

EXP NO: 16 CONTROLLING USER ACCESS

Roll no:231901007

DATE:

1. Privilege for logging on to the Oracle server

- **Answer:** A user needs the **CREATE SESSION** privilege to log on to the Oracle server.
- This is a **system privilege**.

2. Privilege for creating tables

- **Answer:** A user needs the **CREATE TABLE** privilege. - This is a **system privilege**.

3. Granting privileges to others on your table

- **Answer:** The owner of the table can pass along privileges to other users using the **WITH GRANT OPTION** clause when granting privileges.

4. Simplifying privilege management for many users

- **Answer:** Use **roles** to group privileges and assign the role to users. This makes managing privileges easier.

5. Changing your password

- **Answer:** Use the following command to change your password:

```
``sql
ALTER USER <username> IDENTIFIED BY <new_password>;
``
```

6. Grant and exchange privileges on the `DEPARTMENTS` table

Grant access to your `DEPARTMENTS` table:

```
``sql
GRANT SELECT ON DEPARTMENTS TO <other_user>;
```

Name :

Roll No: 2319010

****Grant back access from the other user:****

```sql

-- Other user grants SELECT on their DEPARTMENTS table to you

GRANT SELECT ON DEPARTMENTS TO <your_username>;

```

**7. Query all rows in your `DEPARTMENTS` table**

```sql

SELECT * FROM DEPARTMENTS;

```

**8. Add rows to `DEPARTMENTS` and query the other team's table**

****Team 1 adds `Education` with department number 500:****

```sql

INSERT INTO DEPARTMENTS (DEPARTMENT_ID, DEPARTMENT_NAME)

VALUES (500, 'Education');

```

****Team 2 adds `Human Resources` with department number 510:****

```sql

INSERT INTO DEPARTMENTS (DEPARTMENT_ID, DEPARTMENT_NAME)

VALUES (510, 'Human Resources');

```

****Query the other team's `DEPARTMENTS` table:****

```sql

SELECT * FROM <other_team_username>.DEPARTMENTS;

```

**9. Query the `USER_TABLES` data dictionary**

```sql

SELECT * FROM USER_TABLES;

```

10. Revoke the SELECT privilege on your table

```
```sql
REVOKE SELECT ON DEPARTMENTS FROM <other_user>;
```
```

11. Remove the row inserted in step 8 and save changes

****Team 1 deletes the `Education` row:****

```
```sql
DELETE FROM DEPARTMENTS WHERE DEPARTMENT_ID = 500;
COMMIT;
```
```

****Team 2 deletes the `Human Resources` row:****

```
```sql
DELETE FROM DEPARTMENTS WHERE DEPARTMENT_ID = 510;
COMMIT;
```
```

Notes:

- Replace `` or `` with the actual usernames.
- Ensure that the database objects and table structures (`DEPARTMENTS`) exist for these commands to work.