Trigger 1:  This trigger will display the salary difference between the old values and new values

CREATE OR REPLACE TRIGGER display\_salary\_changes

BEFORE DELETE OR INSERT OR UPDATE ON employee

FOR EACH ROW

WHEN (NEW.EMP\_ID > 0)

DECLARE

sal\_diff number;

BEGIN

sal\_diff := :NEW.salary - :OLD.salary;

dbms\_output.put\_line('Old salary: ' || :OLD.salary);

dbms\_output.put\_line('New salary: ' || :NEW.salary);

dbms\_output.put\_line('Salary difference: ' || sal\_diff);

END;

/

After trigger gets created execute below commands:

Statement 1: UPDATE EMPLOYEE

SET salary = salary+500

WHERE EMP\_ID = e1;

Statement 2: UPDATE employees

SET salary = salary \* 1.1

WHERE EMP\_id = e4;

Procedure to auto update salaries:

DECLARE

total\_rows number(10);

BEGIN

UPDATE EMPLOYEE

SET salary = salary + 5000;

IF sql%notfound THEN

dbms\_output.put\_line('no employees updated');

ELSIF sql%found THEN

total\_rows := sql%rowcount;

dbms\_output.put\_line( total\_rows || ' employees updated ');

END IF;

END;

/

Trigger 2:

CREATE OR REPLACE TRIGGER tax\_deductions

BEFORE DELETE OR INSERT OR UPDATE ON employee

FOR EACH ROW

WHEN (NEW.EMP\_ID > 0)

DECLARE

tax number;

BEGIN

Tax := 0.14;

salary := salary – salary\*tax;

dbms\_output.put\_line('Old salary: ' || :OLD.salary);

dbms\_output.put\_line('New salary: ' || :NEW.salary);

dbms\_output.put\_line('tax deduction total: ' || tax);

END;

/