DATABASE SYSTEMS

ACCESS CONTROL

By Sana Faiz
Sana.faiz.muet83@gmail.com

PRIVILEGES

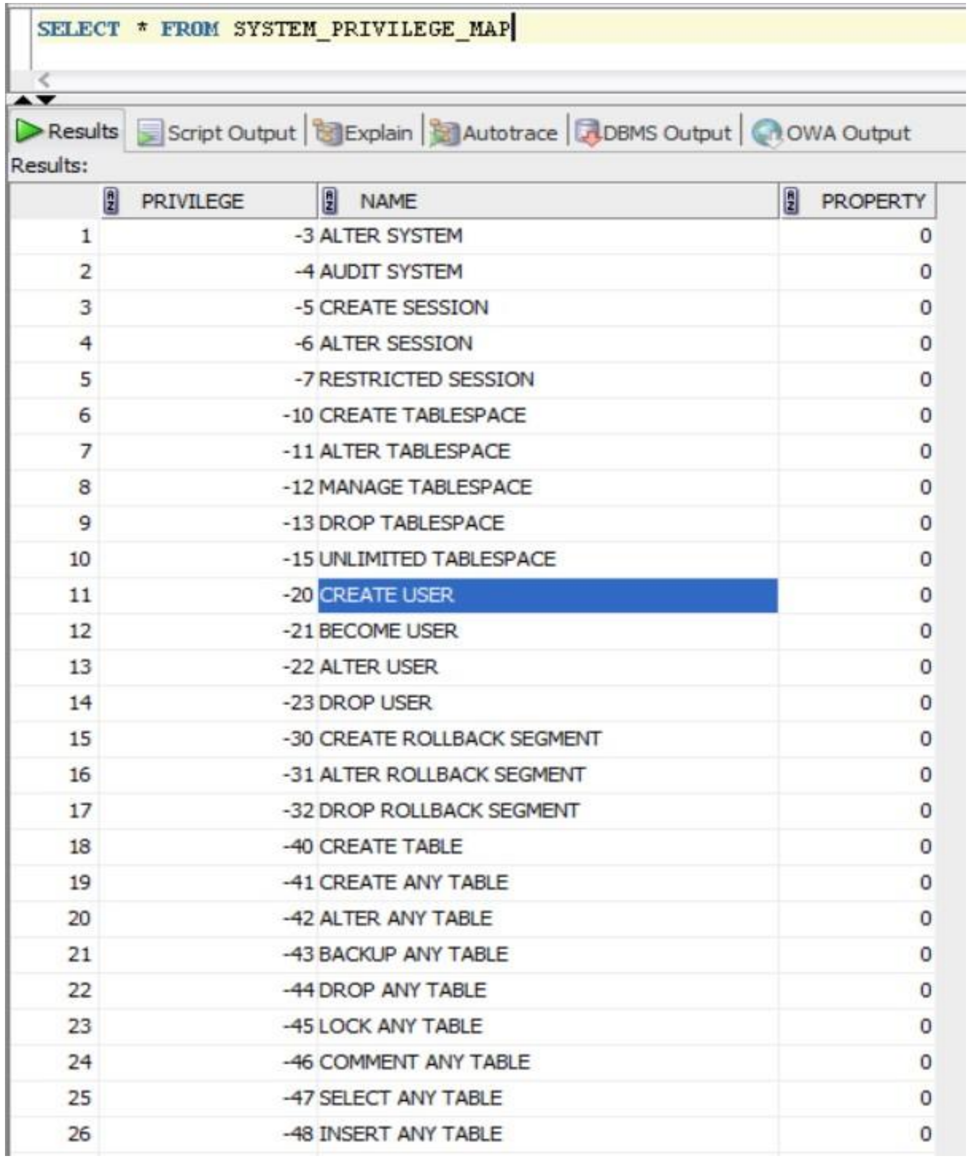
- A privilege is a right to execute a particular type of SQL statement or to access another user's object.
- One should grant privileges to users so that they can accomplish tasks required for their jobs. But privileges should only be given to a user who requires it to accomplish the necessary work. Excessive granting of unnecessary privileges can compromise security.
- There are two types of privileges:

1. **SYSTEM PRIVILEGES.**
2. **OBJECT PRIVILEGES.**

SYSTEM PRIVILEGES

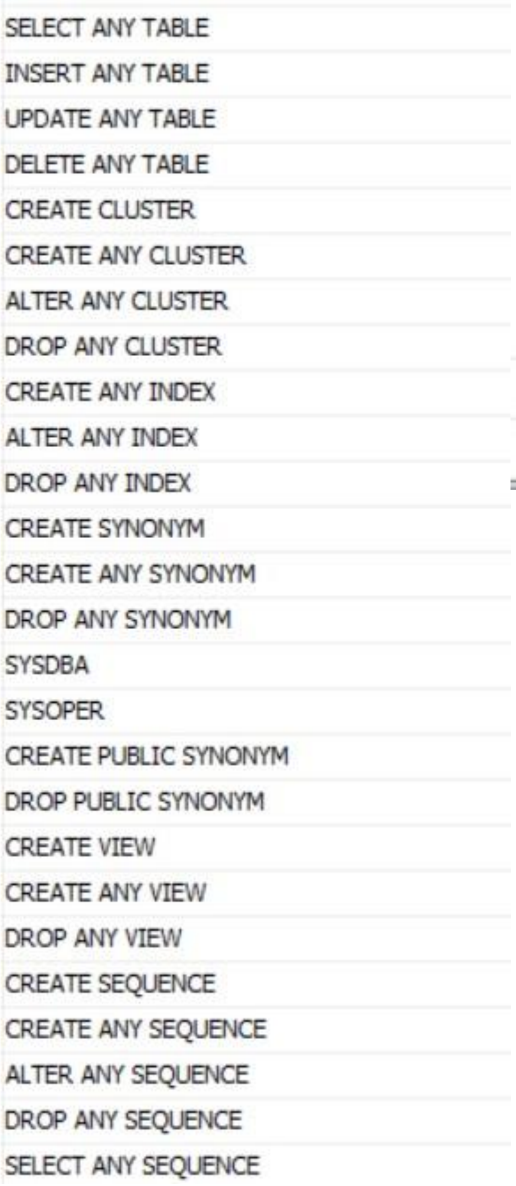
- A system privilege is the right to perform a particular action, or to perform an action on any schema objects of a particular type.
- There are over 200 distinct system privileges to manage.
- Each system privilege allows a user to perform a particular database operation or class of database operations.
- All privileges associated with creation, modification and deletion of DB objects fall under this category.
- In general, you grant system privileges only to administrative personnel and application developers. End users normally do not require and should not have the associated capabilities.
- Remember that system privileges are very powerful. Only grant them when necessary, to roles and trusted users of the database.
- To find the system privileges that have been granted to a user, one can query the DBA_SYS_PRIVS data dictionary view.

LIST OF SYSTEM PRIVILEGES

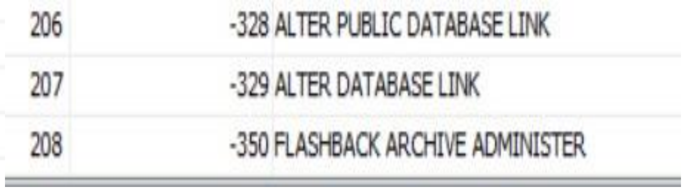


```
SELECT * FROM SYSTEM_PRIVILEGE_MAP
```

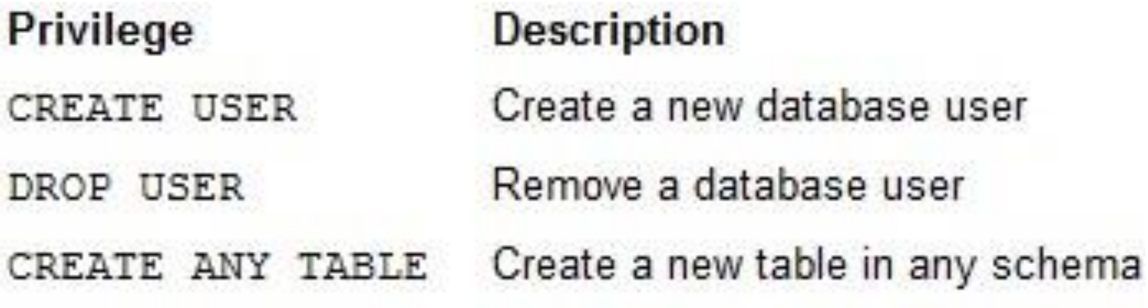
	PRIVILEGE	NAME	PROPERTY
1	-3	ALTER SYSTEM	0
2	-4	AUDIT SYSTEM	0
3	-5	CREATE SESSION	0
4	-6	ALTER SESSION	0
5	-7	RESTRICTED SESSION	0
6	-10	CREATE TABLESPACE	0
7	-11	ALTER TABLESPACE	0
8	-12	MANAGE TABLESPACE	0
9	-13	DROP TABLESPACE	0
10	-15	UNLIMITED TABLESPACE	0
11	-20	CREATE USER	0
12	-21	BECOME USER	0
13	-22	ALTER USER	0
14	-23	DROP USER	0
15	-30	CREATE ROLLBACK SEGMENT	0
16	-31	ALTER ROLLBACK SEGMENT	0
17	-32	DROP ROLLBACK SEGMENT	0
18	-40	CREATE TABLE	0
19	-41	CREATE ANY TABLE	0
20	-42	ALTER ANY TABLE	0
21	-43	BACKUP ANY TABLE	0
22	-44	DROP ANY TABLE	0
23	-45	LOCK ANY TABLE	0
24	-46	COMMENT ANY TABLE	0
25	-47	SELECT ANY TABLE	0
26	-48	INSERT ANY TABLE	0



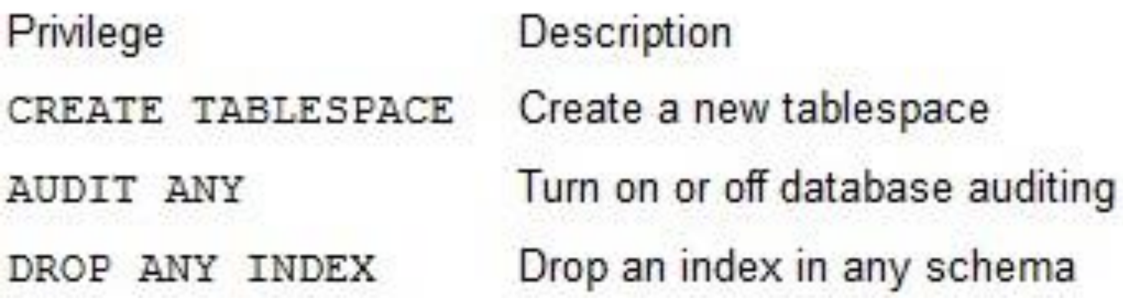
SELECT ANY TABLE
INSERT ANY TABLE
UPDATE ANY TABLE
DELETE ANY TABLE
CREATE CLUSTER
CREATE ANY CLUSTER
ALTER ANY CLUSTER
DROP ANY CLUSTER
CREATE ANY INDEX
ALTER ANY INDEX
DROP ANY INDEX
CREATE SYNONYM
CREATE ANY SYNONYM
DROP ANY SYNONYM
SYSDBA
SYSOPER
CREATE PUBLIC SYNONYM
DROP PUBLIC SYNONYM
CREATE VIEW
CREATE ANY VIEW
DROP ANY VIEW
CREATE SEQUENCE
CREATE ANY SEQUENCE
ALTER ANY SEQUENCE
DROP ANY SEQUENCE
SELECT ANY SEQUENCE



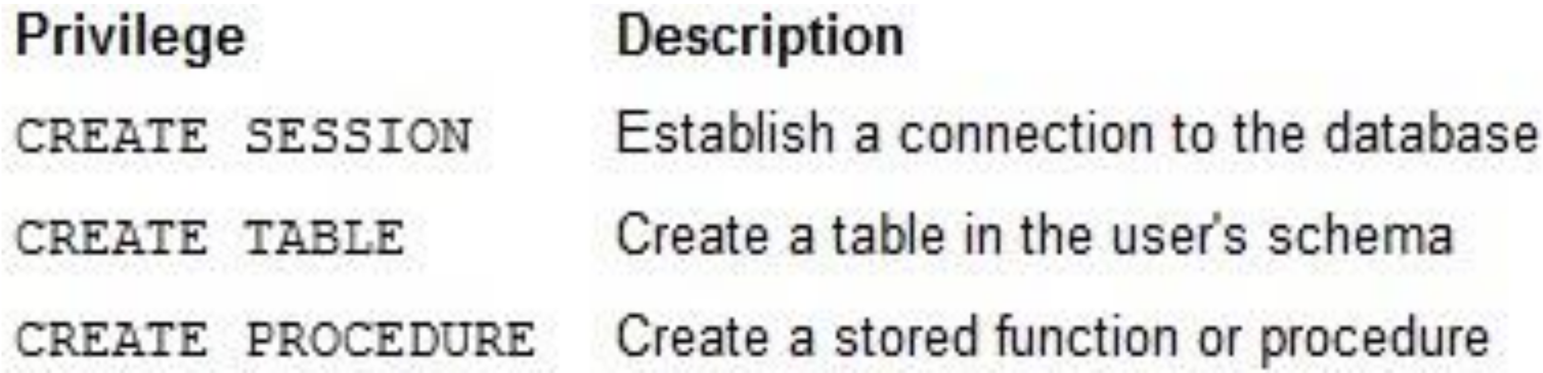
206	-328	ALTER PUBLIC DATABASE LINK
207	-329	ALTER DATABASE LINK
208	-350	FLASHBACK ARCHIVE ADMINISTER



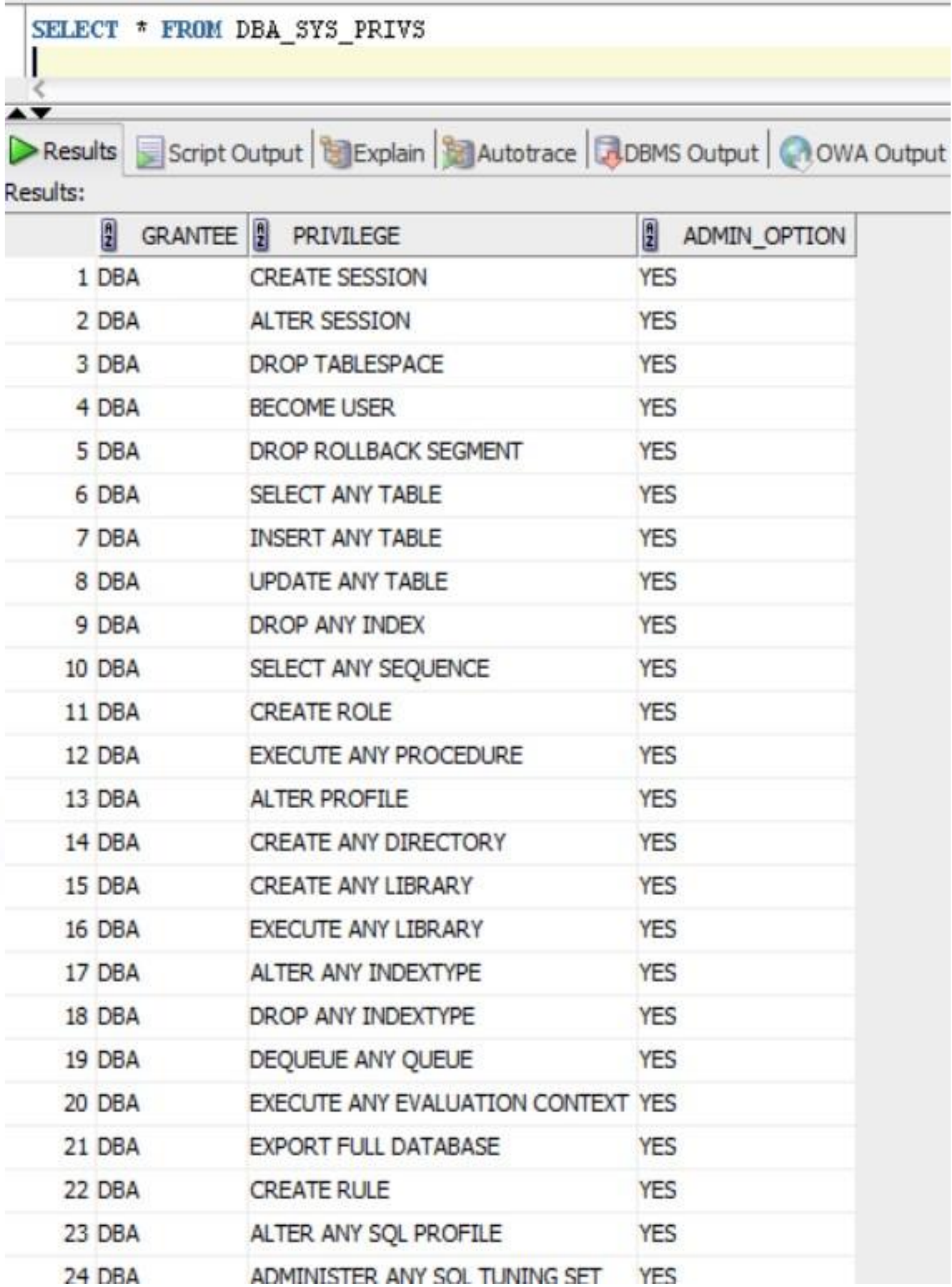
Privilege	Description
CREATE USER	Create a new database user
DROP USER	Remove a database user
CREATE ANY TABLE	Create a new table in any schema



Privilege	Description
CREATE TABLESPACE	Create a new tablespace
AUDIT ANY	Turn on or off database auditing
DROP ANY INDEX	Drop an index in any schema



Privilege	Description
CREATE SESSION	Establish a connection to the database
CREATE TABLE	Create a table in the user's schema
CREATE PROCEDURE	Create a stored function or procedure



```
SELECT * FROM DBA_SYS_PRIVS
```

Results:

	GRANTEE	PRIVILEGE	ADMIN_OPTION
1	DBA	CREATE SESSION	YES
2	DBA	ALTER SESSION	YES
3	DBA	DROP TABLESPACE	YES
4	DBA	BECOME USER	YES
5	DBA	DROP ROLLBACK SEGMENT	YES
6	DBA	SELECT ANY TABLE	YES
7	DBA	INSERT ANY TABLE	YES
8	DBA	UPDATE ANY TABLE	YES
9	DBA	DROP ANY INDEX	YES
10	DBA	SELECT ANY SEQUENCE	YES
11	DBA	CREATE ROLE	YES
12	DBA	EXECUTE ANY PROCEDURE	YES
13	DBA	ALTER PROFILE	YES
14	DBA	CREATE ANY DIRECTORY	YES
15	DBA	CREATE ANY LIBRARY	YES
16	DBA	EXECUTE ANY LIBRARY	YES
17	DBA	ALTER ANY INDEXTYPE	YES
18	DBA	DROP ANY INDEXTYPE	YES
19	DBA	DEQUEUE ANY QUEUE	YES
20	DBA	EXECUTE ANY EVALUATION CONTEXT	YES
21	DBA	EXPORT FULL DATABASE	YES
22	DBA	CREATE RULE	YES
23	DBA	ALTER ANY SQL PROFILE	YES
24	DBA	ADMINISTER ANY SQL TUNING SET	YES

GRANTING SYSTEM PRIVILEGES

- You can grant system privileges to users and roles.
- If you grant system privileges to roles, then you can use the roles to manage system privileges. For example, roles permit privileges to be made selectively available.

GRANTING SYSTEM PRIVILEGES

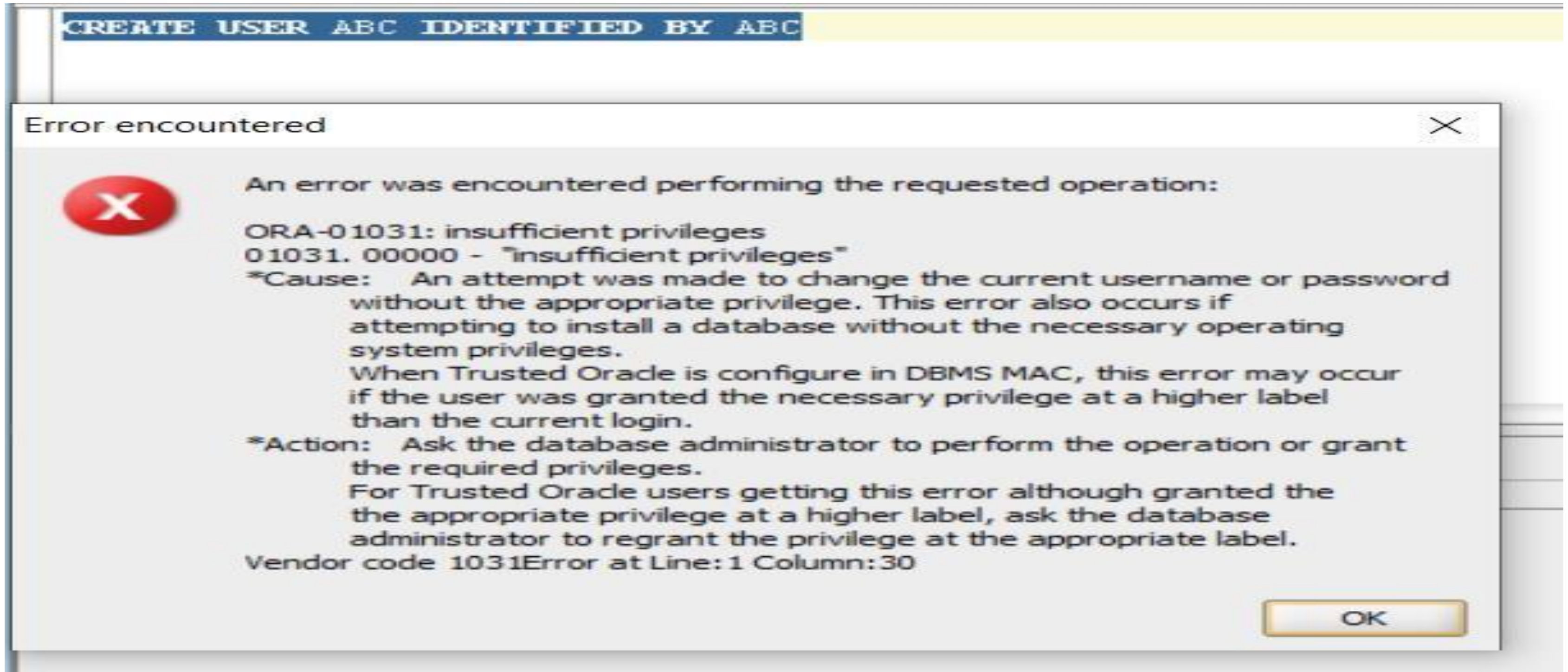
SYNTAX:

```
GRANT sys_privilege [, sys_privilege ...]
TO user [, user, role, PUBLIC ...]
[ WITH ADMIN OPTION ] ;
```

WHO CAN GRANT OR REVOKE SYSTEM PRIVILEGES?

- Only two types of users can grant system privileges to other users or revoke such privileges from them:

1. Users who have been granted a specific system privilege with the **ADMIN OPTION**.
2. Users with the system privilege **GRANT ANY PRIVILEGE**.



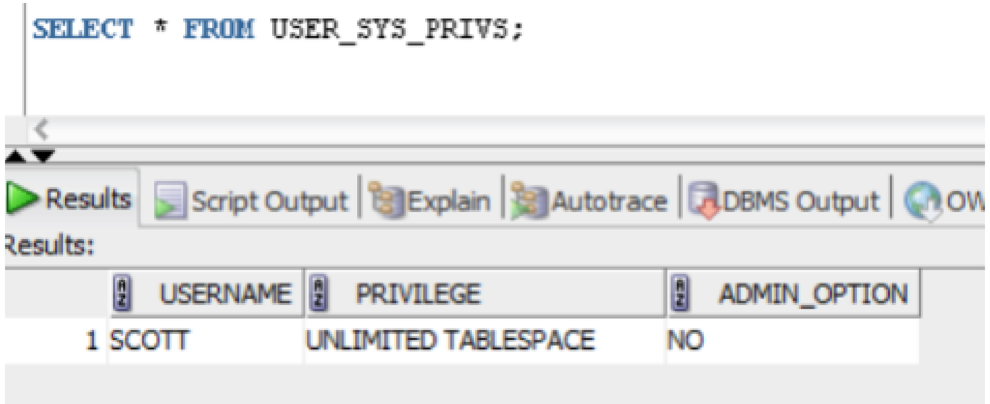
```
CREATE USER ABC IDENTIFIED BY ABC
```

Error encountered

```
An error was encountered performing the requested operation:

ORA-01031: insufficient privileges
01031. 00000 -  "insufficient privileges"
*Cause:    An attempt was made to change the current username or password
           without the appropriate privilege. This error also occurs if
           attempting to install a database without the necessary operating
           system privileges.
           When Trusted Oracle is configure in DBMS MAC, this error may occur
           if the user was granted the necessary privilege at a higher label
           than the current login.
*Action:   Ask the database administrator to perform the operation or grant
           the required privileges.
           For Trusted Oracle users getting this error although granted the
           the appropriate privilege at a higher label, ask the database
           administrator to regrant the privilege at the appropriate label.
Vendor code 1031Error at Line:1 Column:30
```

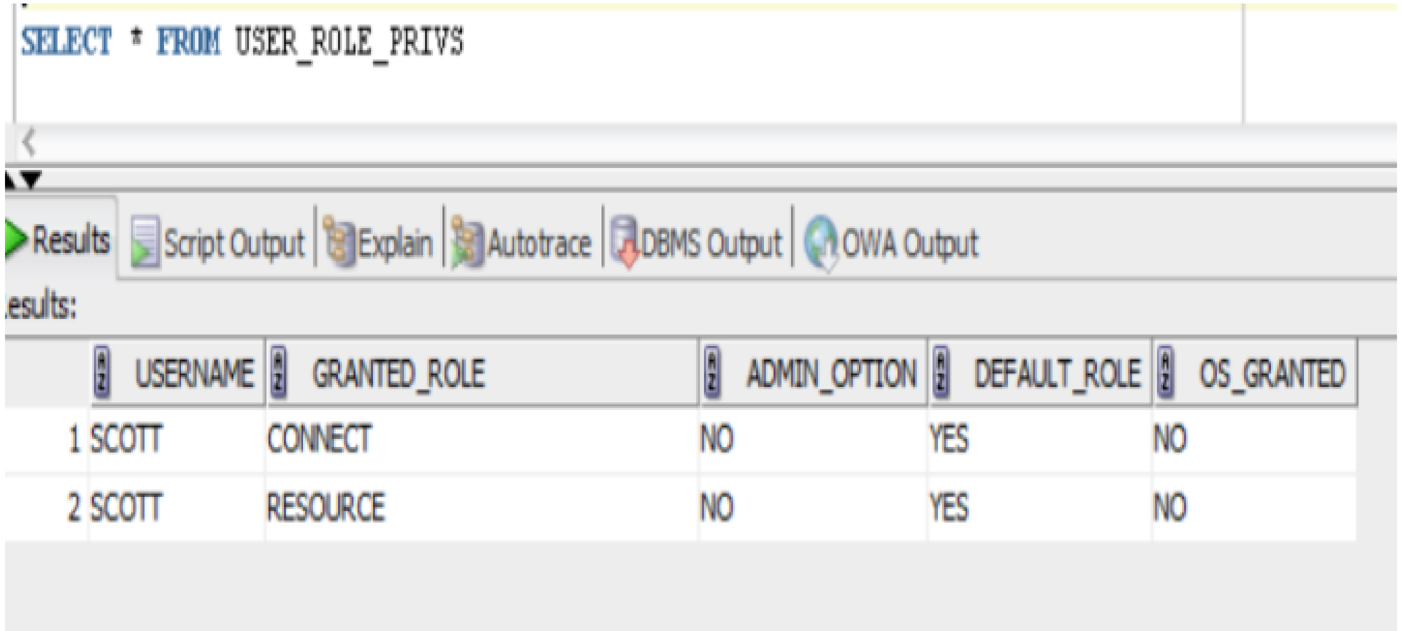
SCOTT'S PRIVILEGES



```
SELECT * FROM USER_SYS_PRIVS;
```

Results:

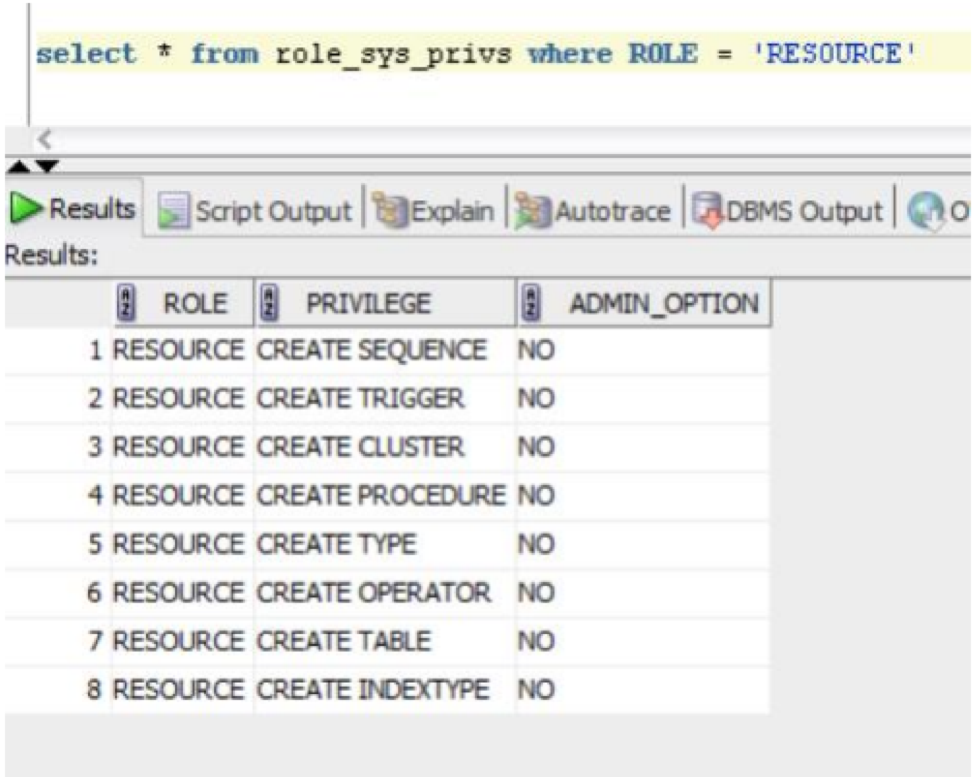
	USERNAME	PRIVILEGE	ADMIN_OPTION
1	SCOTT	UNLIMITED TABLESPACE	NO



```
SELECT * FROM USER_ROLE_PRIVS
```

esults:

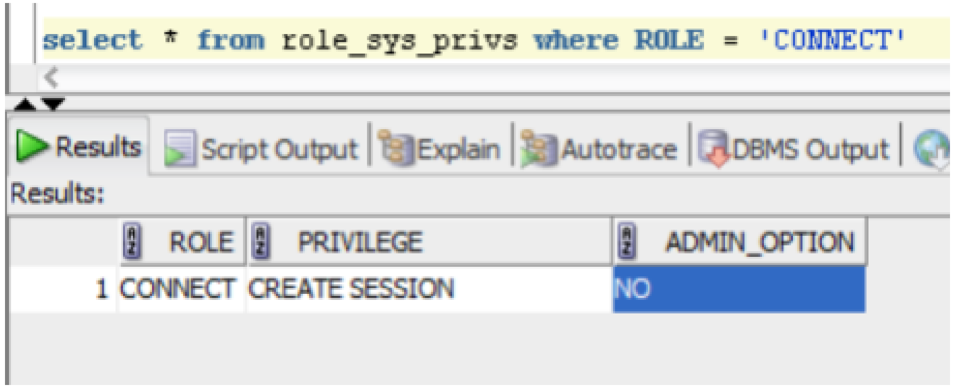
	USERNAME	GRANTED_ROLE	ADMIN_OPTION	DEFAULT_ROLE	OS_GRANTED
1	SCOTT	CONNECT	NO	YES	NO
2	SCOTT	RESOURCE	NO	YES	NO



```
select * from role_sys_privs where ROLE = 'RESOURCE'
```

Results:

	ROLE	PRIVILEGE	ADMIN_OPTION
1	RESOURCE	CREATE SEQUENCE	NO
2	RESOURCE	CREATE TRIGGER	NO
3	RESOURCE	CREATE CLUSTER	NO
4	RESOURCE	CREATE PROCEDURE	NO
5	RESOURCE	CREATE TYPE	NO
6	RESOURCE	CREATE OPERATOR	NO
7	RESOURCE	CREATE TABLE	NO
8	RESOURCE	CREATE INDEXTYPE	NO



```
select * from role_sys_privs where ROLE = 'CONNECT'
```

Results:

	ROLE	PRIVILEGE	ADMIN_OPTION
1	CONNECT	CREATE SESSION	NO

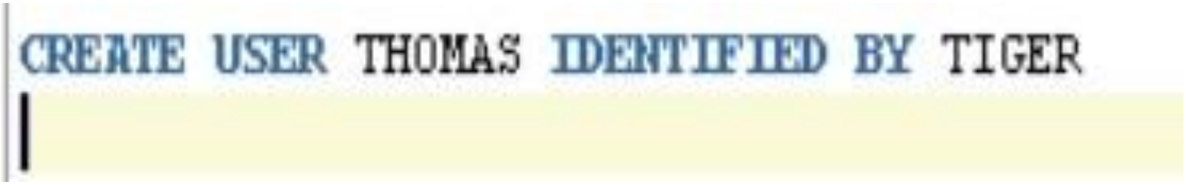
CREATING USERS

- CREATE USER statement is used to create and configure a database user, which is an account through which you can log in to the database, and to establish the means by which Oracle Database permits access by the user.
- For creating a user, one must have the CREATE USER privilege. By default, DBA can create a user.
- To log on to Oracle Database, a user must have the CREATE SESSION system privilege. Therefore, after creating a user, you should grant the user at least the CREATE SESSION system privilege.

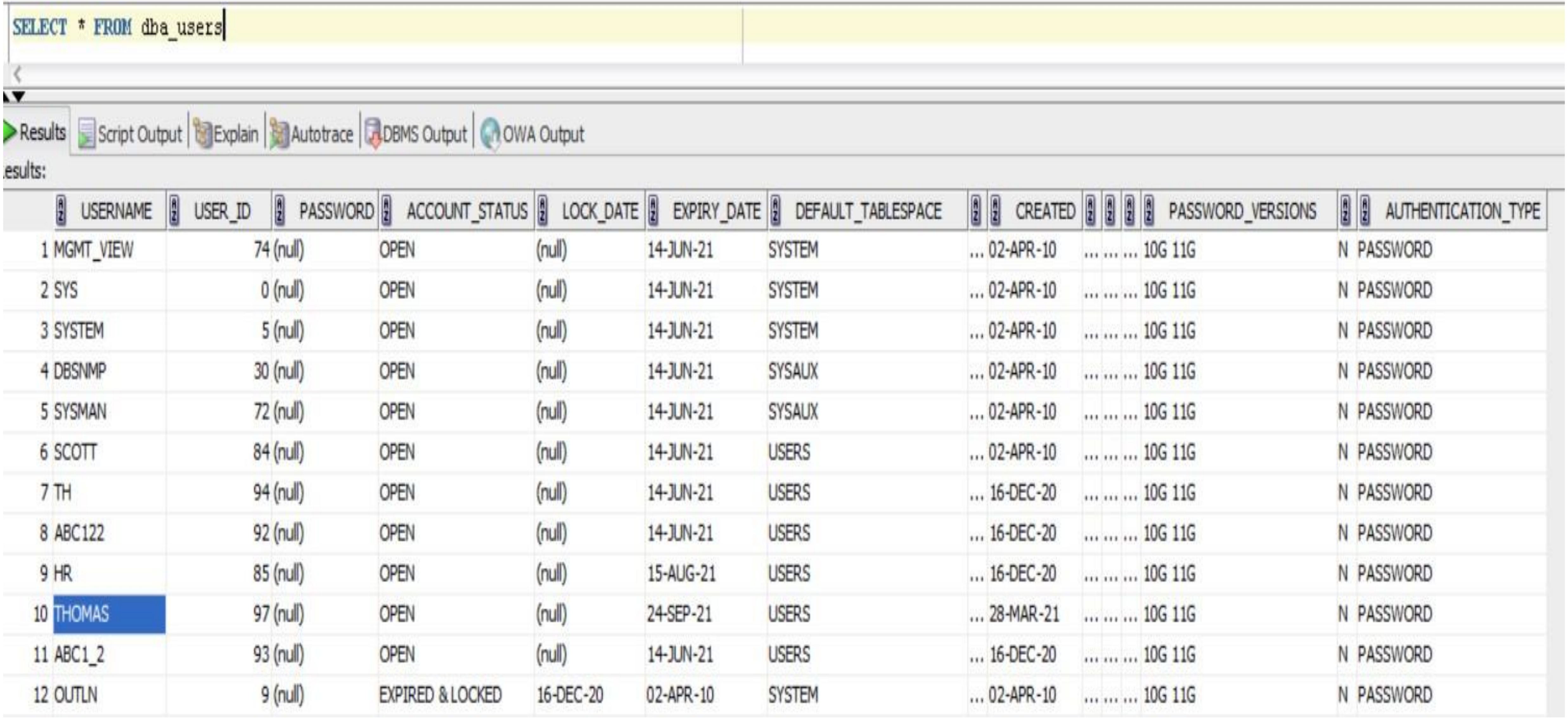
SYNTAX:

CREATE USER user-name

IDENTIFIED BY password [EXPIRE] ;



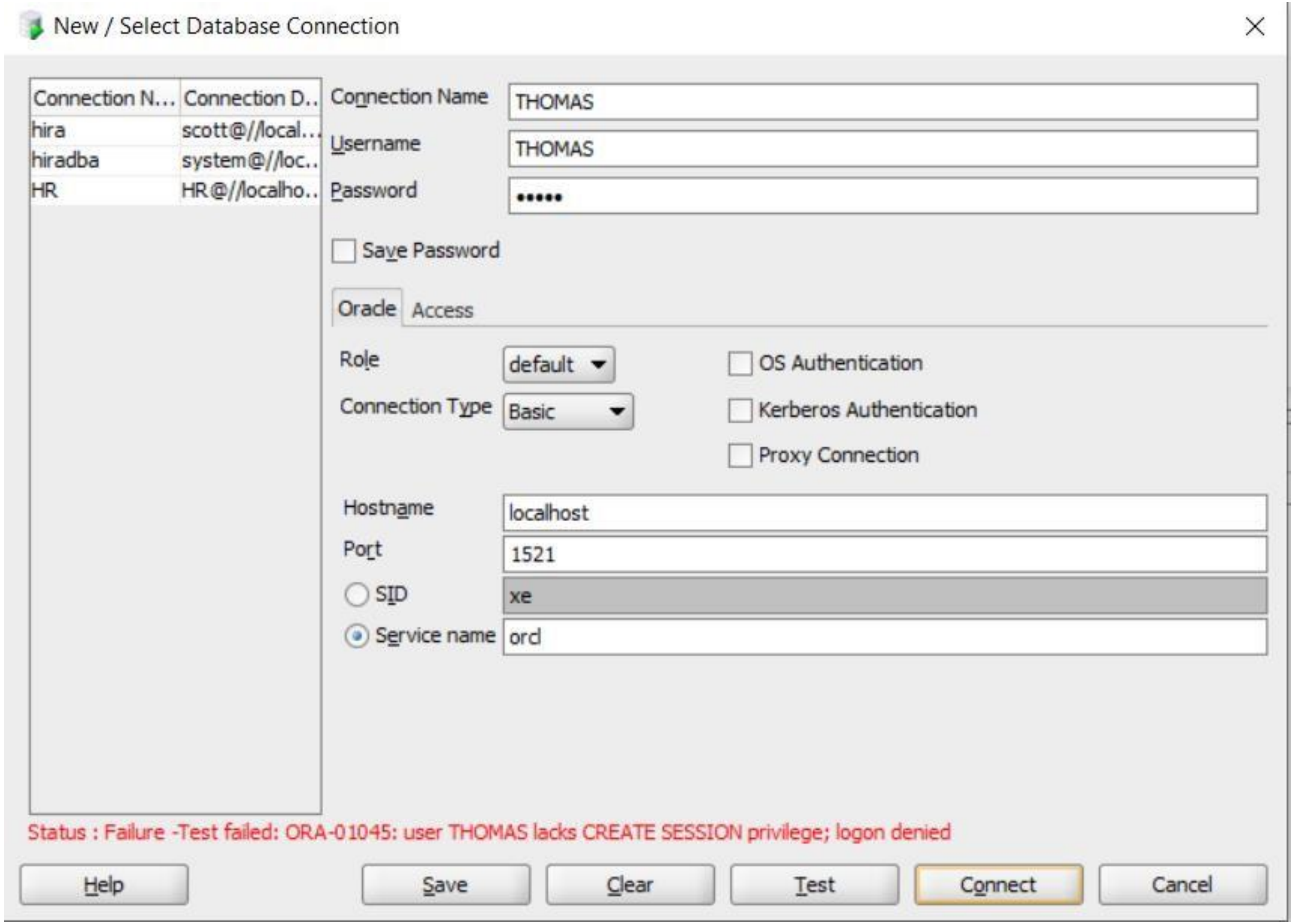
```
CREATE USER THOMAS IDENTIFIED BY TIGER
```



```
SELECT * FROM dba_users
```

esults:

	USERNAME	USER_ID	PASSWORD	ACCOUNT_STATUS	LOCK_DATE	EXPIRY_DATE	DEFAULT_TABLESPACE		CREATED				PASSWORD_VERSIONS		AUTHENTICATION_TYPE
1	MGMT_VIEW	74	(null)	OPEN	(null)	14-JUN-21	SYSTEM	...	02-APR-10	...	...	...	10G 11G	N	PASSWORD
2	SYS	0	(null)	OPEN	(null)	14-JUN-21	SYSTEM	...	02-APR-10	...	...	...	10G 11G	N	PASSWORD
3	SYSTEM	5	(null)	OPEN	(null)	14-JUN-21	SYSTEM	...	02-APR-10	...	...	...	10G 11G	N	PASSWORD
4	DBSNMP	30	(null)	OPEN	(null)	14-JUN-21	SYSAUX	...	02-APR-10	...	...	...	10G 11G	N	PASSWORD
5	SYSMAN	72	(null)	OPEN	(null)	14-JUN-21	SYSAUX	...	02-APR-10	...	...	...	10G 11G	N	PASSWORD
6	SCOTT	84	(null)	OPEN	(null)	14-JUN-21	USERS	...	02-APR-10	...	...	...	10G 11G	N	PASSWORD
7	TH	94	(null)	OPEN	(null)	14-JUN-21	USERS	...	16-DEC-20	...	...	...	10G 11G	N	PASSWORD
8	ABC122	92	(null)	OPEN	(null)	14-JUN-21	USERS	...	16-DEC-20	...	...	...	10G 11G	N	PASSWORD
9	HR	85	(null)	OPEN	(null)	15-AUG-21	USERS	...	16-DEC-20	...	...	...	10G 11G	N	PASSWORD
10	THOMAS	97	(null)	OPEN	(null)	24-SEP-21	USERS	...	28-MAR-21	...	...	...	10G 11G	N	PASSWORD
11	ABC1_2	93	(null)	OPEN	(null)	14-JUN-21	USERS	...	16-DEC-20	...	...	...	10G 11G	N	PASSWORD
12	OUTLN	9	(null)	EXPIRED & LOCKED	16-DEC-20	02-APR-10	SYSTEM	...	02-APR-10	...	...	...	10G 11G	N	PASSWORD



New / Select Database Connection

Connection N...	Connection D...
hira	scott@//local...
hiradba	system@//loc...
HR	HR@//localho...

Connection Name: THOMAS

Username: THOMAS

Password: •••••

□ Save Password

Role: default

Connection Type: Basic

□ OS Authentication

□ Kerberos Authentication

□ Proxy Connection

Hostname: localhost

Port: 1521

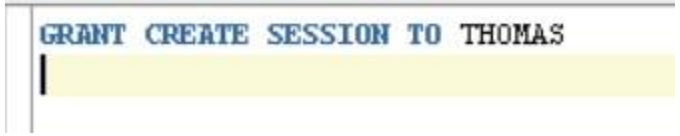
○ SID: xe

● Service name: orcl

Status : Failure -Test failed: ORA-01045: user THOMAS lacks CREATE SESSION privilege; logon denied

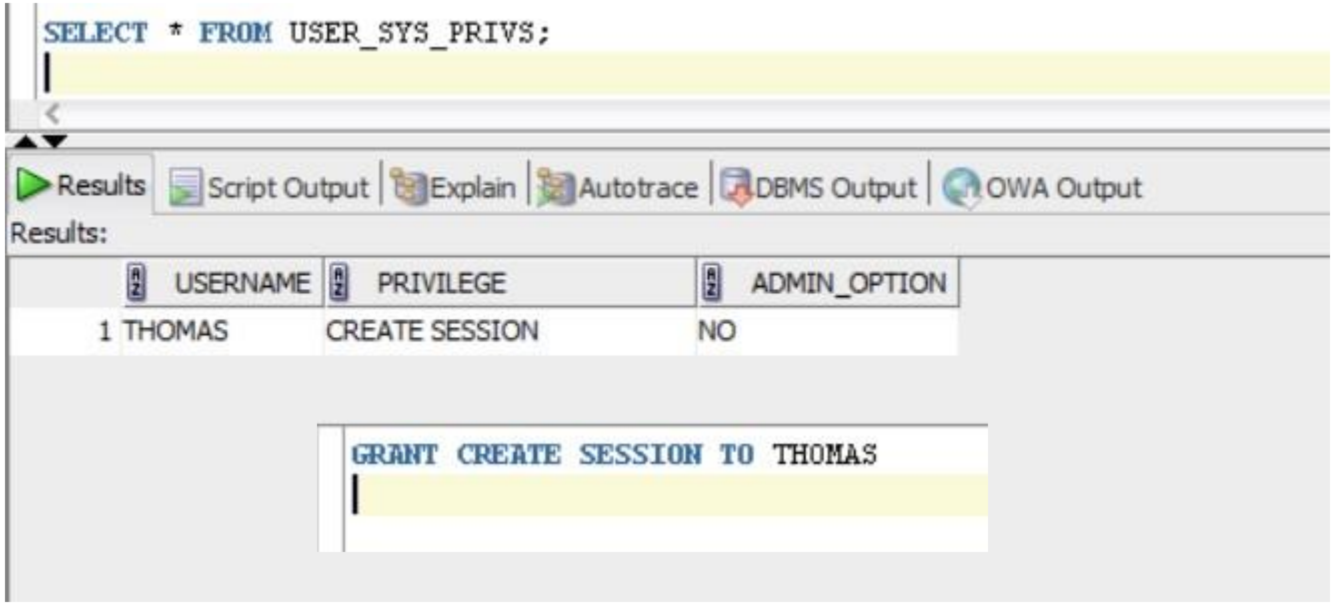
SYNTAX:

GRANT sys_privilege [, sys_privilege ...]
TO user [, user, role, PUBLIC ...]
[WITH ADMIN OPTION] ;



```
GRANT CREATE SESSION TO THOMAS
```

THOMAS'S PRIVILEGES

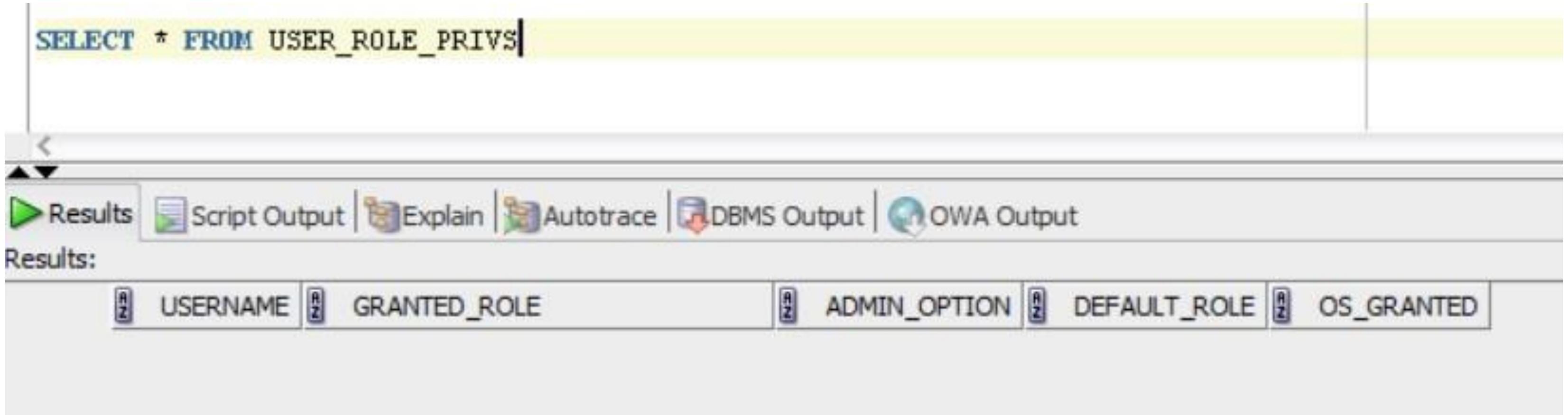


```
SELECT * FROM USER_SYS_PRIVS;
```

Results:

	USERNAME	PRIVILEGE	ADMIN_OPTION
1	THOMAS	CREATE SESSION	NO

```
GRANT CREATE SESSION TO THOMAS
```



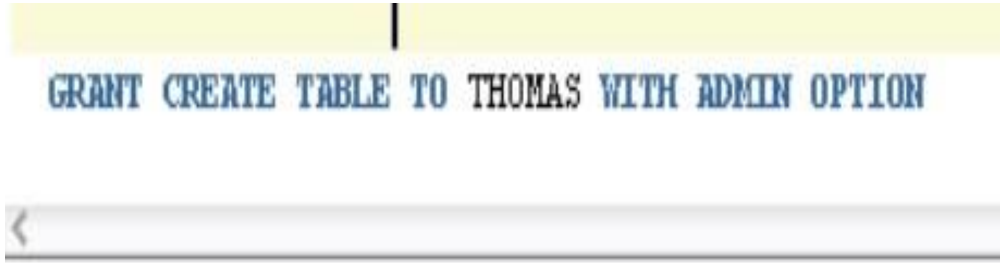
```
SELECT * FROM USER_ROLE_PRIVS
```

Results:

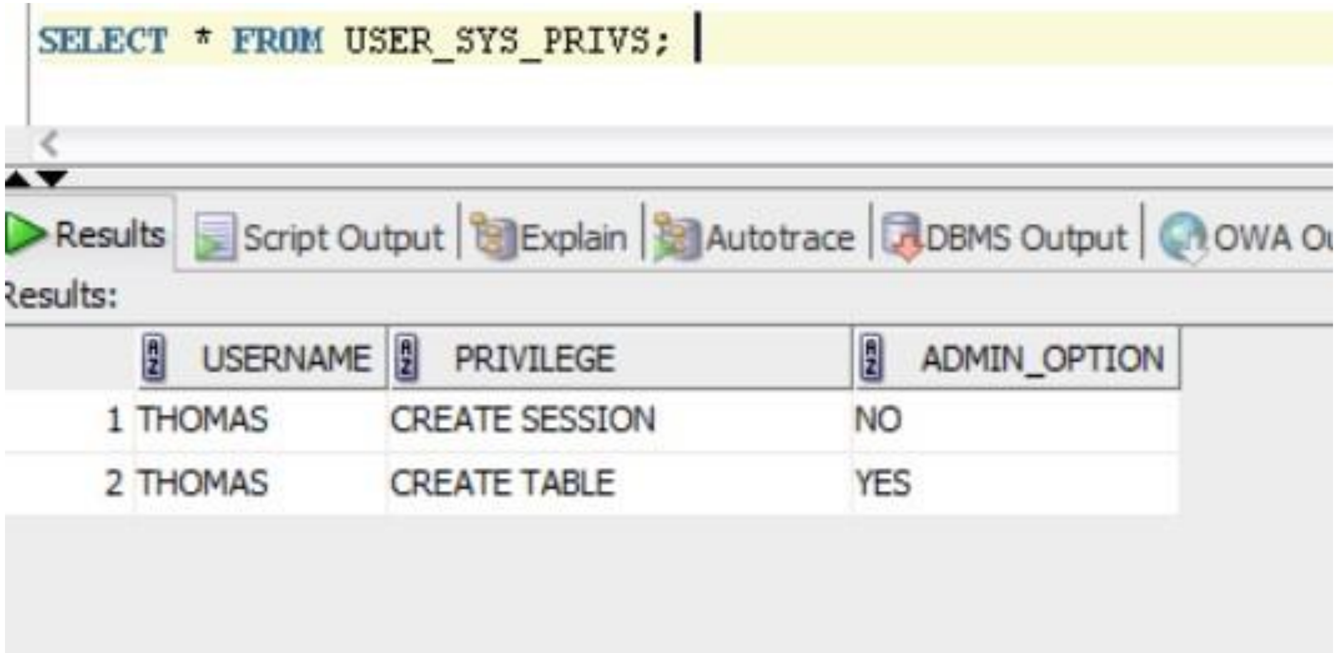
USERNAME	GRANTED_ROLE	ADMIN_OPTION	DEFAULT_ROLE	OS_GRANTED

SYNTAX:

GRANT sys_privilege [, sys_privilege ...]
TO user [, user, role, PUBLIC ...]
[WITH ADMIN OPTION] ;



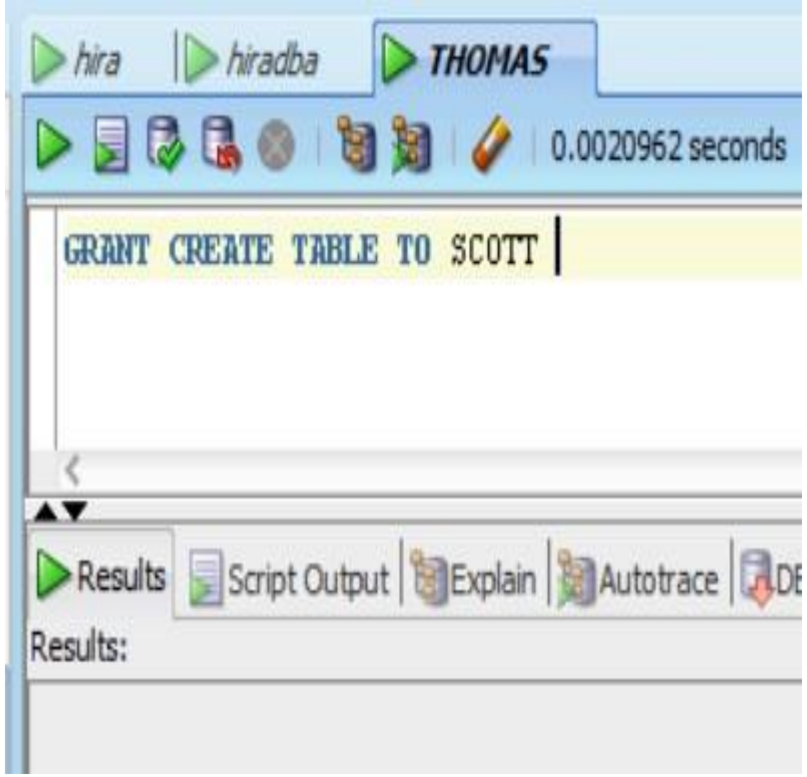
```
GRANT CREATE TABLE TO THOMAS WITH ADMIN OPTION
```



```
SELECT * FROM USER_SYS_PRIVS;
```

Results:

	USERNAME	PRIVILEGE	ADMIN_OPTION
1	THOMAS	CREATE SESSION	NO
2	THOMAS	CREATE TABLE	YES



```
GRANT CREATE TABLE TO SCOTT
```

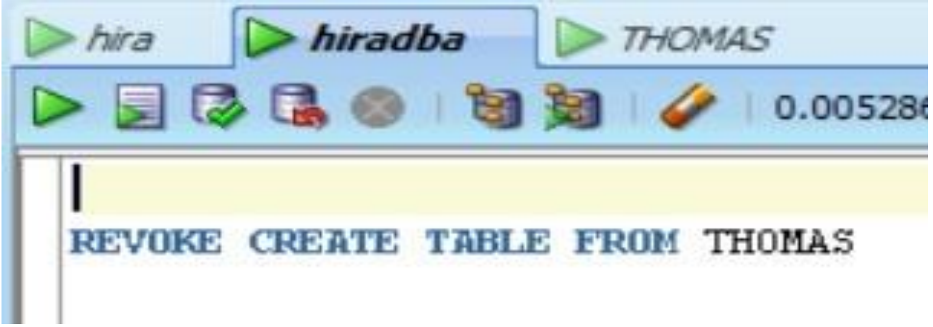
Results:

REVOKING SYSTEM PRIVILEGES

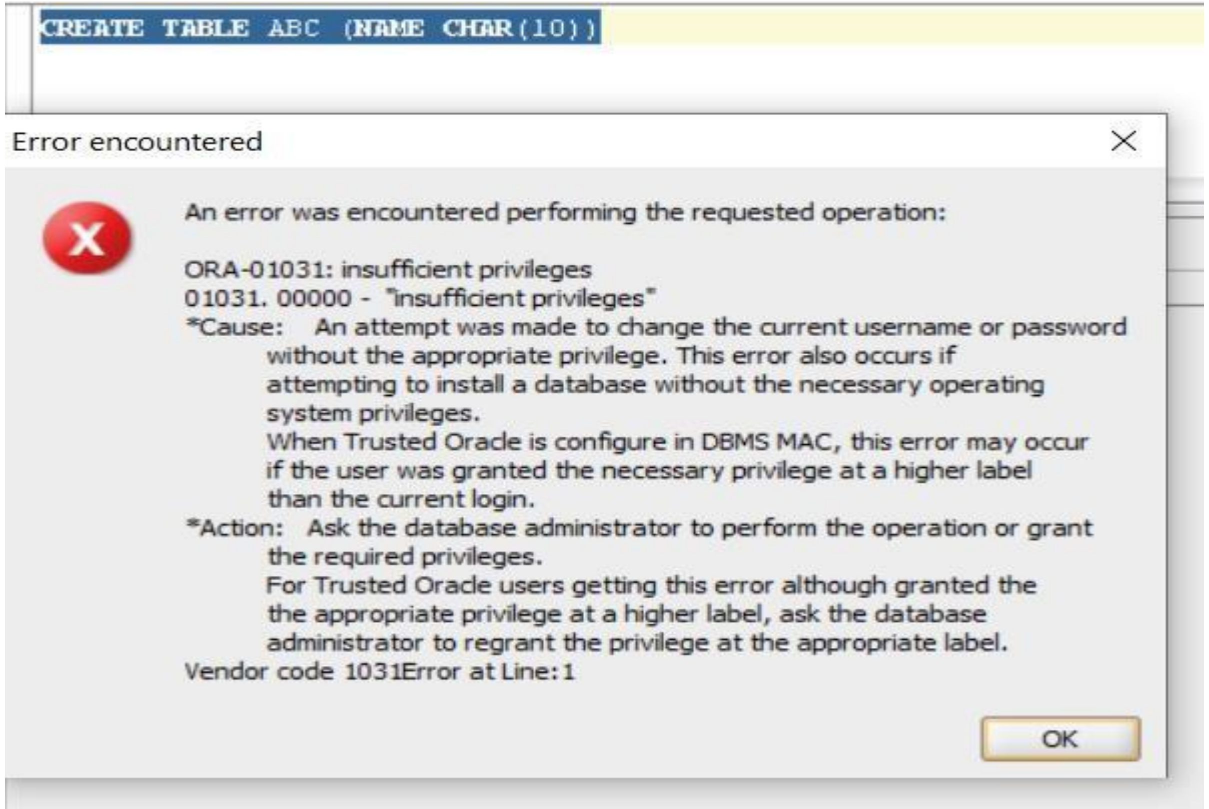
SYNTAX:

REVOKE sys_priv [,…]

FROM user| role ;



```
REVOKE CREATE TABLE FROM THOMAS
```

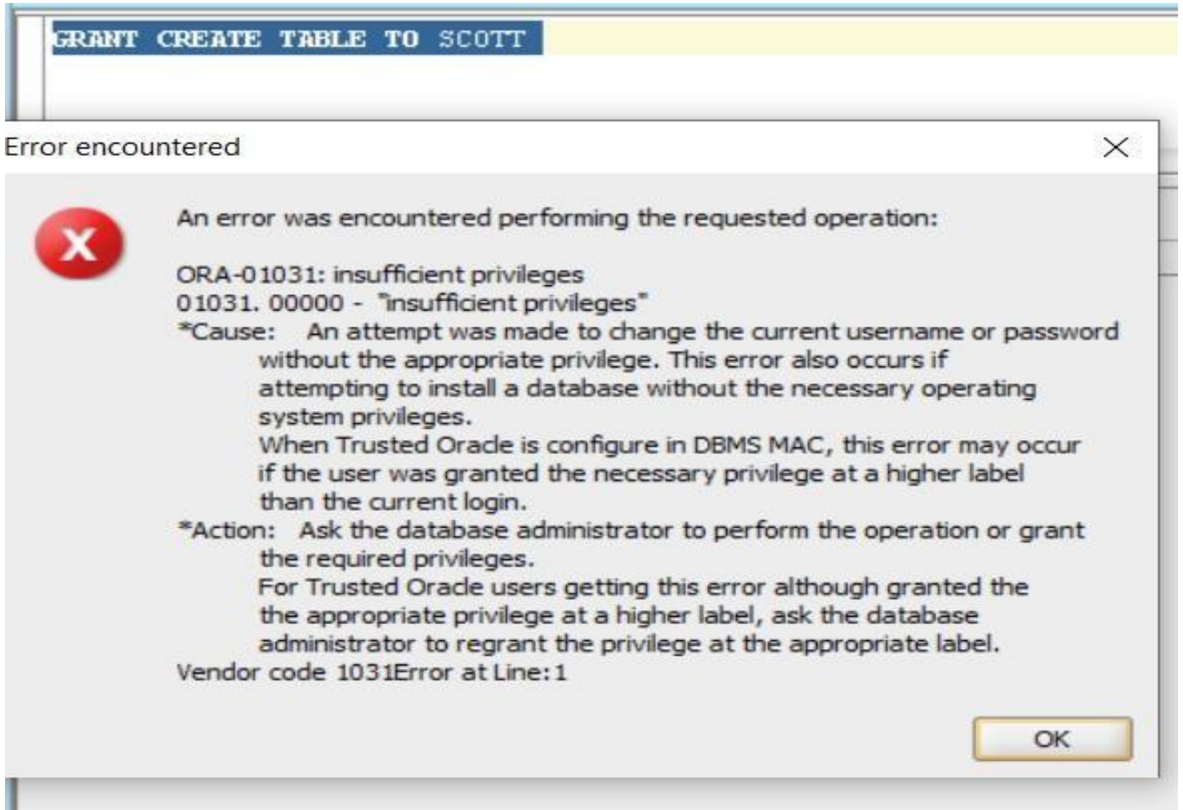


```
CREATE TABLE ABC (NAME CHAR(10))
```

Error encountered

An error was encountered performing the requested operation:

```
ORA-01031: insufficient privileges
01031. 00000 -  "insufficient privileges"
*Cause:    An attempt was made to change the current username or password
           without the appropriate privilege. This error also occurs if
           attempting to install a database without the necessary operating
           system privileges.
           When Trusted Oracle is configure in DBMS MAC, this error may occur
           if the user was granted the necessary privilege at a higher label
           than the current login.
*Action:   Ask the database administrator to perform the operation or grant
           the required privileges.
           For Trusted Oracle users getting this error although granted the
           the appropriate privilege at a higher label, ask the database
           administrator to regrant the privilege at the appropriate label.
Vendor code 1031Error at Line:1
```



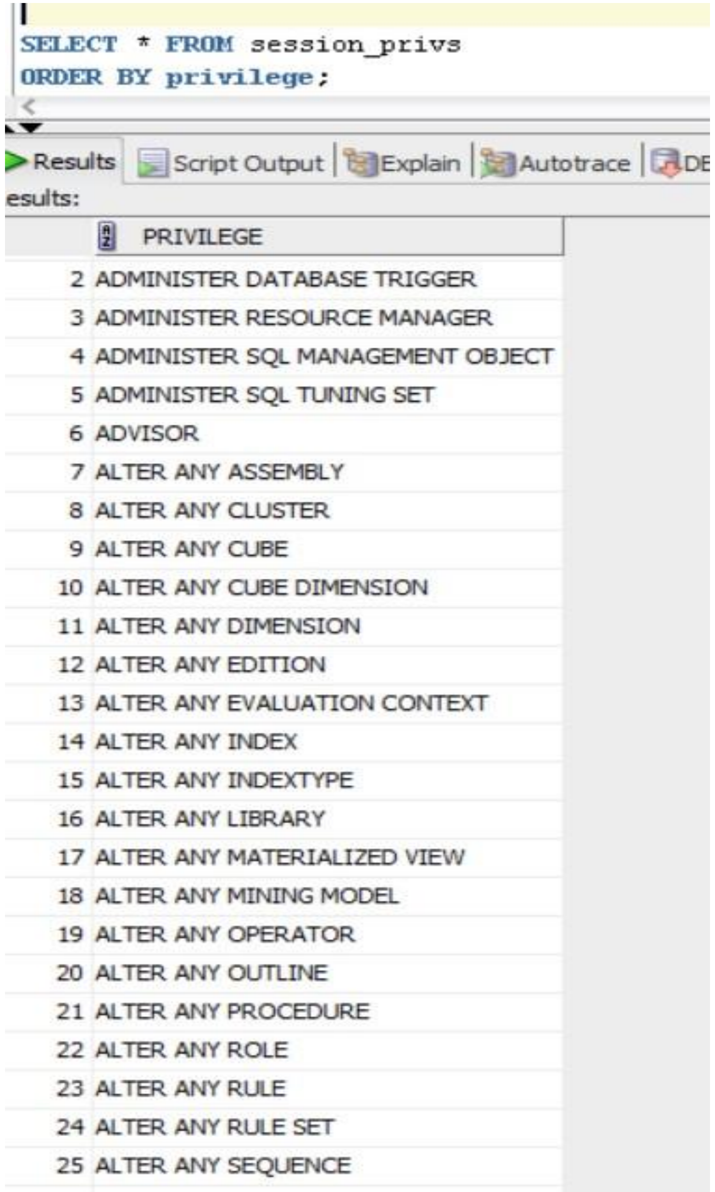
```
GRANT CREATE TABLE TO SCOTT
```

Error encountered

An error was encountered performing the requested operation:

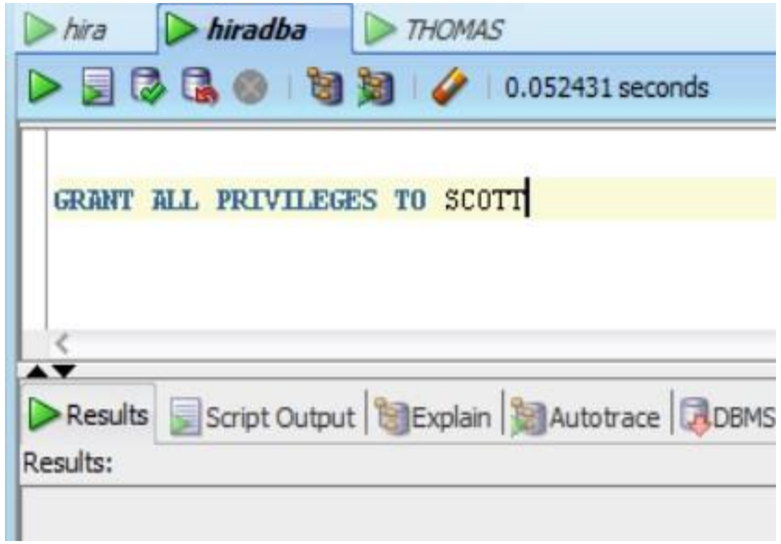
```
ORA-01031: insufficient privileges
01031. 00000 -  "insufficient privileges"
*Cause:    An attempt was made to change the current username or password
           without the appropriate privilege. This error also occurs if
           attempting to install a database without the necessary operating
           system privileges.
           When Trusted Oracle is configure in DBMS MAC, this error may occur
           if the user was granted the necessary privilege at a higher label
           than the current login.
*Action:   Ask the database administrator to perform the operation or grant
           the required privileges.
           For Trusted Oracle users getting this error although granted the
           the appropriate privilege at a higher label, ask the database
           administrator to regrant the privilege at the appropriate label.
Vendor code 1031Error at Line:1
```

GRANTING ALL SYSTEM PRIVILEGES TO A USER

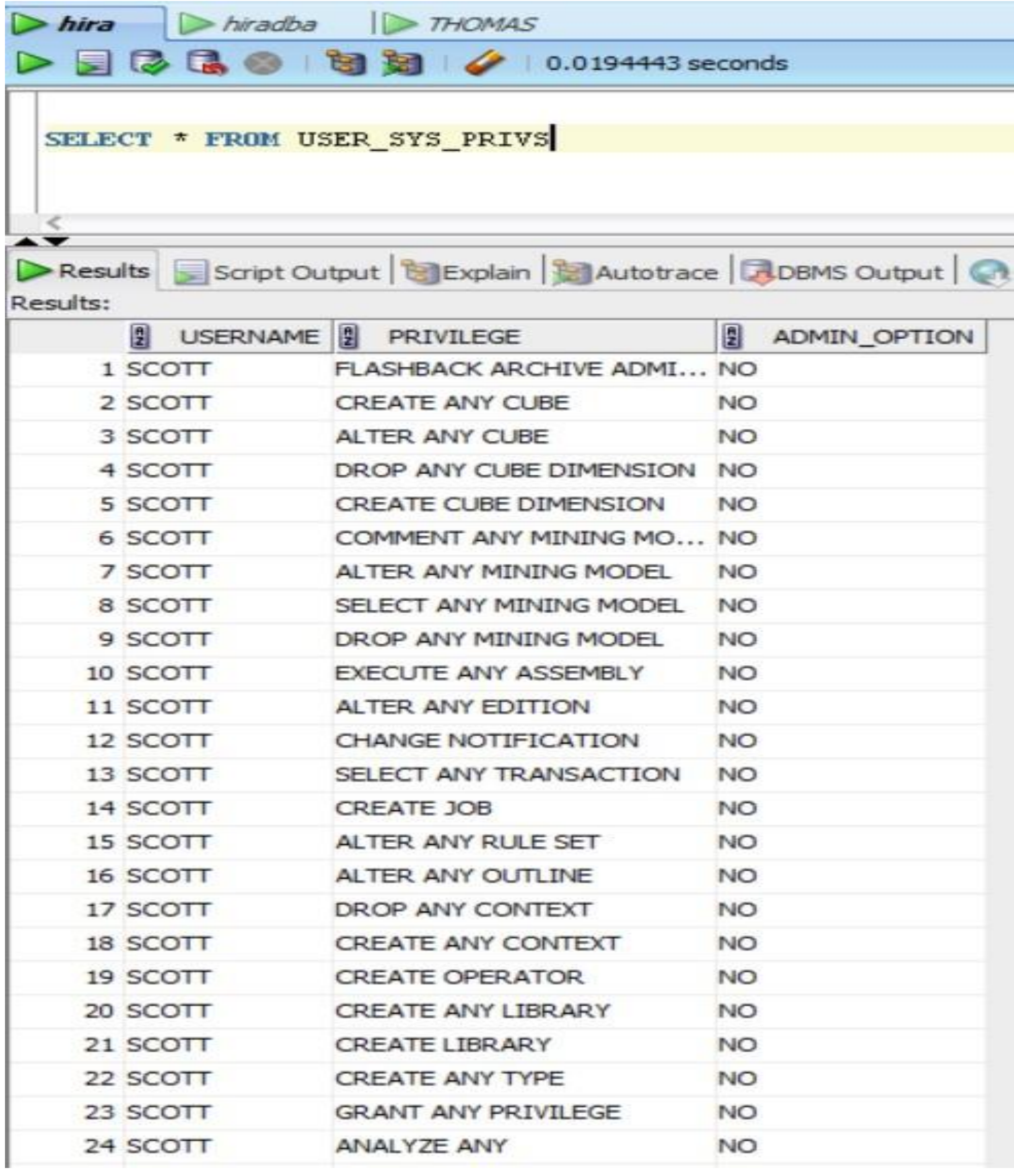


```
SELECT * FROM session_privs
ORDER BY privilege;
```

	PRIVILEGE
2	ADMINISTER DATABASE TRIGGER
3	ADMINISTER RESOURCE MANAGER
4	ADMINISTER SQL MANAGEMENT OBJECT
5	ADMINISTER SQL TUNING SET
6	ADVISOR
7	ALTER ANY ASSEMBLY
8	ALTER ANY CLUSTER
9	ALTER ANY CUBE
10	ALTER ANY CUBE DIMENSION
11	ALTER ANY DIMENSION
12	ALTER ANY EDITION
13	ALTER ANY EVALUATION CONTEXT
14	ALTER ANY INDEX
15	ALTER ANY INDEXTYPE
16	ALTER ANY LIBRARY
17	ALTER ANY MATERIALIZED VIEW
18	ALTER ANY MINING MODEL
19	ALTER ANY OPERATOR
20	ALTER ANY OUTLINE
21	ALTER ANY PROCEDURE
22	ALTER ANY ROLE
23	ALTER ANY RULE
24	ALTER ANY RULE SET
25	ALTER ANY SEQUENCE



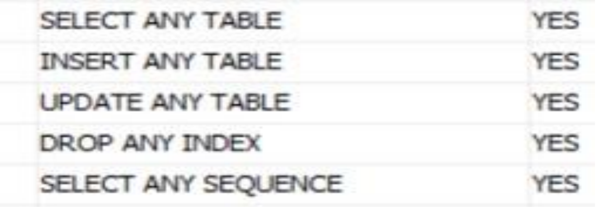
```
GRANT ALL PRIVILEGES TO SCOTT
```



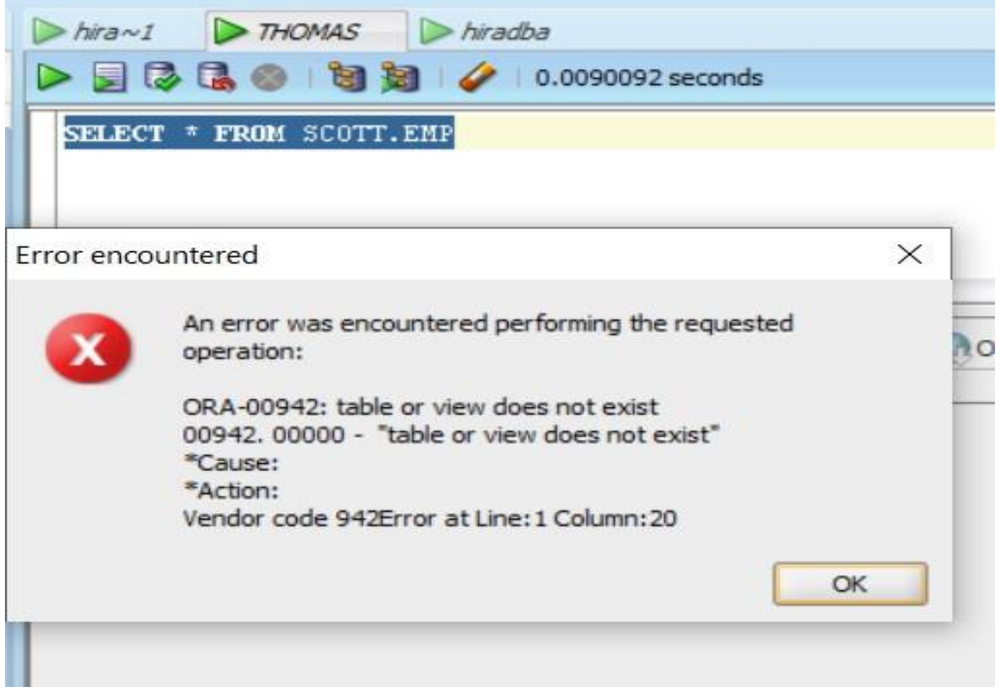
```
SELECT * FROM USER_SYS_PRIVS
```

	USERNAME	PRIVILEGE	ADMIN_OPTION
1	SCOTT	FLASHBACK ARCHIVE ADMI...	NO
2	SCOTT	CREATE ANY CUBE	NO
3	SCOTT	ALTER ANY CUBE	NO
4	SCOTT	DROP ANY CUBE DIMENSION	NO
5	SCOTT	CREATE CUBE DIMENSION	NO
6	SCOTT	COMMENT ANY MINING MO...	NO
7	SCOTT	ALTER ANY MINING MODEL	NO
8	SCOTT	SELECT ANY MINING MODEL	NO
9	SCOTT	DROP ANY MINING MODEL	NO
10	SCOTT	EXECUTE ANY ASSEMBLY	NO
11	SCOTT	ALTER ANY EDITION	NO
12	SCOTT	CHANGE NOTIFICATION	NO
13	SCOTT	SELECT ANY TRANSACTION	NO
14	SCOTT	CREATE JOB	NO
15	SCOTT	ALTER ANY RULE SET	NO
16	SCOTT	ALTER ANY OUTLINE	NO
17	SCOTT	DROP ANY CONTEXT	NO
18	SCOTT	CREATE ANY CONTEXT	NO
19	SCOTT	CREATE OPERATOR	NO
20	SCOTT	CREATE ANY LIBRARY	NO
21	SCOTT	CREATE LIBRARY	NO
22	SCOTT	CREATE ANY TYPE	NO
23	SCOTT	GRANT ANY PRIVILEGE	NO
24	SCOTT	ANALYZE ANY	NO

OBJECT PRIVILEGES GRANTED AS SYSTEM



SELECT ANY TABLE	YES
INSERT ANY TABLE	YES
UPDATE ANY TABLE	YES
DROP ANY INDEX	YES
SELECT ANY SEQUENCE	YES

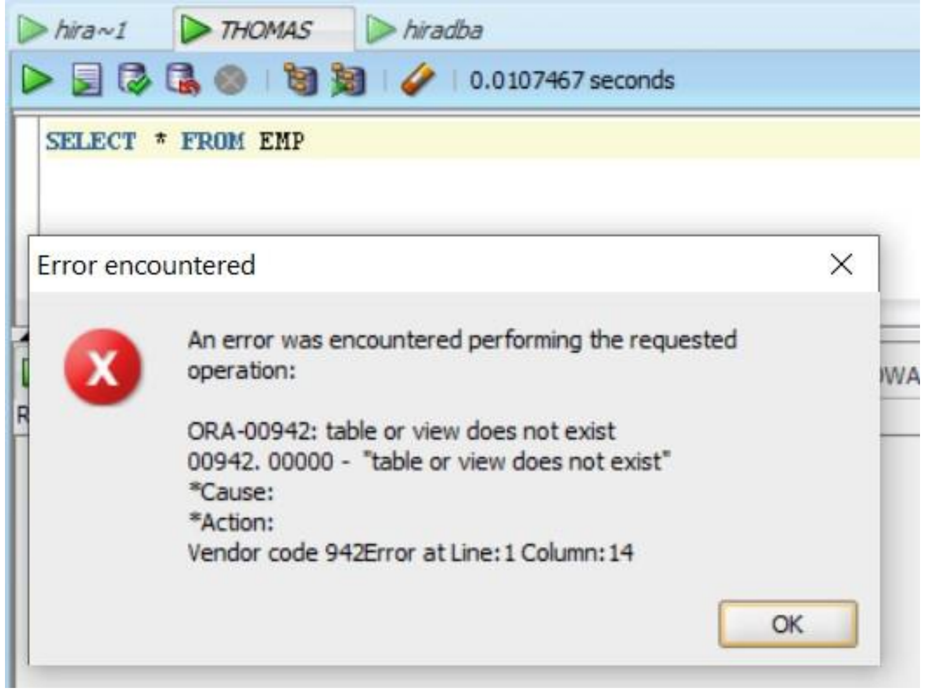


```
SELECT * FROM SCOTT.EMP
```

Error encountered

An error was encountered performing the requested operation:

ORA-00942: table or view does not exist
00942. 00000 - "table or view does not exist"
*Cause:
*Action:
Vendor code 942Error at Line:1 Column:20

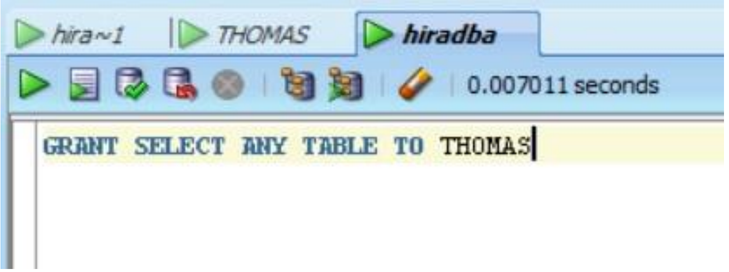


```
SELECT * FROM EMP
```

Error encountered

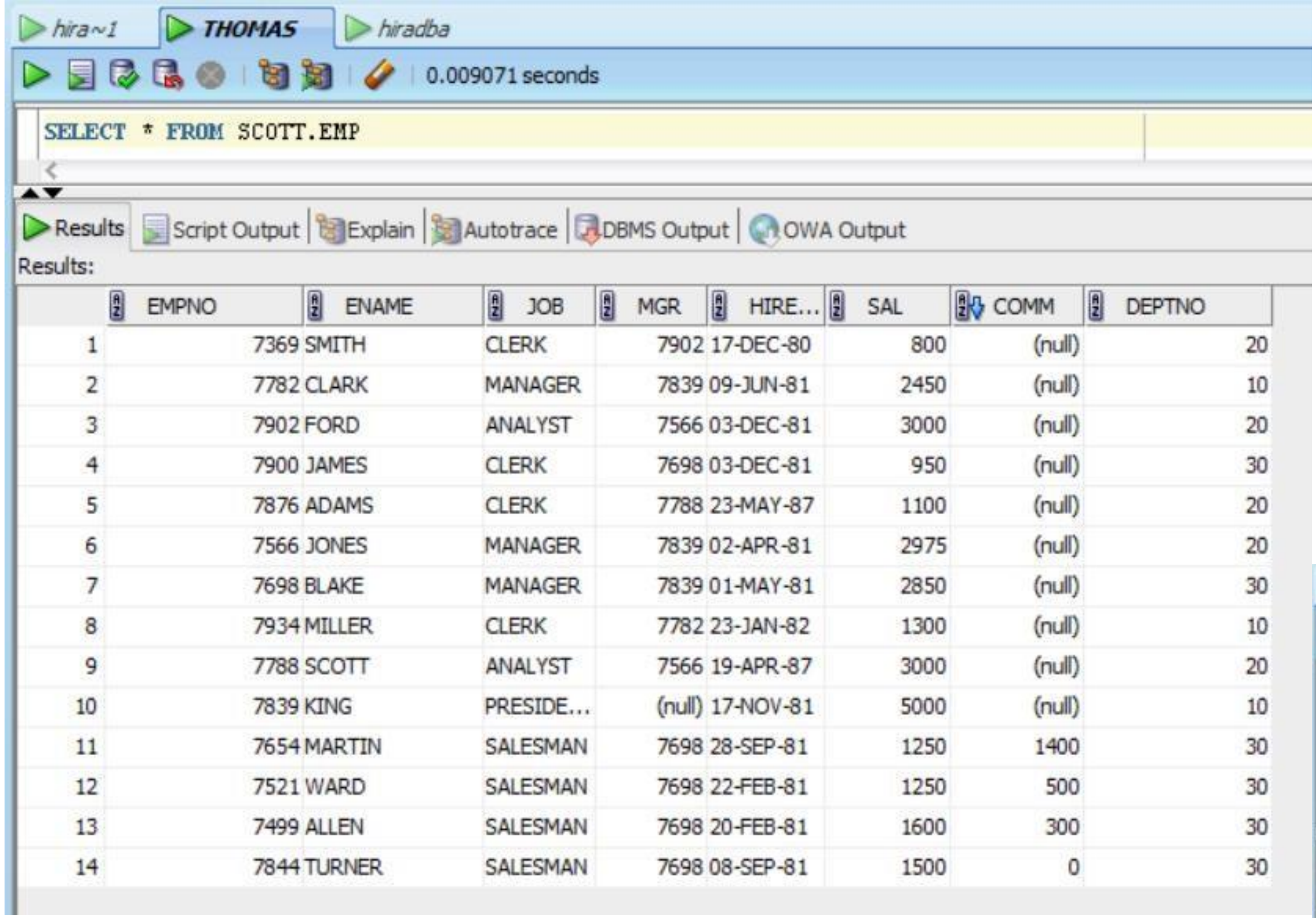
An error was encountered performing the requested operation:

ORA-00942: table or view does not exist
00942. 00000 - "table or view does not exist"
*Cause:
*Action:
Vendor code 942Error at Line:1 Column:14



```
GRANT SELECT ANY TABLE TO THOMAS
```

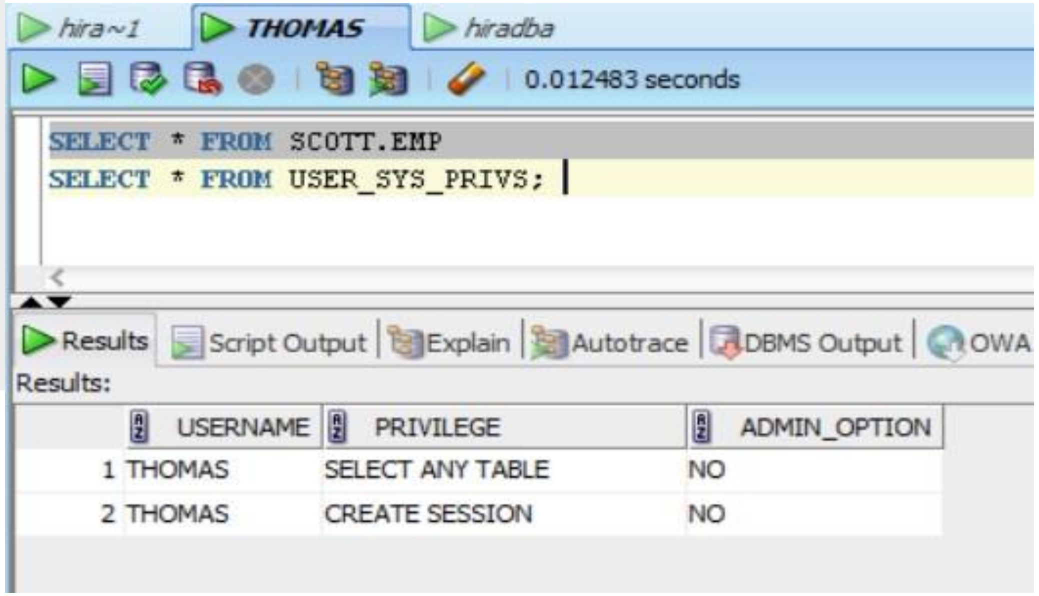
PRIVILEGES



```
SELECT * FROM SCOTT.EMP
```

Results:

	EMPNO	ENAME	JOB	MGR	HIRE...	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	17-DEC-80	800	(null)	20
2	7782	CLARK	MANAGER	7839	09-JUN-81	2450	(null)	10
3	7902	FORD	ANALYST	7566	03-DEC-81	3000	(null)	20
4	7900	JAMES	CLERK	7698	03-DEC-81	950	(null)	30
5	7876	ADAMS	CLERK	7788	23-MAY-87	1100	(null)	20
6	7566	JONES	MANAGER	7839	02-APR-81	2975	(null)	20
7	7698	BLAKE	MANAGER	7839	01-MAY-81	2850	(null)	30
8	7934	MILLER	CLERK	7782	23-JAN-82	1300	(null)	10
9	7788	SCOTT	ANALYST	7566	19-APR-87	3000	(null)	20
10	7839	KING	PRESIDE...	(null)	17-NOV-81	5000	(null)	10
11	7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
12	7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
13	7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
14	7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30



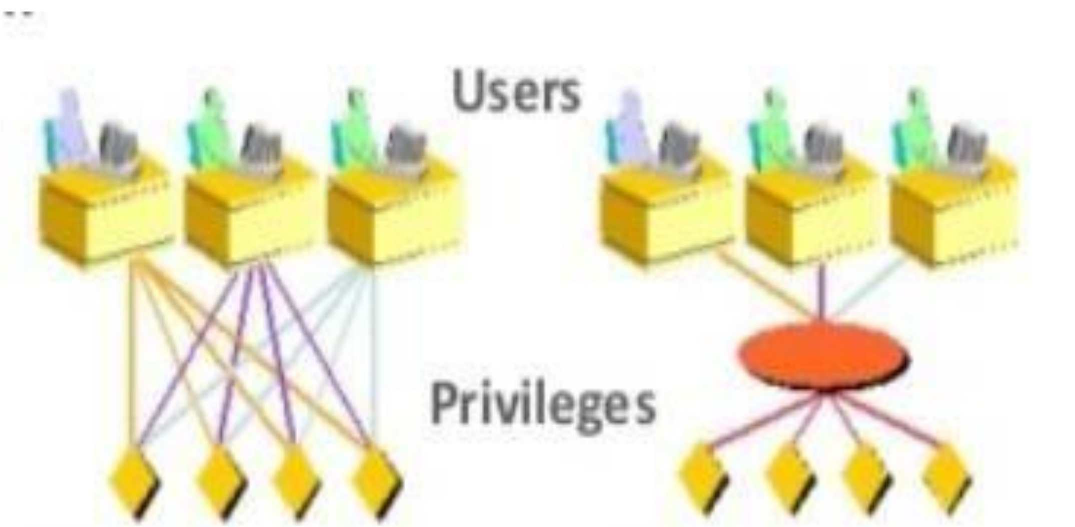
```
SELECT * FROM SCOTT.EMP
SELECT * FROM USER_SYS_PRIVS;
```

Results:

	USERNAME	PRIVILEGE	ADMIN_OPTION
1	THOMAS	SELECT ANY TABLE	NO
2	THOMAS	CREATE SESSION	NO

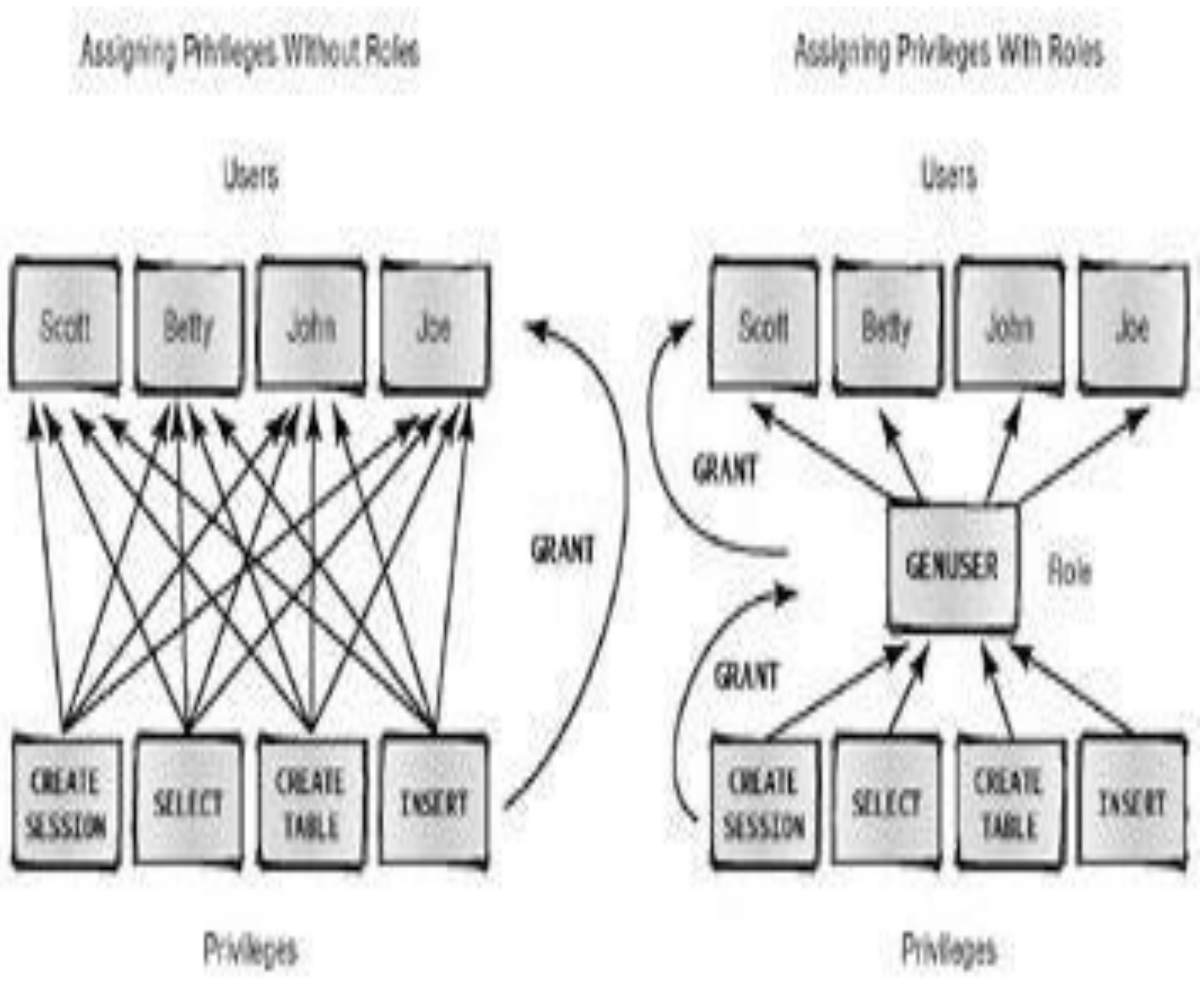
CREATING ROLE

- You can also grant privileges to a role (a named group of privileges), and then grant the role to one or more users.
- Role is a set of privileges that can be granted to users or to other roles.
- Roles ca be used to administer database privileges.
- A new role is initially empty. You add privileges to a role with the GRANT statement.
- A role contains all privileges granted to the role and all privileges of other roles granted to it.
- Because roles allow for easier and better management of privileges, you should normally grant privileges to roles and not to specific users.



Allocating privileges without a role

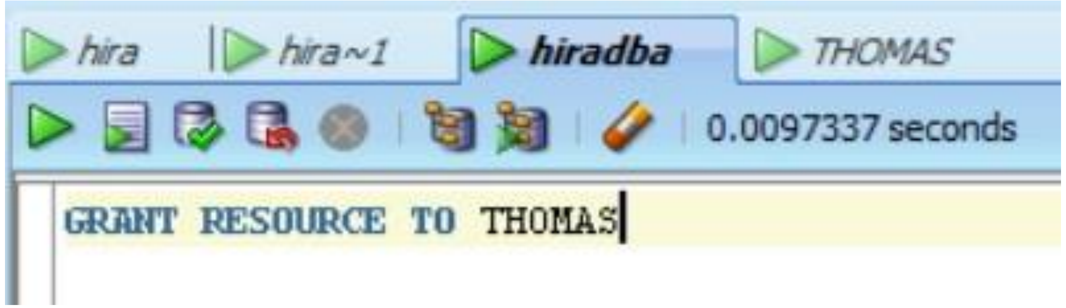
Allocating privileges with a role


Assigning Privileges Without Roles

Assigning Privileges With Roles

SYNTAX:

CREATE ROLE role_name
[IDENTIFIED BY password]
[NOT IDENTIFIED]



```
GRANT RESOURCE TO THOMAS
```

IDENTIFIED BY password option is used to create a local role and indicate that the user, who was granted the role, must provide the password to the database when enabling the role.

NOT IDENTIFIED indicates that the role is authorized by the database and the user, who was granted this role, does not need a password to enable the role. **GRANTING PRIVILEGES TO A ROLE**

GRANT system_privileges | object_privileges TO role_name ;

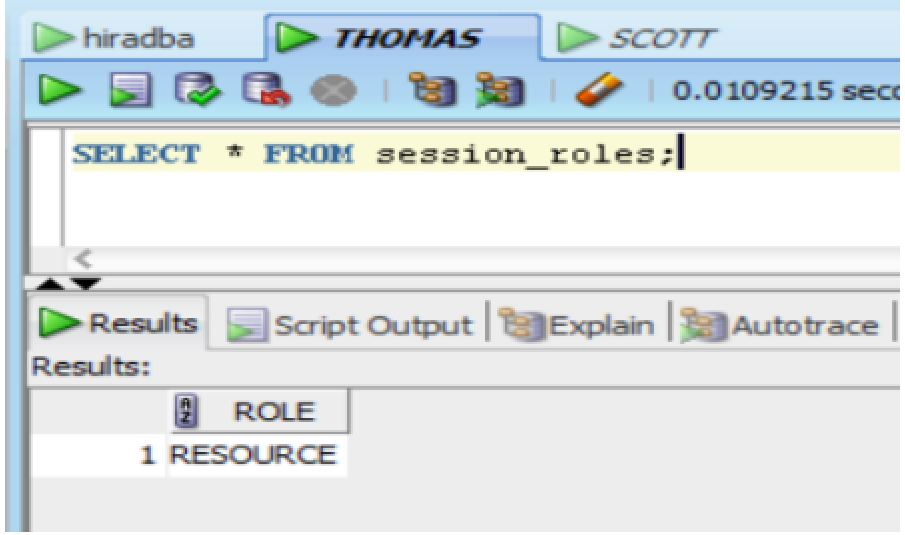
In addition, you can use the GRANT statement to grant privileges of a role to another role:

GRANTING ROLE TO ANOTHER ROLE OR USER GRANT

role_name TO another_role_name ;

STEPS CONCERNING ROLE CREATION AND GRANT OF PRIVILEGES (WITHOUT PASSWORD)

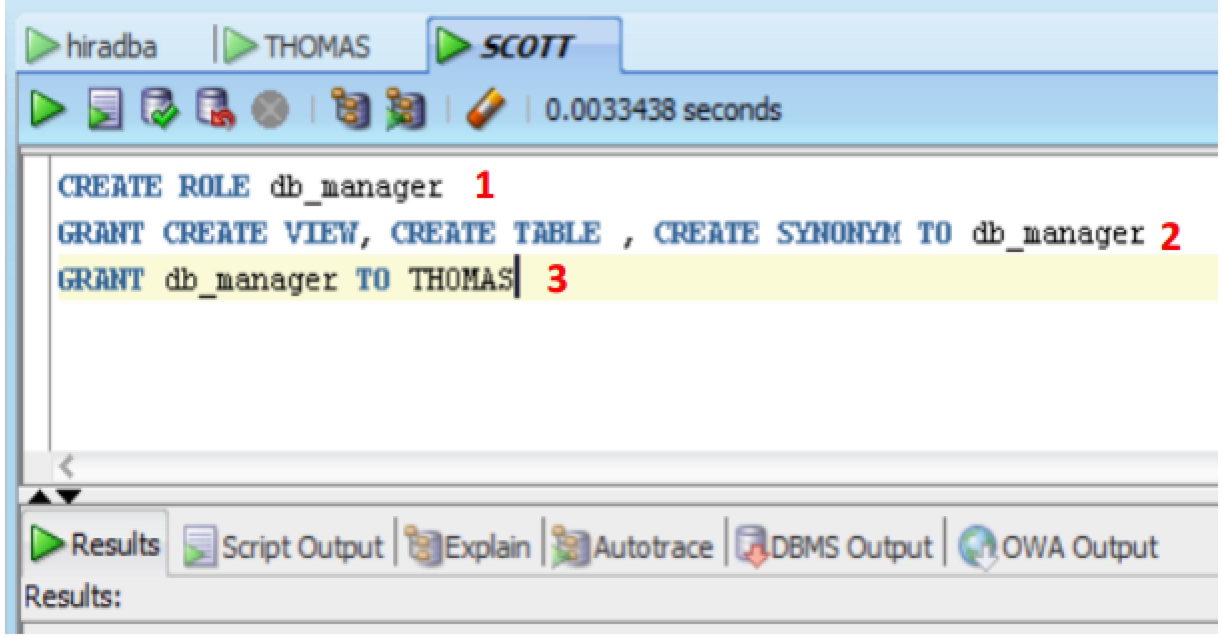
1. CREATE A ROLE.
2. GRANT PRIVILEGES TO ROLE.
3. GRANT ROLE TO USER.
4. GO TO THE USER ACCOUNT AND SET THE ROLE.



```
SELECT * FROM session_roles;
```

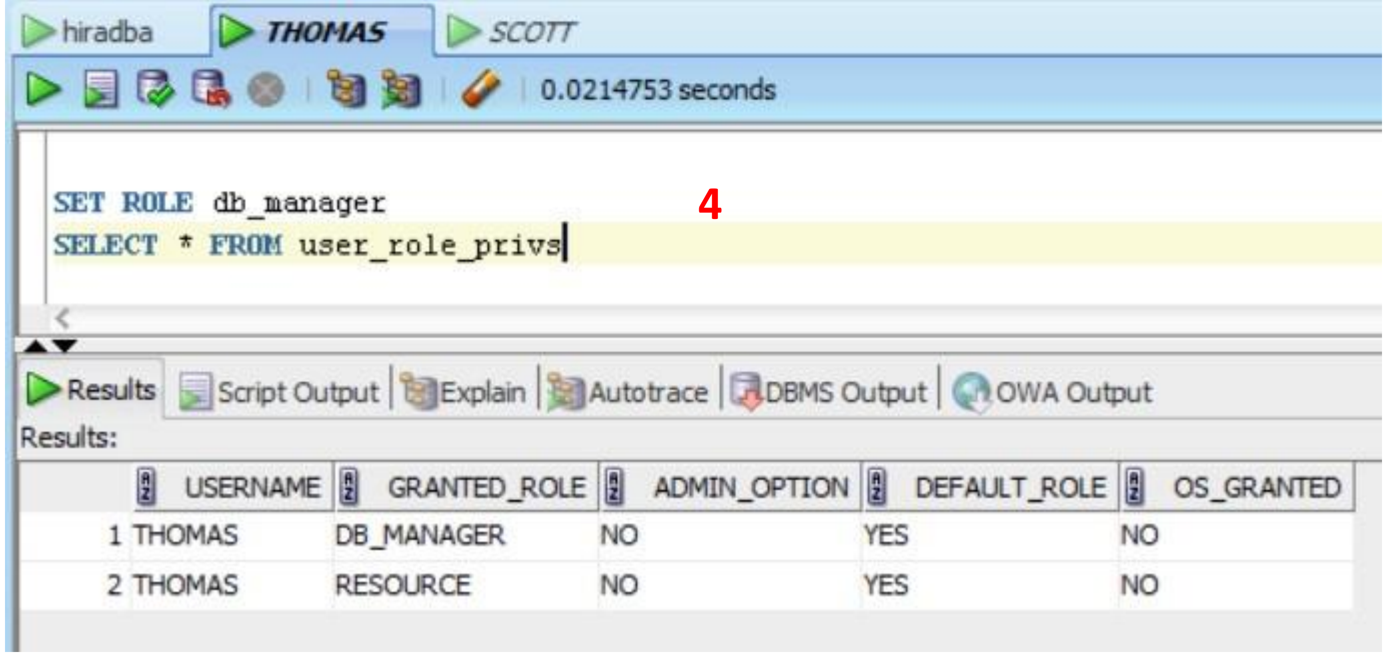
Results:

	ROLE
1	RESOURCE



```
CREATE ROLE db_manager  1
GRANT CREATE VIEW, CREATE TABLE , CREATE SYNONYM TO db_manager 2
GRANT db_manager TO THOMAS  3
```

Results:



```
SET ROLE db_manager
SELECT * FROM user_role_privs
```

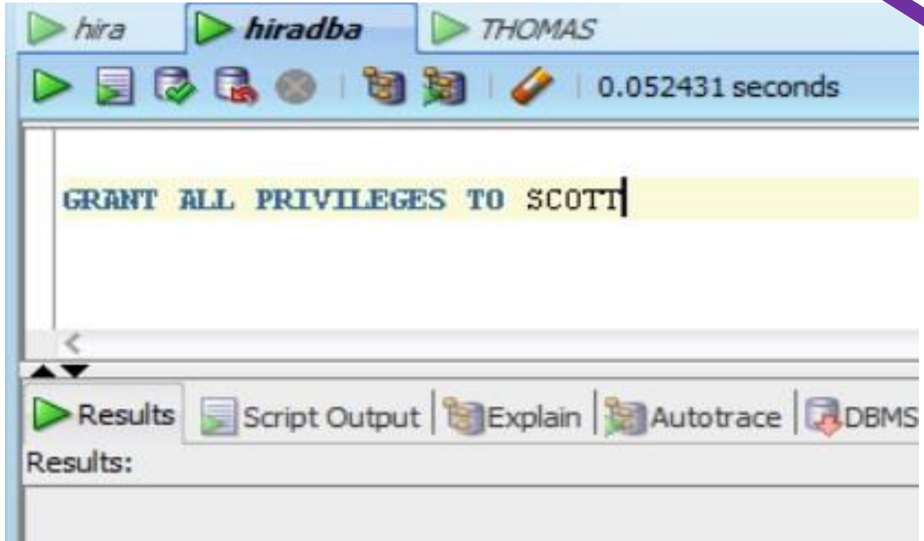
4

Results:

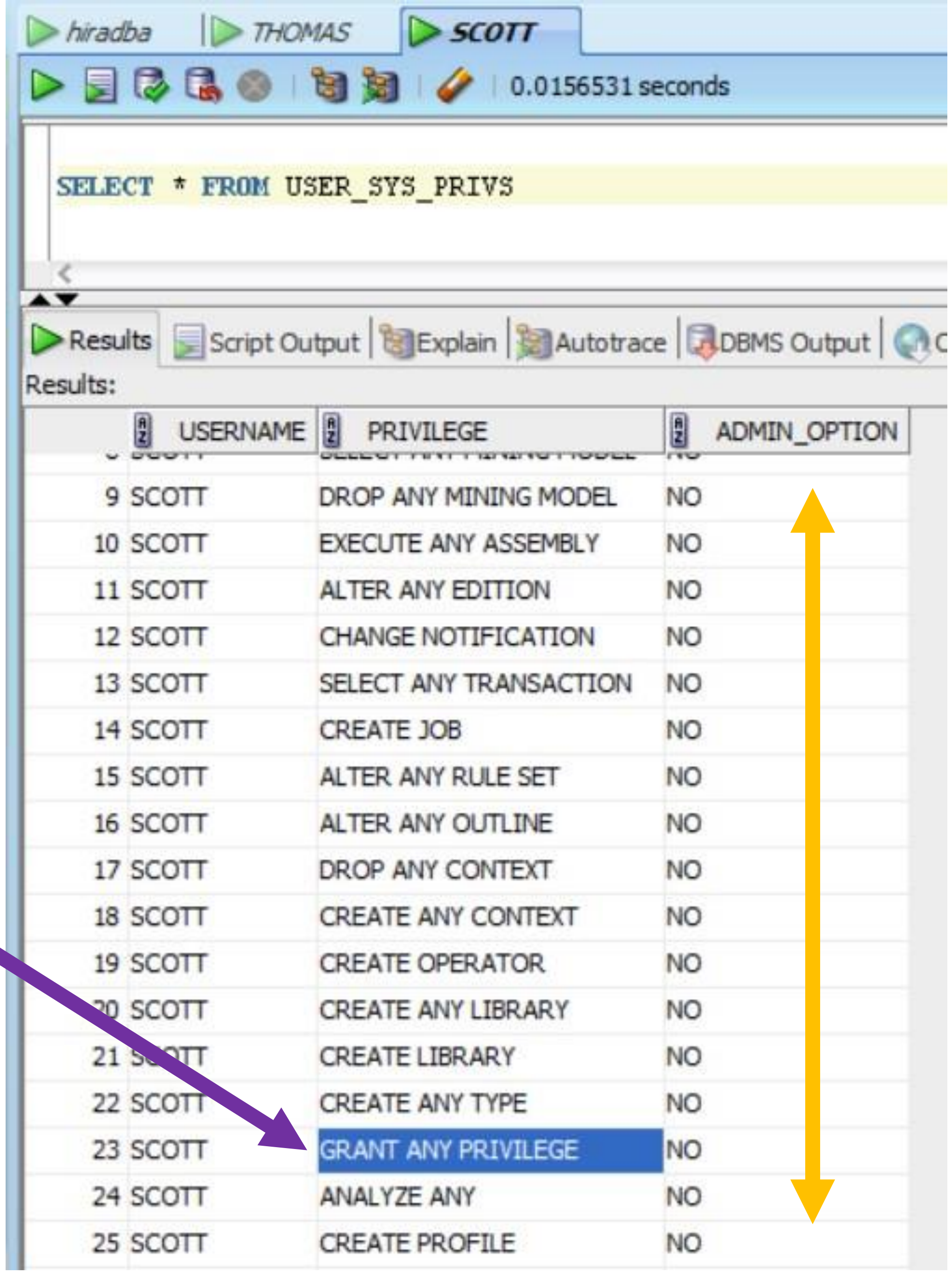
	USERNAME	GRANTED_ROLE	ADMIN_OPTION	DEFAULT_ROLE	OS_GRANTED
1	THOMAS	DB_MANAGER	NO	YES	NO
2	THOMAS	RESOURCE	NO	YES	NO

WHO CAN GRANT PRIVILIGES?

1. Users who have been granted a specific system privilege with the **ADMIN OPTION**.
2. Users with the system privilege **GRANT ANY PRIVILEGE**.



```
GRANT ALL PRIVILEGES TO SCOTT
```



```
SELECT * FROM USER_SYS_PRIVS
```

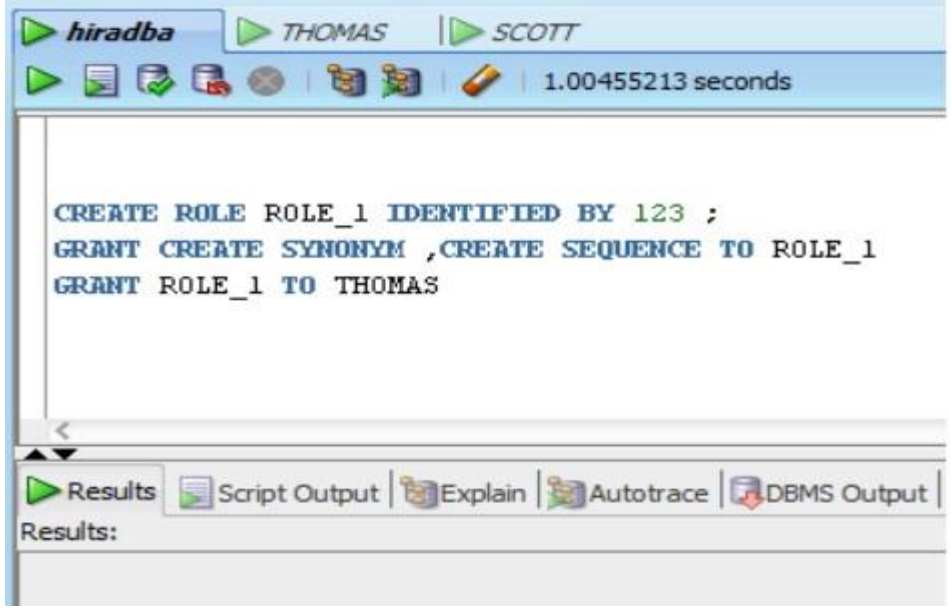
	USERNAME	PRIVILEGE	ADMIN_OPTION
8	SCOTT	SELECT ANY MINING MODEL	NO
9	SCOTT	DROP ANY MINING MODEL	NO
10	SCOTT	EXECUTE ANY ASSEMBLY	NO
11	SCOTT	ALTER ANY EDITION	NO
12	SCOTT	CHANGE NOTIFICATION	NO
13	SCOTT	SELECT ANY TRANSACTION	NO
14	SCOTT	CREATE JOB	NO
15	SCOTT	ALTER ANY RULE SET	NO
16	SCOTT	ALTER ANY OUTLINE	NO
17	SCOTT	DROP ANY CONTEXT	NO
18	SCOTT	CREATE ANY CONTEXT	NO
19	SCOTT	CREATE OPERATOR	NO
20	SCOTT	CREATE ANY LIBRARY	NO
21	SCOTT	CREATE LIBRARY	NO
22	SCOTT	CREATE ANY TYPE	NO
23	SCOTT	GRANT ANY PRIVILEGE	NO
24	SCOTT	ANALYZE ANY	NO
25	SCOTT	CREATE PROFILE	NO

STEPS CONCERNING ROLE CREATION AND GRANT OF PRIVILEGES (WITH PASSWORD)

1. CREATE A ROLE WITH PASSWORD. **SYNTAX:**CREATE ROLE role_name 2. GRANT PRIVILEGES TO ROLE. [IDENTIFIED BY password]

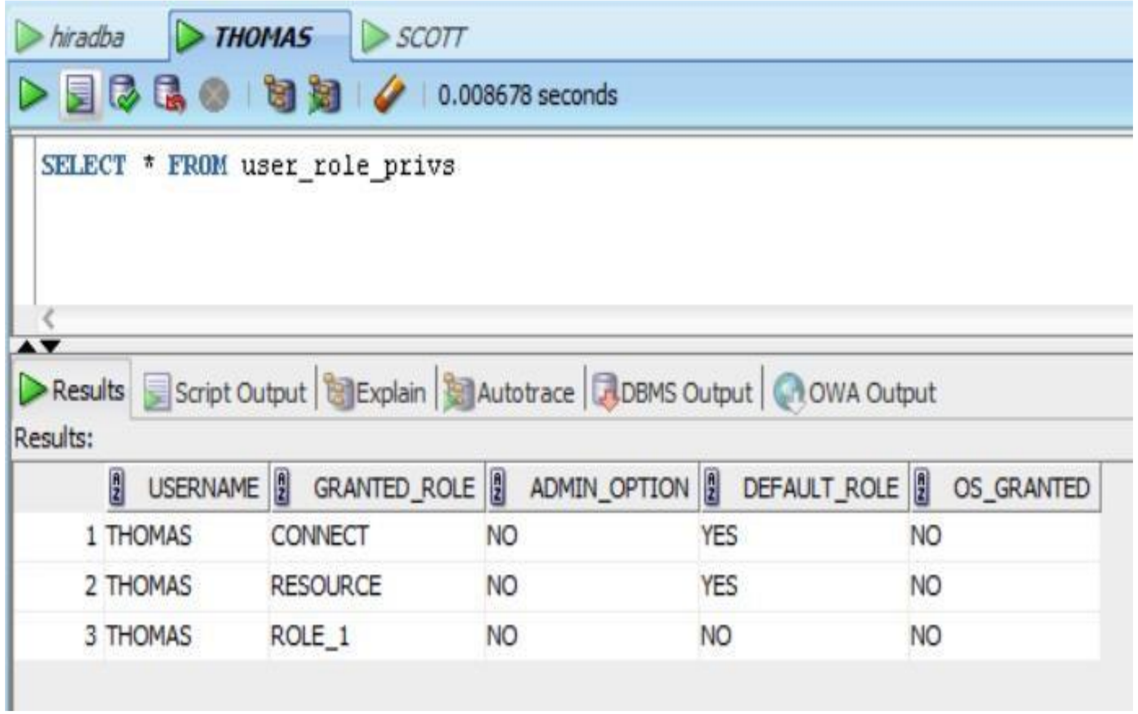
3. GRANT ROLE TO USER.

4. GO TO THE USER ACCOUNT AND SET THE ROLE WITH PASSWORD.



```
CREATE ROLE ROLE_1 IDENTIFIED BY 123 ;
GRANT CREATE SYNONYM ,CREATE SEQUENCE TO ROLE_1
GRANT ROLE_1 TO THOMAS
```

CREATE ROLE ROLE_1 IDENTIFIED BY 123 ;

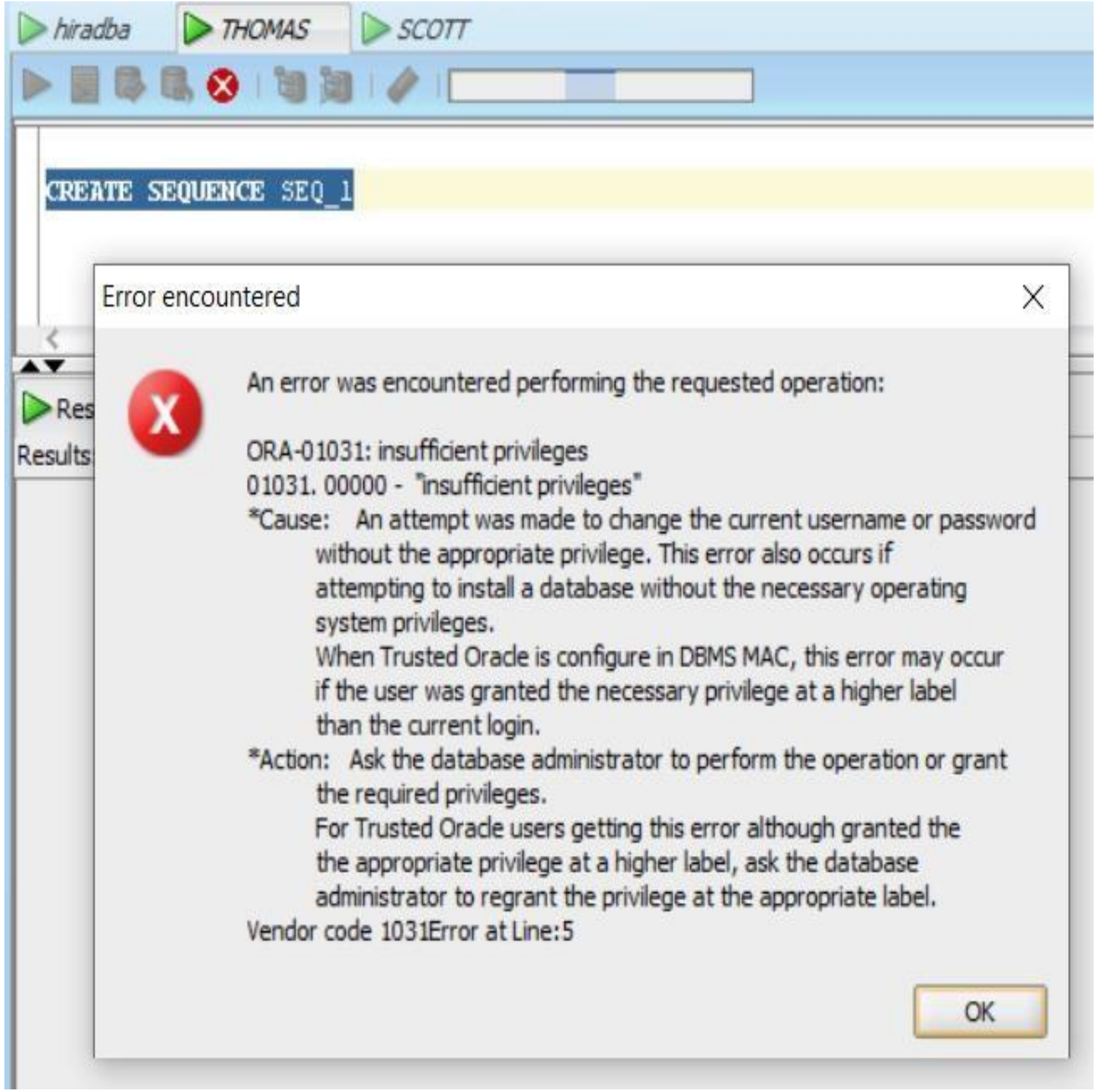


```sql
SELECT * FROM user_role_privs
```

Results:

	USERNAME	GRANTED_ROLE	ADMIN_OPTION	DEFAULT_ROLE	OS_GRANTED
1	THOMAS	CONNECT	NO	YES	NO
2	THOMAS	RESOURCE	NO	YES	NO
3	THOMAS	ROLE_1	NO	NO	NO

RESOURCE and CONNECT are System Roles, while Role_1 is a user defined role.



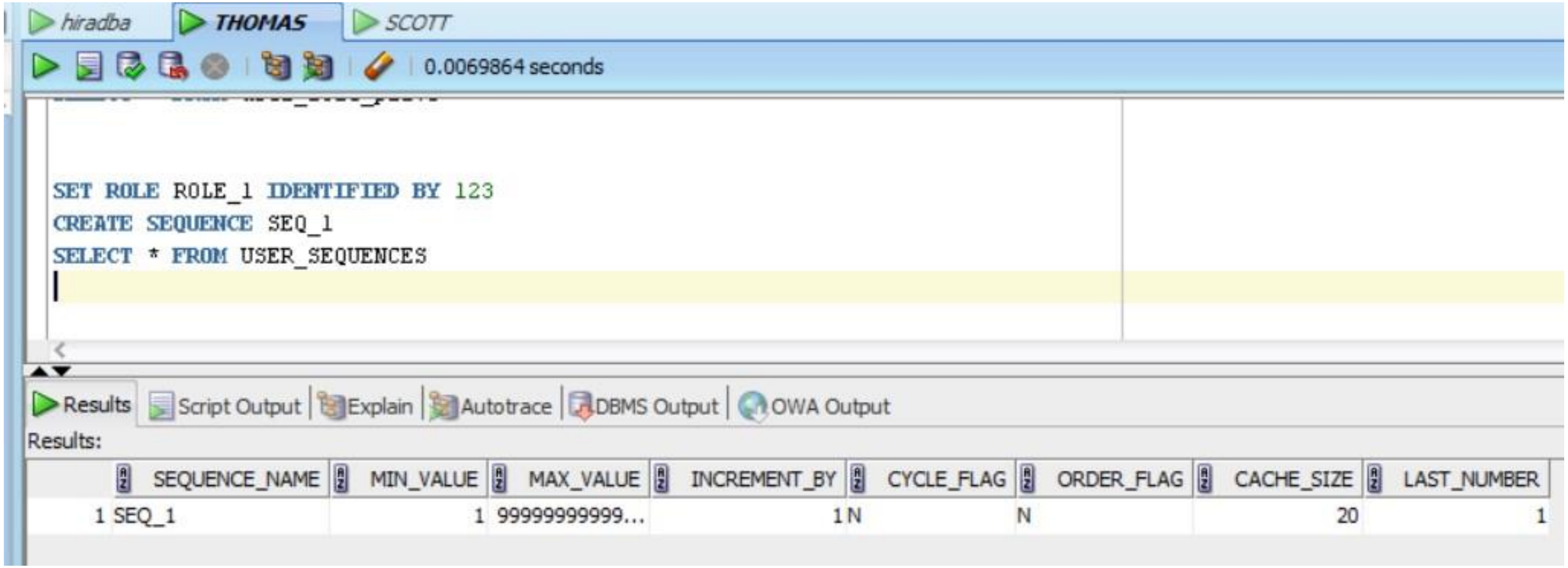
```sql
CREATE SEQUENCE SEQ_1
```

Error encountered

```
An error was encountered performing the requested operation:

ORA-01031: insufficient privileges
01031. 00000 -  "insufficient privileges"
*Cause:    An attempt was made to change the current username or password
           without the appropriate privilege. This error also occurs if
           attempting to install a database without the necessary operating
           system privileges.
           When Trusted Oracle is configure in DBMS MAC, this error may occur
           if the user was granted the necessary privilege at a higher label
           than the current login.
*Action:   Ask the database administrator to perform the operation or grant
           the required privileges.
           For Trusted Oracle users getting this error although granted the
           the appropriate privilege at a higher label, ask the database
           administrator to regrant the privilege at the appropriate label.
Vendor code 1031Error at Line:5
```

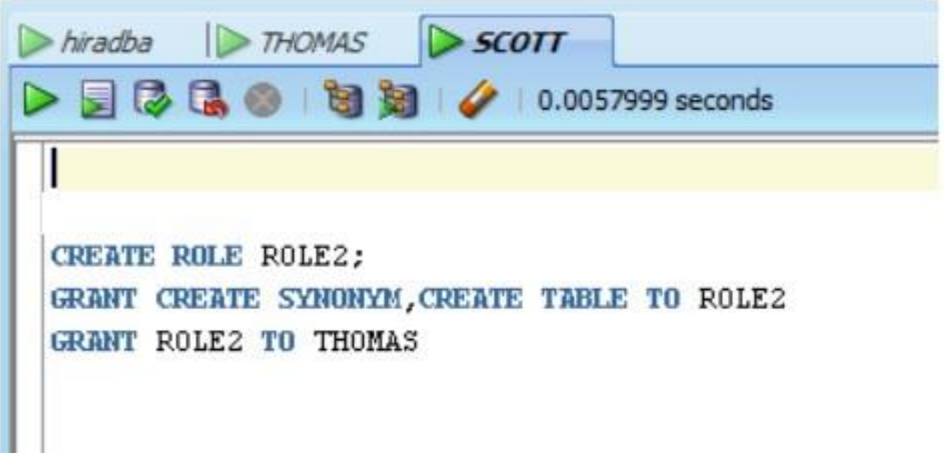
SET ROLE ROLE_1 IDENTIFIED BY 123 ;



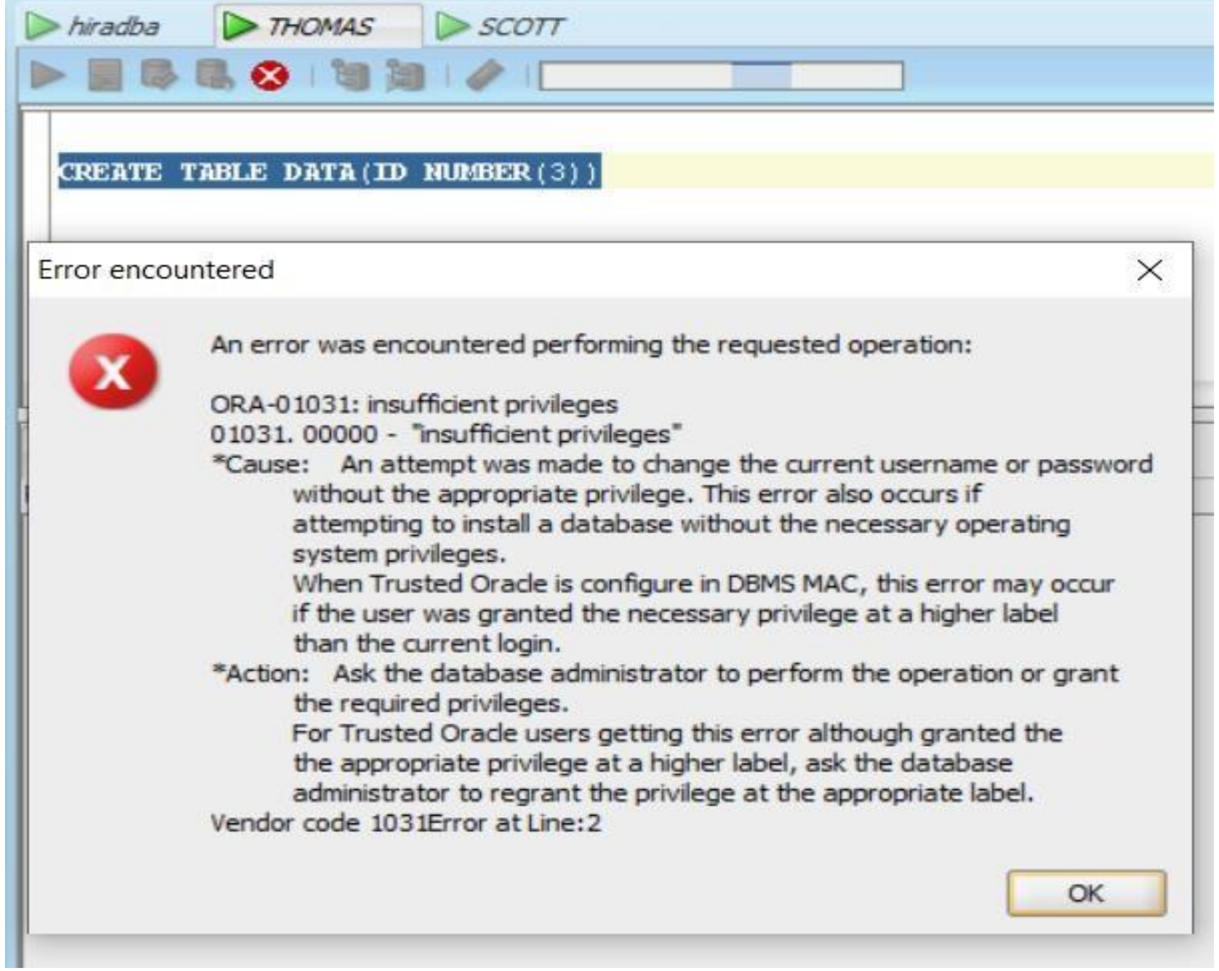
```
SET ROLE ROLE_1 IDENTIFIED BY 123
CREATE SEQUENCE SEQ_1
SELECT * FROM USER_SEQUENCES
```

Results:

	SEQUENCE_NAME	MIN_VALUE	MAX_VALUE	INCREMENT_BY	CYCLE_FLAG	ORDER_FLAG	CACHE_SIZE	LAST_NUMBER
1	SEQ_1	1	99999999999...	1	N	N	20	1



```
CREATE ROLE ROLE2;
GRANT CREATE SYNONYM,CREATE TABLE TO ROLE2
GRANT ROLE2 TO THOMAS
```

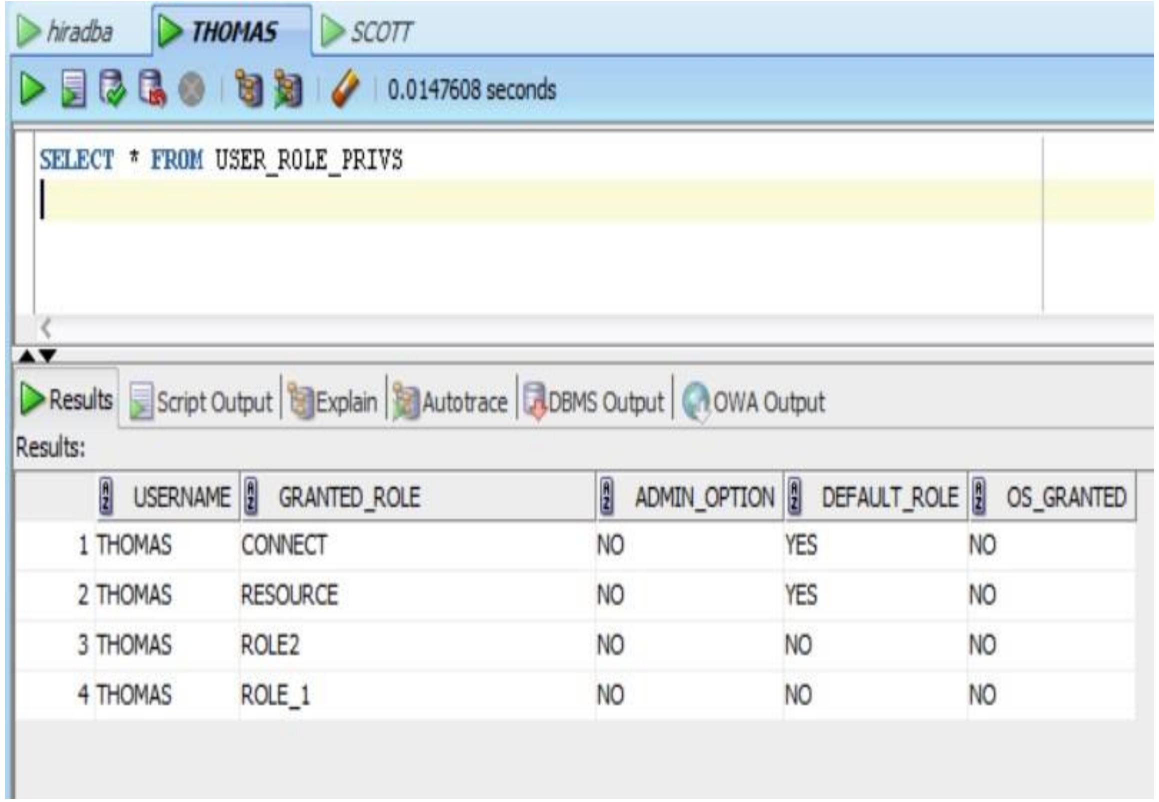


```
CREATE TABLE DATA(ID NUMBER(3))
```

Error encountered

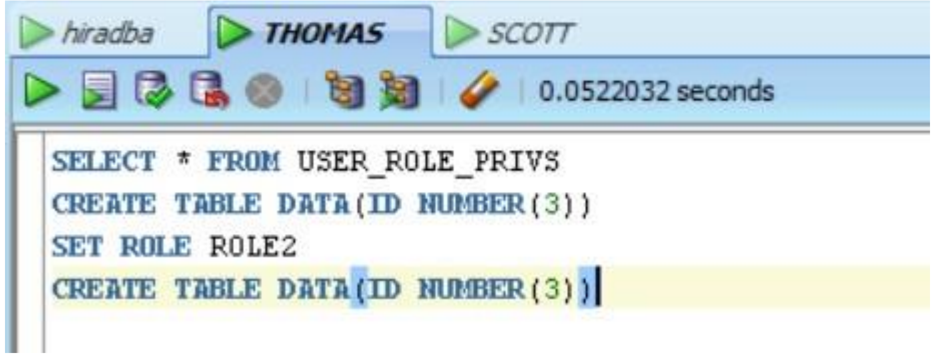
An error was encountered performing the requested operation:

```
ORA-01031: insufficient privileges
01031. 00000 -  "insufficient privileges"
*Cause:    An attempt was made to change the current username or password
           without the appropriate privilege. This error also occurs if
           attempting to install a database without the necessary operating
           system privileges.
           When Trusted Oracle is configure in DBMS MAC, this error may occur
           if the user was granted the necessary privilege at a higher label
           than the current login.
*Action:   Ask the database administrator to perform the operation or grant
           the required privileges.
           For Trusted Oracle users getting this error although granted the
           the appropriate privilege at a higher label, ask the database
           administrator to regrant the privilege at the appropriate label.
Vendor code 1031Error at Line:2
```

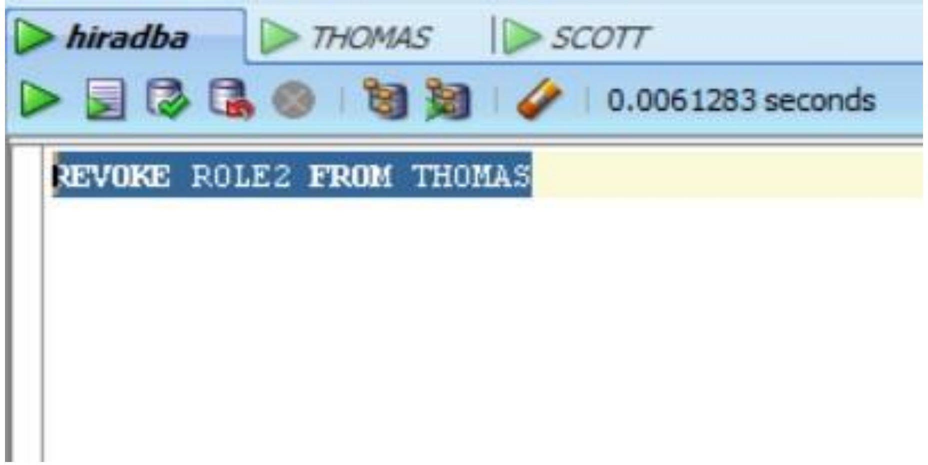


```
SELECT * FROM USER_ROLE_PRIVS
```

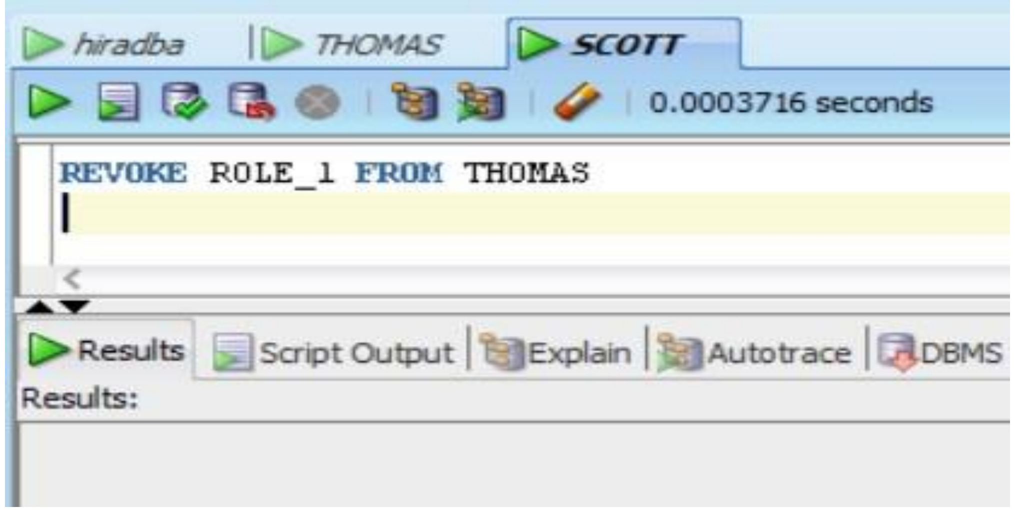
	USERNAME	GRANTED_ROLE	ADMIN_OPTION	DEFAULT_ROLE	OS_GRANTED
1	THOMAS	CONNECT	NO	YES	NO
2	THOMAS	RESOURCE	NO	YES	NO
3	THOMAS	ROLE2	NO	NO	NO
4	THOMAS	ROLE_1	NO	NO	NO



```
SELECT * FROM USER_ROLE_PRIVS
CREATE TABLE DATA(ID NUMBER(3))
SET ROLE ROLE2
CREATE TABLE DATA(ID NUMBER(3))
```

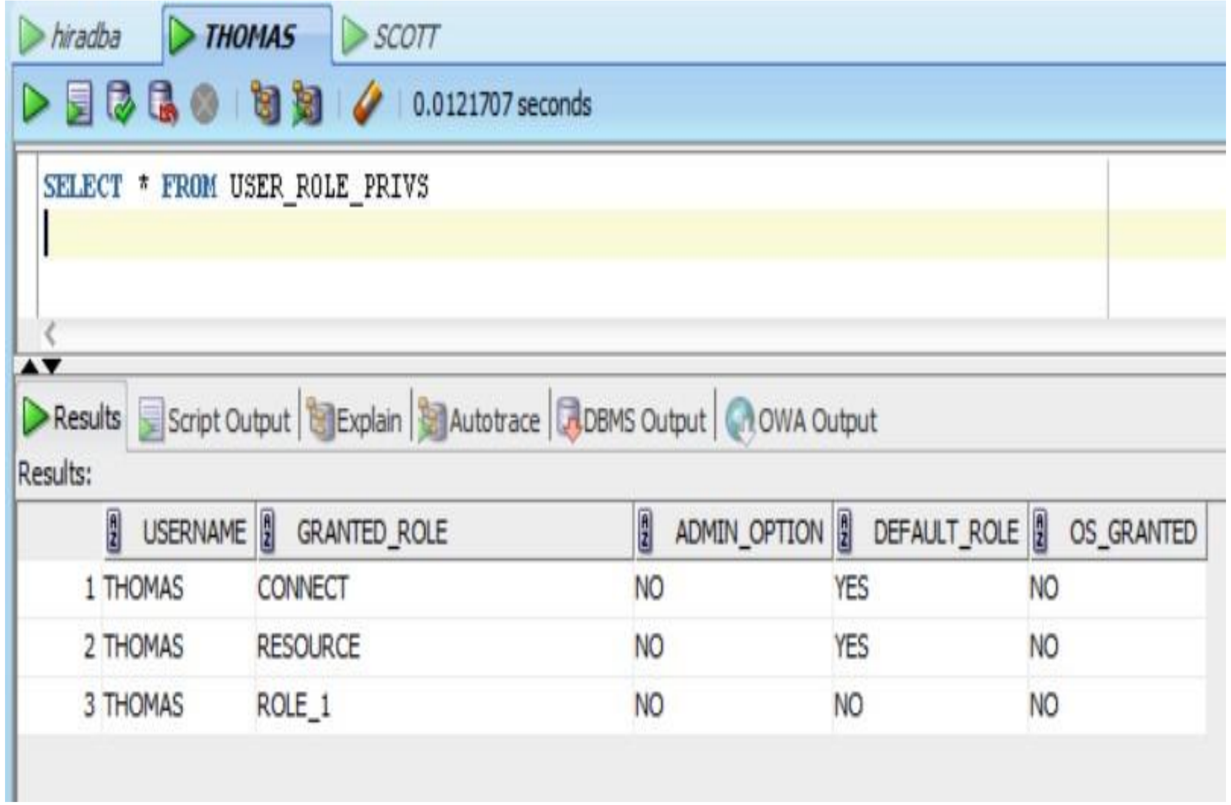


```
REVOKE ROLE2 FROM THOMAS
```



```
REVOKE ROLE_1 FROM THOMAS
```

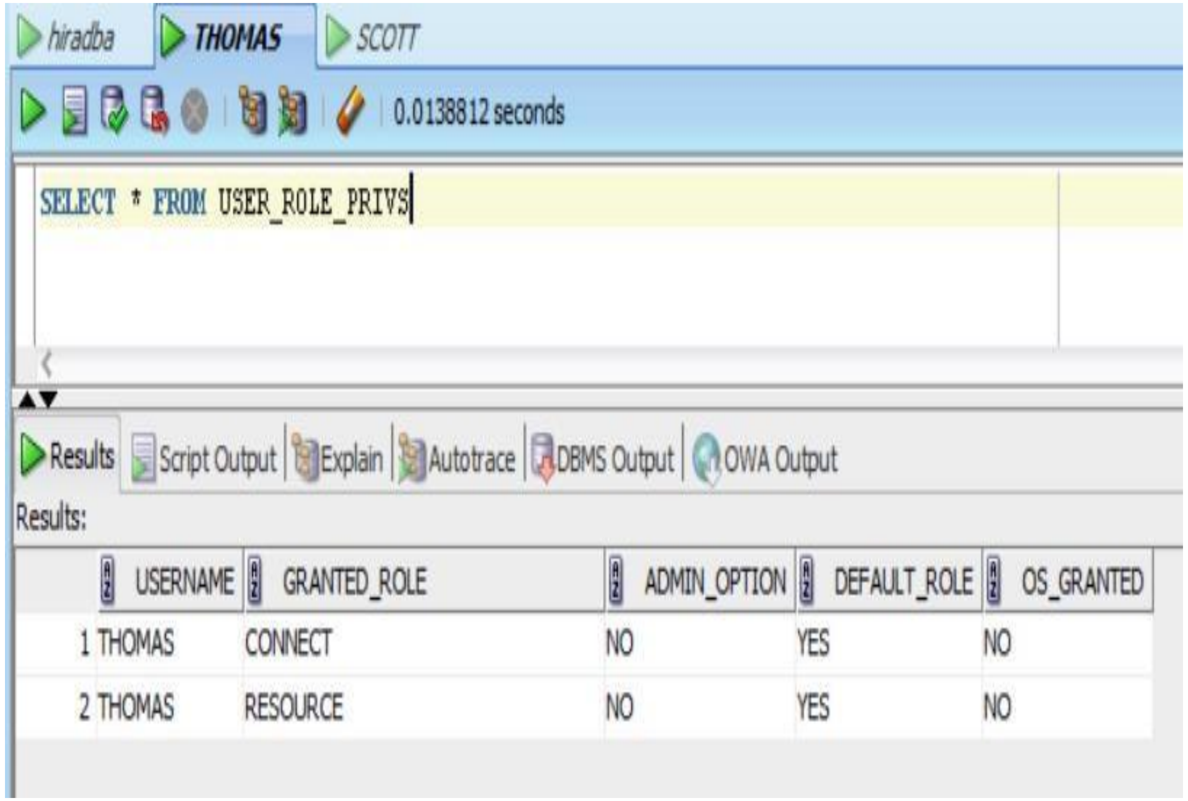
Results:



```
SELECT * FROM USER_ROLE_PRIVS
```

Results:

	USERNAME	GRANTED_ROLE	ADMIN_OPTION	DEFAULT_ROLE	OS_GRANTED
1	THOMAS	CONNECT	NO	YES	NO
2	THOMAS	RESOURCE	NO	YES	NO
3	THOMAS	ROLE_1	NO	NO	NO



```
SELECT * FROM USER_ROLE_PRIVS
```

Results:

	USERNAME	GRANTED_ROLE	ADMIN_OPTION	DEFAULT_ROLE	OS_GRANTED
1	THOMAS	CONNECT	NO	YES	NO
2	THOMAS	RESOURCE	NO	YES	NO

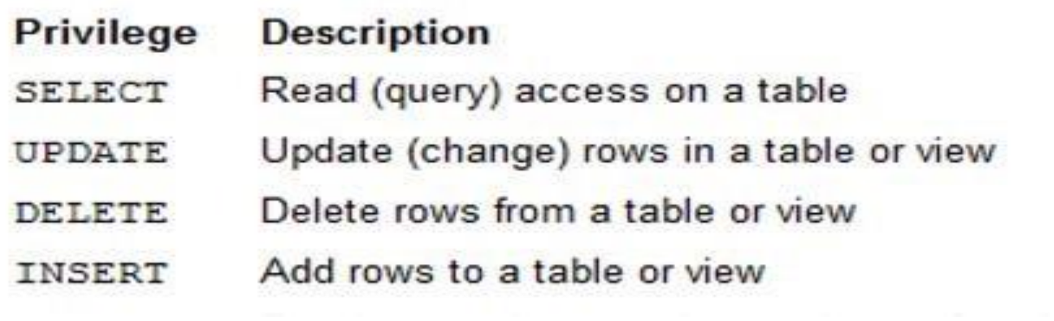
OBJECT PRIVILEGES

- A schema object privilege is the permission to perform a particular action on a specific schema object.
- Different object privileges are available for different types of schema objects. The privilege to delete rows from the departments table is an example of an object privilege.
- Schema object privileges can be granted to and revoked from users and roles. If you grant object privileges to roles, then you can make the privileges selectively available.
- Object privileges allow users to manipulate the contents of database objects in other schemas.
- They are granted to a username in a different schema. In other words, the owner of an object in a schema has all privileges on the object and can grant privileges on the object to another user.

WHO CAN GRANT SCHEMA OBJECT PRIVILEGES?

- A user automatically has all object privileges for schema objects contained in his or her schema.
- A user can grant any object privilege on any schema object he or she owns to any other user or role.
- A user with the **GRANT ANY OBJECT PRIVILEGE** can grant or revoke any specified object privilege to another user with or without the GRANT OPTION of the GRANT statement. Otherwise, the grantee can use the privilege, but cannot grant it to other users.

GRANTING OBJECT PRIVILEGES

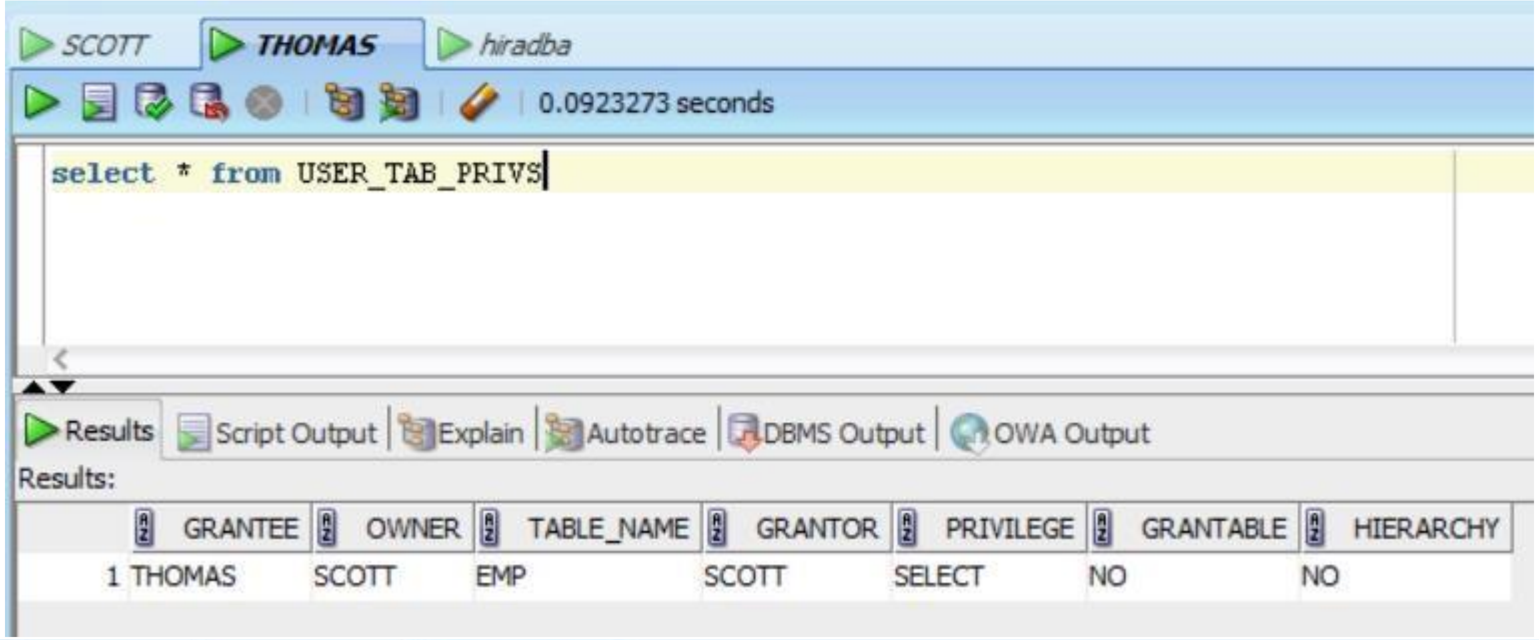


Privilege	Description
SELECT	Read (query) access on a table
UPDATE	Update (change) rows in a table or view
DELETE	Delete rows from a table or view
INSERT	Add rows to a table or view

SYNTAX:

GRANT obj_privilege [(column_list)] [, obj_privilege ...] ON object
TO user [, user, role, PUBLIC ...] [
WITH GRANT OPTION] ;

- The column_list parameter is used if the object is a table and only certain columns of the table are made available for modifications by other users.
- The WITH GRANT OPTION clause allows the grantee to pass the privilege on to yet another user.



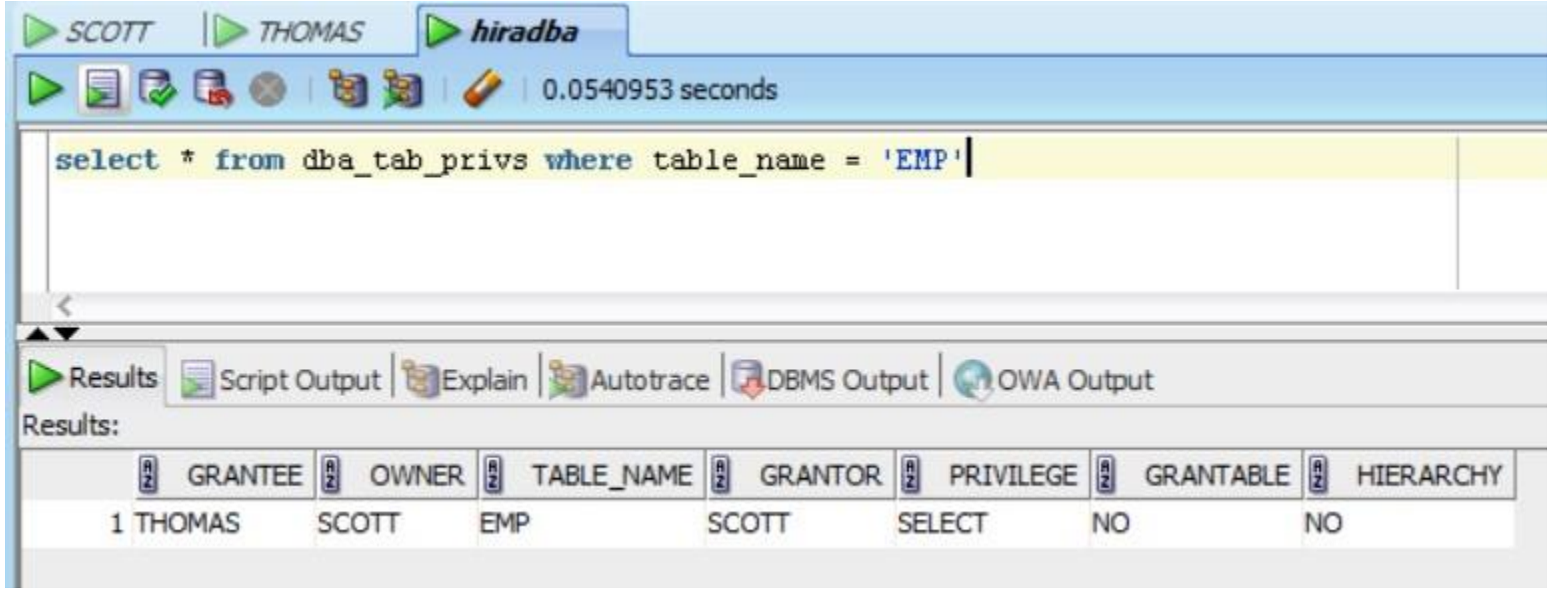
0.0923273 seconds

```
select * from USER_TAB_PRIVS
```

Results

Results:

	GRANTEE	OWNER	TABLE_NAME	GRANTOR	PRIVILEGE	GRANTABLE	HIERARCHY
1	THOMAS	SCOTT	EMP	SCOTT	SELECT	NO	NO



0.0540953 seconds

```
select * from dba_tab_privs where table_name = 'EMP'
```

Results

Results:

	GRANTEE	OWNER	TABLE_NAME	GRANTOR	PRIVILEGE	GRANTABLE	HIERARCHY
1	THOMAS	SCOTT	EMP	SCOTT	SELECT	NO	NO

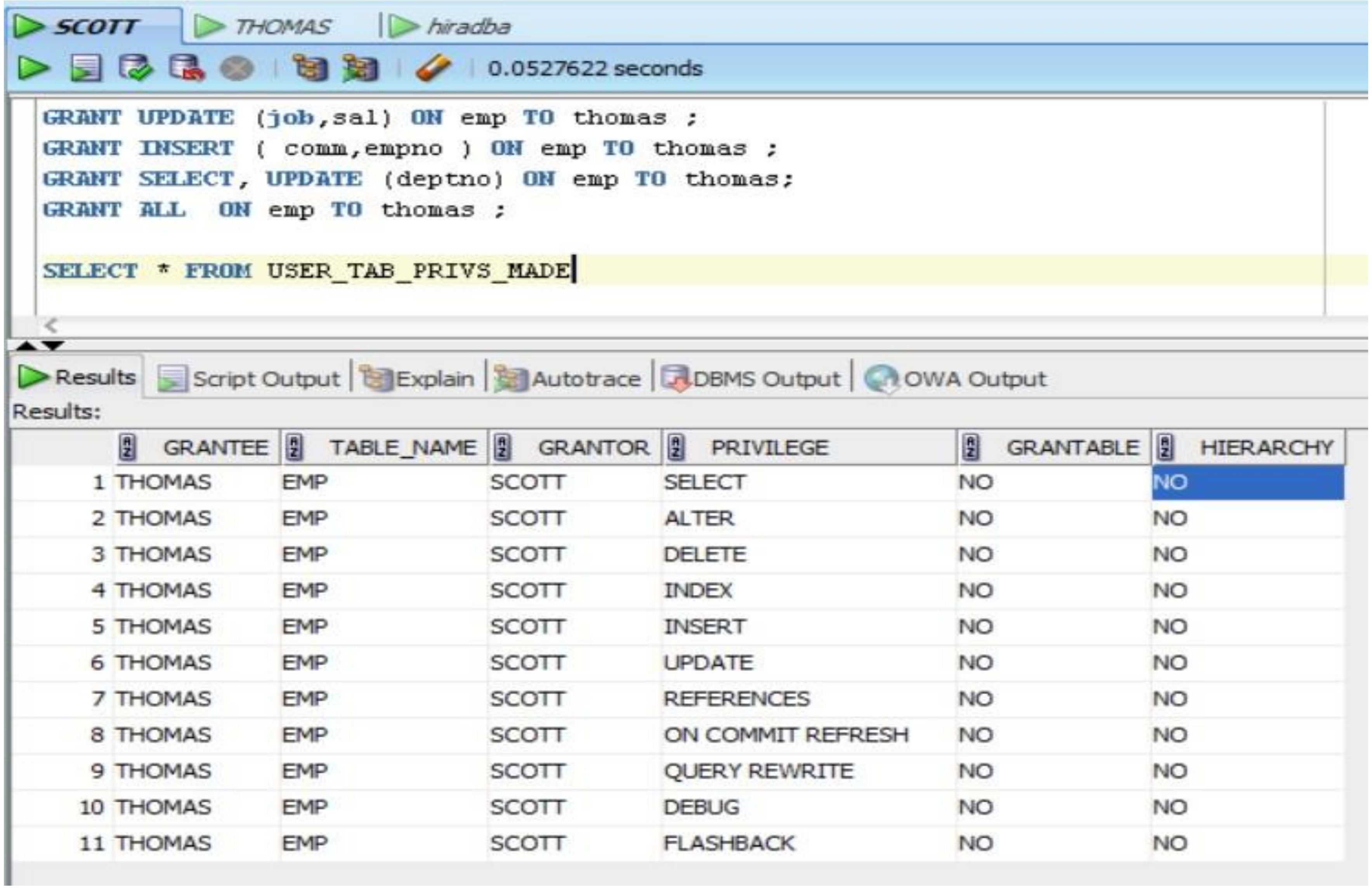
DML OPERATIONS

- One can grant privileges to use the DELETE, INSERT, SELECT, and UPDATE DML operations on a table or view.
- Grant these privileges only to users and roles that need to query or manipulate data in a table.
- One can restrict INSERT and UPDATE privileges for a table to specific columns of the table. With selective INSERT, a privileged user can insert a row with values for the selected columns. All other columns receive NULL or the default value of the column.
- With selective UPDATE, a user can update only specific column values of a row.
- Selective INSERT and UPDATE privileges are used to restrict user access to sensitive data.

EXAMPLES:

1. GRANT SELECT ON emp TO PUBLIC ;
2. GRANT UPDATE (job, sal) ON emp TO thomas ;
3. GRANT UPDATE ON emp TO thomas ;
4. GRANT INSERT (comm,empno) ON emp TO thomas ;
5. GRANT INSERT ON emp TO Thomas ;
6. GRANT SELECT, UPDATE (deptno) ON emp TO Thomas ;
7. GRANT ALL ON emp TO thomas ;

OBJECT PRIVILEGES GRANTED BY SCOTT



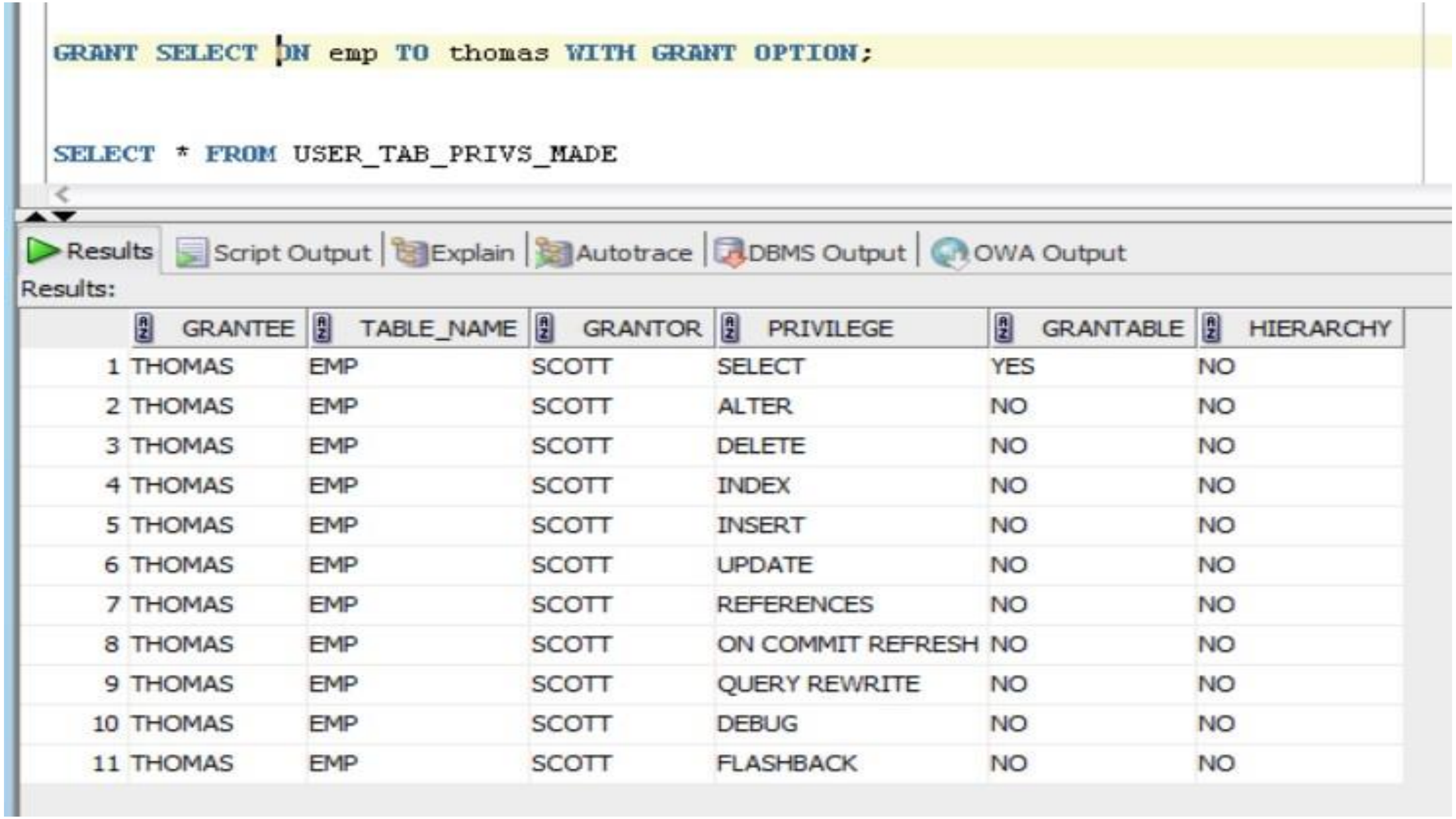
```
GRANT UPDATE (job,sal) ON emp TO thomas ;
GRANT INSERT ( comm,empno ) ON emp TO thomas ;
GRANT SELECT, UPDATE (deptno) ON emp TO thomas;
GRANT ALL  ON emp TO thomas ;

SELECT * FROM USER_TAB_PRIVS_MADE
```

Results:

	GRANTEE	TABLE_NAME	GRANTOR	PRIVILEGE	GRANTABLE	HIERARCHY
1	THOMAS	EMP	SCOTT	SELECT	NO	NO
2	THOMAS	EMP	SCOTT	ALTER	NO	NO
3	THOMAS	EMP	SCOTT	DELETE	NO	NO
4	THOMAS	EMP	SCOTT	INDEX	NO	NO
5	THOMAS	EMP	SCOTT	INSERT	NO	NO
6	THOMAS	EMP	SCOTT	UPDATE	NO	NO
7	THOMAS	EMP	SCOTT	REFERENCES	NO	NO
8	THOMAS	EMP	SCOTT	ON COMMIT REFRESH	NO	NO
9	THOMAS	EMP	SCOTT	QUERY REWRITE	NO	NO
10	THOMAS	EMP	SCOTT	DEBUG	NO	NO
11	THOMAS	EMP	SCOTT	FLASHBACK	NO	NO

GRANT WITH ADMIN OPTION

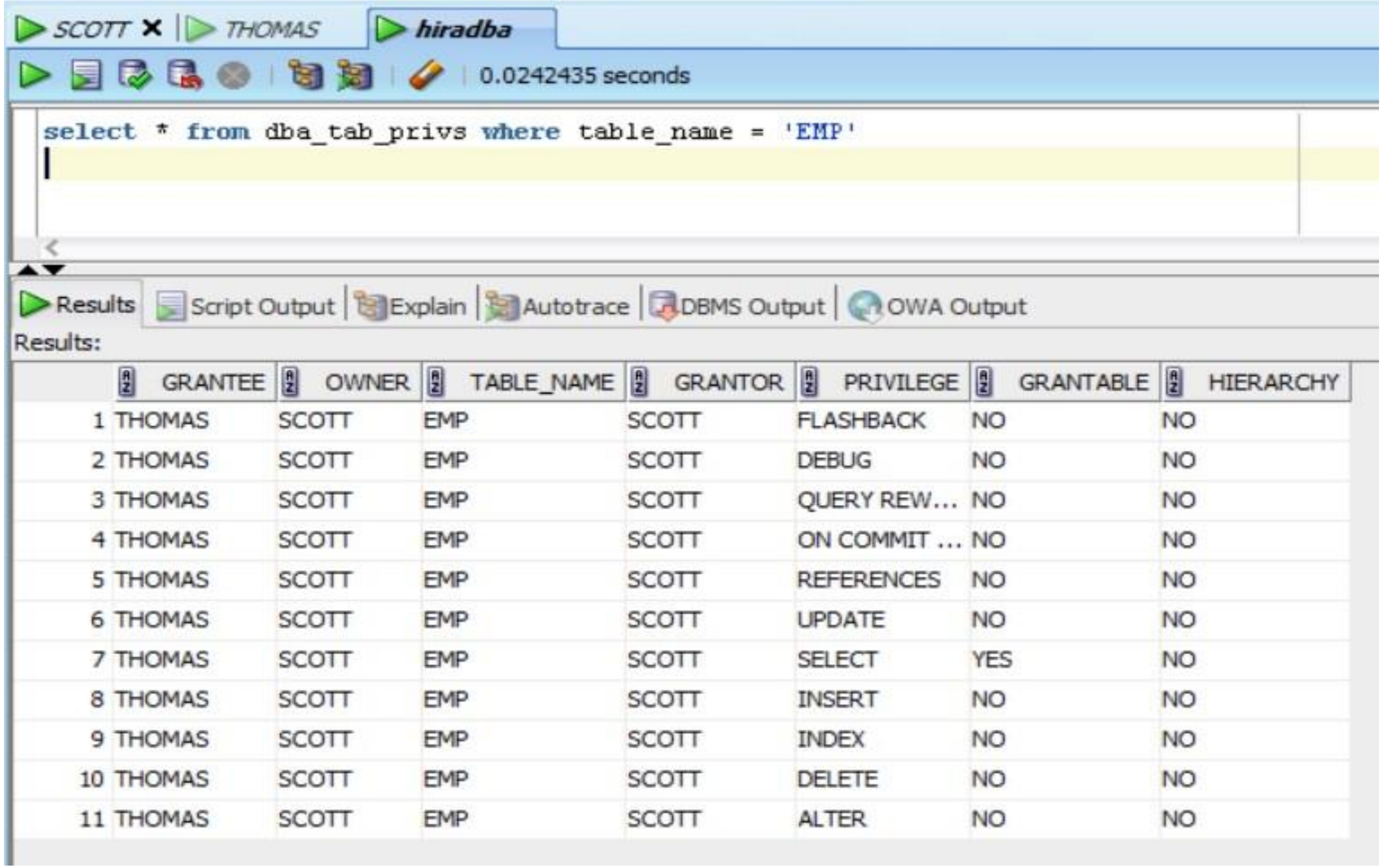


```
GRANT SELECT ON emp TO thomas WITH GRANT OPTION;


SELECT * FROM USER_TAB_PRIVS_MADE
```

Results:

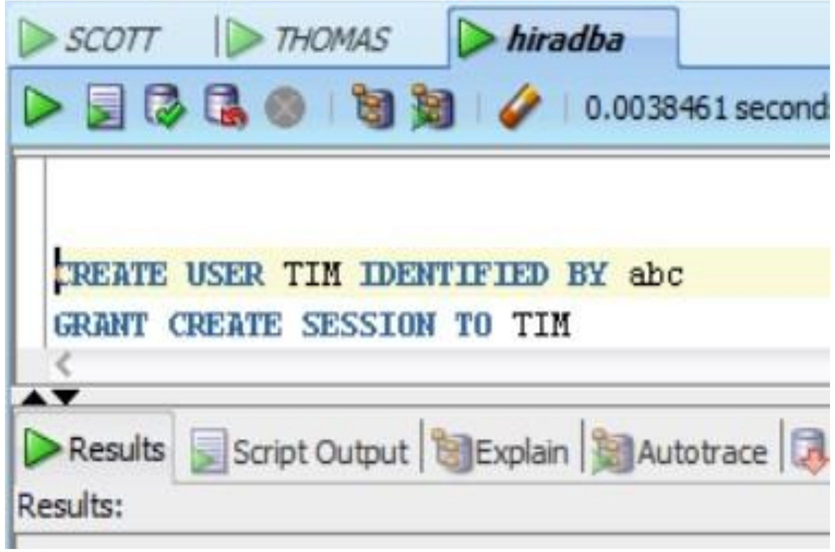
	GRANTEE	TABLE_NAME	GRANTOR	PRIVILEGE	GRANTABLE	HIERARCHY
1	THOMAS	EMP	SCOTT	SELECT	YES	NO
2	THOMAS	EMP	SCOTT	ALTER	NO	NO
3	THOMAS	EMP	SCOTT	DELETE	NO	NO
4	THOMAS	EMP	SCOTT	INDEX	NO	NO
5	THOMAS	EMP	SCOTT	INSERT	NO	NO
6	THOMAS	EMP	SCOTT	UPDATE	NO	NO
7	THOMAS	EMP	SCOTT	REFERENCES	NO	NO
8	THOMAS	EMP	SCOTT	ON COMMIT REFRESH	NO	NO
9	THOMAS	EMP	SCOTT	QUERY REWRITE	NO	NO
10	THOMAS	EMP	SCOTT	DEBUG	NO	NO
11	THOMAS	EMP	SCOTT	FLASHBACK	NO	NO



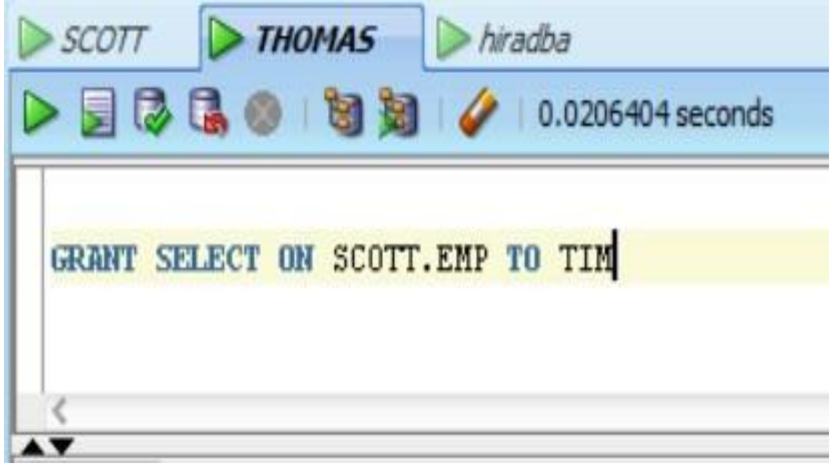
```
select * from dba_tab_privs where table_name = 'EMP'
```

Results:

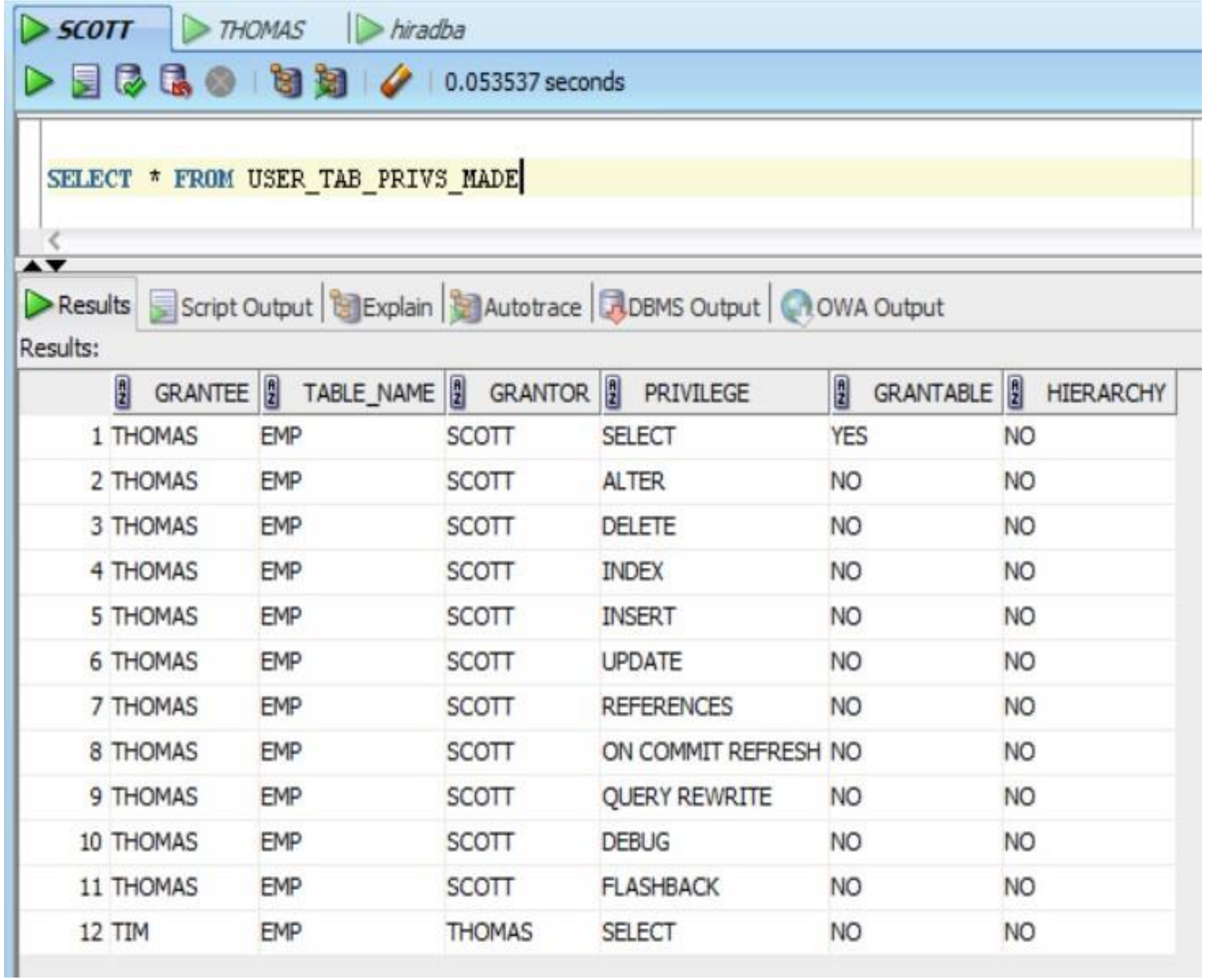
	GRANTEE	OWNER	TABLE_NAME	GRANTOR	PRIVILEGE	GRANTABLE	HIERARCHY
1	THOMAS	SCOTT	EMP	SCOTT	FLASHBACK	NO	NO
2	THOMAS	SCOTT	EMP	SCOTT	DEBUG	NO	NO
3	THOMAS	SCOTT	EMP	SCOTT	QUERY REW...	NO	NO
4	THOMAS	SCOTT	EMP	SCOTT	ON COMMIT ...	NO	NO
5	THOMAS	SCOTT	EMP	SCOTT	REFERENCES	NO	NO
6	THOMAS	SCOTT	EMP	SCOTT	UPDATE	NO	NO
7	THOMAS	SCOTT	EMP	SCOTT	SELECT	YES	NO
8	THOMAS	SCOTT	EMP	SCOTT	INSERT	NO	NO
9	THOMAS	SCOTT	EMP	SCOTT	INDEX	NO	NO
10	THOMAS	SCOTT	EMP	SCOTT	DELETE	NO	NO
11	THOMAS	SCOTT	EMP	SCOTT	ALTER	NO	NO



```
CREATE USER TIM IDENTIFIED BY abc
GRANT CREATE SESSION TO TIM
```



```
GRANT SELECT ON SCOTT.EMP TO TIM
```

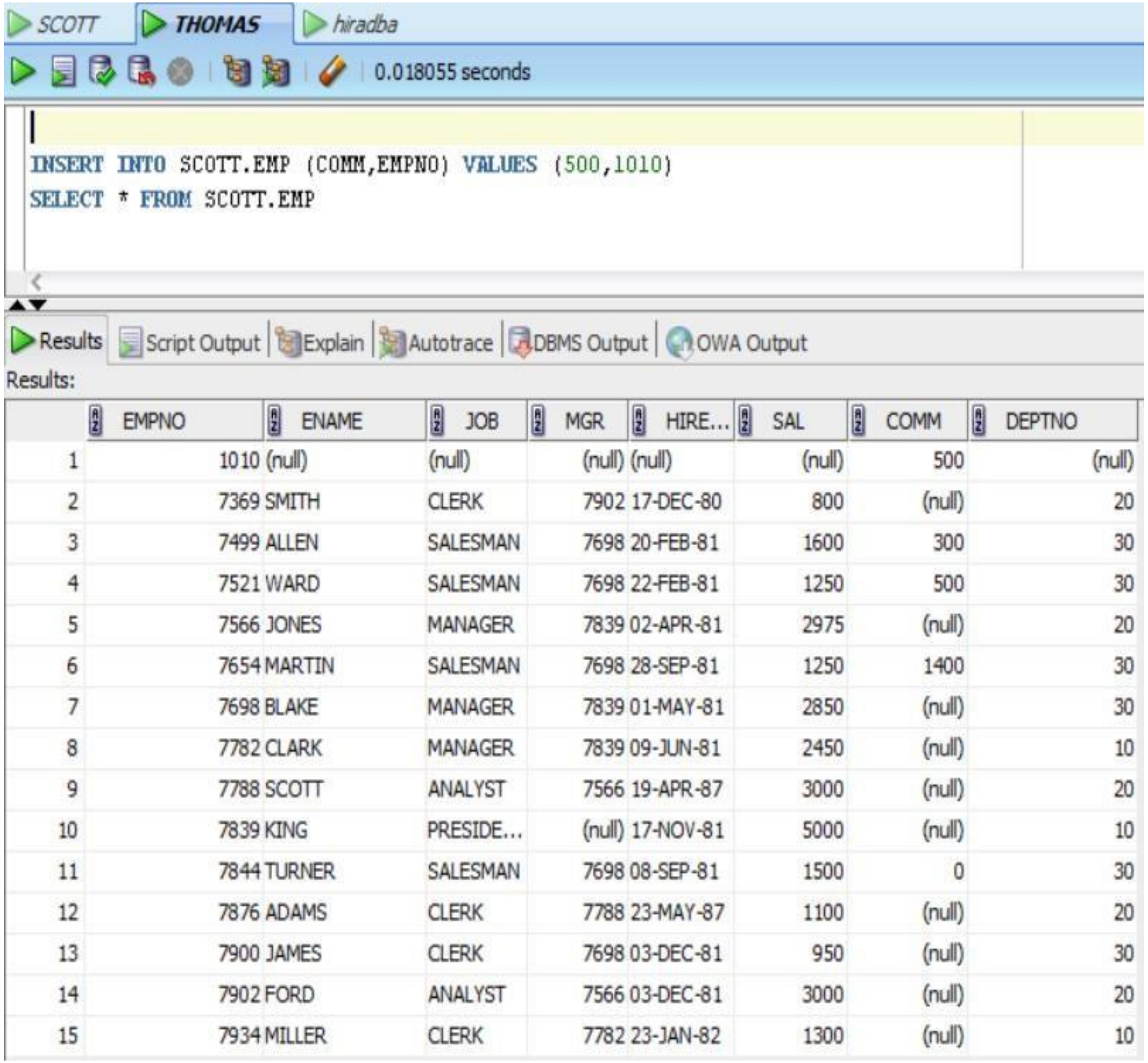


```
SELECT * FROM USER_TAB_PRIVS_MADE
```

Results:

	GRANTEE	TABLE_NAME	GRANTOR	PRIVILEGE	GRANTABLE	HIERARCHY
1	THOMAS	EMP	SCOTT	SELECT	YES	NO
2	THOMAS	EMP	SCOTT	ALTER	NO	NO
3	THOMAS	EMP	SCOTT	DELETE	NO	NO
4	THOMAS	EMP	SCOTT	INDEX	NO	NO
5	THOMAS	EMP	SCOTT	INSERT	NO	NO
6	THOMAS	EMP	SCOTT	UPDATE	NO	NO
7	THOMAS	EMP	SCOTT	REFERENCES	NO	NO
8	THOMAS	EMP	SCOTT	ON COMMIT REFRESH	NO	NO
9	THOMAS	EMP	SCOTT	QUERY REWRITE	NO	NO
10	THOMAS	EMP	SCOTT	DEBUG	NO	NO
11	THOMAS	EMP	SCOTT	FLASHBACK	NO	NO
12	TIM	EMP	THOMAS	SELECT	NO	NO

```
UPDATE scott.emp SET sal = 2500
WHERE empno =  7788 ;
```



```
INSERT INTO SCOTT.EMP (COMM,EMPNO) VALUES (500,1010)
SELECT * FROM SCOTT.EMP
```

Results:

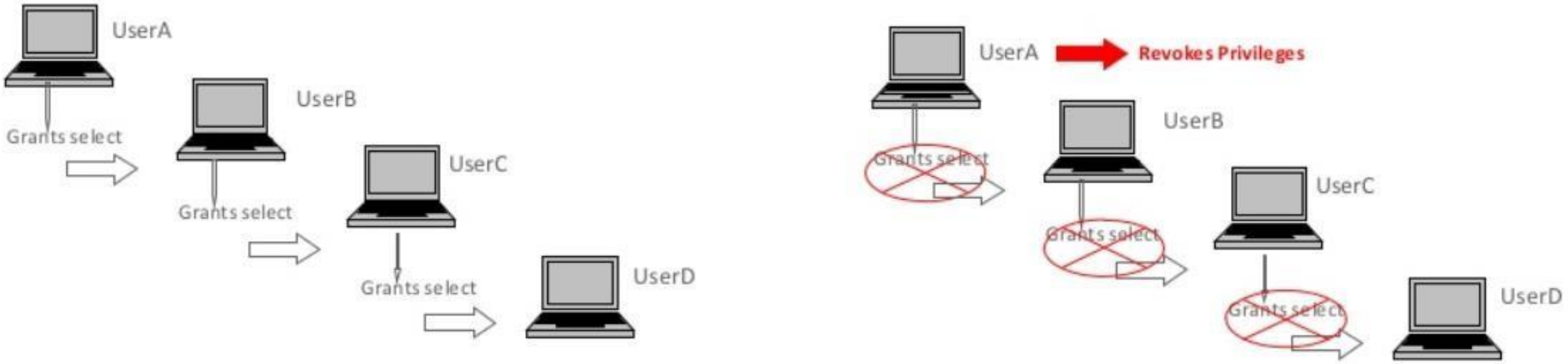
	EMPNO	ENAME	JOB	MGR	HIRE...	SAL	COMM	DEPTNO
1	1010	(null)	(null)	(null)	(null)	(null)	500	(null)
2	7369	SMITH	CLERK	7902	17-DEC-80	800	(null)	20
3	7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
4	7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
5	7566	JONES	MANAGER	7839	02-APR-81	2975	(null)	20
6	7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7	7698	BLAKE	MANAGER	7839	01-MAY-81	2850	(null)	30
8	7782	CLARK	MANAGER	7839	09-JUN-81	2450	(null)	10
9	7788	SCOTT	ANALYST	7566	19-APR-87	3000	(null)	20
10	7839	KING	PRESIDE...	(null)	17-NOV-81	5000	(null)	10
11	7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
12	7876	ADAMS	CLERK	7788	23-MAY-87	1100	(null)	20
13	7900	JAMES	CLERK	7698	03-DEC-81	950	(null)	30
14	7902	FORD	ANALYST	7566	03-DEC-81	3000	(null)	20
15	7934	MILLER	CLERK	7782	23-JAN-82	1300	(null)	10

DDL OPERATIONS

- The ALTER, INDEX, and REFERENCES privileges allow DDL operations to be performed on a table.
- Because these privileges allow other users to alter or create dependencies on a table, you should grant privileges conservatively.
- A user attempting to perform a DDL operation on a table may need additional system or object privileges. For example, to create a trigger on a table, the user requires both the ALTER TABLE object privilege for the table and the CREATE TRIGGER system privilege.
- As with the INSERT and UPDATE privileges, the REFERENCES privilege can be granted on specific columns of a table. The REFERENCES privilege enables the grantee to use the table on which the grant is made as a parent key to any foreign keys that the grantee wishes to create in his or her own tables. This action is controlled with a special privilege because the presence of foreign keys restricts the data manipulation and table alterations that can be done to the parent key.
- A column-specific REFERENCES privilege restricts the grantee to using the named columns (which, of course, must include at least one primary or unique key of the parent table).

REVOKING PRIVILEGES GIVEN WITH GRANT OPTION

- If the owner revokes a privilege from a user who granted privileges to other users, the revoke statement cascades to all privileges granted



REVOKING OBJECT PRIVILEGES

SYNATX:

```
REVOKE { privilege [, privilege...]|ALL } ON object
FROM { user[, user...]|role|PUBLIC }
[ CASCADE CONSTRAINTS ] ;
```

- CASCADE CONSTRAINTS is required to remove any referential integrity constraints made to the object by means of the REFERENCES privilege.

EXAMPLE:

```
REVOKE UPDATE ON emp
FROM thomas
```

Notice that the REVOKE statement did not specify any columns in the EMP table. When revoking UPDATE privileges on a table, columns cannot be specified.