Database 2

ORACLE ENVIRONMENT

LAB 1

Outline

Lab 1	Oracle Administration
1	Types of users in Oracle
2	Types of Oracle Clients
3	Connecting to SQLPLUS
4	Creating user and define user privileges
Lab 2	Privileges
1	Grant System Privilege / Object Privileges
2	Revoke System Privilege / Object Privileges
3	Lock and Expire Accounts
4	Oracle Users and User Table

Outline

Lab 3	SQL Developer
1	Create user using/ Create Table
2	Show Connected user and Disconnect
3	Connect Disconnected User
4	Roles and Granting

Types of Users in Oracle

Two user accounts are automatically created with the database and granted the DBA role. They are both created with the password that you supplied upon installation,

These two user accounts are:

- **SYS**
- **SYSTEM**

Admin user ‘Sys’

- This account can perform all administrative functions.
- All base (underlying) tables and views for the database data dictionary are stored in the SYS schema.
- These base tables and views are critical for the operation of Oracle Database.
- To maintain the integrity of the data dictionary, tables in the SYS schema are manipulated only by the database.
- They should never be modified by any user or database administrator.
- You must not create any tables in the SYS schema.

→ **The SYS user is granted the SYSDBA privilege, which enables a user to perform high-level administrative tasks such as backup and recovery.**

Admin user

Administration tasks as

Shutdown Server.

Backup and Recovery.

Create any User.

Drop any table.

Disconnect any user.

Prevent any user from connecting.

Export and import Data.

Regular User ‘SYSTEM’ Pre Installed User

This account can perform all administrative functions except the following:

- Backup and recovery
- Database upgrade

While this account can be used to perform day-to-day administrative tasks, Oracle strongly recommends creating named users account for administering the Oracle database to enable monitoring of database activity.

Three Clients of ORACLE

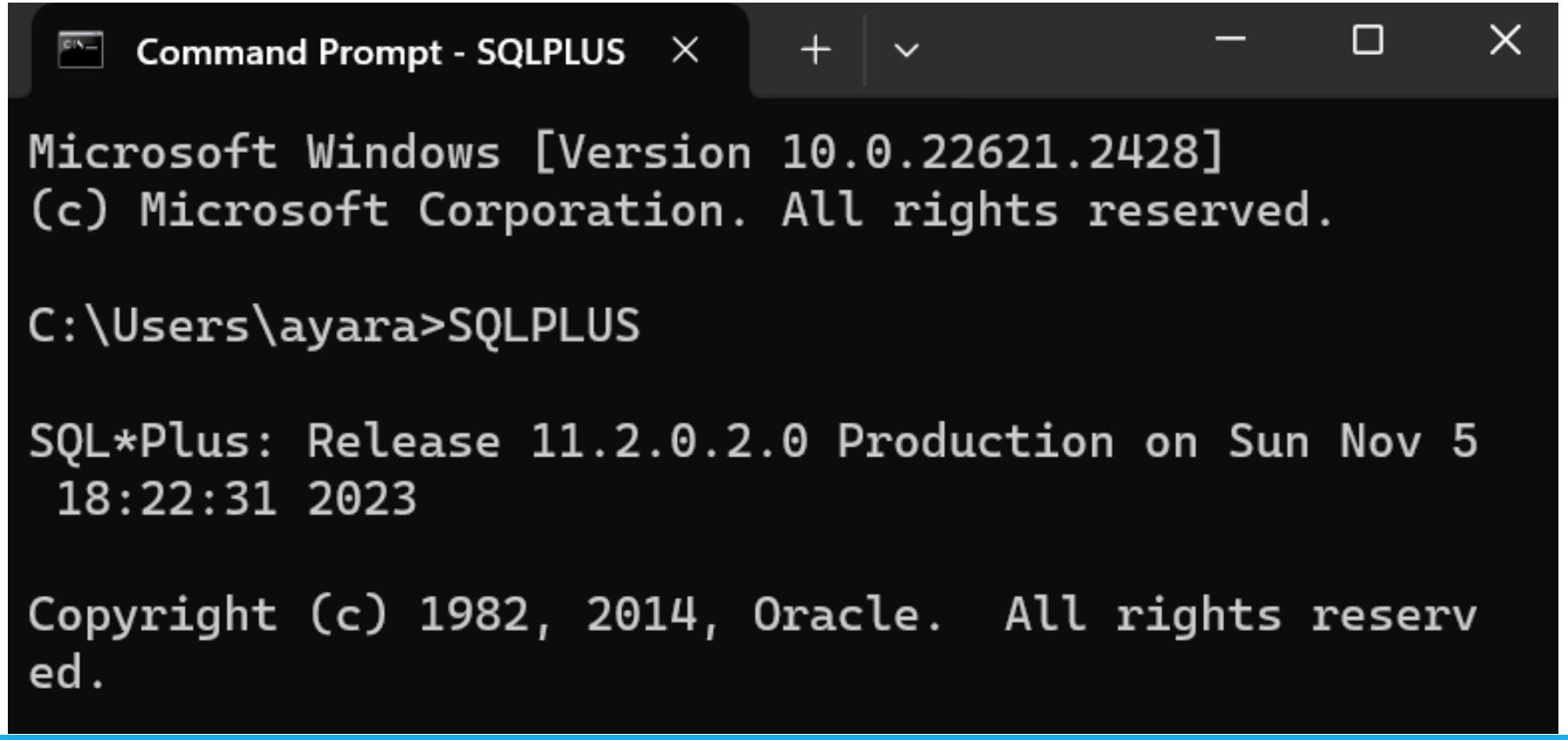
1. **SQL PLUS :** It’s the Primary Client of ORACLE.
   - **CMD** Command Line Based.
2. **ISQLPLUS**: It’s an extension to SQLPLUS.
   - **GUI** --> Graphical User Interface, It's a browser Based.
3. **ORACLE SQL Developer**
   - Developed by Oracle.
   - GUI Based
   - Connect and get data to any other DBMS as from MYSQL server, SQL Server, SYBASE Server, Access, DB2.

Connecting to SQLPLUS

1. CMD
2. SQLPLUS
3. RUN SQL COMMAND LINE

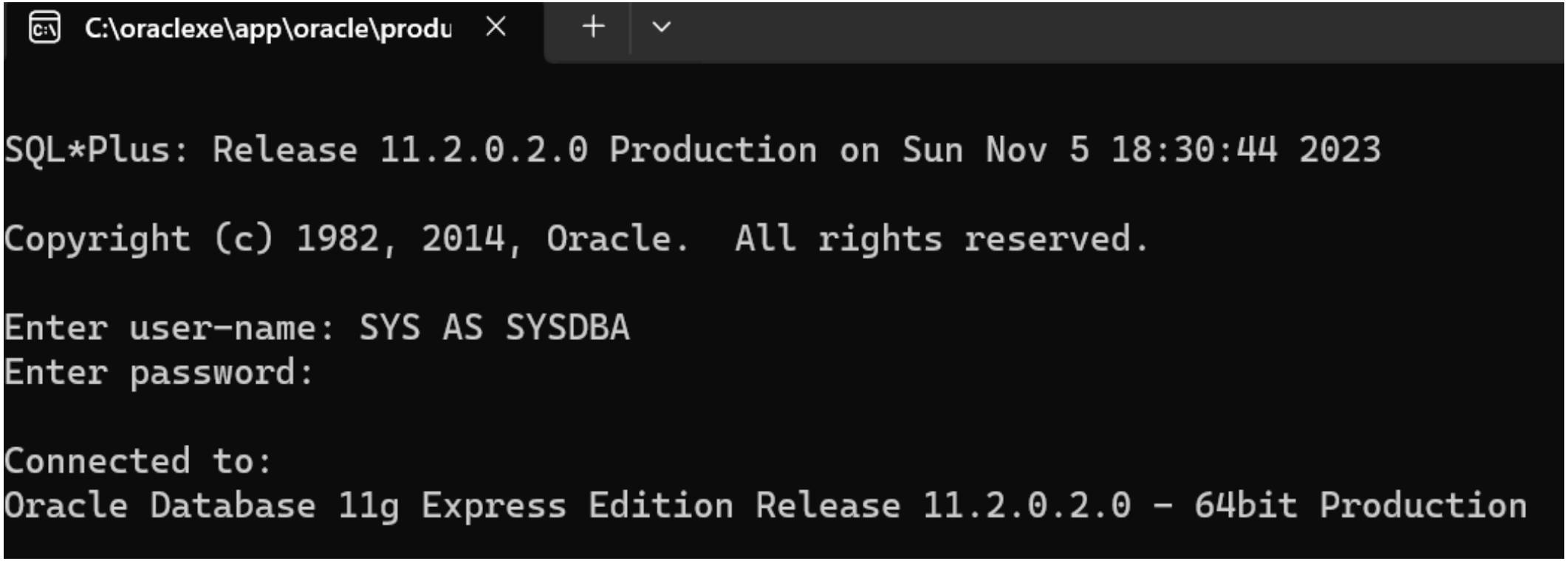
CMD

Open CMD


Command Prompt - SQLPLUS
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.
C:\Users\ayara>SQLPLUS
SQL*Plus: Release 11.2.0.2.0 Production on Sun Nov 5
18:22:31 2023
Copyright (c) 1982, 2014, Oracle. All rights reserv
ed.

SQLPLUS

Open SQLPLUS.exe



SQL COMMAND LINE

Open Run SQL Command Line

- **ADMIN USER**

  CONN / AS SYSDBA

  CONN SYS AS SYSDBA

- **REGULAR USER**

CONN USER / PASSWORD

```
SQL> CONN / AS SYSDBA
Connected.
SQL> CONN SYS AS SYSDBA
Enter password:
Connected.
SQL> CONN ALI / 123
Connected.
SQL> CONNECT ALI / 123
Connected.
```

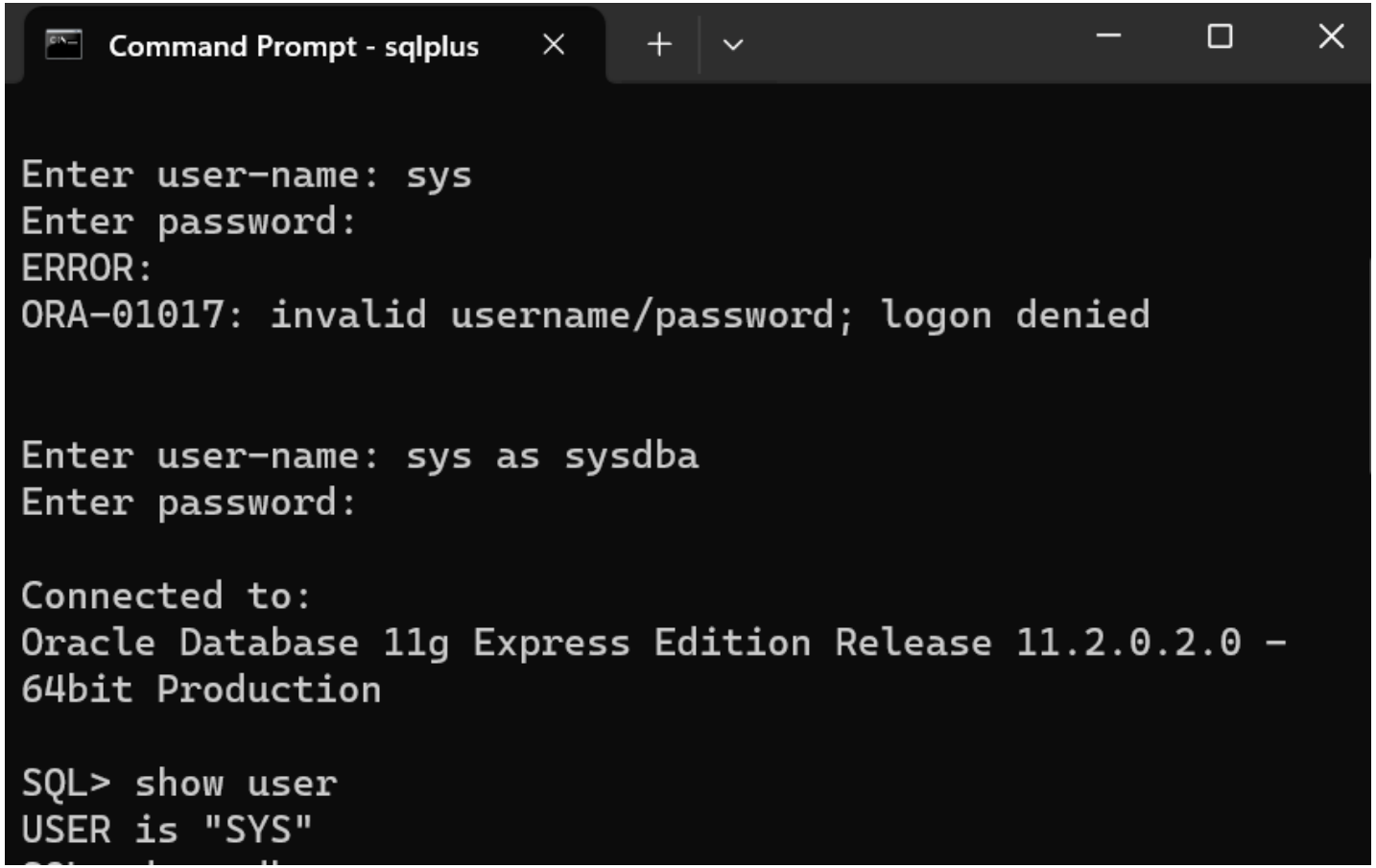
Login by User Admin ‘SYS’

USER NAME:

SYS AS SYSDBA

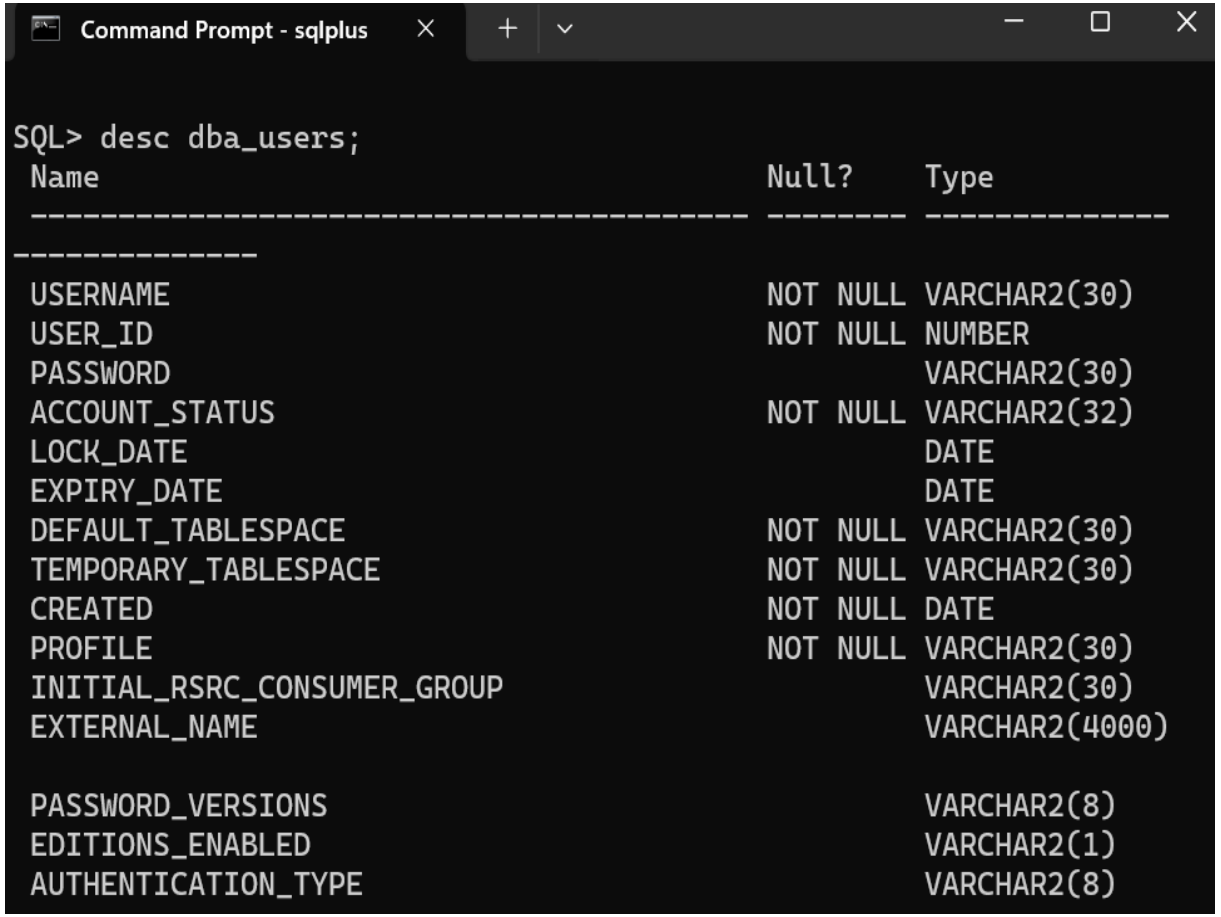
PASSWORD:

ADMIN



Show List of users table columns

DESC DBA_USERS;



Show users from DBA

```
SELECT USERNAME,
CREATED FROM
DBA_USERS;
```

```
SQL> SELECT USERNAME, CREATED FROM DBA_USERS;

USERNAME                       CREATED
------------------------------ ---------
SYS                            29-MAY-14
SYSTEM                         29-MAY-14
ANONYMOUS                      29-MAY-14
APEX_PUBLIC_USER               29-MAY-14
FLOWS_FILES                    29-MAY-14
APEX_040000                    29-MAY-14
OUTLN                          29-MAY-14
DIP                            29-MAY-14
ORACLE_OCM                     29-MAY-14
XS$NULL                        29-MAY-14
MDSYS                          29-MAY-14

USERNAME                       CREATED
------------------------------ ---------
CTXSYS                         29-MAY-14
DBSNMP                         29-MAY-14
XDB                            29-MAY-14
APPQOSSYS                      29-MAY-14
HR                             29-MAY-14

16 rows selected.
```

Create User using SQLPLUS

CREATE USER ALI

IDENTIFIED BY 123

```
SQL> create user ali identified by 123;

User created.
```

Connecting User

User (ALI) can not connect to the server!

He needs privilege….

```
SQL> conn ali/123
ERROR:
ORA-01045: user ALI lacks CREATE SESSION privilege;
logon denied
```

User Privileges

1. System Privileges

Right access to specific users to do **administration tasks**.

Examples

Create, Drop, Disconnect User

Recovery and Backups

Export and Import

2. Object Privileges

Right access to specific users to do **actions on objects**.

Examples

select, insert, delete, update

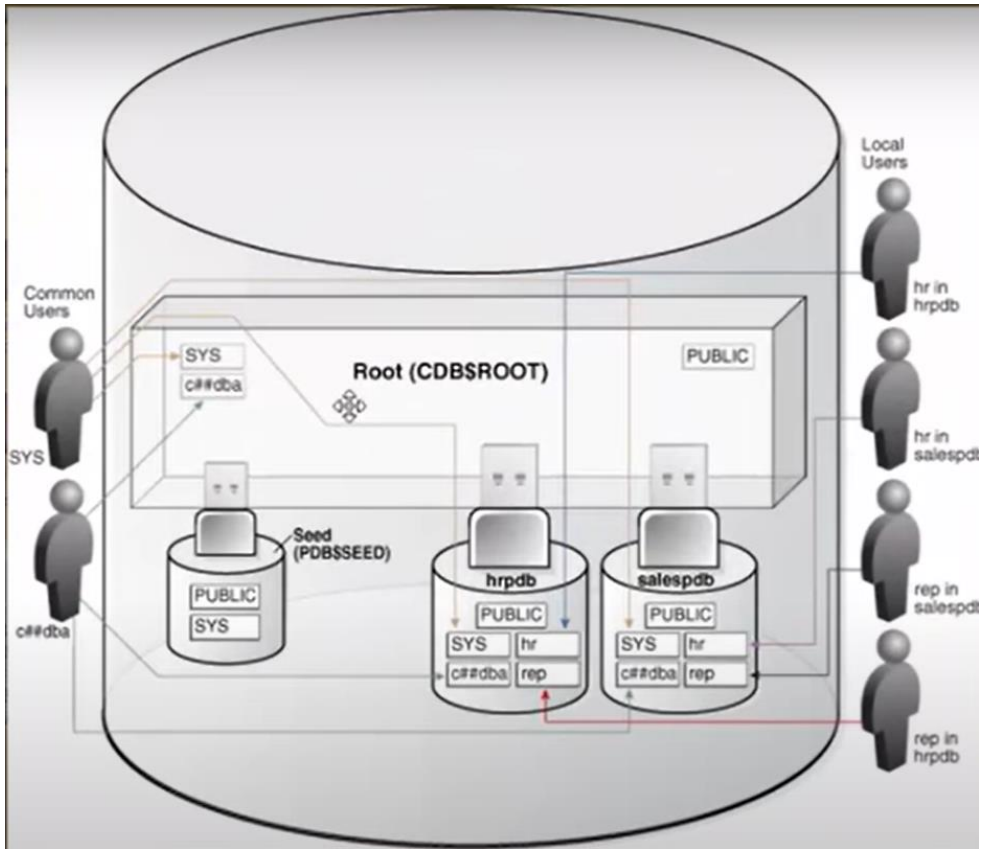
Schema

- A schema is a collection of database objects.
- A schema is owned by a database user and has the same name as that user.
- Schema objects are logical structures created by users.
- Objects such as tables or indexes hold data, or can consist of a definition only, such as a view or synonym.

Database Objects

- Tables
- Triggers
- Function
- procedures
- Indexes
- Sequence
- synonyms

Schema


Local Users
Common Users
SYS
c##dba
Root (CDB$ROOT)
SYS
c##dba
PUBLIC
Seed (PDB$SEED)
PUBLIC
SYS
hrpdb
PUBLIC
SYS
hr
c##dba
rep
salespdb
PUBLIC
SYS
hr
c##dba
rep
hr in hrpdb
hr in salespdb
rep in salespdb
rep in hrpdb

GRANT PRIVILEGES STATMENT

Use the GRANT statement to give privileges to a specific user or role, or to all users, to perform actions on database objects.

The following types of privileges can be granted:

- Delete data from a specific table.
- Insert data into a specific table.
- Create a foreign key reference to the named table or to a subset of columns from a table.
- Select data from a table, view, or a subset of columns in a table.
- Create a trigger on a table.
- Update data in a table or in a subset of columns in a table.
- Run a specified function or procedure.
- Use a sequence generator or a user-defined type

Granting System Privileges to User

GRANT SYSTEM_PRIVILEGES TO USERNAME;

System Privileges	Command
User Session	**GRANT CREATE SESSION TO USERNAME;**
Create Table	**GRANT CREATE TABLE TO USERNAME;**
Table Space	**ALTER USER USERNAME QUOTA SIZE ON SYSTEM;**
Lock Account	**ALTER USER USERNAME ACCOUNT LOCK;**
Unlock Account	**ALTER USER USERNAME ACCOUNT UNLOCK;**
Change Password	**ALTER USER USERNAME IDENTIFIED BY NEWPASSORD;**
EXPIRE PASSWORD	**ALTER USER ALI PASSWORD EXPIRE;**

USER COMMANDS

USER ROLE	Command
Create Table	CREATE TABLE TABLE_NAME (COLOMNE DATA_TYPE);
INSERT Statement	INSERT INTO TABLE_NAME VALUES ();
Select User's tables	SELECT TABLE_NAME FROM USER_TABLE;

GRANT SESSION TO USER

Use the GRANT statement to give privileges to a specific user or role, or to all users, to perform actions on database objects.

```
SQL> GRANT CREATE SESSION TO ALI;

Grant succeeded.
```

Create Table Privileges

```
SQL> GRANT CREATE TABLE TO ALI;

Grant succeeded.

SQL> ALTER USER ALI QUOTA 100M ON SYSTEM;

User altered.
```

ERROR MESSAGES

```
SQL> Create table employee (id int);
Create table employee (id int)
*
ERROR at line 1:
ORA-01031: insufficient privileges

SQL> Create table employees (id int);
Create table employees (id int)
*
ERROR at line 1:
ORA-01536: space quota exceeded for tablespace 'SYSTEM'
```

SHOW USER'S TABLE

Use **USER_TABLES** table to show all tables owned by user.

```
SQL> select table_name from user_tables;

TABLE_NAME
------------------------------
EMPLOYEE
```

GRANT OBJECT PRIVILEGES

GRANT OBJECT_PRIVILEGES ON OBJECT_NAME TO USER;

- **Object_Privileges:**
  - Select, Insert, Update, Delete.
- **Object_Name:**
  - Tables, Functions, Procedures…

GRANT OBJECT PRIVILEGES

GRANT OBJECT_PRIVILEGES ON OBJECT_NAME TO USERNAME;

Object Privileges	Command
Insert	GRANT INSERT ON TABLE TO USERNAME;
Select	GRANT SELECT ON TABLE TO USERNAME;
Update	GRANT UPDATE ON TABLE TO USERNAME;
Delete	GRANT DELETE ON TABLE TO USERNAME;
More Than Privileges	GRANT INSERT, SELECT, DELETE ON TABLE TO USERNAME;
All	GRANT ALL ON TABLE TO USERNAME;

Insert values command

```
SQL> insert into employee values (1);

1 row created.

SQL> insert into employee values (2);

1 row created.

SQL> insert into employee values (3);

1 row created.
```

Grant User object Privilege to Other User ‘GRANT SELECT PRIVILEGE’

Ahmed wants to insert data into Ali’s table.

- Let’s give him this **object privilege**

```
SQL> conn ali / 123
Connected.
SQL> grant select on employee to ahmed;

Grant succeeded.
```

```
SQL> conn ahmed/123
Connected.
SQL> select * from ali.employee;

        ID
----------
         1
         2
         3
```

Grant User object Privilege to Other User ‘GRANT INSERT PRIVILEGE’

Ahmed wants to insert row

into Ali’s table!

- Let’s give him this privilege…

Don’t forget to commit changes.

```
SQL> conn ali / 123
Connected.
SQL> grant insert on employee to ahmed;

Grant succeeded.
```

```
SQL> conn ahmed/123
Connected.
SQL> insert into ali.employee values (4);

1 row created.

SQL> commit;

Commit complete.

SQL> select * from ali.employee;

        ID
----------
         1
         2
         3
         4
```

GRANT All Privileges to User

```
SQL> grant insert, select , delete on employee to ahmed;

Grant succeeded.

SQL> grant all on employee to ahmed;

Grant succeeded.
```

REVOKE PRIVILEGES

Use the REVOKE statement to remove privileges from a specific user or role, or from all users, to perform actions on database objects.

The following types of privileges can be revoked:

- Delete data from a specific table.
- Insert data into a specific table.
- Create a foreign key reference to the named table or to a subset of columns from a table.
- Select data from a table, view, or a subset of columns in a table.
- Create a trigger on a table.
- Update data in a table or in a subset of columns in a table.
- Run a specified routine (function or procedure).
- Use a sequence generator or a user-defined type.

REVOKE SYSTEM / OBJECT PRIVILEGES

REVOKE SYSTEM_PRIVILEGES FROM USERNAME;

System Privileges	Command
User Session	REVOKE CREATE SESSION FROM USERNAME;
Create Table	REVOKE CREATE TABLE FROM USERNAME;

Object Privileges	Command
INSERT	REVOKE INSERT ON OBJECT FROM USERNAME;
More Than Privilege	REVOKE INSERT, SELECT, DELETE ON OBJECT FROM USERNAME;
ALL	REVOKE ALL ON OBJECT FROM USERNAME;

REVOKE SYSTEM PRIVILEGES

```
SQL> revoke create table from ali;

Revoke succeeded.
```

```
SQL> revoke create session from ali;

Revoke succeeded.
```

REVOKE OBJECT PRIVILEGES

```
SQL> REVOKE insert, select , delete on employee from ahmed;

Revoke succeeded.

SQL> REVOKE ALL on employee from ahmed;

Revoke succeeded.
```

LOCK / UNLOCK ACCOUNT PRIVILEGES

```
SQL> CONN / AS SYSDBA
Connected.
SQL> ALTER USER Ali ACCOUNT LOCK;

User altered.

SQL> CONN ali/ 123
ERROR:
ORA-28000: the account is locked
```

```
SQL> CONN / AS SYSDBA
Connected.
SQL> ALTER USER Ali ACCOUNT UNLOCK;

User altered.

SQL> CONN ali/ 123
Connected.
```

CHANGE / EXPIRE PASSWORD

```
SQL> CONN ali/ 123
Connected.
SQL> CONN / AS SYSDBA
Connected.
SQL> ALTER USER Ali IDENTIFIED BY 111;

User altered.

SQL> CONN ali/ 111
Connected.
```

```
SQL> CONN / AS SYSDBA
Connected.
SQL> ALTER USER Ali PASSWORD EXPIRE;

User altered.

SQL> CONN ali/ 111
ERROR:
ORA-28001: the password has expired


Changing password for ali
New password:
Retype new password:
Password changed
Connected.
SQL> CONN ali/ 222
Connected.
```

Thank you!