1. Triggers are blocks of code that are executed from the database declaration automatically. Triggers are normally executed for DML statements such as Edit, Attach or Erase and is shot immediately on Attack, UPDATE or Erase statements allowing the database code execution.

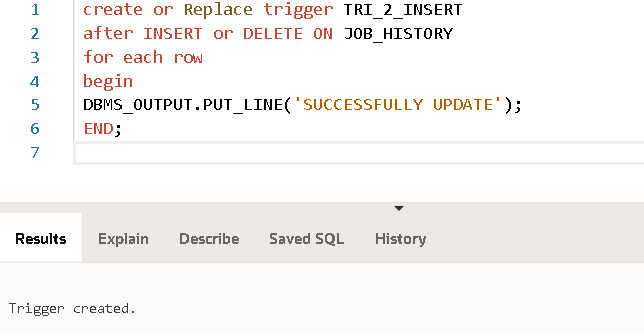
|  |  |
| --- | --- |
| Statement Triggers | Row Level Triggers |
| Only Executed Once | Executed once for each and every row in the transaction |
| “For Each Row” in the build trigger command is omitted | “For Each Row” in the build trigger command is present |
| Used on the table for extra protection on the transactions | Used for Auditing Purpose specifically |
| For Example: On updating 1000 rows in a table, a statement-level trigger will only be executed once on the table | For Example: Will run as many time as the number of rows |

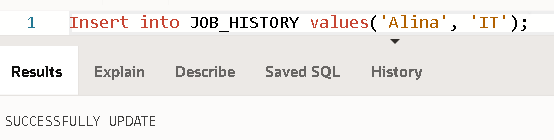
1. An event such as a user or database event or even an SQL Comment that triggers something is called a triggering event. This includes INSERT, DELETE and UPDATE and such events.

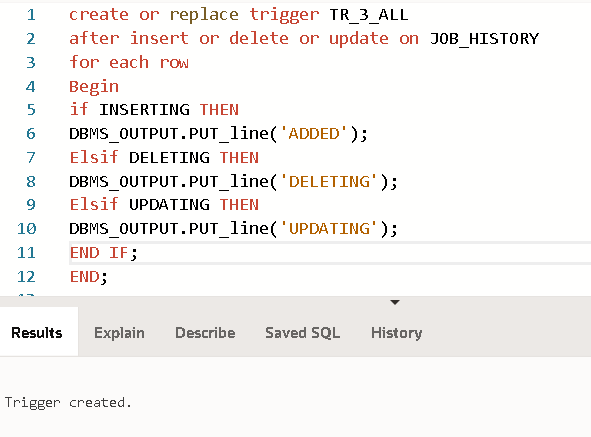
For Example:

1. Applying Trigger on the JOB\_HISTORY table gives us the following message for insert and

delete command.







1. A) IF the employee salary is increased more than 10$, then the rank field is incremented by 1.

B) if the Employee salary is increased by more than 10$, the Rank field is not empty and its value is changed, otherwise the update is rejected.

C) Bonus Attribute in Emploiyee always 4% of salary attribute.

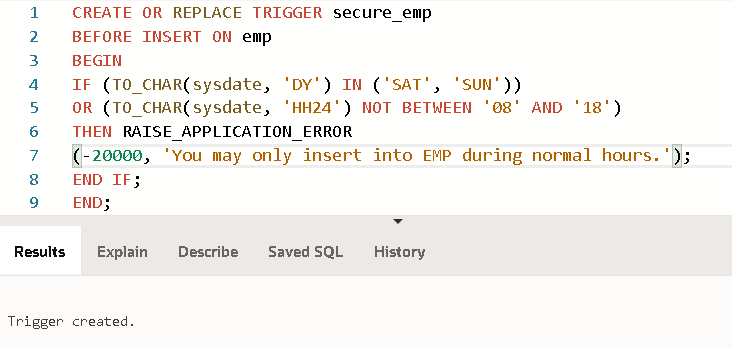
5) A) To store the employee count having salary more than 100,000.

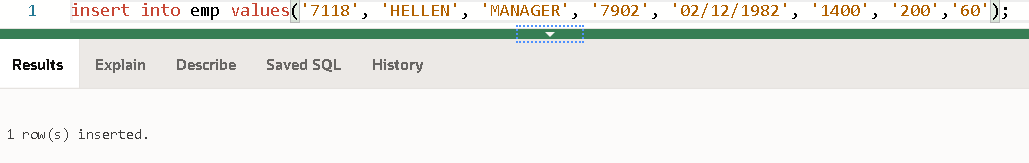
B) Adds a new row to JOB\_HISTORY table when a new employee row is inserted.

C) Shows the action of the trigger for each affected row in JOB\_HISTORY.

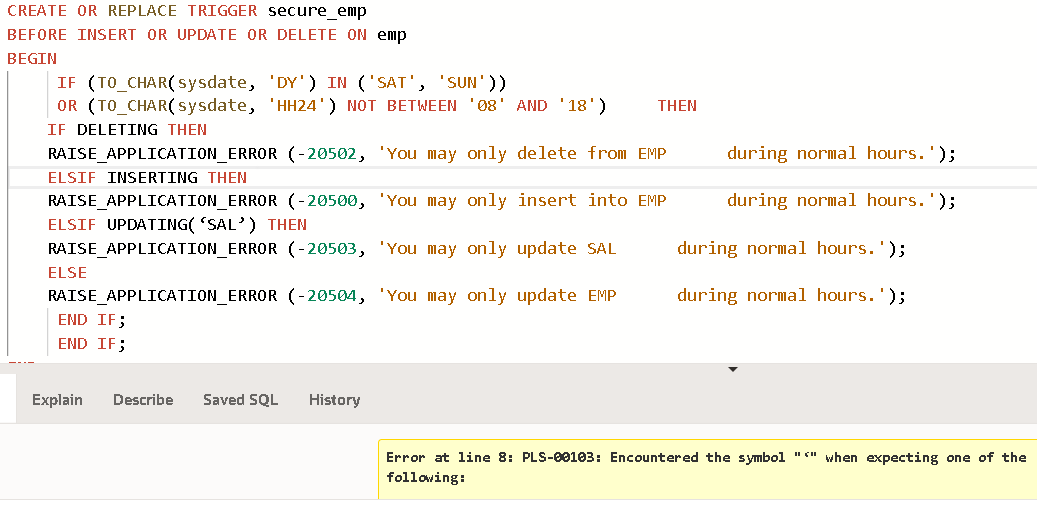
**Class Tasks:**

1. **Create before insert trigger**

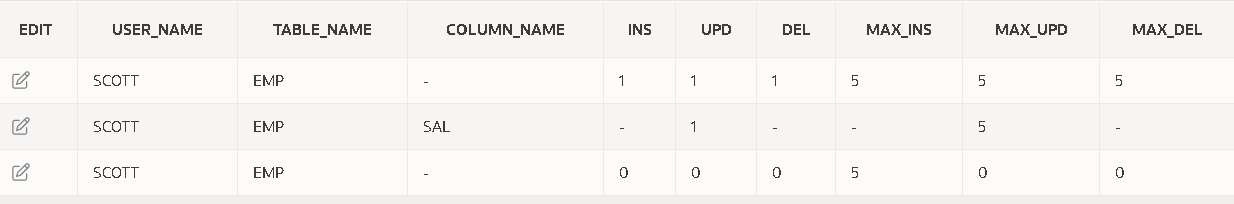




1. **Using conditional predicates**



1. **After update trigger**

**Audit Table**

