**DATABASE SECURITY AND PRIVACY ASSIGNMENT-01**

**CASE STUDY SCENARIO QUESTION:**

Produce an audit trail for DML statement activities on one table using Oracle 10g database feature. Use any table in any schema other than SYS or SYSTEM.

**TEAM MEMBERS:**

* KIRTHIKA SIVAKUMAR (RA1711003020014)
* SWETA PASAYAT (RA1711003020030)
* BARBIE DAS (RA1711003020063)

**CLASS AND SECTION:**

* CSE DEPARTEMENT
* 3RD YEAR
* A SECTION
* SRM IST, RAMAPURAM

**WHAT DOES DML STATEMENT AUDIT TRAIL MEAN?**

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. Oracle Database also provides a set of data dictionary views that you can use to track suspicious activities When you use standard auditing, Oracle Database writes the audit records to either to DBA\_AUDIT\_TRAIL (the SYS.AUD$ table), the operating system audit trail, or to the DBA\_COMMON\_AUDIT\_TRAIL view, which combines standard and fine-grained audit log records. In addition, the actions performed by administrators are recorded in the syslog audit trail when the AUDIT\_SYSLOG\_LEVEL initialization parameter is set.

**DML ACTION AUDITING WITH ORACLE**

To allow auditing on the server you must:

* Set "audit trail = true" in the init.ora file.
* Run the $ORACLE\_HOME/rdbms/admin/cataudit.sql script while connected as SYS.

Assuming that the "fireid" user is to be audited.

* CONNECT sys/password AS SYSDBA
* AUDIT ALL BY fireid BY ACCESS;
* AUDIT SELECT TABLE, UPDATE TABLE, INSERT TABLE, DELETE TABLE BY fireid BY ACCESS;
* AUDIT EXECUTE PROCEDURE BY fireid BY ACCESS;

These options audit all DDL & DML issued by "fireid", along with some system events.

* DDL (CREATE, ALTER & DROP of objects)
* DML (INSERT UPDATE, DELETE, SELECT, EXECUTE).
* SYSTEM EVENTS (LOGON, LOGOFF etc.)

The audit trail is stored in the SYS.AUD$ table. Its contents can be viewed directly or via the following views.

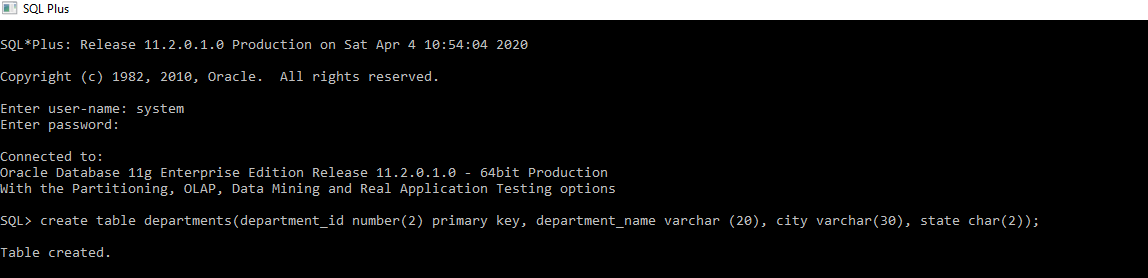
* DBA\_AUDIT\_EXISTS
* DBA\_AUDIT\_OBJECT
* DBA\_AUDIT\_SESSION
* DBA\_AUDIT\_STATEMENT
* DBA\_AUDIT\_TRAIL
* DBA\_OBJ\_AUDIT\_OPTS
* DBA\_PRIV\_AUDIT\_OPTS
* DBA\_STMT\_AUDIT\_OPTS

The audit trail contains a lot of data, but the following are most likely to be of interest.

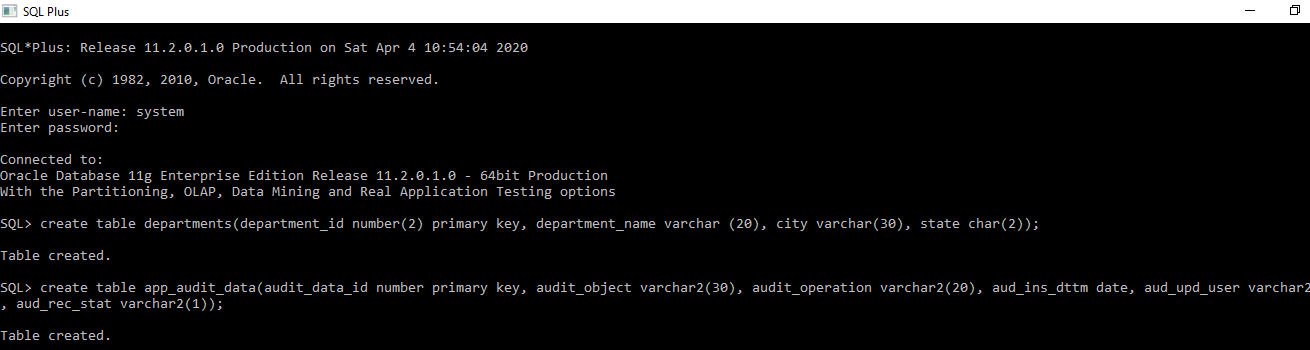
* USERNAME: Oracle Username.
* TERMINAL: Machine that the user performed the action from.
* TIMESTAMP: When the action occurred.
* OBJECT\_OWNER: The owner of the object that was interacted with.
* OBJECT\_NAME: The name of the object that was interacted with.
* ACTION\_NAME: The action that occurred against the object. (INSERT, UPDATE, DELETE, SELECT, EXECUTE)

**STEPS TO IMPLEMENT GIVEN USER CASE:**

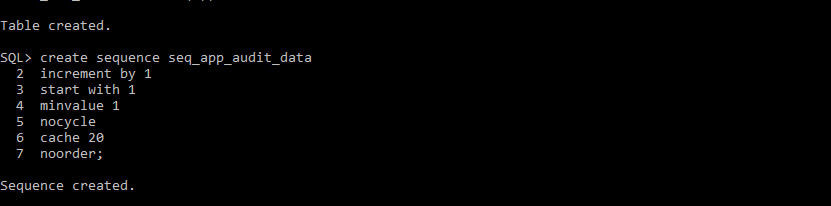
1. Using a user other than SYSTEM or SYS with privileges to create TABLES, SEQUENCES, and TRIGGERS, create the DEPARTMENTS table. An audit trigger can be created on the table which will compare the old and new values of all the columns and in case of a difference will log the old record into an audit table.



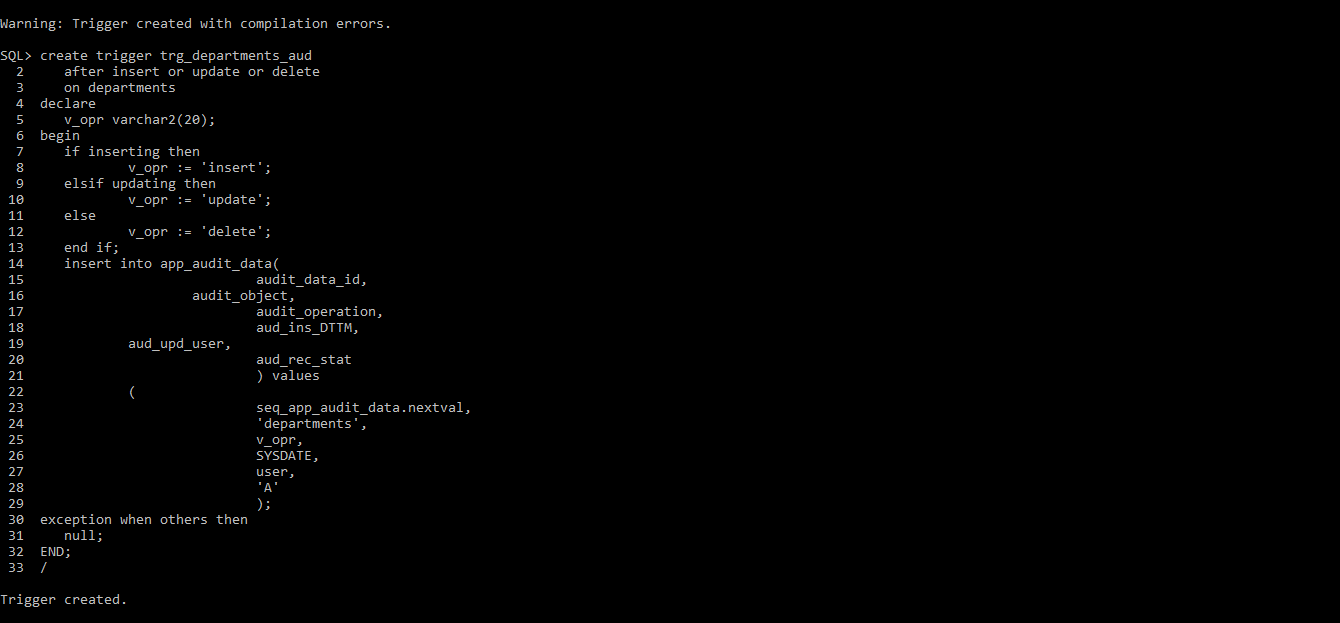
1. Create the auditing table APP\_AUDIT\_DATA having column names AUDIT\_DATA\_ID, AUDIT\_OBJECT, AUDIT\_OPERATION, AUD\_INS\_DTTM, AUD\_UPD\_USER, AUD\_REC\_STAT.



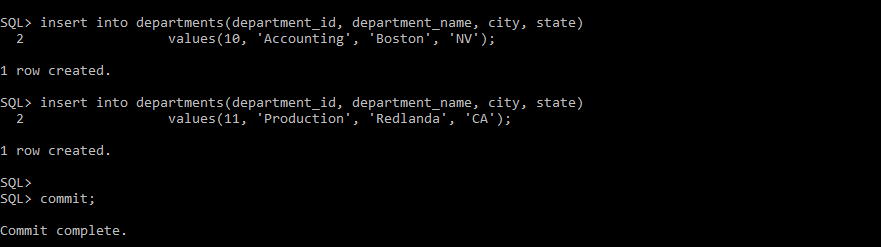
1. Create a sequence object that will be used for the AUDIT\_DATA\_ID column of the APP\_AUDIT\_DATA table. The sequence will generate unique values. This can be useful when you need to create a unique number to act as a primary key.



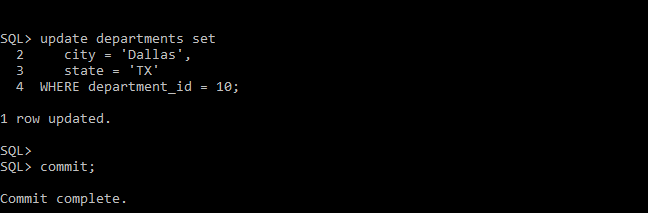
1. Create the trigger on the DEPARTMENTS table that will record the DML operations that occur on it. The name of the trigger has the suffix AIUD, which is an acronym for “After Insert Update Delete” to help distinguish the trigger.



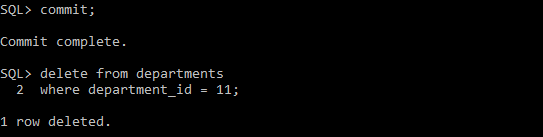
1. Now you can see what is recorded when you insert, modify, or delete rows on the DEPARTMENT table. First try to Insert. When executing these statements, make sure you pause for a minute after each statement and then commit your changes.



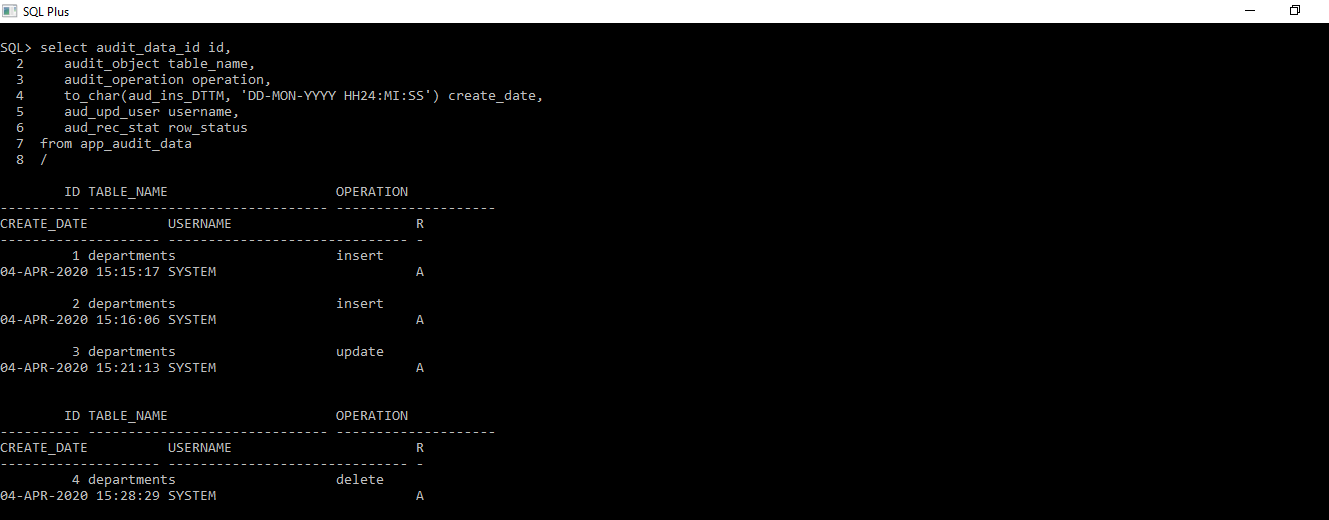
1. Now try an UPDATE on the DEPARTMENT table. The OracleUPDATEstatement is used to update existing records in a table in an Oracle database.



1. Try the Delete statement again on the DEPARTMENTS table. The Oracle DELETE statement is used to delete a single record or multiple records from a table in Oracle. You do not need to list fields in the Oracle DELETE statement since you are deleting the entire row from the table.



1. This last step allows you to view the contents of the auditing table APP\_AUDIT\_DATA. The auditing table does not indicate what column name or column values are being modified. It allows you to view DML activities on the DEPARTMENTS table



**CONCLUSION**

Table auditing means keeping a track of all the DML activities performed on a specific table of the database for example which user Inserted, updated or deleted a row from the table and when. Only DBAs should have maintenance access to the audit trail. If SELECT access is required by any applications this can be granted to any users, or alternatively a specific user may be created for this. A data manipulation language DML statement executed against a table e.g., INSERT, UPDATE, or DELETE. For example, if you define a trigger that fires before an INSERT statement on the customers table, the trigger will fire once before a new row is inserted into the customers table. We use auditing to perform activities such as Enable accountability for actions**.** These include actions taken in a particular schema, table, or row, or affecting specific content. Deter users from inappropriate actions based on that accountability. 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. 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. 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.