntr.depb\_id = deposit\_box.depb\_id  AND EXISTS (SELECT FIRST 1 contract.client\_id  FROM contract  WHERE contract.client\_id = cntr.client\_id  GROUP BY contract.client\_id  HAVING count(contract.depb\_id) > 2)  GROUP BY cntr.client\_id  into :tmp\_client\_id  do begin  for  SELECT contract.client\_id, item.item\_type, item.depb\_id  FROM contract  JOIN item ON contract.depb\_id = item.depb\_id  WHERE contract.client\_id = :tmp\_client\_id  ORDER BY item.item\_type  into :CL\_ID, :Item, :DEB\_ID  do  begin  if (tmp\_current\_cl\_id is null) then  tmp\_current\_cl\_id = :cl\_id;  if (:cl\_id <> :tmp\_current\_cl\_id) then  begin  tmp\_current\_cl\_id = :cl\_id;  tmp\_next\_deb\_id = NULL;  end  if (:Item = :tmp\_item) then  begin  if (:deb\_id <> :tmp\_deb\_id) then  begin  tmp\_next\_deb\_id = :deb\_id;  deb\_id = :tmp\_deb\_id;  end  end  else  begin  tmp\_item = :Item;  if (tmp\_next\_deb\_id is null) then  tmp\_deb\_id = :deb\_id;  else  begin  tmp\_deb\_id = :tmp\_next\_deb\_id;  deb\_id = :tmp\_deb\_id;  end  end  begin  suspend;  end  end  end  end^  SET TERM ; ^  /\* Following GRANT statements are generated automatically \*/  GRANT SELECT ON CONTRACT TO PROCEDURE NEW\_PROCEDURE;  GRANT SELECT ON DEPOSIT\_BOX TO PROCEDURE NEW\_PROCEDURE;  GRANT SELECT ON ITEM TO PROCEDURE NEW\_PROCEDURE;  /\* Existing privileges on this procedure \*/  GRANT EXECUTE ON PROCEDURE NEW\_PROCEDURE TO SYSDBA; |

До сортировки:



После:



Производительность:

Prepare time = 32ms

Execute time = 31ms

Avg fetch time = 1.82 ms

Current memory = 10,099,632

Max memory = 10,166,640

Memory buffers = 2,048

Reads from disk to cache = 0

Writes from cache to disk = 0

Fetches from cache = 1,792

**Выводы:**

В результате выполнения работы были изучены хранимые процедуры. Хранимые процедуры позволяют хранить какие-либо сложные запросы в БД и выполнять их на стороне сервера. Выполнение функций на сервере снижает нагрузку на канал связи, поскольку передается только окончательный результат, при его наличии. На сервере хранимые процедуры хранятся уже в скомпилированном виде, поэтому их выполнение тратится меньше времени.

Процедуры позволяют организовать интерфейс доступа к данным и в случае изменений на серверной стороне, позволяют избежать необходимости переписывания клиентского приложения.

В хранимых процедурах могут быть входные и выходные параметры и локальные переменные, в них могут производиться числовые вычисления и операции над символьными данными, результаты которых могут присваиваться переменным параметрам. В хранимых процедурах могут выполняться стандартные операции с базами данных (как DDL, так и DML). Это расширяет возможности работы с базами данных и позволяет легче реализовывать многие операции.