Nama : Afina Putri Dayanti
NIM : 825200049
Jurusan : Sistem Informasi
Mata Kuliah : Database Design and Management (Praktikum)

Vocabulary

Identify the vocabulary word for each definition below:

Row Trigger	fires once for each row affected by the triggering event.
DML Triggers	A trigger which is automatically fired (executed) whenever a SQL DML statement (INSERT, UPDATE or DELETE) is executed.
Statement Trigger	is fired once on behalf of the triggering event, even if no rows are affected at all.

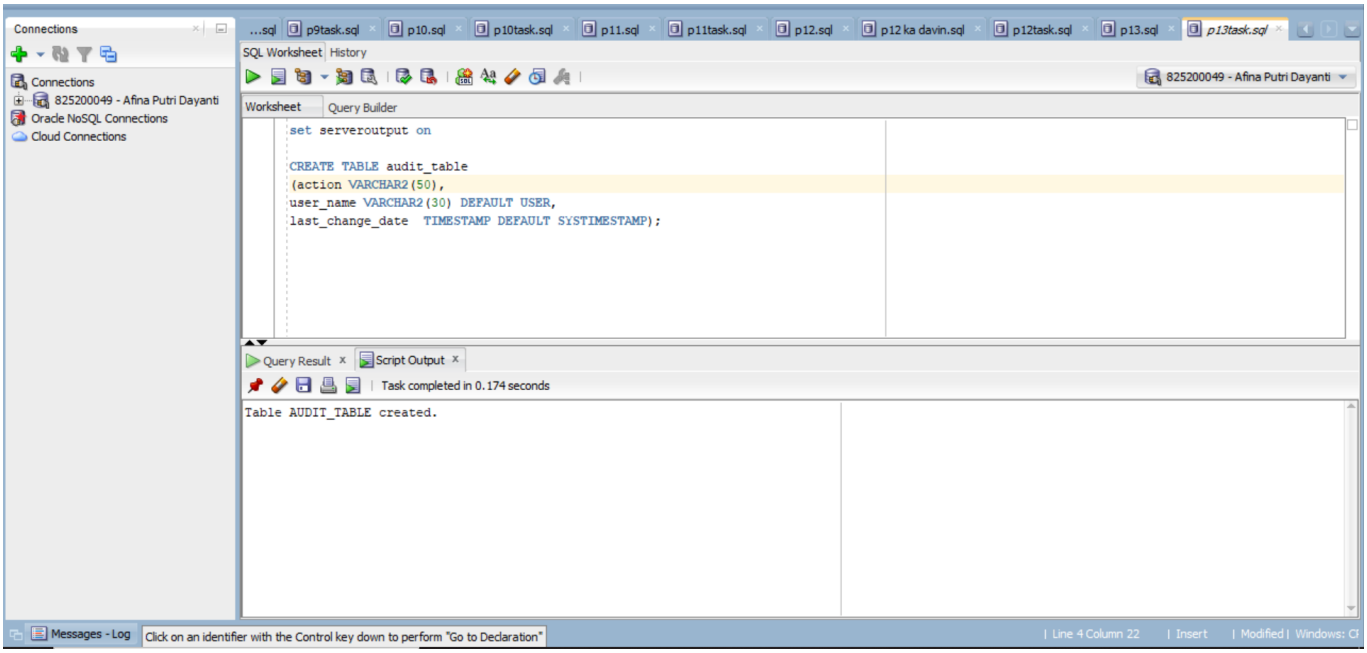
Try It / Solve It

1. When creating a DML statement trigger on a table, what are the components that you must define?
   Answer :
   - Timing : When the trigger fires in relation to the triggering event. – Values are BEFORE, AFTER, or INSTEAD OF
   - Event : Which DML operation causes the trigger to fire. Values are INSERT, UPDATE [OF column], and DELETE
   - Object_name : The table or view associated with the trigger
   - Trigger_body : The action(s) performed by the trigger are defined in an anonymous block

2. A business rule states that each time one or more employees are added to the EMPLOYEES table, an audit record must also be created. This rule could be enforced using application code, but we have decided to enforce it using a DML statement trigger.
   a. Create the AUDIT_TABLE by executing the following SQL statement:

```
CREATE TABLE audit_table
   (action              VARCHAR2(50),
    user_name           VARCHAR2(30) DEFAULT USER,
    last_change_date    TIMESTAMP DEFAULT SYSTIMESTAMP);
```

   Answer :



b. Create a statement-level trigger that inserts a row into the AUDIT_TABLE immediately after one or more rows are added to the EMPLOYEES table. The AUDIT_TABLE row should contain value “Inserting” in the action column. The other two columns should have their default values. Save your trigger code for later.

Answer :

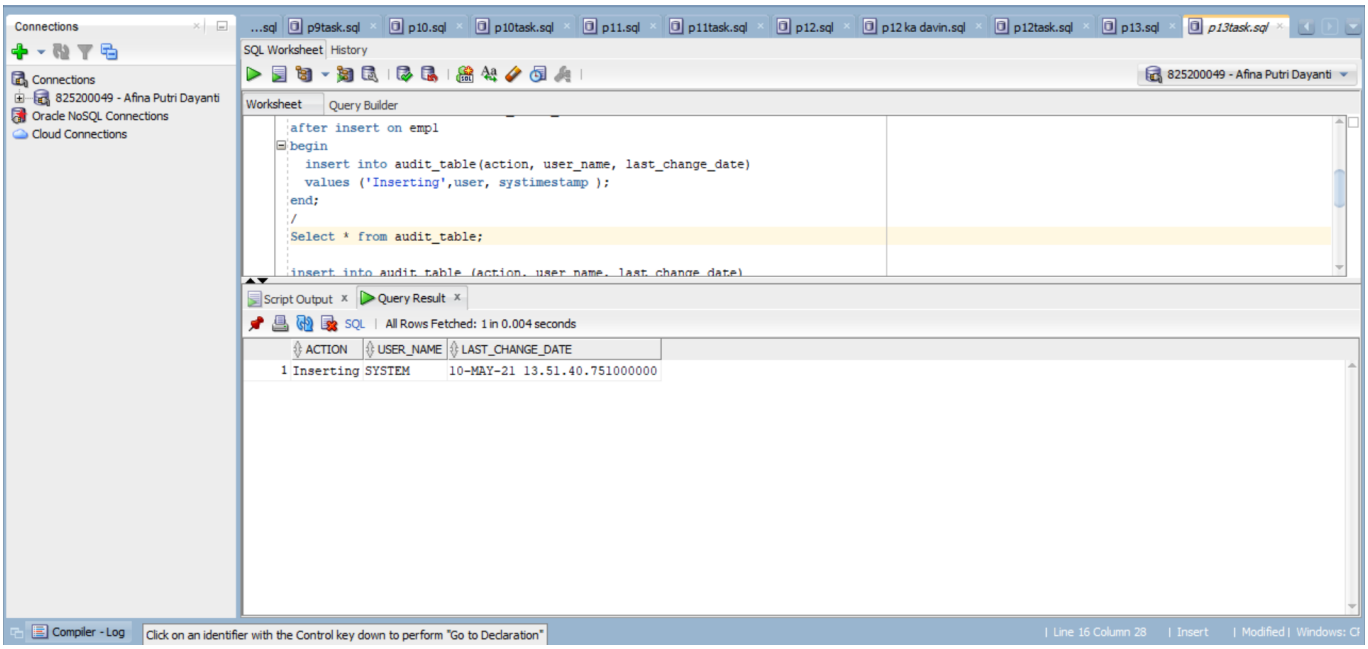
```
create or replace trigger log_audit_table
after insert on emp1
begin
 insert into audit_table(action)
 values ("Inserting");
end;
```

c. Test your trigger by inserting a row into EMPLOYEES, then querying the AUDIT_TABLE to see that it contains a row.

Answer :

```
insert into emp1(
employee_id,
last_name,
first_name,
email,
hire_date,
job_id,
salary,
department_id
) values(
300,
'Smith',
```

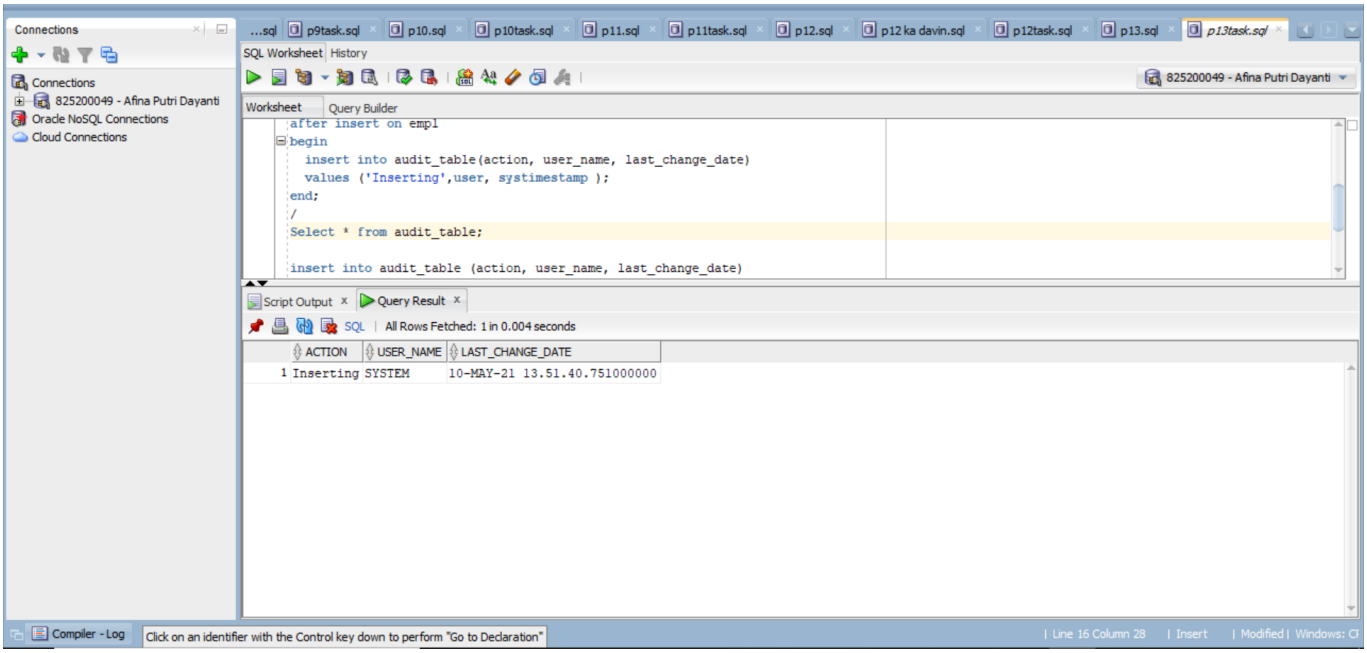
```
'Rob',
'RSMITH',
sysdate,
'IT_PROG',
4500,
60);
/
select * from audit_table;
```



d. Make sure the trigger does not fire with a DELETE by deleting the employee you just entered. Recheck the AUDIT_TABLE to make sure that there is not another new row.

Answer :

```
delete from emp1 where employee_id = 301;
/
select * from audit_table;
```



3. True or false? A row trigger fires at least once even if no rows are affected. What is the difference between a statement trigger and a row trigger?
   Answer : False, because
   - a row trigger fires every time when a row is affected
   - a statement trigger fires only once when a statement modifies multiple rows
   - a row trigger fires for every row that is affected.

4. Imagine that the following DML triggers have been defined on the EMPLOYEES table:
   - A Before Insert statement trigger
   - A Before Update statement trigger
   - An After Delete statement trigger

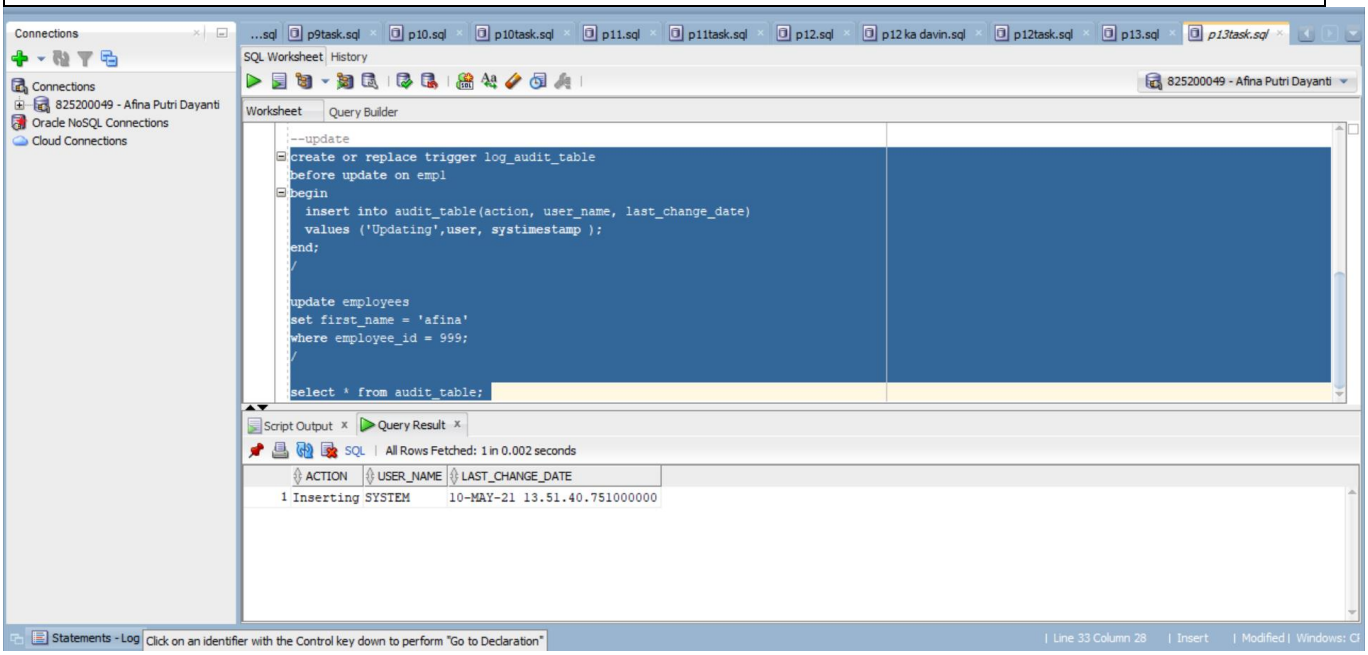
   An UPDATE statement updates three employee rows. How many times will each trigger fire?
   Answer : The update statement will fire once, before the 3 row updates

5. Modify your AUDIT_TABLE trigger from question 2B so that it inserts a row into the audit table immediately before one or more employee salaries are updated. The AUDIT_TABLE rowshould contain value “Updating” in the action column.
   Test your modified trigger by updating the salary of a non-existent employee (employee_id = 999), then querying the AUDIT_TABLE to see that it contains a new row.
   Answer :

```
create or replace trigger log_audit_table
before update on emp1
begin
 insert into audit_table(action, user_name, last_change_date)
 values ('Updating',user, systimestamp );
end;
/
```

```
update emp1
set first_name = 'afina'
where employee_id = 999;
/

select * from audit_table;
```



a. Modify your trigger so that it prevents employees' salaries being updated outside working hours. The trigger should allow UPDATEs at other times (and still insert a row into the AUDIT_TABLE), but should raise an application error if an update is attempted before 8:00 am or after 6:00 pm on any day. (HINT: use HH24:MI to extract the time from SYSDATE).

```
create or replace trigger log_audit_table
before update on emp1
begin
 insert into audit_table(action, user_name, last_change_date)
 values ('Updating', user, systimestamp);
 if (to_char(sysdate, 'HH24') < '12:00' or to_char(sysdate, 'HH24') > '18:00') then
  raise_application_error(-20500,'you may insert into employees table only between 8 am and 6 pm');
 end if;
end;
```

b. You want to test your modified trigger. However, you need to make sure that right now the database time is outside working hours. Remember that the database could be anywhere in the world and therefore the database may not be in your time zone! Find the current database time by executing: SELECT TO_CHAR(SYSDATE,'HH24:MI') FROM dual;

If needed, modify your trigger so that it will raise the application error if you try to update a salary within the next hour. For example, if the database time is 10:30, modify the trigger code to include: …BETWEEN '10:30' AND '11:30'…

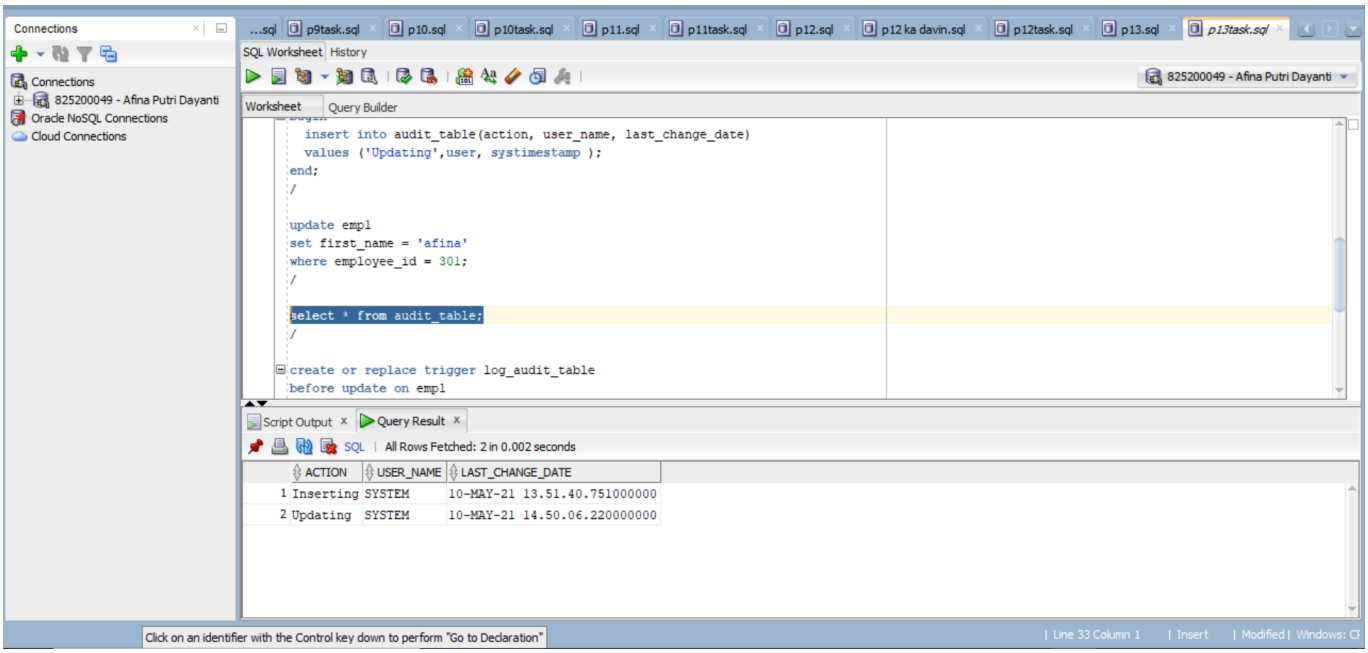
Test your modified trigger by trying to update the salary of employee_id 100 to a new value of 25000. You should see the ORA-20204 error message.

```
UPDATE    employees    SET
    salary = 25000
   WHERE employee_id = 100;
```

Answer :

```
select * from audit_table;
```

14.52



I changed the time to 4 am to 6 pm. (ERROR)