Auditoría de una base de datos Oracle ¿qué es? ¿para qué sirve? ¿cómo se activa? Proyecto AjpdSoft

Os explicamos en este artículo en qué consiste la auditoría de una base de datos Oracle. Para el ejemplo utilizamos Oracle Database 11g. Os mostramos para qué sirve, cómo consultar los datos de auditoría generados por Oracle y cómo desactivar esta opción y volver a activarla.

Definición de Auditoría informática

La auditoría informática es el proceso de recoger, agrupar y evaluar evidencias para determinar si un Sistema de Información salvaguarda el activo empresarial, mantiene la integridad de los datos,

lleva a cabo eficazmente los fines de la organización, utiliza eficientemente los recursos, y cumple con las leyes y regulaciones establecidas. También permite detectar de forma sistemática el uso de los recursos y los flujos de información dentro de una organización y determinar qué información es crítica para el cumplimiento de su misión y objetivos, identificando necesidades, duplicidades, costes, valor y barreras, que obstaculizan flujos de información eficientes.

Auditar consiste principalmente en estudiar los mecanismos de control que están implantados en una empresa u organización, determinando si los mismos son adecuados y cumplen unos determinados objetivos o estrategias, estableciendo los cambios que se deberían realizar para la consecución de los mismos. Los mecanismos de control pueden ser directivos, preventivos, de detección, correctivos o de recuperación ante una contingencia.

Definición de Auditoría para Oracle

En el caso de Oracle Database, la auditoría es un conjunto de características que permite al administrador de la base de datos y a los usuarios hacer un seguimiento del uso de la base de datos. El administrador de base de datos puede definir la actividad de auditoría predeterminada. La información de las auditorías se almacena en el diccionario de datos, en la tablaSYS.AUD$ o en la pista de auditoría del sistema operativo (si lo permite). Lo anterior viene definido en el parámetro audit_trail.

Se pueden auditar tres tipos de acciones: intentos de inicio de sesión, accesos a objetos y acciones de la base de datos. Cuando se realizan auditorías, la funcionalidad de la base de datos es dejar constancia de los comandos correctos e incorrectos. Esto puede modificarse cuando se configura cada tipo de auditoría.

Por ejemplo, se pueden registrar todos los intentos de actualizar los datos de una tabla o sólo los intentos fallidos, también se pueden registrar todos los inicios de sesión en Oracle o sólo los intentos fallidos.

En qué consiste la auditoría en Oracle, tablas y vistas que intervienen

Oracle Databasealmacena en el diccionario de datos, en la tabla SYS.AUD$ o en la pista de auditoría del sistema operativo (si lo permite). Existen varias vistas que se basan en esta tabla (SYS.AUD$) para mostrar distintos resultados, según la información que se quiera obtener:

- ALL_AUDIT_POLICIES
- ALL_AUDIT_POLICY_COLUMNS
- ALL_DEF_AUDIT_OPTS
- ALL_REPAUDIT_ATTRIBUTE
- ALL_REPAUDIT_COLUMN
- APEX_DEVELOPER_AUDIT_LOG
- DBA_AUDIT_EXISTS
- DBA_AUDIT_OBJECT
- DBA_AUDIT_POLICIES

- DBA_AUDIT_POLICY_COLUMNS
- DBA_AUDIT_SESSION
- DBA_AUDIT_STATEMENT
- DBA_AUDIT_TRAIL
- DBA_COMMON_AUDIT_TRAIL
- DBA_FGA_AUDIT_TRAIL
- DBA_OBJ_AUDIT_OPTS
- DBA_PRIV_AUDIT_OPTS
- DBA_REPAUDIT_ATTRIBUTE
- DBA_REPAUDIT_COLUMN
- DBA_STMT_AUDIT_OPTS
- GV_$XML_AUDIT_TRAIL
- KU$_AUDIT_DEFAULT_VIEW
- KU$_AUDIT_OBJ_BASE_VIEW
- KU$_AUDIT_OBJ_VIEW
- KU$_AUDIT_VIEW
- KU$_PROC_AUDIT_VIEW
- KU$_PROCDEPOBJ_AUDIT_VIEW
- KU$_PROCOBJ_AUDIT_VIEW
- KU$_10_1_AUDIT_VIEW
- MGMT$AUDIT_LOG
- MGMT$ESA_AUDIT_SYSTEM_REPORT
- SM$AUDIT_CONFIG
- USER_AUDIT_OBJECT
- USER_AUDIT_POLICIES
- USER_AUDIT_POLICY_COLUMNS
- USER_AUDIT_SESSION
- USER_AUDIT_STATEMENT
- USER_AUDIT_TRAIL
- USER_OBJ_AUDIT_OPTS
- USER_REPAUDIT_ATTRIBUTE
- USER_REPAUDIT_COLUMN
- V_$XML_AUDIT_TRAIL

Estas vistas se pueden ver ejecutando la consulta SQL:

```
SELECT view_name
FROM dba_views
WHERE view_name LIKE '%AUDIT%'
ORDER BY view_name
```

Las principales son:

- **DBA_AUDIT_OBJECT**: guarda la información relativa a la auditoría de
- **DBA_AUDIT_SESSION**: guarda la información relativa a la auditoría de los inicios de sesión de los usuarios.
- **DBA_AUDIT_TRAIL**: muestra la auditoría estándar (de la tabla AUD$).
- **USER_AUDIT_TRAIL**: muestra la auditoría estándar (de la tabla AUD$) relativa al usuario actual.
- **DBA_FGA_AUDIT_TRAIL**: muestra información de auditoría de grano fino (obtenida de FGA_LOG$). La auditoría de grano fino (FGA) extiende la auditoría estándar y, además, captura la sentencia SQL que ha sido ejecutada.

Todo lo anterior estará condicionado al tipo de auditoría que se haya establecido para la base de datos Oracle, como se indica aquí.

Cómo comprobar si una instancia de Oracle tiene activada la auditoría

La activación de la auditoría en Oracle Database viene definida por el valor del parámetro: ***audit_trail***. Para comprobar si la auditoría de la base de datos está activada ejecutaremos el siguiente comando SQL:

```
select name, value
from v$parameter
where name like 'audit_trail'
```

Posibles valores del parámetro ***audit_trail***:

- **none**: desactiva la auditoría de la base de datos.
- **os**: activa la auditoría de la base de datos. Los sucesos auditados se escribirán en la pista de auditoría del sistema operativo, no se auditará en Oracle sino en el sistema operativo anfitrión. Esta opción funcionará dependiendo del sistema operativo.
- **db**: activa la auditoría y los datos se almacenarán en la taba SYS.AUD$ de Oracle.
- **db, extended**: activa la auditoría y los datos se almacenarán en la taba SYS.AUD$ de Oracle. Además se escribirán los valores correspondientes en las columnas SQLBIND y SQLTEXT de la tabla SYS.AUD$.
- **xml**: activa la auditoría de la base de datos, los sucesos será escritos en ficheros XML del sistema operativo.
- **xml, extended**: activa la auditoría de la base de datos, los sucesos será escritos en el formato XML del sistema operativo, además se incluirán los valores de SqlText y SqlBind.

Para **activar la auditoría**:

```
ALTER SYSTEM SET audit_trail = "DB" SCOPE=SPFILE;
```

Para **desactivar la auditoría** ejecutaremos el siguiente comando:

ALTER SYSTEM SET audit_trail = "NONE" SCOPE=SPFILE;

Nota:

- En

  Oracle 9i

  la auditoría viene desactivada por defecto, el valor del parámetro "audit_trail" está a "NONE".

- En

  Oracle 11g

  la auditoría viene activada por defecto, el valor del parámetro "audit_trail" está a "DB".

Los comandos audit y noaudit

Comando audit

Funcionamiento comando audit

El comando ***audit*** permite iniciar los tipos de auditoría que a continuación se detallan. Este comando puede funcionar aunque no esté activada la auditoría de la base de datos, pero no dejará constancia, para que funcione correctamente es necesario que la auditoría esté activada, como se indica aquí.

- **Auditorías de inicio de sesión**: cada intento de conexión con la base de datos por parte de un usuario (bien una aplicación externa o las aplicaciones del propio Oracle) puede ser auditado. El comando para iniciar la auditoría de los intentos de inicio de sesión es:

  audit session;

  El comando anterior auditará tanto los intentos fallidos como los aciertos.

  Para auditar sólo los intentos **fallidos** utilizaremos el comando:

  audit session whenever not successful;

  Para auditar sólo las conexiones **correctas** utilizaremos el comando:

  audit session whenever successful;

- **Auditorías de acción**: cualquier acción que afecte a un objeto de la base de datos (tabla, enlace de base de datos, espacio de tablas, sinónimo, segmento de anulación, usuario, índice, etc.) puede auditarse. Las posibles acciones que pueden auditarse (create, alter, drop) sobre estos objetos pueden agruparse para simplificar la cantidad de esfuerzo administrativo necesario para determinar y mantener las opciones de configuración de la auditoría. Por ejemplo, para auditar todos los comandos que afectan a los roles puede emplearse el comando SQL:

audit role;

Este comando activará la auditoría de las acciones: *create role*, *alter role*, *drop role* y *set role*.

También se puede ser más selectivo, por ejemplo, si queremos auditar a un usuario concreto cuando realiza la acción "update" ejecutaremos el siguiente comando:

audit update table by nombre_usuario;

De esta forma se activará la auditoría para el usuario "nombre_usuario" sólo cuando ejecute el comando "update" para cualquier tabla.

- **Auditorías de objeto**: además de las acciones a nivel de sistema sobre objetos, también es posible auditar las acciones de manipulación de datos sobre objetos. Se pueden auditar operaciones de *select*, *insert*, *update* y *delete* sobre tablas. Este tipo de auditoría es similar a la anterior de auditoría de acción, la única diferencia es que el comando "audit" incorpora un parámetro nuevo "by session" (el registro de auditoría se escribirá una única vez por sesión) o "by access" (el registro de auditoría se escribirá cada vez que se acceda al objeto auditado).

Por ejemplo, para auditar los "insert" realizados sobre la tabla "facturacion" por acceso, el comando será:

audit insert on FACTURACION by access;

Nota: al indicar "by access" hay que tener cuidado pues registrará un suceso de auditoría por cada insert, esto puede afectar al rendimiento. De ser así siempre será mejor optar por "by session" que sólo registrará un suceso de auditoría por sesión, aunque es menos exaustivo.

Otro ejemplo, para auditar todas las acciones realizadas en la tabla "contabilidad" por sesión utilizaremos el siguiente comando:

audit all on CONTABILIDAD by session;

El comando anterior auditará todas las acciones realizadas sobre la tabla FACTURACION (select, insert, update, delete), pero sólo un registro de auditoría por cada sesión.

Otro ejemplo, para auditar las eliminaciones de registros de la tabla "nóminas":

audit delete NOMINAS by access;

Prerequisitos para poder ejecutar audit

Para activar la auditoría de las instrucciones SQL con el comando ***audit*** se necesita el privilegio de sistema **AUDIT SYSTEM**.

El usuario que se desee pueda activar la auditoría de objetos de un esquema, tiene que ser el propietario del objeto o disponer del privilegio de sistema AUDIT ANY. Además, si el objeto que

eligió para la auditoría se ubica en un directorio, incluso habiéndolo creado uno mismo, se necesita el privilegio de sistema AUDIT ANY.

Para obtener los resultados de la auditoría hay que definir correctamente el parámetro de inicialización audit_trail. Se podrán definir las opciones de auditoría con el comando ***audit*** pero, si no está activada la auditoría en la base de datos, Oracle no generará los regisros de auditoría.

Sintaxis comando audit

```
AUDIT
  { sql_statement_clause | schema_object_clause | NETWORK }
  [ BY { SESSION | ACCESS } ]
  [ WHENEVER [ NOT ] SUCCESSFUL ] ;
```

- sql_statement_clause: activa la auditoría para una sentencia SQL concreta.
- schema_object_clause: activa la auditoría para un objeto concreto de la base de datos.
- WHENEVER SUCCESSFUL: activa la auditoría sólo para operaciones e instrucciones SQL en objetos de esquema que se completen con éxito.
- WHENEVER NOT SUCCESSFUL: activa la auditoría sólo para operaciones e instrucciones SQL en objetos de esquema que originen error.

La sintaxis de este comando tiene muchas más opciones, disponibles en la ayuda de Oracle.

Comando noaudit

Funcionamiento comando noaudit

La instrucción ***noaudit*** se utiliza para detener la actividad de auditoría que se había activado previamente con la instrucción audit. Esta instrucción no influye en el parámetro audit_trail.

La instrucción ***noaudit*** debe tener la misma sintaxis que la instrucción audit que queramos detener. Por ejemplo, si hemos auditado un usuario con:

audit session by alonso;

Auditará los inicios de sesión para el usuario de Oracle "alonso", tanto los fallidos como los correctos. Para desactivar esta auditoría ejecutaremos el comando:

noaudit session by alonso;

Hay que tener en cuenta que el comando ***noaudit*** sólo desactiva la auditoría de su comando audit análogo. Por ejemplo, si ejecutamos este comando:

audit session by alonso;

Que auditará los inicios de sesión para el usuario "alonso". Y luego este otro:

audit session;

Que aditará los inicios de sesión para todos los usuarios.

Si ejecutásemos ahora el comando:

noaudit sesión;

Desactivaría el comando "audit session", pero seguría auditándose al usuario "alonso", puesto que aún estaría activo el comando "audit session by alonso".

Prerequisitos para poder ejecutar noaudit

Para detener la auditoría de las instrucciones SQL con el comando ***noaudit*** se necesita el privilegio de sistema **AUDIT SYSTEM**.

El usuario que se desee pueda detener la auditoría de objetos de un esquema, tiene que ser el propietario del objeto o disponer del privilegio de sistema AUDIT ANY. Además, si el objeto que eligió para la auditoría se ubica en un directorio, incluso habiéndolo creado uno mismo, se necesita el privilegio de sistema AUDIT ANY.

Sintaxis comando noaudit

```
NOAUDIT
   { sql_statement_clause | schema_object_clause | NETWORK}
   [ WHENEVER [ NOT ] SUCCESSFUL ] ;
```

- sql_statement_clause: detiene la auditoría de una sentencia SQL concreta.
- schema_object_clause: detiene la auditoría para un objeto concreto de la base de datos.
- WHENEVER SUCCESSFUL: detiene la auditoría sólo para operaciones e instrucciones SQL en objetos de esquema que se completen con éxito.
- WHENEVER NOT SUCCESSFUL: detiene la auditoría sólo para operaciones e instrucciones SQL en objetos de esquema que originen error.

La sintaxis de este comando tiene muchas más opciones, disponibles en la ayuda de Oracle.

Consultar los datos de auditoría guardados

Dependiendo del tipo de auditoría que queramos consultar utilizaremos una u otra consulta SQL.

- Para el caso de la auditoría de **inicio de sesión** utilizaremos la siguiente consulta SQL:

```
select OS_Username Usuario_SO,
  Username Usuario_Oracle, Terminal ID_Terminal,
  DECODE (Returncode, '0', 'Conectado', '1005', 'Fallo - Null',
  1017, 'Fallo', Returncode) Tipo_Suceso,
  TO_CHAR(Timestamp, 'DD-MM-YY HH24:MI:SS') Hora_Inicio_Sesion,
  TO_CHAR(Logoff_Time, 'DD-MM-YY HH24:MI:SS') Hora_Fin_Sesion
from DBA_AUDIT_SESSION;
```

- Para el caso de la auditoría de **acción** utilizaremos la siguiente consulta SQL:

```
select OS_Username Usuario_SO,
  Username Usuario_Oracle, Terminal ID_Terminal,
  Owner Propietario_Objeto,
  Obj_Name Nombre_Objeto,
  Action_Name Accion,
  DECODE (Returncode, '0', 'Realizado', 'Returncode')
Tipo_Suceso,
  TO_CHAR (Timestamp,  'DD-MM-YY HH24:MI:SS') Hora
from DBA_AUDIT_OBJECT;
```

Montando el escenario, probando la auditoría de Oracle

En primer lugar accederemos a Oracle con un usuario con privilegios suficientes para crear usuarios y establecer permisos. Con este usuario crearemos un usuario en Oracle, para ello utilizaremos los cuatro comandos SQL siguientes:

CREATE USER "ALONSO" PROFILE "DEFAULT"
IDENTIFIED BY "xxx" DEFAULT TABLESPACE "USERS"
ACCOUNT UNLOCK;

GRANT "CONNECT" TO "ALONSO";

GRANT CREATE TABLE TO "ALONSO";

ALTER USER "RAFAEL"
QUOTA 100M ON "USERS";

A continuación accederemos a Oracle con el usuario "alonso" y crearemos una tabla para las pruebas, llamada "facturas", con el comando:

create table facturas (
codigo number primary key,
fecha date default sysdate);

Accederemos a Oracle con un usuario con privilegios suficientes para activar sucesos de auditoría y ejecutaremos el comando:

audit session by alonso;

para auditar el inicio de sesión de los usuarios.

Y ejecutaremos el comando:

audit all on facturas by access;

para auditar auditar los cambios realizados en la tabla "facturas" (update, delete, insert, select).

Volveremos a acceder a Oracle con el usuario "alonso" y realizaremos algunas inserciones, eliminaciones y select en la tabla "facturas":

insert into facturas (codigo) values (1);

insert into facturas (codigo) values (2);

insert into facturas (codigo) values (3);

*select * from facturas;*

delete facturas where codigo=2;

update facturas set codigo=33 where codigo=2;

Ahora volveremos a acceder con un usuario con privilegios suficientes para consultar las vistas de

auditoría y ejecutaremos la sentencia SQL para ver los accesos del usuario "alonso":

```
select OS_Username Usuario_SO,
  Username Usuario_Oracle, Terminal ID_Terminal,
  DECODE (Returncode, '0', 'Conectado', '1005', 'Fallo -
Null',
  1017, 'Fallo', Returncode) Tipo_Suceso,
  TO_CHAR(Timestamp, 'DD-MM-YY HH24:MI:SS')
Hora_Inicio_Sesion,
  TO_CHAR(Logoff_Time, 'DD-MM-YY HH24:MI:SS') Hora_Fin_Sesion
from DBA_AUDIT_SESSION
where Username="ALONSO";
```

y esta otra para ver las acciones realizadas a la tabla "facturas":

```
select OS_Username Usuario_SO,
  Username Usuario_Oracle, Terminal ID_Terminal,
  Owner Propietario_Objeto,
  Obj_Name Nombre_Objeto,
  Action_Name Accion,
  DECODE (Returncode, '0', 'Realizado', 'Returncode') Tipo_Suceso,
  TO_CHAR (Timestamp,  'DD-MM-YY HH24:MI:SS') Hora
from DBA_AUDIT_OBJECT
where Obj_Name = "FACTURAS";
```

Anexo

- **Para ejecutar las sentencias y comandos SQL anteriores se pueden utilizar**:
  - Aplicaciones de terceros como TOAD.
  - Aplicación AjpdSoft Administración Bases de Datos, gratuita y disponible en esta web.
  - Utilizar la utilidad de ejecución de consultas SQL vía web (a partir de la versión 10g) del propio Oracle: **Hoja de Trabajo de SQL**.
- **Cambiar la tabla SYS.AUD$ a otro tablespace**: la auditoría de la base de datos, si no se realiza con control, puede provocar efectos negativos en el rendimiento, además, la tabla AUD$ puede crecer considerablemente, por lo que es recomendable moverla a un tablespace distinto al de defecto: SYSTEM. Para realizar esto seguiremos los siguientes pasos:

  1. En primer lugar crearemos el tablespace (si no lo hemos hecho ya) al que moveremos la tabla, podemos utilizar la siguiente consulta SQL para hacerlo:

     create tablespace nombre_tablespace
     logging
     datafile 'D:/oracle/auditoria/auditoria.dbf'
     size 100m
     autoextend on
     next 32m maxsize 10000m;

  2. Desactivaremos la auditoría de la base de datos con el comando:

     ALTER SYSTEM SET audit_trail = "NONE" SCOPE=SPFILE;

  3. A continuación moveremos la tabla al tablespace creado, con el comando

SQL:

```
ALTER TABLE aud$
MOVE TABLESPACE nombre_tablespace;
```

Nota: este proceso conviene realizarlo con el usuario **SYS**, accediendo como **SYSOPER**.

- Estructura de la tabla **SYS.AUD$**:

Campo	Tipo de datos	Tamaño	Permite nulos
SESSIONID	NUMBER	22	N
ENTRYID	NUMBER	22	N
STATEMENT	NUMBER	22	N
TIMESTAMP#	DATE	7	Y
USERID	VARCHAR2	30	Y
USERHOST	VARCHAR2	128	Y
TERMINAL	VARCHAR2	255	Y
ACTION#	NUMBER	22	N
RETURNCODE	NUMBER	22	N
OBJ$CREATOR	VARCHAR2	30	Y
OBJ$NAME	VARCHAR2	128	Y
AUTH$PRIVILEGES	VARCHAR2	16	Y
AUTH$GRANTEE	VARCHAR2	30	Y
NEW$OWNER	VARCHAR2	30	Y
NEW$NAME	VARCHAR2	128	Y
SES$ACTIONS	VARCHAR2	19	Y
SES$TID	NUMBER	22	Y
LOGOFF$LREAD	NUMBER	22	Y
LOGOFF$PREAD	NUMBER	22	Y
LOGOFF$LWRITE	NUMBER	22	Y
LOGOFF$DEAD	NUMBER	22	Y
LOGOFF$TIME	DATE	7	Y
COMMENT$TEXT	VARCHAR2	4000	Y
CLIENTID	VARCHAR2	64	Y
SPARE1	VARCHAR2	255	Y
SPARE2	NUMBER	22	Y
OBJ$LABEL	RAW	255	Y
SES$LABEL	RAW	255	Y
PRIV$USED	NUMBER	22	Y

SESSIONCPU	NUMBER	22	Y
NTIMESTAMP#	timestamp	11	Y
PROXY$SID	NUMBER	22	Y
USER$GUID	VARCHAR2	32	Y
INSTANCE#	NUMBER	22	Y
PROCESS#	VARCHAR2	16	Y
XID	RAW	8	Y
AUDITID	VARCHAR2	64	Y
SCN	NUMBER	22	Y
DBID	NUMBER	22	Y
SQLBIND	CLOB	4000	Y
SQLTEXT	CLOB	4000	Y
OBJ$EDITION	VARCHAR2	30	Y

- Posibles acciones a auditar, contenido de la vista **audit_actions**:

ACTION	NAME
0	UNKNOWN
1	CREATE TABLE
2	INSERT
3	SELECT
4	CREATE CLUSTER
5	ALTER CLUSTER
6	UPDATE
7	DELETE
8	DROP CLUSTER
9	CREATE INDEX
10	DROP INDEX
11	ALTER INDEX
12	DROP TABLE
13	CREATE SEQUENCE
14	ALTER SEQUENCE
15	ALTER TABLE
16	DROP SEQUENCE
17	GRANT OBJECT
18	REVOKE OBJECT
19	CREATE SYNONYM
20	DROP SYNONYM
21	CREATE VIEW
22	DROP VIEW
23	VALIDATE INDEX
24	CREATE PROCEDURE
25	ALTER PROCEDURE
26	LOCK
27	NO-OP

28	RENAME
29	COMMENT
30	AUDIT OBJECT
31	NOAUDIT OBJECT
32	CREATE DATABASE LINK
33	DROP DATABASE LINK
34	CREATE DATABASE
35	ALTER DATABASE
36	CREATE ROLLBACK SEG
37	ALTER ROLLBACK SEG
38	DROP ROLLBACK SEG
39	CREATE TABLESPACE
40	ALTER TABLESPACE
41	DROP TABLESPACE
42	ALTER SESSION
43	ALTER USER
44	COMMIT
45	ROLLBACK
46	SAVEPOINT
47	PL/SQL EXECUTE
48	SET TRANSACTION
49	ALTER SYSTEM
50	EXPLAIN
51	CREATE USER
52	CREATE ROLE
53	DROP USER
54	DROP ROLE
55	SET ROLE
56	CREATE SCHEMA
57	CREATE CONTROL FILE
59	CREATE TRIGGER
60	ALTER TRIGGER
61	DROP TRIGGER
62	ANALYZE TABLE
63	ANALYZE INDEX
64	ANALYZE CLUSTER
65	CREATE PROFILE
66	DROP PROFILE
67	ALTER PROFILE
68	DROP PROCEDURE
70	ALTER RESOURCE COST
71	CREATE MATERIALIZED VIEW LOG

72	ALTER MATERIALIZED VIEW LOG
73	DROP MATERIALIZED VIEW LOG
74	CREATE MATERIALIZED VIEW
75	ALTER MATERIALIZED VIEW
76	DROP MATERIALIZED VIEW
77	CREATE TYPE
78	DROP TYPE
79	ALTER ROLE
80	ALTER TYPE
81	CREATE TYPE BODY
82	ALTER TYPE BODY
83	DROP TYPE BODY
84	DROP LIBRARY
85	TRUNCATE TABLE
86	TRUNCATE CLUSTER
91	CREATE FUNCTION
92	ALTER FUNCTION
93	DROP FUNCTION
94	CREATE PACKAGE
95	ALTER PACKAGE
96	DROP PACKAGE
97	CREATE PACKAGE BODY
98	ALTER PACKAGE BODY
99	DROP PACKAGE BODY
100	LOGON
101	LOGOFF
102	LOGOFF BY CLEANUP
103	SESSION REC
104	SYSTEM AUDIT
105	SYSTEM NOAUDIT
106	AUDIT DEFAULT
107	NOAUDIT DEFAULT
108	SYSTEM GRANT
109	SYSTEM REVOKE
110	CREATE PUBLIC SYNONYM
111	DROP PUBLIC SYNONYM
112	CREATE PUBLIC DATABASE LINK
113	DROP PUBLIC DATABASE LINK

114	GRANT ROLE
115	REVOKE ROLE
116	EXECUTE PROCEDURE
117	USER COMMENT
118	ENABLE TRIGGER
119	DISABLE TRIGGER
120	ENABLE ALL TRIGGERS
121	DISABLE ALL TRIGGERS
122	NETWORK ERROR
123	EXECUTE TYPE
128	FLASHBACK
129	CREATE SESSION
130	ALTER MINING MODEL
131	SELECT MINING MODEL
133	CREATE MINING MODEL
157	CREATE DIRECTORY
158	DROP DIRECTORY
159	CREATE LIBRARY
160	CREATE JAVA
161	ALTER JAVA
162	DROP JAVA
163	CREATE OPERATOR
164	CREATE INDEXTYPE
165	DROP INDEXTYPE
166	ALTER INDEXTYPE
167	DROP OPERATOR
168	ASSOCIATE STATISTICS
169	DISASSOCIATE STATISTICS
170	CALL METHOD
171	CREATE SUMMARY
172	ALTER SUMMARY
173	DROP SUMMARY
174	CREATE DIMENSION
175	ALTER DIMENSION
176	DROP DIMENSION
177	CREATE CONTEXT
178	DROP CONTEXT
179	ALTER OUTLINE
180	CREATE OUTLINE
181	DROP OUTLINE
182	UPDATE INDEXES
183	ALTER OPERATOR
197	PURGE USER_RECYCLEBIN

198	PURGE DBA_RECYCLEBIN
199	PURGE TABLESPACE
200	PURGE TABLE
201	PURGE INDEX
202	UNDROP OBJECT
204	FLASHBACK DATABASE
205	FLASHBACK TABLE
206	CREATE RESTORE POINT
207	DROP RESTORE POINT
208	PROXY AUTHENTICATION ONLY
209	DECLARE REWRITE EQUIVALENCE
210	ALTER REWRITE EQUIVALENCE
211	DROP REWRITE EQUIVALENCE
212	CREATE EDITION
213	ALTER EDITION
214	DROP EDITION
215	DROP ASSEMBLY
216	CREATE ASSEMBLY
217	ALTER ASSEMBLY
218	CREATE FLASHBACK ARCHIVE
219	ALTER FLASHBACK ARCHIVE
220	DROP FLASHBACK ARCHIVE

- **Algunos resultados tras la ejecución de comandos SQL**:
  - Al ejecutar un "audit", si todo es correcto Oracle devolverá: "Audit succeeded".
  - Al ejecutar un "noaudit", si todo es correcto Oracle devolverá: "Noaudit succeeded".
  - Al ejecutar un "audit session by nombre_usuario", si no existe el usuario devolverá:

    ORA-01435: user does not exist

  - Al ejecutar un "audit", si el usuario con el que lo ejecutamos no tiene permisos suficientes mostrará el error:

    ORA-01031: privilegios insuficientes

Artículos relacionados

- Instalar Oracle Database 11g Standard Edition en Windows XP Profesional.
- Instalar Oracle Database 10g en Windows XP.
- Instalar Oracle Database 10g Express Edition XE en Linux Ubuntu 6.06.

- Manual para instalar Oracle 9i en Windows con capturas de pantalla.
- Instalación de Oracle Client en Windows XP.
- Instalar y realizar aplicación web con Oracle Application Express.
- Instalación y configuración de Windows XP Service Pack 3.
- Concepto y ejemplo de creación de disparadores (triggers) en Oracle.
- Tipos de datos / Datatypes en Oracle.
- Cómo acceder a Oracle con Delphi sin utilizar código fuente.
- Consultas SQL de Oracle para obtener datos de una tabla.
- Oracle Database.
- AjpdSoft Administración Bases de Datos.
- AjpdSoft Monitor Espacio Oracle Código Fuente Delphi.
- Definición ODBC.
- Definición SQL.

Créditos

Artículo realizado íntegramente por Alonsojpd miembro fundador del proyecto AjpdSoft.

Nota: *Revisado por AjpdSoft el 18-08-2009.*