Assignment 6 – Theory Summer 2018

***J. Packy Laverty***

***laverty@rmu.edu***

Contents

[**Assignment 6 Theory 4**](#_Toc508290613)

[**1.0 Oracle Architecture 4**](#_Toc508290614)

[**1.1 Database, Instances, Server Processes and User Sessions 5**](#_Toc508290615)

[**1.2 Oracle Connections 6**](#_Toc508290616)

[**1.2.1 Client Server Connections 6**](#_Toc508290617)

[**1.3 Oracle Environmental Variables 8**](#_Toc508290618)

[**1.4 Questions – Oracle Architecture 9**](#_Toc508290619)

[**2.0 Introduction to Oracle Security 10**](#_Toc508290620)

[**2.1 An Integrated Approach to Security 11**](#_Toc508290621)

[**2.2 Oracle Authentication and Authorization 11**](#_Toc508290622)

[**2.3 Oracle CONNECT Command 12**](#_Toc508290623)

[**2.4 Oracle Authentication Methods 12**](#_Toc508290624)

[**2.4.1 Authentication by Operating Systems 12**](#_Toc508290625)

[**2.4.2 Authentication by the Network and LDAP Directories 13**](#_Toc508290626)

[**2.4.3 Authentication by Oracle Database 13**](#_Toc508290627)

[**2.5 Basic Oracle security Commands. 14**](#_Toc508290628)

[**2.6 Oracle Built-in Default Users 15**](#_Toc508290629)

[**2.6.1 Questions - Introduction to Oracle Security 16**](#_Toc508290630)

[**2.7 Create User Command 17**](#_Toc508290631)

[**2.7.1 IDENTIFIED BY 18**](#_Toc508290632)

[**2.7.2 DEFAULT TABLE SPACE 18**](#_Toc508290633)

[**2.7.3 User Quotas 19**](#_Toc508290634)

[**2.7.4 ALTER or DROP USER 20**](#_Toc508290635)

[**2.7.5 Questions – CREATE USER 20**](#_Toc508290636)

[**2.8 Oracle Profiles 21**](#_Toc508290637)

[**2.8.1 Creating Oracle Profiles 22**](#_Toc508290638)

[**2.8.2 Password Complexity and Resource Settings 22**](#_Toc508290639)

[**2.8.3 Questions – Oracle Profiles 23**](#_Toc508290640)

[**2.9 GRANT Command and Privileges 24**](#_Toc508290641)

[**2.9.1 System and Object Privileges 25**](#_Toc508290642)

[**2.9.2 Grant Command Examples 27**](#_Toc508290643)

[**2.9.3 WITH GRANT OPTION 28**](#_Toc508290644)

[**2.9.4 Questions - GRANT Command and Privileges 29**](#_Toc508290645)

[**2.10 Revoke Command 30**](#_Toc508290646)

[**2.11 CREATE ROLE 31**](#_Toc508290647)

[**2.11.1 How to Assign Privileges to Roles 31**](#_Toc508290648)

[**2.11.2 Oracle Pre-Defined Roles 33**](#_Toc508290649)

[**2.11.3 Assigning Privileges and Users to a Role 34**](#_Toc508290650)

[**2.11.4 Viewing Role Privilege 34**](#_Toc508290651)

[**2.11.5 Assigning Users to Roles 35**](#_Toc508290652)

[**2.11.6 Questions - Oracle ROLES 36**](#_Toc508290653)

[**3.0 Oracle’s Data Dictionary 37**](#_Toc508290654)

[**3.1 USER\_CATALOG (CAT) and USER\_OBJECTS (OBJ) 38**](#_Toc508290655)

[**3.2 USER\_TABLES (TABS) 39**](#_Toc508290656)

[**3.3 USER\_TAB\_COLUMNS (COLS) 40**](#_Toc508290657)

[**3.4 USER\_CONSTRAINTS and USER\_CONS\_COLUMNS 41**](#_Toc508290658)

[**3.5 USER\_INDEXES (IND) and USER\_IND\_COLUMNS 44**](#_Toc508290659)

[**3.6 USER\_VIEWS and USER\_SYNONYMS (SYN) 45**](#_Toc508290660)

[**3.7 Questions – Oracle System Tables and Data Dictionary 46**](#_Toc508290661)

[**4.0 Oracle Securing Data and Database Auditing 48**](#_Toc508290662)

[**4.1 Securing Data - Transparent data encryption 48**](#_Toc508290663)

[**4.1.1 What is Transparent data encryption 48**](#_Toc508290664)

[**4.1.2 When Should You Encrypt Data? 48**](#_Toc508290665)

[**4.1.3 How Transparent Data Encryption Works 48**](#_Toc508290666)

[**4.2 Securing Data - Oracle Virtual Private Database (VPD). 50**](#_Toc508290667)

[**4.3 Securing Data - Oracle Label Security (OLS) 52**](#_Toc508290668)

[**4.4 Securing Data - Oracle Database Vault 53**](#_Toc508290669)

[**4.5 Questions - Securing Data 53**](#_Toc508290670)

[**4.5 Database Auditing? 54**](#_Toc508290671)

[**4.5.1 What is Database Auditing? 54**](#_Toc508290672)

[**4.5.2 Why is Database Auditing Used? 55**](#_Toc508290673)

[**4.5.3 Type of Audit Records. 55**](#_Toc508290674)

[**4.6 Questions - Oracle Database Auditing 57**](#_Toc508290675)

**Enter your Name Here 🡺 Puja Ghosh**

# Assignment 6 Theory

**Assignment 6 is organized into two documents: Assignment 6 Theory and Assignment 6 Hands-on.**

**Assignment 6 Theory will be allocated 30% of the assignment points**

**Assignment 6 Hands-on will be allocated 70% of the Assignment points.**

# 1.0 Oracle Architecture

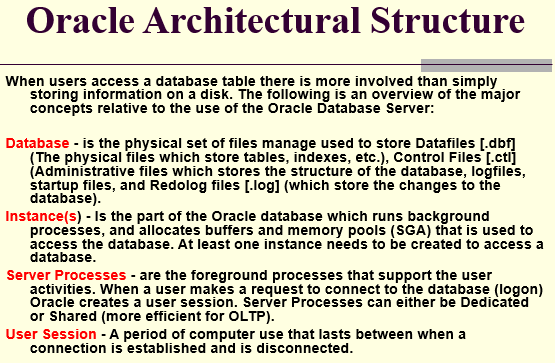
Video - Oracle Basics: Understanding Oracle Architecture - https://www.youtube.com/watch?v=09jjir0O6zA

Video - Oracle Architecture in less than 10 minutes - https://www.youtube.com/watch?v=yk8esAZKz4k

Video - Difference between Oracle Instance and Oracle Database - https://www.youtube.com/watch?v=2ow4\_D5VLnE

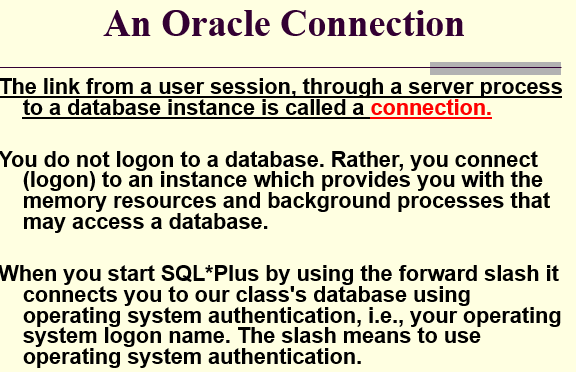
Video - Oracle 11g Administration Tutorial - Data Dictionary Views (Theory) - https://www.youtube.com/watch?v=RFm9G1f10-8

## 1.1 Database, Instances, Server Processes and User Sessions



## 1.2 Oracle Connections

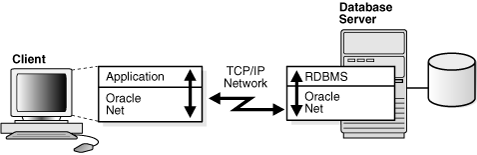
Database connection - https://en.wikipedia.org/wiki/Database\_connection



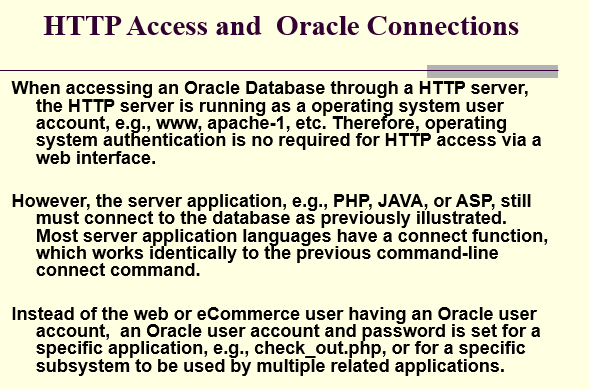
### 1.2.1 Client Server Connections

Oracle Net, a component of Oracle Net Services, enables a network session from a client application to an Oracle Database server. When a network session is established, Oracle Net acts as the data courier for both the client application and the database. It is responsible for establishing and maintaining the connection between the client application and database, as well as exchanging messages between them. Oracle Net is able to perform these jobs because it is located on each computer in the network.

Oracle Net enables connections from traditional client/server applications to Oracle Database servers. Figure 1-1 shows how Oracle Net enables a network connection between a client and a database server. Oracle Net is a software component that resides on both the client and the database server. Oracle Net is layered on top of network Oracle protocol support, rules that determine how applications access the network and how data is subdivided into packets for transmission across the network. In Figure 1-1, Oracle Net communicates with TCP/IP to enable computer-level connectivity and data transfer between the client and the database.

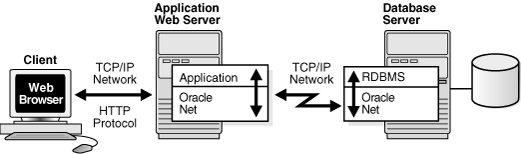


1.2.2 Web Client Application Connections

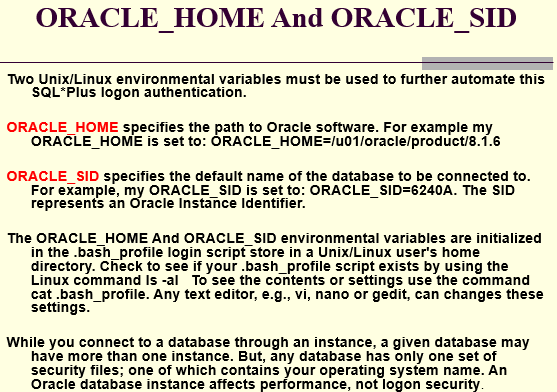


Internet connections from client Web browsers to an Oracle Database server are similar to client/server applications, except that the connection request goes to an application Web server.

Figure 1-3 shows the basic architecture for Web client connections, including a client Web browser, an application Web server, and an Oracle Database server. The browser on the client communicates with HTTP protocol to the Web server to make a connection request. The Web server sends the request to an application where it is processed. The application then uses Oracle Net to communicate with the Oracle Database server that also is configured with Oracle Net.



## 1.3 Oracle Environmental Variables



## 1.4 Questions – Oracle Architecture

1. From a security point-of-view what is a database? Answer =>

As database is set of organized data which can retrieve in many ways for use, security purpose is the important thing to consider. Data security refers to the process of protecting data from unauthorized access and data corruption throughout its life cycle. There are various ways one can implement database security like restricting unauthorized access,physical security of database server back up equipment etc. Summer 2018

2. What is the difference between an Oracle database and an instance? Answer =>

database is the set of files where application data and meta data is stored. An instance is the software that oracle uses to manipulate the data in the database.

3. What is the function of an Oracle server process? Answer =>

server processes are the foreground processes that support the user activities. When a user makes a request to connect to the database,oracle creates a user session. Server processes can either be dedicated or shared.

Summer 2018

4. What is the function of an Oracle user session? Answer =>

A user session begins when a user logs in and ends when a user logs out.

The user name and password they type, determines the type of user they are.

5. What is the relationship between an Oracle user session and an Oracle connection? Answer =>

The relation ship between these two is that an oracle connection is a link from a user session through a server process to a database instance. Summer 2018

6. If you increase the number of database instances, would this make your database more secure? Clearly explain your answer. Answer =>

A given database has more than one instance. But ,any database has only one set of security files; one of which contains operating system name. An oracle database instance affects performance, not logon security.

7. How does an Oracle Connection related to Oracle Security? Answer =>

oracle net plays an important role in oracle connection and it security. Oracle net, a component of net services, enables a network session from a client application to an oracle database server. Oracle net enables a network connection between a client and a database server.

8. What is the function of the ORACLE\_SID environmental variable? Answer =>

ORACLE\_SID works unix/linux, the oracle\_sid identifies the name of a specific database instance and tells the operating system which Oracle Instance to apply commands. Oracle\_sid is stored in the registry.Summer 2018

# 2.0 Introduction to Oracle Security

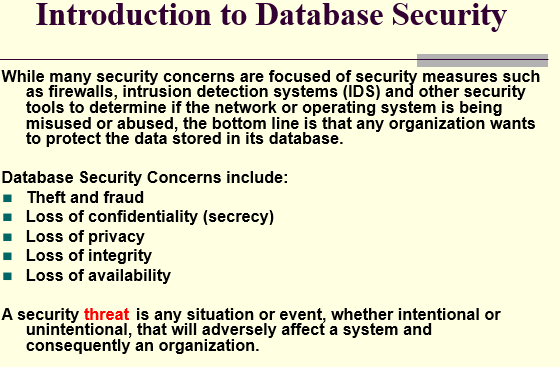
Oracle Security Requirements, Threats, and Concepts - <https://docs.oracle.com/cd/B19306_01/network.102/b14266/reqthret.htm#DBSEG1000>

Oracle Security Checklists and Recommendations - <https://docs.oracle.com/cd/B19306_01/network.102/b14266/checklis.htm#CHDBEJBJ>

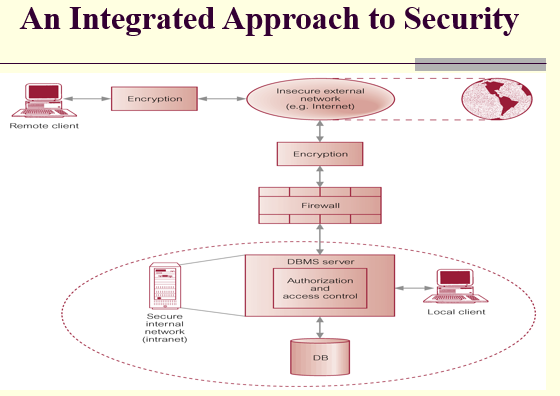
Oracle Security Policies and Tips - https://docs.oracle.com/cd/B19306\_01/network.102/b14266/politips.htm#DBSEG3000

Video - Security in the Database Oracle 11g - <https://www.youtube.com/watch?v=Z4Ng4aT0vac>

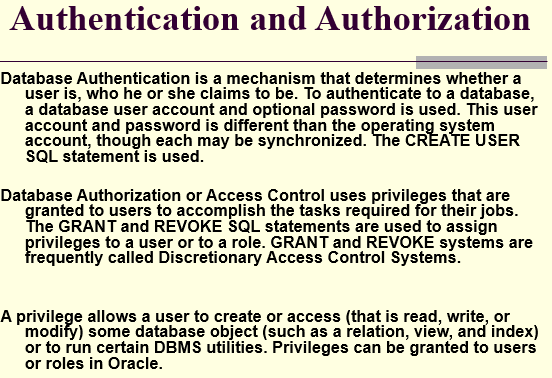
Video - Security Boot Camp: Oracle Database Security Vulnerabilities Explained - https://www.youtube.com/watch?v=NLHERbNaAkA



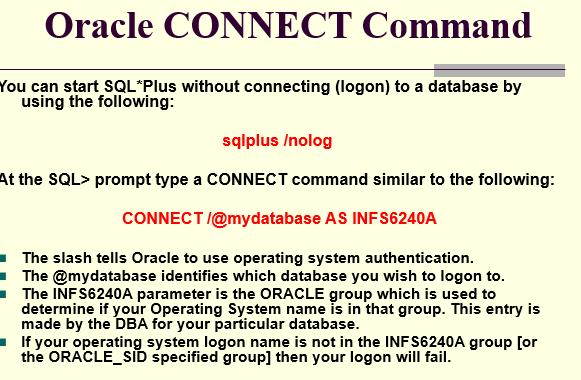
## 2.1 An Integrated Approach to Security



## 2.2 Oracle Authentication and Authorization



## 2.3 Oracle CONNECT Command



## 2.4 Oracle Authentication Methods

Oracle Authentication Methods - https://docs.oracle.com/cd/B19306\_01/network.102/b14266/authmeth.htm#BABCGGEB

Authentication means verifying the identity of someone (a user, device, or an entity) who wants to access data, resources, or applications. Validating that identity establishes a trust relationship for further interactions. Authentication also enables accountability by making it possible to link access and actions to specific identities. After authentication, authorization processes can allow or limit the levels of access and action permitted to that entity or user.

Oracle allows a single database instance to use any or all authentication methods. Oracle requires special authentication procedures for database administrators, because they perform special database operations. Oracle also encrypts passwords during transmission to ensure the security of network authentication.

### 2.4.1 Authentication by Operating Systems

Some operating systems permit Oracle to use information they maintain to authenticate users. This has the following benefits:

* Once authenticated by the operating system, users can connect to Oracle more conveniently, without specifying a user name or password. For example, an operating-system-authenticated user can invoke SQL\*Plus and skip the user name and password prompts by entering the following: SQLPLUS /
* With control over user authentication centralized in the operating system, Oracle need not store or manage user passwords, though it still maintains user names in the database.
* Audit trails in the database and operating system can use the same user names.

### 2.4.2 Authentication by the Network and LDAP Directories

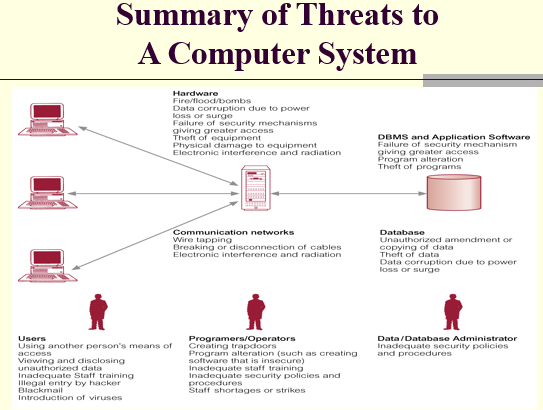
**Authentication Using SSL** - The Secure Socket Layer (SSL) protocol is an application layer protocol. It can be used for user authentication to a database, and it is independent of global user management in Oracle Internet Directory. That is, users can use SSL to authenticate to the database even without a directory server in place.

**Authentication Using Third-Party Services** - Authentication over a network makes use of third-party network authentication services. Prominent examples include Kerberos, Public Key Infrastructure (PKI), the Remote Authentication Dial-In User Service (RADIUS), and directory-based services, as described in the following subsections.

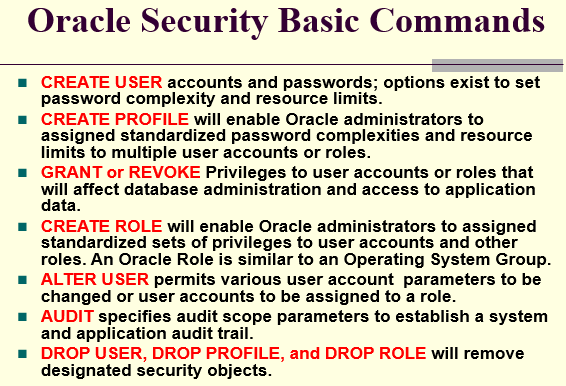
If network authentication services are available to you, then Oracle can accept authentication from the network service. If you use a network authentication service, then some special considerations arise for network roles and database links.

### 2.4.3 Authentication by Oracle Database

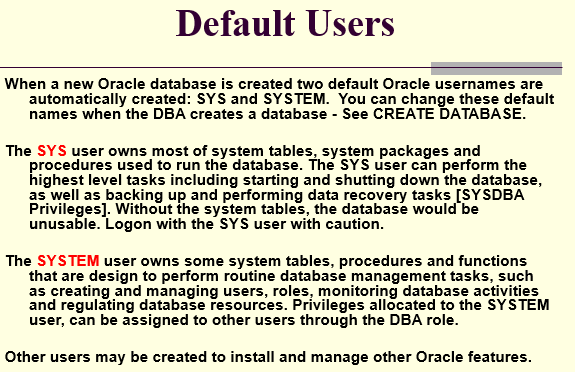
Oracle Database can authenticate users attempting to connect to a database, by using information stored in that database itself. To set up Oracle Database to use database authentication, you must create each user with an associated password. The user must provide this user name and password when attempting to establish a connection. This process prevents unauthorized use of the database, because the connection will be denied if the user provides an incorrect password. Oracle Database stores user passwords in the data dictionary in an encrypted format to prevent unauthorized alteration. Users can change their passwords at any time.



## 2.5 Basic Oracle security Commands.



## 2.6 Oracle Built-in Default Users



### 2.6.1 Questions - Introduction to Oracle Security

1. Oracle provides DBMS authentication and authorization security services. What other security service are also important in **an integrated approach**. Answer =>

other security services are such as firewalls, intrusion detection systems and other security tools to determine if the network or operating system is being misused or abused, the bottom line is that any organization wants to protect the data stored in its database.

Summer 2018

2. Which Oracle SQL command is an important component of Oracle's **Authentication** services? Answer =>

CREATE USER: accounts and password ,options to exist to set password complexity and resourse limits. CRAETE PROFILE will enable oracle administrators to assigned standardized password complexities resourse limits to multiple users accounts or roles.

Summer 2018

3. Which Oracle SQL command is an important component of Oracle's **Authorization** services? Answer =>

GRANT OR REVOKE previleges to user accounts or roles that will effect database administration and access to application data.

CREATE ROLE will enable oracle administrator to assigned standardized sets of privilages to user accounts andother roles

Summer 2018

4. Explain the process and advantages of Oracle authentication by operating system. Answer =>

few operating system permits oracle to use to maintain to authenticate users. It has some benefits in it. Users can connect to Oracle more comveniently, if authenticated through operating system. As users can authenticate through oracle through centralized operating system, oracle need not store oracle user passwords,except mainting its name.

Summer 2018

5. Explain the process and advantages of Oracle authentication by network or LDAP services. Answer =>

authentication can be happened in two ways through SSL and third party services. SSL is an application layer protocol. Users can use SSL to authenticate to the database even without a directory server in place. Oracle can accept authenticated from the network service if authentication service available.

6. Explain the process and advantages of Oracle DBMS authentication. Answer =>

database can authenticate user by using information stored in that database itself. User needs to provide user name and password when attempting to establish connection. This process does not allow any unauthorized user because any incorrect password prevents user to get authentication.

7. You access a web application as a customer, e.g., Amazon. How will a customer be authenticated and authorized to use the backend Oracle database? Answer =>

8. What are the differences between the Oracle users named SYS and SYSTEM? Answer =>

SYS user owns most of the system table,system packages and procedures used to run the database. The SYS user can perform the highest level of tasks including starting and shutting down the database, as well as backing up and performing data recovery tasks.

SYSTEM user owns some system tables, procedures and functions that are design to perform routine database management task, such as creating and managing users,roles,monitoring database activities and regulating database resources.

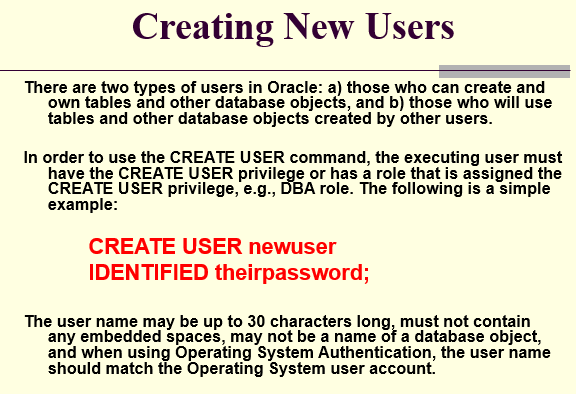
## 2.7 Create User Command

Video - Oracle SQL Tutorial - Creating a User (Theory) - <https://www.youtube.com/watch?v=eVGUaB7srIw>

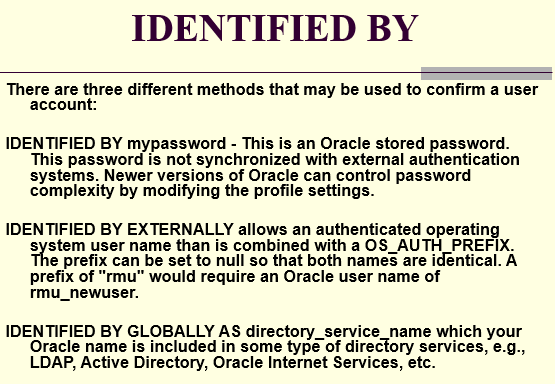
Video - Oracle user management | User creation and privilege management | password management in oracle - https://www.youtube.com/watch?v=WQfTsr6x5UY

Video - How to create NEW USER account using Create User statement in Oracle database - <https://www.youtube.com/watch?v=4QbK2Y-1LZw>

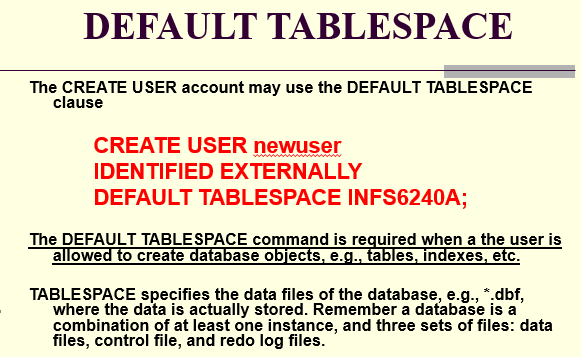
Video - Oracle SQL Tutorial - Creating a User (Hands-On) - https://www.youtube.com/watch?v=VSoA5vj8H3k

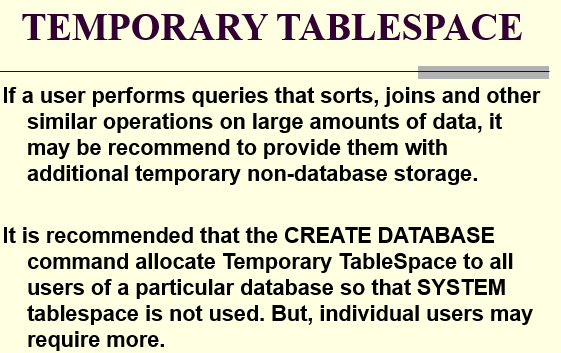


### 2.7.1 IDENTIFIED BY

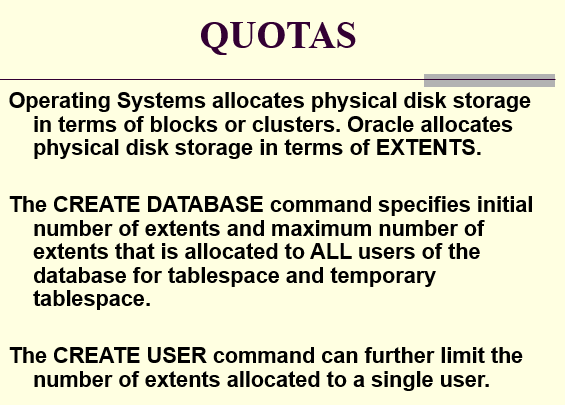


### 2.7.2 DEFAULT TABLE SPACE





### 2.7.3 User Quotas



### 2.7.4 ALTER or DROP USER

Use the ALTER USER command to modify a user account. Any attributes that you do not specify remain unchanged. For example, if you do not specify an expiration time, the previously set expiration time continues to apply. Use the ALTER USER command to set or change the attributes of a user account:

* Set or change a password.
* Set an expiration time for the password or for the account itself.
* Set session timeout, query timeout, and rowset limits.
* Add a user to a user group. (The ALTER USER command cannot be used to remove a user from a group; to do that, you must use the ALTER GROUP command.)
* Unlock an account after the maximum number of logon attempts is exceeded.

Use the DROP USER statement to remove a database user and optionally remove the user's objects.

### 2.7.5 Questions – CREATE USER

1. What is the function or purpose of an **IDENITFIED BY** clause in Oracle Authentication Security? Answer =>

It is an oracle stored password. This clause creates a local user and indicates that user must specify password to log on to the database.

Summer 2018

2. What is the function or purpose of an **IDENITFIED BY EXTERNALLY** clause in Oracle Authentication Security? Answer =>

IDENTIFIED BY EXTERNALLY creates that a user must be authenticated by an external service such as an operating system or a third party service. In this case, Oracle Database relies on authentication by the operating system or third-party service to ensure that a specific external user has access to a specific database user.

3. Which Oracle SQL command can limit the amount storage space of an Oracle Database? Answer =>

CREATE DATABASE command speficies number of extents or storage and maximum number of extents that is allocated to ALL users of the database for tablespace and temporary tablespace.

4. Which Oracle SQL command can limit the amount storage space for tables and databases owned or created by a specified user? Answer =>

CRAETE USER can limit the amount storage space for tables and databses owned or created by a specified user.

Summer 2018

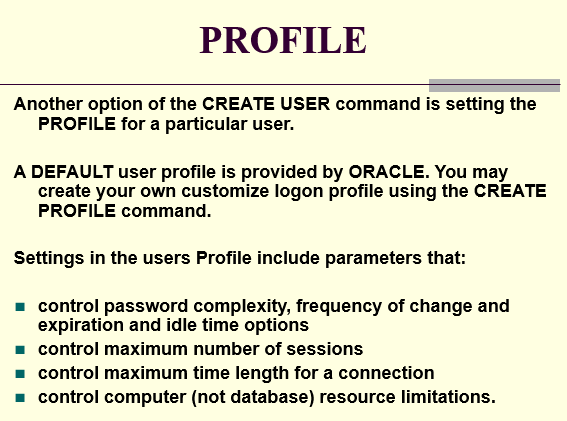
## 2.8 Oracle Profiles

Oracle Profiles - <https://docs.oracle.com/cd/B19306_01/network.102/b14266/authoriz.htm#i1009023>

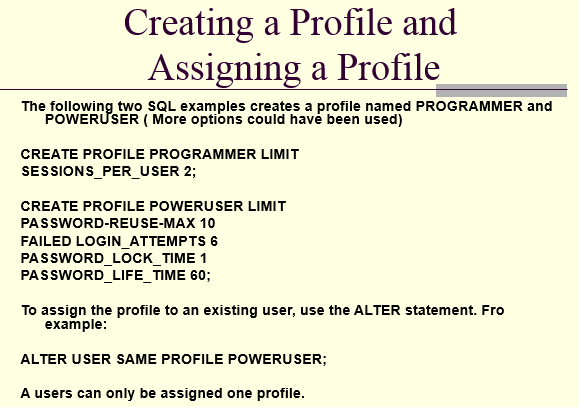
Oracle recommends that you use the Database Resource Manager rather than this SQL statement to establish resource limits. Use the CREATE PROFILE statement to create a profile, which is a set of limits on database resources. If you assign the profile to a user, then that user cannot exceed these limits. A profile is a database object - a named set of resource limits to:

* Restrict database usage by a system user – profiles restrict users from performing operations that exceed reasonable resource utilization. Examples of resources that need to be managed: Disk storage space, I/O bandwidth to run queries, CPU power, Connect time, and CPUu time
* Enforce password practices – how user passwords are created, reused, and validated.
* Profiles are assigned to users as part of the CREATE USER or ALTER USER commands.
* User accounts can have only a single profile.
* A default profile can be created – a default already exists within Oracle named DEFAULT – it is applied to any user not assigned another profile.
* Assigning a new profile to a user account supersedes any earlier profile.
* Profiles cannot be assigned to roles or other profiles. Summer 2018

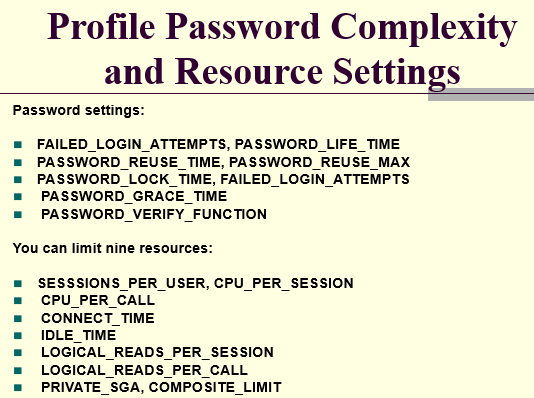
**Profile can be assign in two ways either during USER creation or by using ALTER statement.**



### 2.8.1 Creating Oracle Profiles



### 2.8.2 Password Complexity and Resource Settings



### 2.8.3 Questions – Oracle Profiles

1. What is the function or purpose of Oracle PROFILE? Answer =>

Oracle PROFILE is a set of limit assigned to a user on system resources. PROFILE is much hepful in many large organization when you have lots of database users. It will allow you to control the amount of resources used by each database user by assigning some limits in it.

2. Explain the difference function or purpose between an Oracle PROFILE and an Oracle Privilege? Answer =>

Oracle Privileges is the way to control authorization on database objects when you have multiple users. Every object has an owner. Privileges control whether a user can modify an object owned by another user. Oracle PRIFILE is a assigned limit on system resources.

3. List two methods that you may use to assign an Oracle PROFILE to a user? Answer =>

Two methods to assign an Oracle PROFILE to a user CREATE USER and ALTER USER.

4. How many PROFILES may be assigned to a USER? Answer =>

A User can have only single profile.

5. What are the advantages of using Oracle PROFILES? Answer =>

Oracle PROFILES are important because when large complex database and multiple users , it is necessary to set limit to maintain equality among all users.

## 2.9 GRANT Command and Privileges

Oracle Access Control on Tables, Views, Synonyms, or Rows - https://docs.oracle.com/cd/B19306\_01/network.102/b14266/accessre.htm#CHDDGEJG

Video - How to grant System and Object Privileges in Oracle Database - https://www.youtube.com/watch?v=f4LJg4qEp40

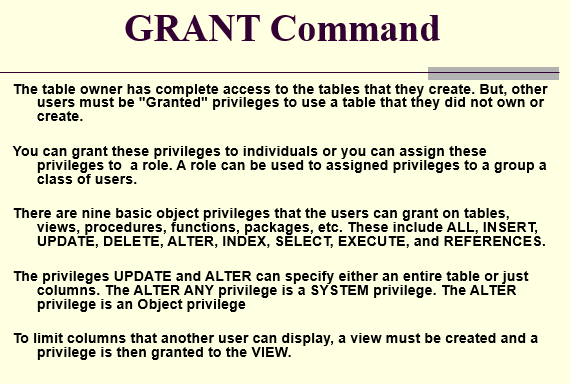
Video - Oracle SQL Tutorial - Data Control Language - System Privileges (Theory) - <https://www.youtube.com/watch?v=MUqZ5U7VIMk>

Video - Oracle SQL Tutorial - Object Privileges (Theory) - <https://www.youtube.com/watch?v=MPx1as5_ePA>

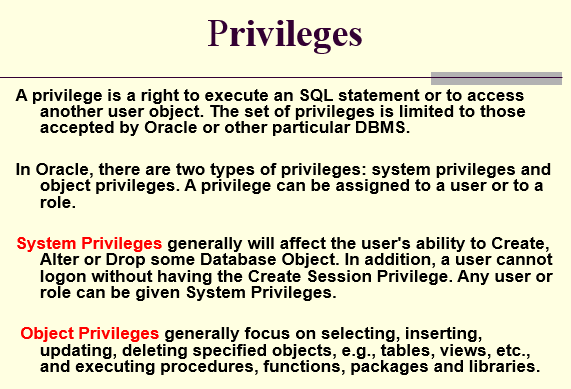
Oracle Authorization: Privileges, Roles, Profiles, and Resource Limitations - <https://docs.oracle.com/cd/B19306_01/network.102/b14266/authoriz.htm#DBSEG5000>

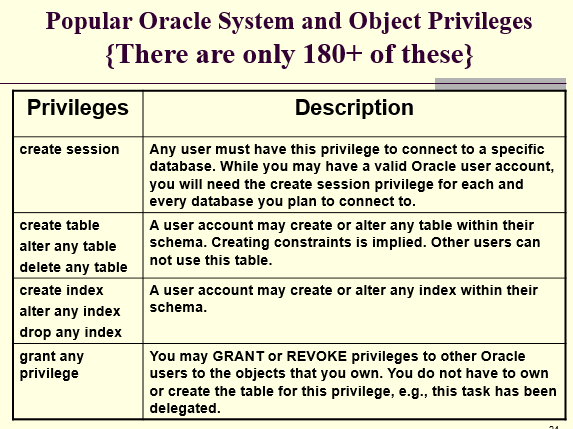
**Oracle System privileges -** [**https://docs.oracle.com/database/121/TTSQL/privileges.htm#TTSQL339**](https://docs.oracle.com/database/121/TTSQL/privileges.htm#TTSQL339)

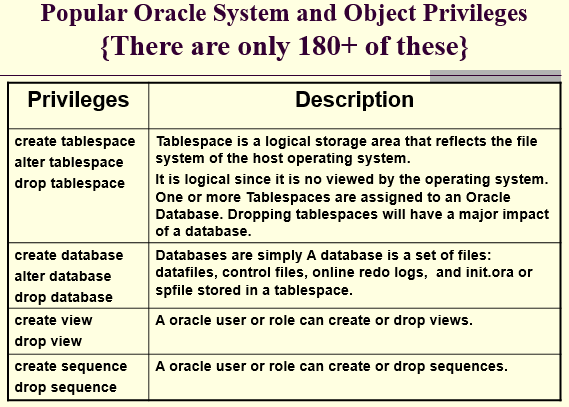
**Object privileges - https://docs.oracle.com/database/121/TTSQL/privileges.htm#TTSQL341**

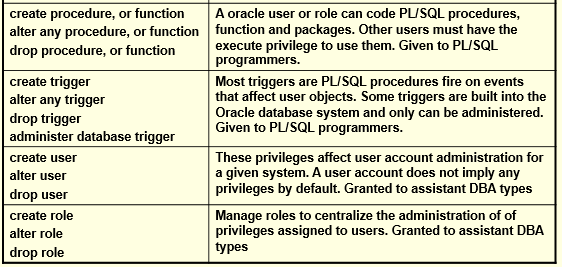


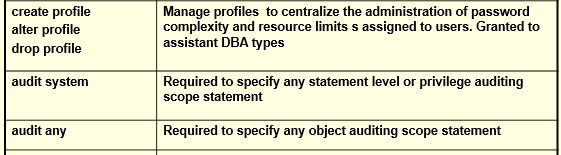
### 2.9.1 System and Object Privileges

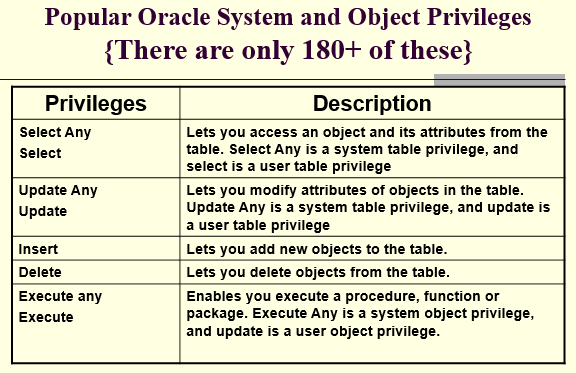




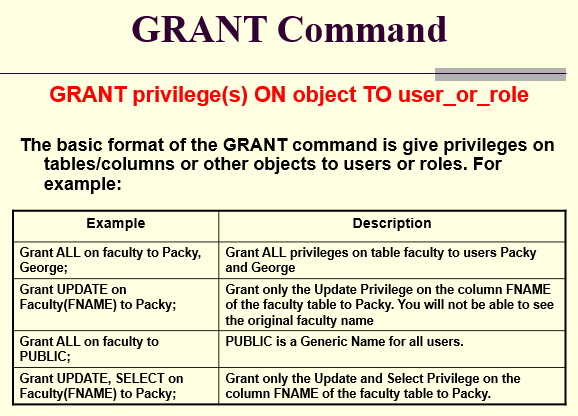




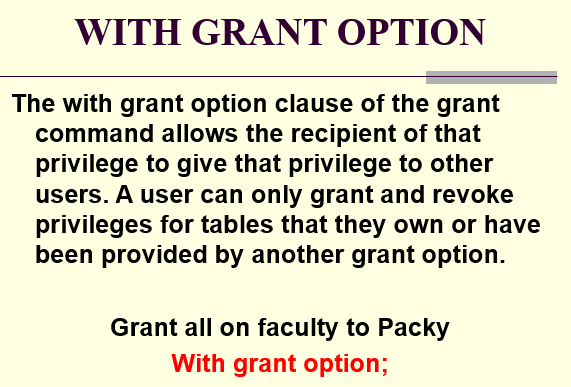




### 2.9.2 Grant Command Examples



### 2.9.3 WITH GRANT OPTION



### 2.9.4 Questions - GRANT Command and Privileges

1. Oracle will authorized an authenticated user or group to preform Oracle administrative functions or provide access to Oracle objects, e.g., a database, table column, etc. What are System Privileges? Answer =>

System privilege is the right to perform a particular action, or to perform an action on any schema objects of a particular type. For example, the privileges to create tablespace and to delete the rows of any table in a database are system privileges. There are over 60 distinct system privileges.

Summer 2018

2. Oracle will authorized an authenticated user or group to preform Oracle administrative functions or provide access to Oracle objects, e.g., a database, table column, etc. What are Object Privileges? Answer =>

An object privilege is a permission granted to an oracle database user account or role to perform some action on a database object.

Summer 2018

3. List 8 or more security related privileges should be assigned to an Oracle security job. Answer =>

Summer 2018

4. Should the CREATE TABLE privilege be assigned to a new Oracle database security job? Explain your reason. Answer =>

Summer 2018

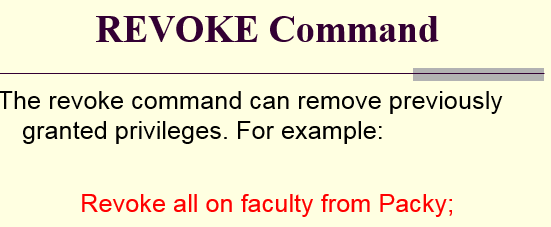
5. List 8 or more privileges should be assigned to a role which manages application development of for an inventory database. Answer =>

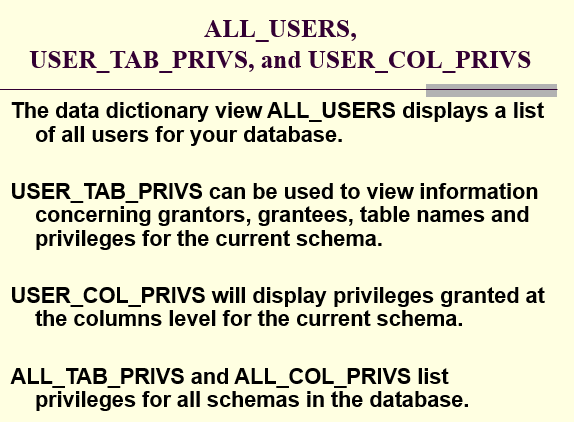
6. List 8 or more privileges should be assigned to an Oracle DBA of an inventory database. Answer =>

7. What are the security concerns associated the WITH GRANT OPTION clause of the SQL GRANT command? Answer =>

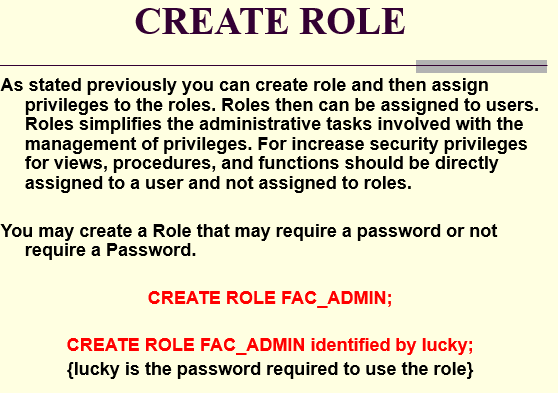
The owner of an object can grant it to another user by specifying the WITH GRANT OPTION clause in the GRANT statement. In this case, the new grantee can then grant the same level of access to other users or roles.

## 2.10 Revoke Command





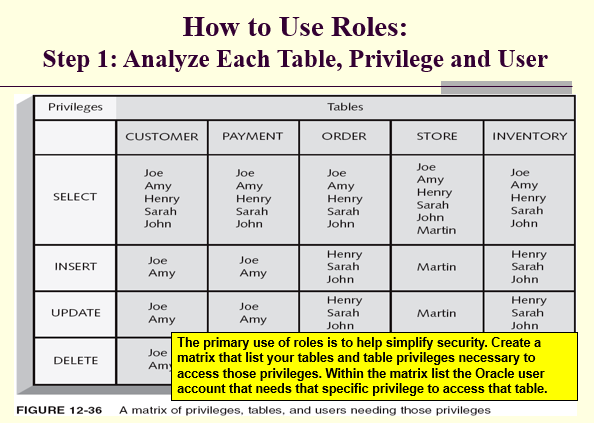
## 2.11 CREATE ROLE

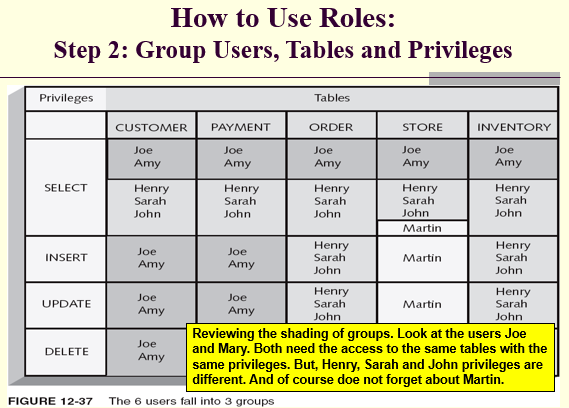


Video - Oracle SQL Tutorial - Roles in Oracle (Theory) - https://www.youtube.com/watch?v=uyg5jz1I7kU

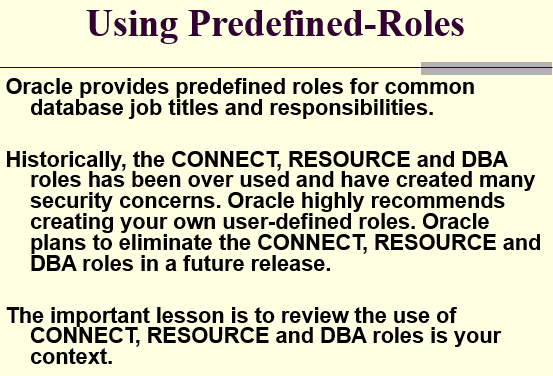
Video - Oracle Database Profiles - <https://www.youtube.com/watch?v=8k7mrStMN0Y>

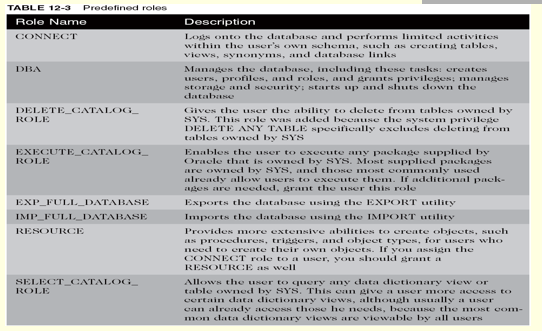
### 2.11.1 How to Assign Privileges to Roles



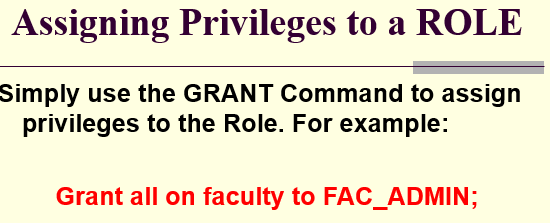


### 2.11.2 Oracle Pre-Defined Roles

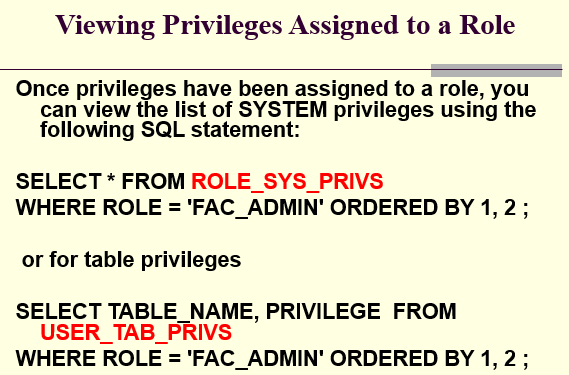




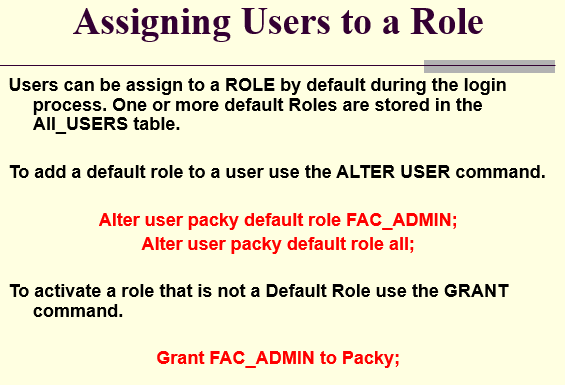
### 2.11.3 Assigning Privileges and Users to a Role



### 2.11.4 Viewing Role Privilege



### 2.11.5 Assigning Users to Roles



### 2.11.6 Questions - Oracle ROLES

1. What is are the advantages of assigning privileges to an Oracle Role rather than directly to an individual Oracle user? Answer =>

Assigning roles is much more simple when there large number of people working on. As it is easier to give a group of people same privileges and access to system, than assigning it individually.

2. How many Oracle roles may be assigned to an Oracle user? Answer =>

One or more can be assigned to an oracle user. A user assigned to a ROLE by default during the logon process.

3. How many Oracle profiles may be assigned to an Oracle user? Answer =>

An oracle user can have single profile.

4. Provide an SQL example of for creating an Oracle Role. Answer =>

ROLE can be created that may require a password or not.

CREATE ROLE role name;

CREATE ROLE role name IDENTIFIED BY password;

5. Provide an SQL example of assigning Oracle Privileges to an Oracle Role. Answer =>

GRANT all on faculty to FAC\_ADMIN

6. Provide an SQL example of assigning Oracle users to an Oracle Role. Answer =>

ALTER USER to add a default role to an oracle user.

Summer 2018

# 3.0 Oracle’s Data Dictionary

Video - Oracle 11g Administration Tutorial - Data Dictionary Views (Theory) - <https://www.youtube.com/watch?v=RFm9G1f10-8>

Oracle System Tables - https://www.techonthenet.com/oracle/sys\_tables/index.php

What are Oracle Data Dictionaries? - <https://chartio.com/resources/tutorials/how-to-list-all-tables-in-oracle/>

Oracle Data Dictionary and Dynamic Performance Views - https://docs.oracle.com/database/121/CNCPT/datadict.htm#CNCPT002

Oracle’s data dictionary is a collection of system tables that contain information about the structure of database. When you successfully execute a Create Table or Create Index statement the appropriate tables, columns, constraints and indexes are created. The information that describes these objects is stored in the data dictionary, e.g., the name of the tables, columns, data types, etc. In addition, information is also stored concerning the space utilization, performance, users, security and roles in the database. A series of views are available that will provide access for three types of Oracle users: ALL (Public and User), USER, and DBA views. The DBA views include all objects, e.g., tables, indexes, etc., for all users and has access to more system performance information. The USER views will display all of the objects owned by the current user. The ALL views will display the USER owned tables as well as access had been GRANTed to that User or the PUBLIC (See Grant and Revoke for more information). The following is summary of the more popular Select statements that a normal user may use to document their database design. Summer 2018

## 3.1 USER\_CATALOG (CAT) and USER\_OBJECTS (OBJ)

The USER\_CATALOG will display all tables, views, synonyms and sequences for which has been created. Not all objects are provided in this view, but certainly the more popular objects are. The following will display the table name (or view, synonym name) owned by particular user. CAT is a synonym, an alias, for the User\_Catalog

**Select Table\_Name, Table\_Type from User\_Catalog;**  Or

**Select Table\_Name, Table\_Type from Cat;**

A second view, User\_Objects, provides more detailed information to all types of objects. In addition to the User\_Catalog objects, indexes, database clusters, links, functions, packages, procedures and triggers are included

|  |  |
| --- | --- |
| **USER\_OBJECTS (OBJ)** | |
| **Column Name** | **Description** |
| Object\_Name | The Name of the object |
| Object\_ID | A unique, Oracle-assigned ID for the object. |
| Object\_Type | The object type, e.g., TABLE, INDEX, VIEW, etc. |
| Created | The date the object was created. (I wonder if I can use this field to find the procrastinators. |
| Last\_DLL\_Time | The timestamp for the last DDL command used on the object, e.g. when was the last time someone updated, selected, or deleted a row. The Alter, Grant and Revoke command will also affect this time. |
| Timestamp | Same as created, but stored as a character column |
| Status | Valid or Invalid |

For example:

**Select Object\_Name, Created, Last\_DLL\_Time from User\_Objects**

**Ordered by Last\_DLL\_Time**

Will display a list of all objects created by you and will order the objects by the time they were last used. You can code any Select statement and clause to retrieve the information from Oracle’s Data Dictionaries. Also, you can code the same statement to retrieve information from the ALL\_Objects view, which represents all the tables the User owns and all of other tables that the user has been granted access.

## 3.2 USER\_TABLES (TABS)

How many rows are in my tables? Which tables that up the most disk space?

Another useful data dictionary view is USER\_TABLES or Tabs for short:

|  |  |
| --- | --- |
| **USER\_TABLES (TABS)** | |
| **Column Name** | **Description** |
| Table\_Name | The Name of the table |
| Num\_Rows | Number of rows |
| TableSpace\_Name | An Oracle database is an operating system file. This is the name of the operating system file. A Table space can contain many tables, indexes or clusters. Tablespace has a fixed size and can only be increased by a DBA. Tablespace is added by attaching a second operating system file or by expanding the size of the original operating system file. The Create Table Space command allows one or more files to be assigned immediately to the table space. |
| Min\_Extents | When a table is created an area is set aside for it. Extents are units of disk space measured in K that controls how a table will grow and use up space inside the Tablespace. Min\_Extents is the minimum size of the table. If min\_extents is 10K then a newly created table uses 10K of disk space even though nothing is stored inside it. |
| Next\_Extent | When rows are inserted to a point that the original min\_extents have been used, then Oracle can allocate more extents (disk space) to the table. Next\_extents controls how table will grow. |
| Max\_Extents | The maximum size of the table as measured in units of next\_extents. This number is not related to tablespace. Before a given table may grow to its max\_extents, Oracle may run out of tablespace. The total number of extents allocated to any given table is called a segment. Therefore, Tablespace contains tables, an each table have a physical unit of space assigned to it called a segment. A table segment was created by adding extents (units of disk space) to the table. |
| Pct\_free  Pct\_used | Precentage of the Segement used or free |
| Blocks  Empty\_Blocks | Physical measurement of disk space allocate to a segment. A block size varies from 512 characters to 2K depending upon the installation of Oracle into the operating system. |

For example, the following will display the number of rows stored in each table.

**Select Table\_name, Num\_rows, from tabs;**

## 3.3 USER\_TAB\_COLUMNS (COLS)

What are the column names of my tables? Default values to be used for a given column? What are the maximum or minimum values for a given column?

|  |  |
| --- | --- |
| **USER\_TAB\_COLUMNS (COLS)** | |
| **Column Name** | **Description** |
| Table\_Name | The Name of the table |
| Column\_Name | The name of column |
| Column\_Id | Unique name of a column assigned by Oracle |
| Data\_Type | Number, Char, Varchar, etc. |
| Data\_Length | For example Char(5) would have a length of 5 |
| Data\_Precision | The number of places to the right of the decimal point |
| Data\_Scale | Total number of decimal positions. For example: number(5,2) would have a scale of 5 and a precision of 2 |
| Nullable | Nulls not allowed? |
| Data\_Default | The default value to be store in the column if a value was provided when a row was inserted. |
| Num\_Distinct | Number of distinct values for that column in the table. Foe example, if there was only five departments used in the faculty table for 35 rows of faculty members, this value would be 5 |
| Low\_Value  High\_Value | Lowest and Highest value for that column in the table |

While much of this information is available from the DESCRIBE command, the DESCRIBE command cannot display default values and the various statistics, i.e., Num\_Distinct, Low\_Value, and High\_Value.

## 3.4 USER\_CONSTRAINTS and USER\_CONS\_COLUMNS

What are my Primary and Foreign Key Relationships? What are the Check Options stored with a table?

There are two tables needed for this task. First, the view User\_Constraints is used to determine the type of constraint: Check Constraint (C) or NOT NULLS, Primary Key Constraint (P), Foreign Key Constraint (R), Unique Constraint (U), with the CHECK option (V) constraint. Once you know the type constraint you then can identify the name of table and the column involved.

You will notice that these tables will specify the name of an OWNER. OWNER refers to the owner of the constraint, which may be different from the owner who created the table. Suppose the table COURSE has a primary key name COURSE\_PK based upon the column named CNO. In addition, the table has a foreign key constraint name FACULTY\_COURSE\_FK based upon column COURSE(FNO) which references FACULTY(FID)

|  |  |
| --- | --- |
| **USER\_CONSTRAINTS** | |
| **Column Name** | **Description** |
| Owner | The Owner of the constraint |
| Constraint\_Name | The name of the constraint. Using the above example there would be one row in this view named COURSE\_PK and another row name FACULTY\_COURSE\_FK |
| Constraint\_Type | Check Constraint (C) or NOT NULLS, Primary Key Constraint (P), Foreign Key Constraint (R), Unique Constraint (U), with the CHECK option (V) |
| Table\_Name | The name of the table associated with the constraint. In both cases this would be the table named COURSE |
| Search\_Condition | The search condition used with CHECK option, e.g., FPAY >10000 and FPAY< 90000. The previous example has not check option |
| R\_Constraint\_Name | The name of the constraint referenced by a FOREIGN KEY. Since a FOREIGN key must reference another table’s PRIMARY KEY, this column would only be used for FOREIGN KEY. In this case the PRIMARY KEY constraint name of the faculty table is FACULTY-PK |
| R\_OWNER | The owner of the constraint of the referenced table. Only used with FOREGIN KEY constraints |
| Delete\_Rule | Only used with Foreign Key tables. If the value is Cascade, if the parent row is deleted, i.e., the faculty row is deleted, then all dependent rows are deleted, i.e., all the courses taught by that faculty member. |
| Status | Enabled or Disabled |

|  |  |
| --- | --- |
| **USER\_CONS\_COLUMNS** | |
| **Column Name** | **Description** |
| Owner | The Owner of the constraint |
| Constraint\_Name | The name of the constraint. Using the above example there would be one row in this view named COURSE\_PK and another row name FACULTY\_COURSE\_FK |
| Table\_Name | The name of the table associated with the constraint. In both cases this would be the table named COURSE |
| Column\_Name | The name of the column associated with the constraint. The column CNO would be the name of the PRIMARY KEY constraint of the COURSE table and FNO would the name of the FOREIGN KEY constraint of the COURSE table |
| Position | The position of the column within the constraint definition. Since a constraint can have many columns (a composite key) then this system view must be separated from the constraint definition. |

To show all primary key constraints for your tables you can use the following:

Select Table\_Name, Constraint\_Name

From User\_Constraints

Where Constraint\_type = ‘P’;

The previous SQL script will not display the name of the column used in the primary key. The next example will

Select C.Table\_Name, Column\_Name, Position

From User\_Constraints C, User\_Cons\_Columns Ccols

Where Constraint\_type = ‘P’ and

C.Constraint\_Name = Ccols.Constraint\_Name;

To show all foreign key relationships for your tables you can use the following:

Select Table\_Name, Constraint\_Name

From User\_Constraints

Where Constraint\_type = ‘R’;

The previous SQK script will not display the name of the column used in the foreign key. Nor will it display information about the parent table. The next example will

Select C.Table\_Name, Column\_Name, Position,

From User\_Constraints C, User\_Cons\_Columns Ccols

Where Constraint\_type = ‘F’ and

C.Constraint\_Name = Ccols.Constraint\_Name;

{Need to test}

Select C.Table\_Name, Column\_Name, Position, P.Table\_Name

From User\_Constraints C, User\_Constraints P, User\_Cons\_Columns Ccols

Where Constraint\_type = ‘F’ and

C.Constraint\_Name = Ccols.Constraint\_Name and

C.R\_Constaint\_Name = C.Constraint\_Name

## 3.5 USER\_INDEXES (IND) and USER\_IND\_COLUMNS

The Primary key and Unique Constraints automatically have indexes created by Oracle. Other indexes may be created to provide better performance to access, order and group other columns. Similar to a constraint the column names are stored in a separate view because an index may have many columns.

Since an Index uses disk space, the USER\_INDEX view contains entries likes USER\_TABLES, i.e., tablespace, Initial\_Extent, Min\_Extents, Max\_Extents. This information will not be discussed here. When a table space is first created, by default have of the space is reserved for index space. You can however, change the allocation of table space between tables and indexes.

For sake of discussion consider the following SQL script:

CREATE INDEX FACULTY\_FNAME\_IND ON FACULTY(FNAME);

|  |  |
| --- | --- |
| **USER\_INDEXES (IND)** | |
| **Column Name** | **Description** |
| Table\_Owner | The Owner of the Table |
| Index\_Name | The name of the index. In this case the name of the index would be FACULTY\_FNAME\_IND |
| Table\_Name | The name of the table associated with the index. In this the tabvle name would be the table named FACULTY |
| Uniqueness | UNIQUE or NONUNIQUEe |
| Status | Enabled or Disabled |

|  |  |
| --- | --- |
| **USER\_IND\_COLUMNS** | |
| **Column Name** | **Description** |
| Index\_Name | The name of the index. In this case the name would be FACULTY\_FNAME\_IND |
| Table\_Name | The name of the table associated with the index. In this case it would be the table named FACULTY |
| Column\_Name | The name of the column associated with the index. In this case it would be column FNAME |
| Position | The position of the column within the index definition. Since an index can have many columns (a composite key) then this system view must be separated from the indexed definition. |
| Column\_Length | The indexed length of columns |

To display a list of indexes created for your project:

**Select Table\_Name, Index\_Name, Uniqueness**

**From USER\_INDEXES;**

or

**Select Table\_Name, Index\_Name, Position, Column\_name**

**From USER\_INDEXES I, USER\_IND\_COLUMNS C**

**Where I.Index\_Name = C.Index\_Name;**

## 3.6 USER\_VIEWS and USER\_SYNONYMS (SYN)

A view is simply a predefined SELECT statement and may be treated like a table. A synonym is another name for an entire table or view. For example:

Create View Cis\_Faculty

Select FID, Fname from faculty where FDEP=’CIS’;

|  |  |
| --- | --- |
| **USER\_VIEWS** | |
| **Column Name** | **Description** |
| View\_Name | The name of the view In this case the name would be CIS\_FACULTY |
| Text\_Length | The length of the view’s base query, in characters. |
| Text | In this case Select FID, Fname from faculty where FDEP=’CIS’; |

For example:

Create Synonym faculty\_members for faculty;

|  |  |
| --- | --- |
| **USER\_SYNONYMS** | |
| **Column Name** | **Description** |
| Synonym\_Name | The name of the synonym. In this case the name would be FACULTY\_MEMBERS |
| Table\_Name | In this case it would be FACULTY |

## 3.7 Questions – Oracle System Tables and Data Dictionary

1. What is the Oracle Data Dictionary? Answer =>

Oracle’s data dictionary is a collection of system tables that contains information about the structure of the database. When executing a create table or create index statement the appropriate table,column , constraints and indexes are created. The information that describes these objects is stored in the data dictionary , like the name of the tables,columns data types etc.

2. List the SQL statement can be used to retrieve data from an Oracle System Table. Answer =>

SELECT \* FROM table name;

3. List the Oracle System Table which stores information about primary or foreign keys Answer =>

ALL\_CONSTRAINTS stores all users constaints information.

USER\_CONSTRAINTS describes all constraints by a current user.

4. List the Oracle System Table which stores information about Check options Answer =>

5. List the Oracle System Table which stores information about all tables, views, synonyms, and sequences. Answer =>

USER\_CATALOG stores information about all tables, views, synonyms and sequences.

6. List the Oracle System Table which stores information about all users. Answer =>

DBA\_USERS

ALL\_USERS

7. List the Oracle System Table which stores information about details of a table column, e.g., data type, etc. Answer =>

USER\_TAB\_COLUMNS

8. List the Oracle System Table which stores information about privileges assigned to a user. Answer =>

USER\_TAB\_PRIVA current user

DBA\_TAB\_PRIVS

9. List the Oracle System Table which stores information about which tables that may be using thee most disk space?

Answer =>

DBA\_TS\_QUOTAS

USER\_TS\_QUOTAS

10. List the Oracle System Table which stores information about privileges assigned to a role. Answer =>

ROLE\_SYS\_PRIVS

# 4.0 Oracle Securing Data and Database Auditing

## 4.1 Securing Data - Transparent data encryption

### 4.1.1 What is Transparent data encryption

Transparent data encryption encrypts data in one or more database table columns, or it can encrypt an entire tablespace. Transparent data encryption is the quickest and easiest way to encrypt data. Encrypted data can only be read by its recipient. You use encryption to protect data in a potentially unprotected environment, such as data you have placed on backup media that is sent to an offsite storage location.

The encryption data includes the following components:

1. An algorithm to encrypt the data. The encryption algorithm is used by Oracle databases to encrypt data. Oracle Database supports several industry-standard encryption and hashing algorithms, including the Advanced Encryption Standard (AES) encryption algorithm, which has been approved by the National Institute of Standards and Technology (NIST).
2. A key to encrypt and decrypt data. When you encrypt data, Oracle Database uses the key and clear text data as input into the encryption algorithm. Conversely, when you decrypt data, the key is used as input into the algorithm to reverse the process and retrieve the clear text data. Oracle Database uses a symmetric encryption key to perform this task, in which the same key is used to both encrypt and decrypt the data. The encryption key is stored in the data dictionary, but encrypted with another master key.

### 4.1.2 When Should You Encrypt Data?

In most cases, you encrypt sensitive data on your site to meet a regulatory compliance. For example, sensitive data such as credit card numbers, Social Security numbers, or patient health information must be encrypted.

Historically, users have wanted to encrypt data because they want to restrict data access from their database administrators. However, this problem is more of an access control problem, not an encryption problem. You can address this problem by using Oracle Database Vault to control the access to your application data from database administrators.

In most cases, you encrypt sensitive data such as credit cards, and Social Security numbers to prevent access when backup tapes or disk drives are lost or stolen. In recent years industry regulations such as the Payment Card Industry (PCI) Data Security Standard and the Healthcare Insurance Portability and Accountability Act (HIPAA) have become a driving factor behind increased usage of encryption for protecting credit card and health care information, respectively.

### 4.1.3 How Transparent Data Encryption Works

Transparent data encryption enables you to encrypt individual table columns or an entire tablespace. When a user inserts data into an encrypted column, transparent data encryption automatically encrypts the data. When users select the column, the data is automatically decrypted.

To encrypt data by using transparent data encryption, you create the following components:

1. A wallet to store the master encryption key. The wallet is an operating system file located outside the database. The database uses the wallet to store the master encryption key. To create the wallet, you can use Enterprise Manager or the ALTER SYSTEM command. The wallet is encrypted using a password as the encryption key. You create the password when you create the wallet. Access to the contents (or master key) of the wallet is thus restricted to only those who know the password. After the wallet is created, you must open the wallet using the password so that the database can access the master encryption key.
2. A location for the wallet. You can specify the wallet location in the sqlnet.ora file.

Afterward, when a user enters data, Oracle Database performs the following steps:

1. Retrieves the master key from the wallet.
2. Decrypts the encryption key using the master key.
3. Uses the encryption key to encrypt the data the user entered.
4. Stores the data in encrypted format in the database.

## 4.2 Securing Data - Oracle Virtual Private Database (VPD).

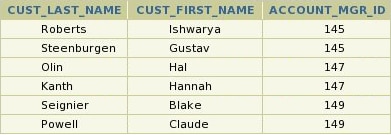
This feature restricts row and column level data access by creating a policy that enforces a WHERE clause for all SQL statements that query the database. You create and manage the VPD policy at the database table or view level, which means that you do not modify the applications that access the database.

VPD is used when the standard object privileges and associated database roles are insufficient to meet application security requirements. VPD policies can be simple or complex depending on your security requirements. VPD can be used in combination with the "application context" feature to enforce sophisticated row and/or column level security requirements for privacy and regulatory compliance. A simple VPD example might restrict access to data during business hours and a more complex VPD example might read an application context during a login trigger and enforce row level security against the ORDERS table.

No matter how users connect to the protected table (via an application, a Web interface or SQL\*Plus), the result is the same. There is no "application security problem" anymore, since the access policy is attached to the table, and cannot be bypassed.

  
Example: A customer can only see his orders in the 'orders' table (below), when he is listed in the 'customers' table (above).

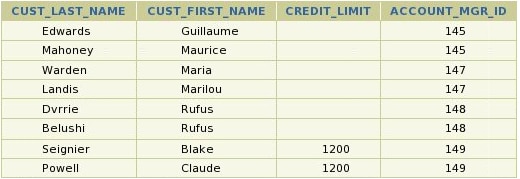
With "Column Relevance", VPD can be configured such that the policy is enforced **only** when a critical column is selected:



VPD Column Relevance (active)

Example: The account manager with the account\_mgr\_id "149" can see all rows from the customers table, but not the credit limits. As soon as she queries the 'credit\_limit' column, she can only see her own customers.

The most advanced configuration ("Column Hiding") of VPD allows for the most effective combination of ease-of-use and security: She still has access to all public information in the 'customers' table, but confidential information remains hidden:

  
Example: All 'credit\_limit' data cells are empty except those of her own customers.

## 4.3 Securing Data - Oracle Label Security (OLS)

This feature secures your database tables at the row level, and assigns these rows different levels of security based on security labels. You then create a security authorization for users based on the OLS labels.

Oracle Label Security is based on multi-level security (MLS) requirements that are found in government and defense organizations. You can easily restrict sensitive information to only authorized users. Oracle Label Security is based on Oracle Virtual Private Database (VPD). However, unlike VPD, Oracle Label Security provides the access mediation functions, data dictionary tables, and policy based architecture out of the box, eliminating customized coding and providing a consistent label based access control model that can be used by multiple applications. Oracle Label Security policies can be applied to one or more application tables. Oracle Label Security works by comparing the row label with a user's label authorizations. Oracle Label Security software is installed by default, but not automatically enabled.

Oracle Label Security provides several benefits for controlling row level management.

* It enables row level data classification and provides out of the box access mediation based on the data classification and the user label authorization or security clearance.
* It enables you to assign label authorizations or security clearances to both database users and application users.
* It provides both a graphical user interface and APIs for defining and storing data classification labels and user label authorizations.
* It integrates with Oracle Database Vault and Oracle Advanced Security Data Redaction, enabling security clearances to be use in both Database Vault command rules and Data Redaction policy definitions.

## 4.4 Securing Data - Oracle Database Vault

Oracle Database Vault restricts access to specific areas in an Oracle database from any user, including users who have administrative access. For example, you can restrict administrative access to employee salaries, customer medical records, or other sensitive information.

This enables you to apply fine-grained access control to your sensitive data in a variety of ways. It hardens your Oracle Database instance and enforces industry standard best practices in terms of separating duties from traditionally powerful users. Most importantly, it protects your data from super-privileged users but still allows them to maintain your Oracle databases. Oracle Database Vault is an integral component of your enterprise.

With Oracle Database Vault, you address the most difficult security problems remaining today: protecting against insider threats, meeting regulatory compliance requirements, and enforcing separation of duty.

## 4.5 Questions - Securing Data

1. Explain the function or purpose of Oracle Transparent Data Encryption to secure data? Answer =>

Oracle Database authenticates and authorizes to secure data in the database, but not in the operating system files where the data is stored. Oracle database provides transparent data encryption to protect those files. The purpose of this feature is to protect sensitive data in the database columns stored in the operating system files by encrypting it. It stores encryption keys in a security module external to the database to prevent unauthorized decryption.

2. When is it appropriate to use Oracle Transparent Data Encryption to secure data? Answer =>

usually it is necessary to encrypt such as credit card numbers, social security number or patient health information to prevent access when back up tapes or disk drives are stolen or lost. Now a days industry regulation such as payment card industry Data security standard and the healthcare insurance portability and Accountability Act have become a driving factor behind increased usage of encryption for protecting credit card and health care information.

3. How does Oracle Transparent Data Encryption protect DBAs from accessing user data? Answer =>

transparent data encryption is key-based access control system. If the encrypted data is retrieved somehow, it can not be understood until authorized decryption occurs,that is for the user who is authorized to the table. When a table contains encrypted columns, a single key is used regardless of the number of encrypted columns. The keys for all tables containing encrypted columns are encrypted with the database server master key and stored in a dictionary table in the database. This master key is stored in an external security module to prevent this from unauthorized access. For this external module, oracle uses an oracle wallet, which is also used to generate encryption keys and perform encryption and decryption.

4. Explain the function or purpose of an Oracle Virtual Private Database (VPD) to secure data? Answer =>

Oracle Virtual Private Database restricts row and column level data access by creating a policy that enforces a WHERE clause for all SQL statements. VPD policy can be managed at the database table or view level, which do not need to modify the application that access the database. VPD policy can be simple or complex , based on the requirement. A simple VPD example might restrict access to data during business hours and a more complex VPD example might read an application context during a login trigger and enforce row level security.

5. When is it appropriate to use Oracle Virtual Private Database (VPD) to secure data? Answer =>'

When the standard object privileges and associated database roles are insufficient , VPD is used to meet application security requirements.

6. Explain the function or purpose of Oracle Label Security (OLS) to secure data? Answer =>

The purpose of Oracle Level Security is to secure database tables at the row level, and assigns these rows different levels of security based on the security levels. A security authorization can be created for users based on the OLS level. It enables row level data classification and provides out of the box access mediation based on the data classification and the user level authorization or security database. It also allows to assign label authorizations or security clearences to both database users and application users. It provides both graphical user interface and APIs for defining and storing data classification labels and user label authorization. It combines with Oracle Database Vault and Oracle Advanced Security Data Redaction, helps to enable security clearences to be use in both Database Vault command rules and Data Redaction policy definitions.

7. When is it appropriate to use Oracle Label Security (OLS) to secure data? Answer =>

OLS is based on multi-level security requirements that are found in government and defense organization. Data in any field can be viewed as sensititive. Allowing information to be seen or used by inappropriate persons can be embarrassing, damaging to indivisuls, careers, organisations agencies goverments and coutries. Restricting access to entire tables or segregating sensitive data into separate databases can create an awkward working environment which may be costly in hardware, software , user time and administration. OLS security prevents the need for such measures by enabling row leve access control.

8. Explain the function or purpose of Oracle Database Vault to secure data? Answer =>

Oracle Database Vault restricts access to specific ares in an oracle database from any user, who have administrative access, reducing the risk of insider and outsider threats and addressing common compliance requirements. Oracle database Vault provides powerful cyber security controls to help protect application data from unauthorized or even authorized user access and improve compliance with privacy and regulatory requirements.

9. When is it appropriate to use Oracle Database Vault to secure data? Answer =>

it is sometimes necessary to protect data from the access to authorized user in the database . some data isnot even sharable with administrator access such as employee salaries, customer medical information, and other sensitive information. It protects data from such a way without disturbing administrative job.

10. How does Oracle Database Vault DBAs from accessing user data? Answer =>

Oracle database vault keeps the DBA from viewing application data, a top concern for customers who must protect sensitive business information or privacy data related to partners, employees and customers. it keeps DBA from performing tasks outside their authorized responsibilities. First you have make modifications to basic database parameters as the system user. Then Modifying a Rule set to disallow remote Alter System command execution. When the RULE set was created , the auditing option was set to Audit on Failure. You have do some action to verify the audit.

## 4.5 Database Auditing?

### 4.5.1 What is Database Auditing?

Auditing is the monitoring and recording of selected user database actions. In standard auditing, you use initialization parameters and the AUDIT and NOAUDIT SQL statements to audit SQL statements, privileges, and schema objects, and network and multitier activities.

There are also activities that Oracle Database always audits, regardless of whether auditing is enabled. These activities are administrative privilege connections, database startups, and database shutdowns

Another type of auditing is fine-grained auditing. Fine-grained auditing enables you to audit at the most granular level, data access, and actions based on content, using Boolean measurement, such as value > 1000. You can use fine-grained auditing to audit activities based on access to or changes in a column. You can create security policies to trigger auditing when someone accesses or alters specified elements in an Oracle database, including the contents within a specified object. You can create policies that define specific conditions that must take place for the audit to occur. For example, you can audit a particular table column to find out when and who tried to access it during a specified period of time. Furthermore, you can create alerts that are triggered when the policy is violated, and write this data to a separate audit file.

### 4.5.2 Why is Database Auditing Used?

You typically use auditing to perform the following activities:

1. Enable accountability for actions. These include actions taken in a particular schema, table, or row, or affecting specific content.
2. Deter users from inappropriate actions based on that accountability.
3. Investigate suspicious activity. For example, if a user is deleting data from tables, then a security administrator might decide to audit all connections to the database and all successful and unsuccessful deletions of rows from all tables in the database.
4. Notify an auditor of actions by an unauthorized user. For example, an unauthorized user could change or delete data, or a user has more privileges than expected, which can lead to reassessing user authorizations.
5. Detect problems with an authorization or access control implementation. For example, you can create audit policies that you expect will never generate an audit record because the data is protected in other ways. However, if these policies do generate audit records, then you will know the other security controls are not properly implemented.
6. Address auditing requirements for compliance. Regulations such as the following have common auditing-related requirements, such as Sarbanes-Oxley Act and Health Insurance Portability and Accountability Act (HIPAA)
7. Monitor and gather data about specific database activities. For example, the database administrator can gather statistics about which tables are being updated, how many logical I/O operations are performed, or how many concurrent users connect at peak times.

### 4.5.3 Type of Audit Records.

Oracle Database records audit activities in audit records. Audit records provide information about the operation that was audited, the user performing the operation, and the date and time of the operation. Audit records can be stored in either a data dictionary table, called the database audit trail, or in operating system files, called an operating system audit trail.

**Statement Auditing**

Enables you to audit SQL statements by type of statement, not by the specific schema objects on which they operate. Typically broad, statement auditing audits the use of several types of related actions for each option. For example, AUDIT TABLE tracks several DDL statements regardless of the table on which they are issued. You can also set statement auditing to audit selected users or every user in the database.

**Privilege Auditing**

Enables you to audit the use of powerful system privileges that enable corresponding actions, such as AUDIT CREATE TABLE. Privilege auditing is more focused than statement auditing, which audits only a particular type of action. You can set privilege auditing to audit a selected user or every user in the database.

**Schema Object Auditing**

Enables you to audit specific statements on a particular schema object, such as AUDIT SELECT ON employees. Schema object auditing is very focused, auditing only a single specified type of statement (such as SELECT) on a specified schema object. Schema object auditing always applies to all users of the database.

**Fine-Grained Auditing**

Enables you to audit at the most granular level, data access and actions based on content, using any Boolean measure, such as value > 1,000,000. Enables auditing based on access to or changes in a column

## 4.6 Questions - Oracle Database Auditing

1. What is Database Auditing? Answer =>

Auditing is always about accountability, and is frequently done to protect and preserve privacy for the information stored in database. Concern about privacy policies and practices has been rising steadily with ubiquitous use of database in businesses and on the ineternet. Oracle database privides a depth of auditing that readily enables system adminitrators to implement enhanced protections, early detection of suspicious activities, and finely-tuned security responses.

2. List four or more reasons why a database should be audited? Answer =>

it enbles future accountability for cuurent actions taken in a particular schema , table, or row or affecting specific content.

It helps to detect suspicious activity. For example, if some user is deleting data from tables, then the security administrator might decide to audit all connections to the database and all successful and unsuccessful deletions of rows from all tables in the database. It also allows auditor to detect any unauthorized user is manipulating or deleting data and that if any user has more privileges than expected. It allows to monitor and gather data about specific database activities.

3. Describe the function or purpose of Statement Database auditing? Answer =>

It enables to audit SQL statements by type of statement, not by the specific schema objects on which they operate. Statement auditing broadly audits the use of several types of related action for each option. AUDIT TABLE tracks several DDL statements,without any information of the table on which they are issued.

4. Describe the function or purpose of Privilege Database auditing? Answer =>

Privilege Database auditing audits powerful system privileges which enable corresponding actions, like AUDIT CREATE TABLE. As privilege auditing is more focused than statement auditing, it audits a particular type of action. Privilege auditing can be set to audit a selected user.

5. Describe the function or purpose of Schema Object auditing? Answer =>

Selective auditing of specific statements on a particular schema object. Schema object auditing audits only a specific statement on a specific schema object . this audit applies to all users of the database.

6. Describe the function or purpose of Fine-grain Database auditing? Answer =>

Fine grained Auditing enables audit policies to be associated with columns in application tables along with conditions necessary for an audit record to be generated. It can be used to create audit records when a table is accessed during specific periods or specific columns are accessed.