**Triggers**

Triggers can be defined on the **table, view, schema, or database** with which the event is associated. Triggers are PL/SQL blocks of code that are stored in the database and automatically execute (fire) in response to specific events that occur on a particular table or view. These events can be data manipulation language (DML) statements like INSERT, UPDATE, or DELETE, or data definition language (DDL) statements like CREATE, ALTER, or DROP Triggers can be written for the following purposes:

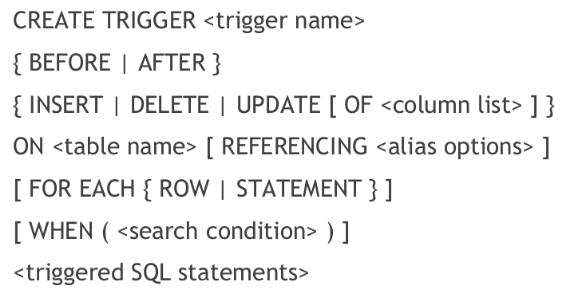
* Gain strong Control over Security
* Enforcing referential integrity
* Event logging and storing information on table access
* **Auditing data modifications** by logging changes to a separate table.
* Synchronous replication of tables
* Preventing invalid transactions

**1. Timing:**

* **BEFORE Triggers:** Fire *before* the triggering DML statement is executed. They are often used to validate or modify data before it's written to the database.
* **AFTER Triggers:** Fire *after* the triggering DML statement has been executed and successfully committed. They are often used for auditing, logging, or performing actions based on the changes that have occurred.

**2. Scope (Level):**

* **Row-Level Triggers:** Execute once for *each row* that is affected by the triggering DML statement. These triggers can access and manipulate the individual row being processed (using the :NEW and :OLD pseudorecords). They are defined with the FOR EACH ROW clause.
* **Statement-Level Triggers (default):** Execute *once* for the entire triggering DML statement, regardless of how many rows are affected. These triggers cannot directly access individual row data using :NEW and :OLD. They are defined without the FOR EACH ROW clause.

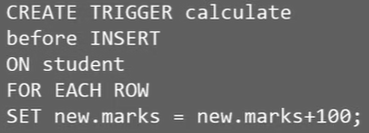


A grey background with white text

AI-generated content may be incorrect. A grey background with white text

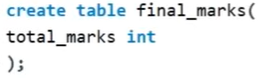
AI-generated content may be incorrect.

-------------------------------------------------------------------------------------------------------------------------------------------

 A blue background with white text

AI-generated content may be incorrect.

-------------------------------------------------------------------------------------------------------------------------------------------



A grey background with white text

AI-generated content may be incorrect. A blue and white rectangle with white text

AI-generated content may be incorrect.



-------------------------------------------------------------------------------------------------------------------------------------------

**new** – for new values

**old** – for deleted or updated

For Oracle - :OLD & :NEW

**The inserted Logical Table**

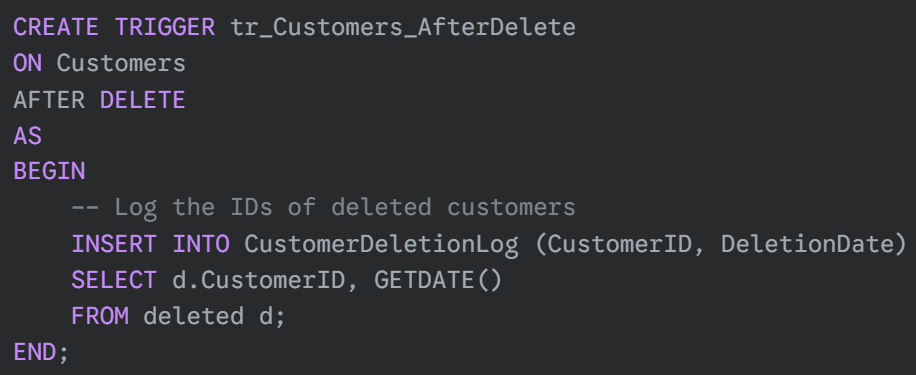
* **Purpose:** The inserted table holds a copy of the row(s) that were ***just inserted*** or the ***new values*** of the row(s) that were ***just updated***.
* **Structure:** The inserted table has the same column structure as the **table** or **view** on which the trigger is defined.

A screen shot of a computer

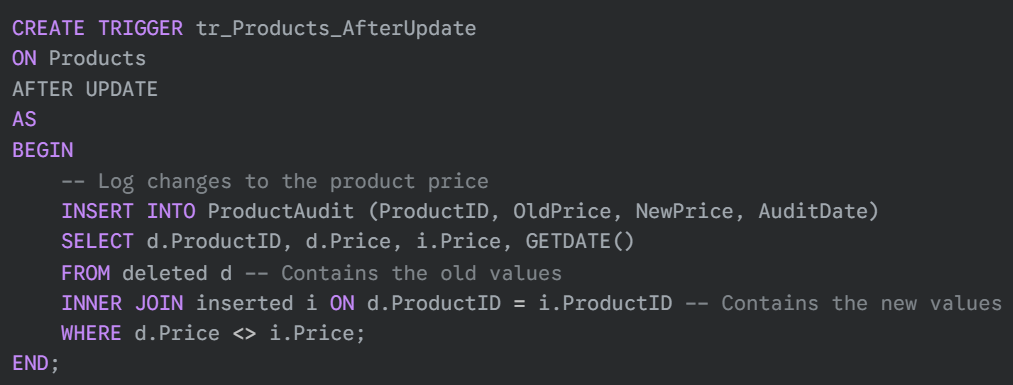
AI-generated content may be incorrect.

**The deleted Logical Table**

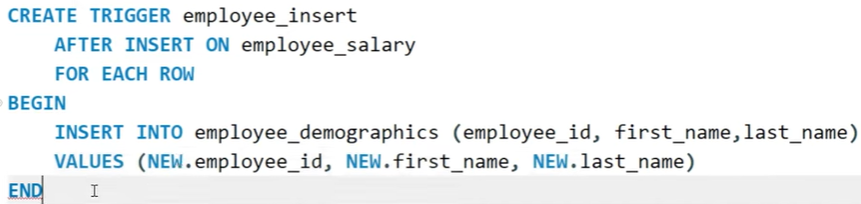
* **Purpose:** The deleted table is used in DELETE and UPDATE triggers. It holds a copy of the row(s) that were ***just deleted*** or the ***original values*** of the row(s) that were ***just updated*** (the state of the row before the UPDATE occurred).
* **Structure:** Like inserted, the deleted table has the same column structure as the **table** or **view** on which the trigger is defined.



**Temporary and Read-Only:** Both inserted and deleted are temporary, read-only tables. They exist only within the scope of the trigger execution. You cannot modify their contents.



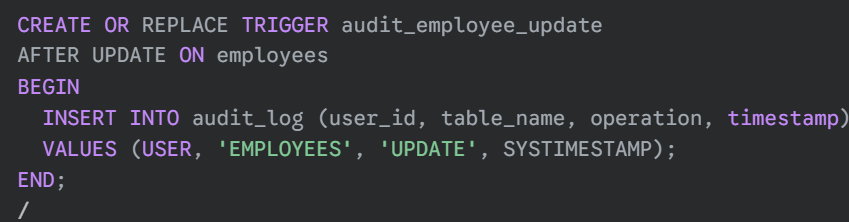
**Examples:**



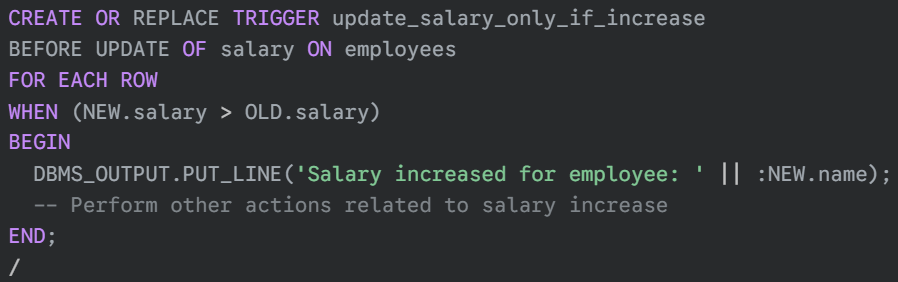
A computer screen with white text

AI-generated content may be incorrect.

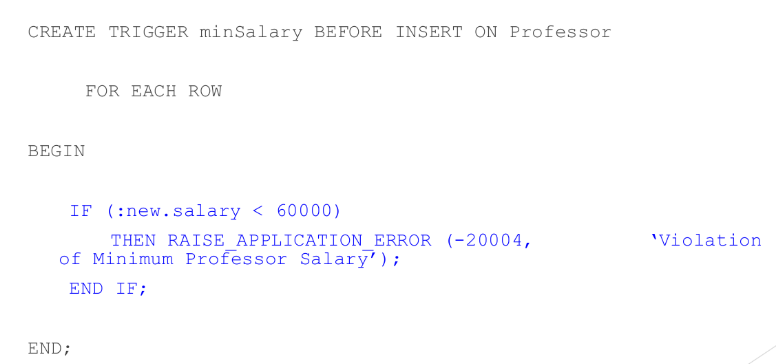
**Statement-Level Trigger:**



It fires only once after the entire UPDATE statement has completed.



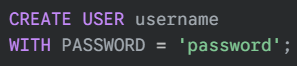
This row-level trigger fires only when the salary column is being updated and the new salary is greater than the old salary.



A close-up of a message

AI-generated content may be incorrect.

**Creating a contained database user with a password**

****

* **Oracle syntax:**

****

**Creating a database user linked to an existing server-level login**

**A black background with white text

AI-generated content may be incorrect.**

**DCL Commands Without Roles (Granting/Revoking Directly to Users):**

**Allow user to connect to database**

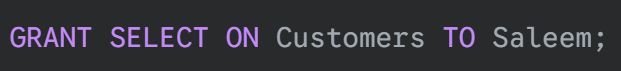
**A close up of a sign

AI-generated content may be incorrect.**

**Revoke database login permission**

****

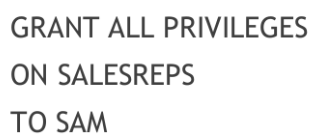
**Granting SELECT permission to a user:**

****

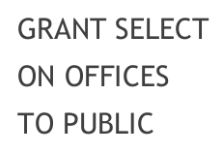
**Granting multiple permissions on a view**

****

**Assign all privileges**

****

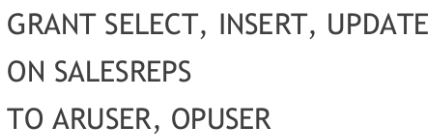
**Give all users SELECT access**

****

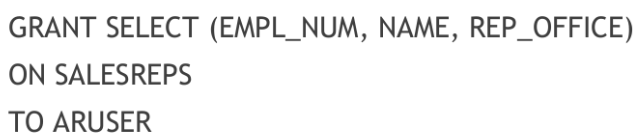
**Grant update privilege on specific columns**

****

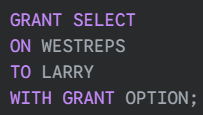
**Grant read and write access as well as update on all columns**

****

**Grant users read-only access on specific columns**

****

**Empower LARRY to also give other users or roles the same SELECT permission on only WESTREPS**

****

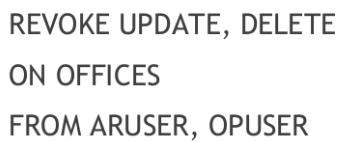
**Revoke LARRY's SELECT permission and also any SELECT permissions LARRY granted**

****

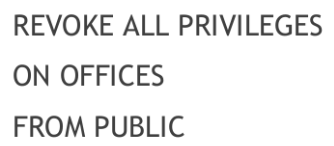
**Revoking SELECT permission from a user**

****

**Revoke update and delete access from multiple users**

****

**Revoke all permission from all users**

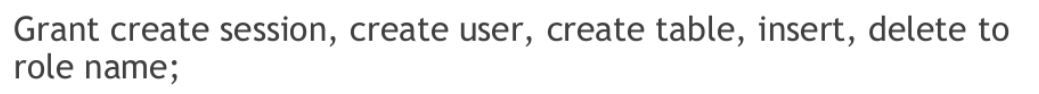
****

**DCL Commands With Roles (Granting/Revoking to Roles and assigning them to Users):**

**Create role which is identified by individual user passwords**

****

**Creating a role with specific permissions**

****

**Give role to users**

****

**Revoke specific privilege from a role**

****

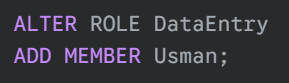
**Grant specific privilege to a role**

****

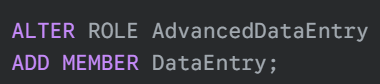
**Drop role**

****

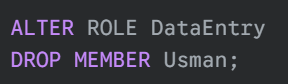
**1. Adding a database user to a role:**



**2. Adding another database role as a member of a role:**



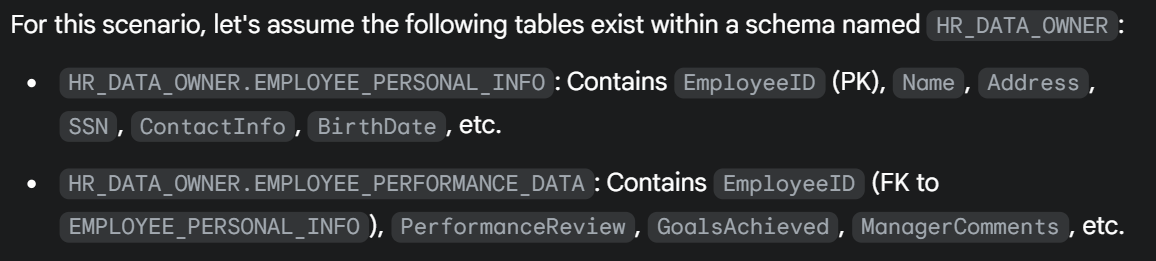
**3. Removing a database user from a role:**



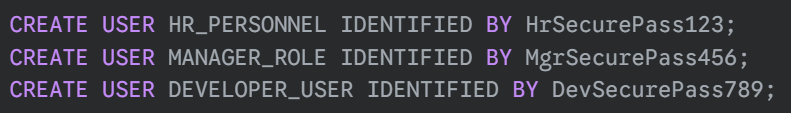
**-- Questions --**

You are the database administrator (DBA) for a company that stores sensitive employee information in a relational database. As the DBA, you are responsible for managing user permissions and access control to ensure data security. The company has different roles for employees, such as HR personnel, managers, and developers. HR personnel should only have access to employee personal information, while managers should be able to access both personal and work-related performance data. Developers should not have access to any sensitive information. Constraints: The HR department needs the ability to view and update employee personal information, but they should not be able to delete any records. The company's managers need to view all employee records, including sensitive performance related data. However, they should not be allowed to modify the data. The company's HR need to view the employee records, including only personal but not viewing sensitive performance related data. However, they should not be allowed to modify the data.

**We'll use Oracle SQL syntax for the DCL (Data Control Language) commands.**



**1. What DCL command would you use to create a user with its valid password?**



A black background with white text

AI-generated content may be incorrect.

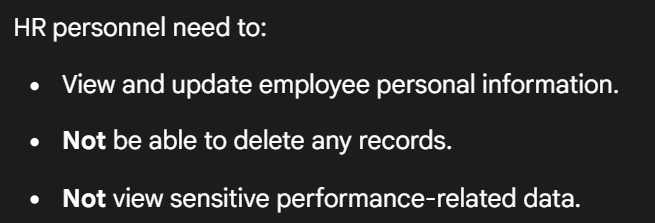
**2. What are necessary permissions for the newly created user? Grant these permissions.**

The necessary permissions are **CREATE SESSION** for all users to allow them to connect to the database. For **HR\_PERSONNEL** and **MANAGER\_ROLE** who will be interacting with tables.

A black background with white text

AI-generated content may be incorrect.

**3. What DCL command would you use to grant the HR personnel the required permissions on the relevant table?**





**4. What DCL command would you use to provide the managers with read-only access to the employee performance data?**

A black background with white text

AI-generated content may be incorrect.

A black screen with white text

AI-generated content may be incorrect.

**5. Which DCL command would you use to revoke all permissions granted to this user?**

