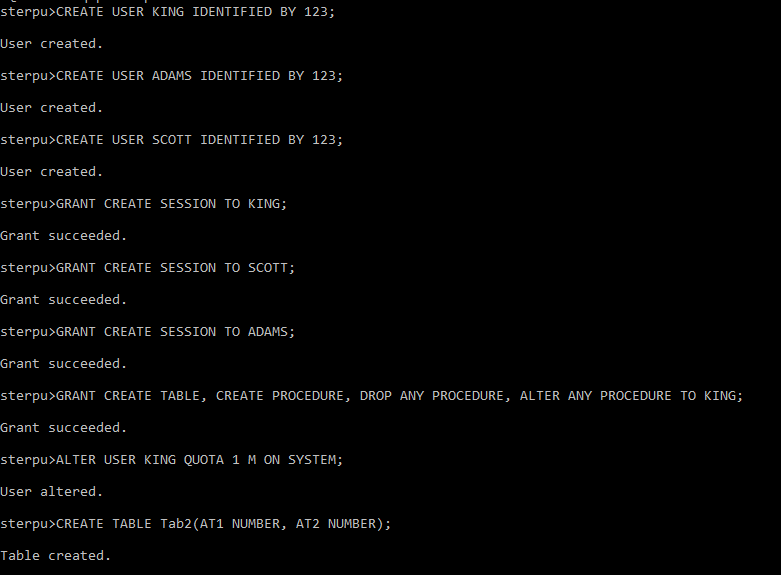
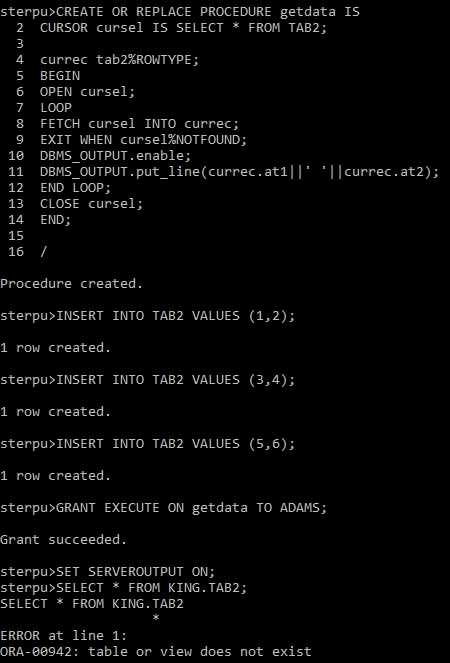
|  |  |
| --- | --- |
| МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ | |
| ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САМАРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С.П. КОРОЛЕВА» | |
|  | |
|  | |
| Лабораторная работа №19 | |
|  | |
|  | Выполнил:  Стерпу Е.К., гр. 6411  Проверил:  Додонов М.В. | |
|  | |
| Самара 2018 | |

**Цель:** обучение методам и средствам разграничения доступа к данным с использованием программных модулей, хранящихся в базах данных СУБД *Oracle*.

1. Создать вспомогательные объекты для выполнения лабораторной работы.



1. Как пользователь *King* создать процедуру выборки данных из таблицы *Tab*2и предоставить право на ее исполнение пользователю *ADAMS*. Пользователь *ADAMS*, выполняя процедуру, может получить результат выборки из *KING*.*Tab*2, но попытка явного выполнения выборки отвергается системой.



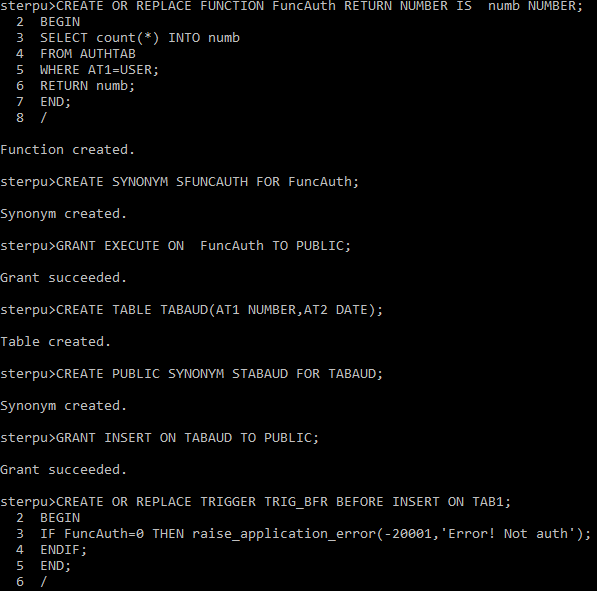
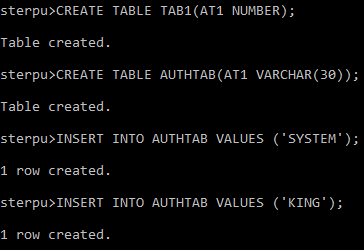
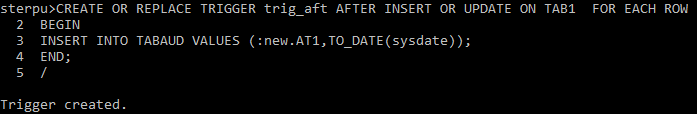
EXECUTE.KING.getdata;

1 2

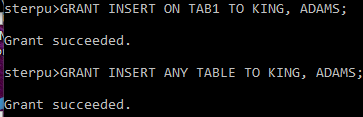
3 4

5 6

1. Создать триггер для выполнения функций разграничения доступа и фискальных действий. Как пользователь *SYSTEM* создать таблицу *Tab*1с одним столбцом *At*1, в который предполагается запись данных уполномоченными пользователями. Список уполномоченных пользователей размещается в таблице *AuthTab*. Процесс ввода данных в таблицу *Tab*1контролируется триггером *TRIG*\_*BFR*, который автоматически выполняется до выполнения операции вставки строк в таблицу *Tab*1. Наличие полномочий у пользователя на выполнение операции вставки проверяется вызовом функции *FuncAuth*. Для регистрации действий уполномоченных пользователей создать таблицу *TabAud*. В таблице *TabAud* регистрировать вводимое в таблицу *Tab*1значение атрибута и дату вставки записи. Регистрация действий осуществляется после выполнения вставки строки в таблицу *Tab*1.

1. Пользователь *KING* уполномочен выполнять вставку строк в таблицу *Tab*1, а пользователь *ADAMS* — нет. С позиций системы пользователи наделяются одинаковыми привилегиями.



1. Для пользователя *KING* операция вставки строк выполняется успешно. Для пользователя *ADAMS* — автоматически запускается триггер, который генерирует исключительную ситуацию по нарушению условий авторизации доступа.

sterpu> INSERT INTO SYST.TAB1 VALUES(1);

sterpu> INSERT INTO SYST.TAB1 VALUES(3);

ORA-20001:Error! Not auth

sterpu> SELECT \* FROM TAB1;

AT1

----------

1

sterpu> SELECT \* FROM TABAUD;

AT1 AT2

---------- --------

1 11.04.18

1. Создать процедуру *definer*\_*proc* с правами создателя, выдающую значения атрибутов контекста: *CURRENT*\_*USER* (имя пользователя, с привилегиями которого работает сеанс), *SESSION*\_*USER* (имя пользователя, зарегистрировавшегося и первоначально создавшего этот сеанс. Это значение в течение сеанса неизменно), *CURRENT*\_*SCHEMA* (имя стандартной схемы, которая будет использоваться при разрешении неуточненных ссылок на объекты). Дать пользователю scott права на выполнение этой процедуры.

sterpu> CREATE OR REPLACE PROCEDURE definer\_proc IS

2 BEGIN

3 DBMS\_OUTPUT.enable;

4 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','CURRENT\_USER'));

5 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','SESSION\_USER'));

6 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','CURRENT\_SCHEMA'));

7 END;

8 /

sterpu> GRANT EXECUTE ON definer\_proc TO SCOTT;

1. Создать процедуру *invoker*\_*proc* с правами вызывающего, выдающую значения атрибутов контекста: *CURRENT*\_*USER* (имя пользователя, с привилегиями которого работает сеанс), *SESSION*\_*USER* (имя пользователя, зарегистрировавшегося и первоначально создавшего этот сеанс. Это значение в течение сеанса неизменно), *CURRENT*\_*SCHEMA* (имя стандартной схемы, которая будет использоваться при разрешении неуточненных ссылок на объекты).Дать пользователю scott права на выполнение этой процедуры.

sterpu> CREATE OR REPLACE PROCEDURE invoker\_proc AUTHID СURRENT\_USER AS

2 BEGIN

3 DBMS\_OUTPUT.enable;

4 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','CURRENT\_USER'));

5 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','SESSION\_USER'));

6 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','CURRENT\_SCHEMA'));

7 END;

8 /

sterpu> GRANT EXECUTE ON invoker\_proc TO SCOTT;

1. Соедениться как пользователь scott и выполнить представленные выше процедуры и сравнить выдаваемые ими результаты.

SQL> set sqlp sterpu>

sterpu> SET SERVEROUTPUT ON;

EXECUTE SYST.definer\_proc;

SYST

SCOTT

SYST

EXECUTE SYST.invoker\_proc;

SCOTT

SCOTT

SCOTT

1. Внутри процедуры, работающей с правами создателя, текущий пользователь и схема, с привилегиями которой работает сеанс – *KING*. Пользователь, зарегистрировавшийся и начавший сеанс – *SCOTT*. Это значение в течение сеанса не меняется.

SQL> set sqlp sterpuk>

sterpuk> CREATE OR REPLACE PROCEDURE definer\_proc IS

2 BEGIN

3 DBMS\_OUTPUT.enable;

4 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','CURRENT\_USER'));

5 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','SESSION\_USER'));

6 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','CURRENT\_SCHEMA'));

7 END;

8 /

sterpuk> GRANT EXECUTE ON definer\_proc TO SCOTT;

sterpus>EXECUTE KING.DEFINER\_PROC;

KING

SCOTT

KING

sterpuk> CREATE OR REPLACE PROCEDURE invoker\_proc AUTHID CURRENT\_USER AS

2 BEGIN

3 DBMS\_OUTPUT.enable;

4 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','CURRENT\_USER'));

5 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','SESSION\_USER'));

6 DBMS\_OUTPUT.put\_line(SYS\_CONTEXT('USERENV','CURRENT\_SCHEMA'));

7 END;

8 /

sterpuk> GRANT EXECUTE ON invoker\_proc TO SCOTT;

EXECUTE KING.INVOKER\_PROC; => SCOTT || SCOTT || SCOTT