

Lab 4. Privilegii si roluri (partea I)

Cuvinte cheie: <ul style="list-style-type: none">Privilegiu (sistem, obiect)	<ul style="list-style-type: none">RolIerarhia privilegiilor
--	--

Recapitulare:

In laboratorul trecut (lab. 3) am studiat restricționarea accesului la resurse computaționale. Ne reamintim:

„Următorul pas, după crearea conturilor de utilizatori si stabilirea spațiilor de stocare, il reprezintă impunerea unor limitări referitoare la accesul la resurse pentru utilizatori. Scopul este de a asigura o funcționare performanta a bazei de date, evitarea monopolului de către un utilizator asupra resurselor, s.a.

Parametrii de performanta care sunt adesea regasiti in aceste configurări se refera la:

- timpul de execuție estimat al instrucțiunilor; consumul de CPU;
- gradul de paralelism acceptat in sisteme multi-procesor;
- numărul de sesiuni deschise per utilizator;timp de inactivitate (idle).”

In acest laborator vom studia un alt tip de restricționare a activitatii utilizatorilor bazei de date prin privilegii si roluri.

Un privilegiu este un drept de a executa un tip particular de instrucțiune SQL sau de a accesa obiectele altui utilizator. Exista 2 categorii mari de privilegii: sistem si obiect, ce sunt rezumate in tabelul 1.

Acordarea unui privilegiu si respectiv retragerea unui privilegiu catre utilizatori se realizează după sintaxa.

GRANT privilegiu1,privilegiu2,...privilegiun [ON obiect] TO utilizator;
REVOKE privilegiu1,privilegiu2,...privilegiun [ON obiect] FROM utilizator;

La orice moment de timp un utilizator isi poate afla privilegiile din sesiunea curenta cu interogarea:

```
SELECT * FROM session_privs;
SQL> select * from session_privs;

PRIVILEGE
-----
CREATE SESSION
CREATE TABLE
CREATE ANY TABLE
```

Conectați ca SYS/SYSDBA putem afla privilegiile oricarui utilizator cu interogarea:
SELECT substr(grantee,1,20) grantee, owner, substr(table_name,1,15) table_name, grantor,privilege
FROM DBA_TAB_PRIVS WHERE grantee like '%ELEARN%';

GRANTEE	OWNER	TABLE_NAME	GRANTOR	PRIVILEGE
ELEARN_PROFESOR1	ELEARN_APP_ADMIN	CURS	ELEARN_APP_ADMIN	UPDATE
ELEARN_PROFESOR1	ELEARN_APP_ADMIN	CURS	ELEARN_APP_ADMIN	SELECT
ELEARN_PROFESOR1	ELEARN_APP_ADMIN	CURS	ELEARN_APP_ADMIN	INSERT

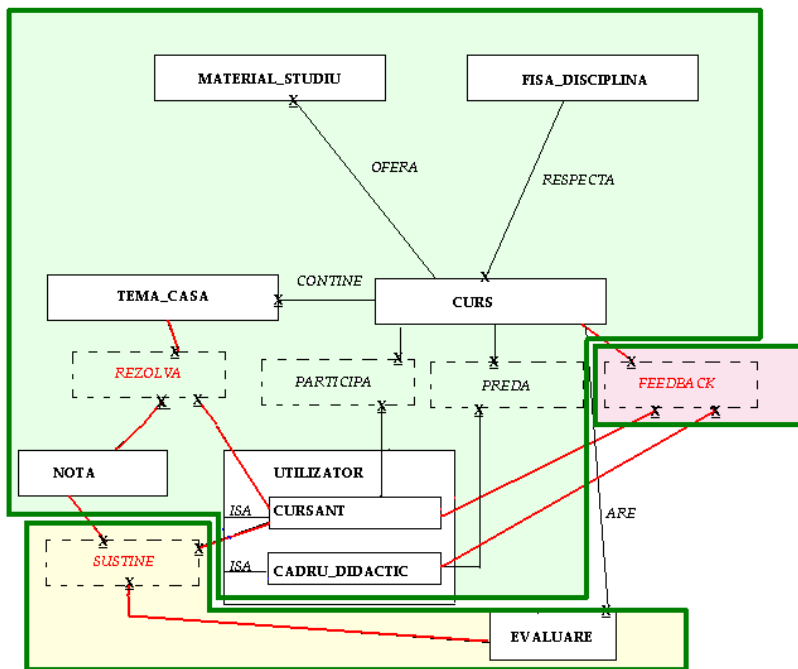
Obs: deși SYS i-a dăruit privilegiul, ca grantor apare proprietarul obiectului, adica ELEARN_APP_ADMIN

Conectați ca SYS/SYSDBA putem afla privilegiile tuturor utilizatorilor pe un anumit obiect al bazei de date cu interogarea:

SELECT substr(grantee,1,15) grantee, owner, substr(table_name,1,15) table_name, grantor,privilege
FROM DBA_TAB_PRIVS WHERE table_name='CURS';

Privilegiile se clasifica in: privilegii sistem si privilegii obiect (v. tabelu). Privilegiile sistem pot fi acordate cu optiunea **WITH ADMIN OPTION**, iar privilegiile obiect pot fi acordate cu optiunea **WITH GRANT OPTION**, optiuni ce permit detinatorilor lor sa acorde la randul lor privilegiul catre alti utilizatori.

Se considera exemplul aplicației de e-learning început în laboratorul 3 din care reamintim schemele:



Utilizatorul	Schimbam parola	Obiectele din schema utilizatorului, conform design-ului aplicației
ELEARN_APP_ADMIN	alter user ELEARN_APP_ADMIN identified by 12345;	MATERIAL_STUDIU, FISA_DISCIPLINA, TEMA_CASA, CURS, NOTA, UTILIZATOR, CURSANT, CADRU_DIDACTIC, REZOLVA, PARTICIPA, PREDĂ
ELEARN_student1	alter user ELEARN_student1 identified by 12345;	
ELEARN_student2		
ELEARN_student3		
ELEARN_student4		
ELEARN_student5		
ELEARN_student6		
ELEARN_student7		
ELEARN_student8		
ELEARN_student9		
ELEARN_student10		
ELEARN_profesor1	alter user ELEARN_profesor1 identified by 12345;	FEEDBACK
ELEARN_profesor2		FEEDBACK
ELEARN_asistent3	alter user ELEARN_asistent3 identified by 12345;	FEEDBACK
ELEARN_GUEST		
OP\$MM-33C58500149B/ELEARN_CAT		EVALUARE, SUSTINE

Tabl 1. Tipuri de privilegii , cu comentarii si exemple (conectare sys as sysdba)

Tip de privilegiu	Comentarii	Exemple de privilegii	Exemplu real
1. Privilegii sistem	1.1. Permit utilizatorului sa se conecteze la baza de date	CREATE SESSION	GRANT CREATE SESSION TO ELEARN_APP_ADMIN;
	1.2. Permit utilizatorului sa creeze orice obiect care are un anumit tip si apartine schemei proprii (peste care el este owner)	CREATE TABLE CREATE SEQUENCE CREATE VIEW	<p>Utilizatorul ELEARN_APP_ADMIN incearca sa creeze tabela CURS in schema proprie:</p> <pre>CREATE TABLE CURS (id number(6) primary key, denumire varchar2(30) NOT NULL, an_studiu number(1) NOT NULL, semestru number(1) NOT NULL, nr_credite number(1) NOT NULL, forma_evaluare VARCHAR2(10) DEFAULT 'EXAMEN', ore_curs number(2) DEFAULT 28, ore_laborator number(2) DEFAULT 14, ore_seminar number(2) DEFAULT 0);</pre> <p>CREATE TABLE CURS * ERROR at line 1: ORA-01031: insufficient privileges</p> <p>SYS AS SYSDBA ii acorda privilegiul de a crea tabele utilizatorului ELEARN_APP_ADMIN in schema acestuia: GRANT CREATE TABLE TO ELEARN_APP_ADMIN;</p> <p>Utilizatorul ELEARN_APP_ADMIN incearca din nou sa creeze tabela CURS in schema proprie: Table created.</p> <p>Din acest moment, utilizatorul ELEARN_APP_ADMIN este owner (proprietar) peste tabela CURS si poate executa cu succes operatii LDD si LMD pe aceasta:</p> <pre>ALTER TABLE CURS drop column ore_seminar;</pre> <pre>DROP TABLE CURS ; -- apoi o recreaza</pre> <pre>INSERT INTO CURS VALUES (1,'SECURITATEA BAZELOR DE DATE',6,1,5,'E',28,14,0);</pre> <p>Reținem ca proprietarului asupra unei tabele nu ii putem revoca privilegiile asupra ei.</p>

	<p>1.3. Permit utilizatorului sa efectueze o operatie LDD sau o operatie LMD anume asupra oricarui obiect al schemei oricarui utilizator, care are un anumit tip (tabel, trigger,s.a)</p> <p>select name from system_privilege_map where name like '%ANY%' order by name;</p>	<p>CREATE ANY TABLE</p> <p>ALTER ANY TABLE</p> <p>DROP ANY TABLE DROP ANY VIEW DROP ANY TRIGGER</p> <p>SELECT ANY TABLE INSERT ANY TABLE UPDATE ANY TABLE DELETE ANY TABLE</p> <p>EXECUTE ANY PROCEDURE</p>	<p>Utilizatorul ELEARN_APP_ADMIN incearca sa creeze o tabela in schema utilizatorului ELEARN_profesor1:</p> <p>CREATE TABLE ELEARN_profesor1.FEEDBACK (id number(6) primary key, mesaj varchar2(200), timp date);</p> <p>CREATE TABLE ELEARN_profesor1.FEEDBACK * ERROR at line 1: ORA-01031: insufficient privileges</p> <p>SYS AS SYSDBA ii acorda privilegiul de a crea tabele utilizatorului ELEARN_APP_ADMIN in schema oricarui utilizator:</p> <p>GRANT CREATE ANY TABLE TO ELEARN_APP_ADMIN;</p> <p>Utilizatorul ELEARN_APP_ADMIN incearca din nou sa creeze tabela FEEDBACK in schema utilizatorului ELEARN_profesor1:</p> <p>--> TOT EROARE DE INSUFFICIENT PRIVILEGES , pentru ca avem un primary key --> are nevoie si de CREATE ANY INDEX</p> <p>SYS AS SYSDBA ii acorda privilegiul de a crea tabele si indecsi utilizatorului ELEARN_APP_ADMIN in schema oricarui utilizator:</p> <p>GRANT CREATE ANY TABLE TO ELEARN_APP_ADMIN; GRANT CREATE ANY INDEX TO ELEARN_APP_ADMIN;</p> <p>Utilizatorul ELEARN_APP_ADMIN incearca din nou sa creeze tabela FEEDBACK in schema utilizatorului ELEARN_profesor1:</p> <p>SQL> CREATE TABLE ELEARN_profesor1.FEEDBACK</p> <p>Table created.</p> <p>De cercetat cine apare a fi owner al tablei ELEARN_profesor1.FEEDBACK :</p> <p>select owner, object_name from all_objects where owner like '%ELEARN%';</p> <table><tr><th>OWNER</th><th>OBJECT_NAME</th></tr><tr><td>ELEARN_APP_ADMIN</td><td>CURS</td></tr><tr><td>ELEARN_APP_ADMIN</td><td>SYS_C0011548</td></tr><tr><td>ELEARN_PROFESOR1</td><td>FEEDBACK</td></tr><tr><td>ELEARN_PROFESOR1</td><td>SYS_C0011559</td></tr></table> <p>Morala: proprietar va fi userul in schema căruia este creat obiectul, indiferent cine il creaza.</p>	OWNER	OBJECT_NAME	ELEARN_APP_ADMIN	CURS	ELEARN_APP_ADMIN	SYS_C0011548	ELEARN_PROFESOR1	FEEDBACK	ELEARN_PROFESOR1	SYS_C0011559
OWNER	OBJECT_NAME												
ELEARN_APP_ADMIN	CURS												
ELEARN_APP_ADMIN	SYS_C0011548												
ELEARN_PROFESOR1	FEEDBACK												
ELEARN_PROFESOR1	SYS_C0011559												

			<p>De cercetat dintre ELEARN_profesor1 si ELEARN_APP_ADMIN, ce operatii sunt finalizate cu succes in cazul fiecaruia (La acest moment utilizatorul ELEARN_profesor1 are doar privilegiul CREATE SESSION):</p> <p>ALTER TABLE ELEARN_profesor1.FEEDBACK drop column timp; --ii reuseste doar lui ELEARN_profesor1</p> <p>DROP TABLE ELEARN_profesor1.FEEDBACK ; ----ii reuseste doar lui ELEARN_profesor1 -- apoi o recreaza ELEARN_APP_ADMIN</p> <p>INSERT INTO ELEARN_profesor1.FEEDBACK VALUES (1, 'materia este foarte interesanta si utila', SYSDATE); -- ii reuseste doar lui ELEARN_profesor1</p> <p>Morala: chiar daca un utilizator X nu are privilegiile explicite pentru operatii LDD, LMD nici macar pentru schema sa proprie, daca alt utilizator Y ii creaza un obiect in schema , atunci asupra obiectului respectiv utilizatorul X va putea face cu succes operatii LMD si LDD.</p>
2. Privilegii asupra obiectelor schemei	<p>Au efect in cadrul unui anumit obiect numit al unei scheme a bazei de date.</p> <p>Aplicați asupra sinonimului unui obiect au același efect cu a fi aplicați direct pe obiect.</p> <p>Privilegiul ALL asupra unui obiect dintr-o schema da drepturi depline asupra lui.</p>	<p>2.1) Privilegii tabel</p> <p>LMD</p> <p>SELECT ON schema.tabela INSERT ON schema.tabela UPDATE ON schema.tabela DELETE ON schema.tabela</p> <p>LDD</p> <p>ALTER ON schema.tabela INDEX ON schema.tabela</p>	<p>Pe baza <i>matricii entitate –utilizator</i> , se dau următoarele privilegii asupra obiectelor schemei: de select,insert si update pe tabela ELEARN_APP_ADMIN.CURS către utilizatorul ELEARN_profesor1.</p> <p>GRANT SELECT,INSERT,UPDATE ON ELEARN_APP_ADMIN.CURS TO ELEARN_profesor1;</p> <p>Testam, conectati ca ELEARN_profesor1: INSERT INTO ELEARN_APP_ADMIN.CURS VALUES (2,'RETELE DE CALCULATOARE',3,2,5,'E',28,28,0);</p> <p>SQL> INSERT INTO ELEARN_APP_ADMIN.CURS VALUES 1 row created.</p> <p>Am vazut mai sus(pct 1.3) ca esueaza comanda data de ELEARN_APP_ADMIN: ALTER TABLE ELEARN_profesor1.FEEDBACK drop column timp</p> <p>Li acordam acest privilegiu de catre SYS/ AS SYSDBA: GRANT ALTER ON ELEARN_profesor1.FEEDBACK TO ELEARN_APP_ADMIN;</p> <p>Testam: Mai întâi, ELEARN_profesor1 face DELETE FROM FEEDBACK (altfel da eroare de resource busy pentru ca nu pot șterge o coloana care are date) si COMMIT Apoi, ELEARN_APP_ADMIN reda comanda de ALTER TABLE, cu succes Apoi, ELEARN_APP_ADMIN da urmatoarele comenzi cu succes:</p>

			ALTER TABLE ELEARN_profesor1.FEEDBACK add cod_student NUMBER(4); ALTER TABLE ELEARN_profesor1.FEEDBACK add timp date;
		2.2) Privilegii vizualizare SELECT ON schema.viz INSERT ON schema.viz UPDATE ON schema.viz DELETE ON schema.viz	<p>Propunem următorul scenariu drept aplicatie:</p> <p>SYS AS SYSDBA acorda privilegiile urmatoare utilizatorului ELEARN_APP_ADMIN: GRANT CREATE ANY TABLE TO ELEARN_APP_ADMIN; GRANT CREATE ANY INDEX TO ELEARN_APP_ADMIN;</p> <p>Utilizatorul ELEARN_APP_ADMIN creaza tabelele: CREATE TABLE ELEARN_profesor1.FEEDBACK (id number(6) primary key, mesaj varchar2(200), cod_student number(4), timp date); CREATE TABLE ELEARN_profesor2.FEEDBACK (id number(6) primary key, mesaj varchar2(200), cod_student number(4), timp date); CREATE TABLE ELEARN_asistent3.FEEDBACK (id number(6) primary key, mesaj varchar2(200), cod_student number(4), timp date);</p> <p>Un student poate insera feedback pentru profesorii care i-au predat la clasa. Proiectantul aplicatiei de e-learning a prevazut o vizualizare prin intermediul careia sa se realizeze inserarile corespunzatoare in tabelele de baza, printr-un trigger instead of. Astfel, ELEARN_APP_ADMIN dorește sa creeze vizualizarea VIZ_FEEDB:</p> <pre>CREATE OR REPLACE VIEW VIZ_FEEDB AS SELECT MESAJ,COD_STUDENT,'PROF1' AS cod_prof FROM ELEARN_profesor1.FEEDBACK UNION SELECT MESAJ,COD_STUDENT,'PROF2' AS cod_prof FROM ELEARN_profesor2.FEEDBACK UNION SELECT MESAJ,COD_STUDENT,'ASIST3' AS cod_prof FROM ELEARN_asistent3.FEEDBACK;</pre> <p>Initial ELEARN_APP_ADMIN va primi eroare de prrilegii insuficiente.</p>

			<p>Pentru a rezolva aceasta problema, SYS/AS SYSDBA ii acorda urmatoarele privilegii:</p> <ul style="list-style-type: none"> • privilegiu sistem de creare de view in schema proprie (1.1): GRANT CREATE VIEW TO ELEARN_APP_ADMIN ; ==> Nu este suficient, continua sa primeasca eroare de privilegii insuficiente • privilegiu obiect de select pe fiecare din tabelele ce intervin in cererea din vizualizare, with grant option: GRANT SELECT ON ELEARN_profesor1.FEEDBACK TO ELEARN_APP_ADMIN WITH GRANT OPTION; GRANT SELECT ON ELEARN_profesor2.FEEDBACK TO ELEARN_APP_ADMIN WITH GRANT OPTION; GRANT SELECT ON ELEARN_asistent3.FEEDBACK TO ELEARN_APP_ADMIN WITH GRANT OPTION; ==> acum vizualizarea este creata cu succes de catre ELEARN_APP_ADMIN . <p>Ii dam unui student privilegiul de a interoga vizualizarea: GRANT SELECT ON ELEARN_APP_ADMIN.VIZ_FEEDB TO ELEARN_student1;</p> <p>Studentul interogheaza vizualizarea cu succes:</p> <p>SELECT * FROM ELEARN_APP_ADMIN.VIZ_FEEDB WHERE cod_student=1;</p> <p>In ceea ce privește inserarea de date in tabelele de baza prin intermediul vizualizarii: * intrucat vizualizarea contine operatorul UNION vizualizarea este complexa → nu se pot insera date direct prin view, ci este necesar un trigger INSTEAD OF</p> <p>Este necesar ca utilizatorul ELEARN_APP_ADMIN sa poata crea triggeri in schema proprie. SYS AS SYSDBA ii da privilegiul: GRANT CREATE TRIGGER TO ELEARN_APP_ADMIN;</p> <p>In plus, ELEARN_APP_ADMIN trebuie sa primească privilegii de a insera in tabelele de baza ale vizualizării, cu grant option: GRANT INSERT ON ELEARN_profesor1.FEEDBACK TO ELEARN_APP_ADMIN WITH GRANT OPTION; GRANT INSERT ON ELEARN_profesor2.FEEDBACK TO ELEARN_APP_ADMIN WITH GRANT OPTION; GRANT INSERT ON ELEARN_asistent3.FEEDBACK TO ELEARN_APP_ADMIN WITH GRANT OPTION;</p> <p>ELEARN_APP_ADMIN creeaza un trigger INSTEAD OF astfel:</p>
--	--	--	--

			<pre>CREATE OR REPLACE TRIGGER TR_FEEDB INSTEAD OF INSERT ON VIZ_FEEDB FOR EACH ROW BEGIN IF :NEW.COD_PROF='PROF1' THEN INSERT INTO ELEARN_profesor1.FEEDBACK VALUES(SYSDATE- TO_DATE('2000-01-01', 'yyyy-mm-dd'),:NEW.MESAJ,101,SYSDATE); END IF; IF :NEW.COD_PROF='PROF2' THEN INSERT INTO ELEARN_profesor2.FEEDBACK VALUES(SYSDATE- TO_DATE('2000-01-01', 'yyyy-mm-dd'),:NEW.MESAJ,101,SYSDATE); END IF; IF :NEW.COD_PROF='ASIST3' THEN INSERT INTO ELEARN_asistent3.FEEDBACK VALUES(SYSDATE- TO_DATE('2000-01-01', 'yyyy-mm-dd'),:NEW.MESAJ,101,SYSDATE); END IF; END; / Trigger created. Acum SYS AS SYSDBA ii acorda privilegiul de insert prin view catre studentul ELEARN_student1: GRANT INSERT ON ELEARN_APP_ADMIN.VIZ_FEEDB TO ELEARN_student1; Studentul executa următorul insert: INSERT INTO ELEARN_APP_ADMIN.VIZ_FEEDB VALUES('UN CURS INTERESANT',101,'PROF1'); SQL> INSERT INTO ELEARN_APP_ADMIN.VIZ_FEEDB VALUES 1 row created. COMMIT; -- neaparat ! Verifica profesorul ELEARN_profesor1 daca a primit vreun feedback: SELECT ID, SUBSTR(MESAJ,1,20) MESAJ, COD_STUDENT,TIMP FROM FEEDBACK; SQL> SELECT ID, SUBSTR(MESAJ,1,20) MESAJ, COD_STUDENT,TIMP FROM FEEDBACK; <table><thead><tr><th>ID</th><th>MESAJ</th><th>COD_STUDENT</th><th>TIMP</th></tr></thead><tbody><tr><td>4620</td><td>UN CURS INTERESANT</td><td>101</td><td>24-AUG-12</td></tr></tbody></table></pre>	ID	MESAJ	COD_STUDENT	TIMP	4620	UN CURS INTERESANT	101	24-AUG-12
ID	MESAJ	COD_STUDENT	TIMP								
4620	UN CURS INTERESANT	101	24-AUG-12								

		<p>2.3) Privilegii procedura</p> <p>EXECUTE ON schema.proc</p>	<p>Exista doua situatii: Situatia 1) ca si la view si la triggerul create anterior, creatorul procedurii primeste drept de a crea o procedura in schema proprie si in plus el primeste privilegii corespunzatoare pe obiectele prelucrate in interiorul procedurii, WITH GRANT OPTION. In acest caz, apelantul are nevoie doar de privilegiul execute asupra procedurii.</p> <p>Astfel, ELEARN_APP_ADMIN dorește sa creeze procedura DELETE_SPAM care sa stearga comentariile de tip SPAM din tabela de FEEDBACK a profesorului. Ca input procedura primește minim 1, maxim 3 cuvinte cheie de SPAM ce vor fi cautate in mesaje:</p> <p>CREATE OR REPLACE PROCEDURE DELETE_SPAM(key1 VARCHAR2, key2 VARCHAR2 default 'reclama', key3 VARCHAR2 default 'publicitate') AS</p> <p>BEGIN DELETE FROM ELEARN_profesor1.FEEDBACK WHERE MESAJ LIKE '% key1 '%' OR MESAJ LIKE '% key2 '%' OR MESAJ LIKE '% key3 '%; DBMS_OUTPUT.PUT_LINE('Au fost sterse: SQL%ROWCOUNT ' mesaje de tip spam din tabela profesorului 1'); COMMIT; END; /</p> <p>CREATE OR REPLACE PROCEDURE DELETE_SPAM(licitate'> DELETE_SPAM AS * ERROR at line 1: ORA-01031: insufficient privileges</p> <p>Pentru a rezolva aceasta problema, SYS/AS SYSDBA ii acorda urmatoarele privilegii lui ELEARN_APP_ADMIN:</p> <ul style="list-style-type: none"> * privilegiu de a crea proceduri in schema proprie: GRANT CREATE PROCEDURE TO ELEARN_APP_ADMIN; * privilegii asupra obiectelor prelucrate de procedura, CU GRANT OPTION: GRANT DELETE ON ELEARN_profesor1.FEEDBACK TO ELEARN_APP_ADMIN WITH GRANT OPTION; <p>Acum , ELEARN_APP_ADMIN va reuși sa creeze procedura cu succes. Mai mult, utilizatorul ELEARN_APP_ADMIN poate executa procedura cu succes:</p>
--	--	--	---

			<pre>SQL> SET SERVEROUTPUT ON SQL> EXEC DELETE_SPAM('AVANTAJOS');</pre> <p>Au fost sterse:0 mesaje de tip spam din tabela profesorului 1</p> <p>PL/SQL procedure successfully completed.</p> <p>Presupunem ca proiectantul aplicației a stabilit ca acest task de ștergere a mesajelor de spam sa poată fi realizat si de un asistent. Cand incearca sa execute, primește eroare:</p> <pre>SQL> EXEC ELEARN_APP_ADMIN.DELETE_SPAM('AVANTAJOS'); BEGIN ELEARN_APP_ADMIN.DELETE_SPAM('AVANTAJOS'); END;</pre> <p style="text-align: center;">*</p> <pre>ERROR at line 1: ORA-06550: line 1, column 7: PLS-00201: identifier 'ELEARN_APP_ADMIN.DELETE_SPAM' must be declared ORA-06550: line 1, column 7: PL/SQL: Statement ignored</pre> <p>Prin urmare, asistentul ELEARN_asistent3 primește de la SYS AS SYSDBA următoarele privilegii:</p> <p>GRANT EXECUTE ON ELEARN_APP_ADMIN.DELETE_SPAM TO ELEARN_asistent3;</p> <p>Acum, si asistentul va putea executa cu succes procedura.</p> <pre>EXEC ELEARN_APP_ADMIN.DELETE_SPAM('AVANTAJOS');</pre> <pre>SQL> SET SERVEROUTPUT ON SQL> EXEC ELEARN_APP_ADMIN.DELETE_SPAM('AVANTAJOS');</pre> <p>Au fost sterse:0 mesaje de tip spam din tabela profesorului 1</p> <p>PL/SQL procedure successfully completed.</p> <p>Situatia 2) in cazul in care creatorul procedurii are privilegii corespunzatoare pe obiectele prelucrate in interiorul procedurii, dar fara grant option, ATUNCI apelantul va avea nevoie sa aiba el insusi drepturile respective, altfel neputand executa cu succes procedura.</p>
--	--	--	--

Recapitularea situațiilor întâlnite in exemple:

userul X creeaza un obiect de tip view (trigger, procedura)

<i>in schema proprie</i>			<i>in schema altui user Y</i>		
<i>accesseaza obiecte din schema proprie</i>	<i>accesseaza obiecte din schema lui Y (select Y.D, insert Y.D)</i>		<i>accesseaza obiecte din schema proprie</i>	<i>accesseaza obiecte din schema lui Y (select Y.D, insert Y.D)</i>	
<i>Ce privilegii are nevoie X</i>	CREATE VIEW	CREATE VIEW	CREATE ANY VIEW	CREATE ANY VIEW	
	SELECT ON Y.D INSERT ON Y.D	SELECT ON Y.D WITH GRANT OPTION INSERT ON Y.D WITH GRANT OPTION		SELECT ON Y.D INSERT ON Y.D	SELECT ON Y.D WITH GRANT OPTION INSERT ON Y.D WITH GRANT OPTION
<i>Ce privilegii are nevoie apelantul Z</i>	SELECT ON VIZ INSERT ON VIZ	SELECT ON VIZ INSERT ON VIZ	SELECT ON VIZ INSERT ON VIZ	SELECT ON VIZ INSERT ON VIZ	SELECT ON VIZ INSERT ON VIZ
		SELECT ON Y.D INSERT ON Y.D		SELECT ON Y.D INSERT ON Y.D	