**Assignment 1:**

The triggers were created in the below sequence .

CREATE OR REPLACE TRIGGER t1 BEFORE

INSERT ON EMPLOYEES

FOR EACH ROW

BEGIN

dbms\_output.put\_line('trigger t1 executed');

END;

CREATE OR REPLACE TRIGGER t2 BEFORE

INSERT ON EMPLOYEES

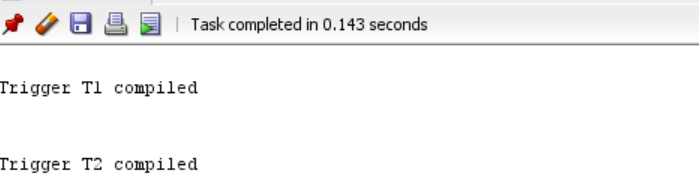
FOR EACH ROW

BEGIN

dbms\_output.put\_line('Trigger T2 executed');

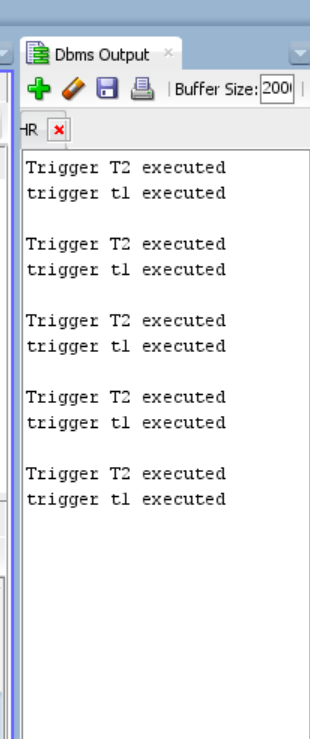
END;

Compiled in the following sequence.



Mutiple insert operation were performed on the table EMPLOYEES.

Below are the results:



Thus,it appears as if the triggers were executed in the descending order of their creation/compilation timestamp.

BUT THIS ORDER CAN NEVER BE GUARANTED.

Theoretically, TRIGGER FIRING SEQUENCE IS

--BEFORE STATEMENT

--BEFORE ROW

--AFTER ROW

--AFTER STATEMENT

To ensure that trigger T2 always gets executed only after trigger T1 execution,

**1)We can modify the firing time of trigger T2 to ‘AFTER’.**

CREATE OR REPLACE TRIGGER t2 AFTER

INSERT ON EMP

FOR EACH ROW

BEGIN

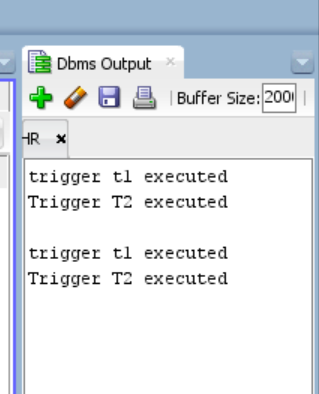
dbms\_output.put\_line('Trigger T2 executed');

END;

Trigger T2 was complied.

Post this, mutiple insert operation were performed on the table EMPLOYEES.

Below are the results,which is per expectation:



**2) We can use the keyword ‘FOLLOWS’**

CREATE OR REPLACE TRIGGER t2 BEFORE

INSERT ON EMP

FOR EACH ROW

FOLLOWS t1

BEGIN

dbms\_output.put\_line('Trigger T2 executed');

END;

Trigger T2 was complied.

Post this, mutiple insert operation were performed on the table EMPLOYEES.

Below are the results,which is per expectation:

