**CSE2007 DBMS LAB**

**SLOT: L39+L40**

**NAME - AMAN SAHU**

**REG. NO – 22BCE7224**

**EXERCISE NO.-11**

1.Develop a trigger to convert a employee name into upper case when a user is trying to insert or update an employee information.Also validate the input using exceptions if the user is entered other than character data.

CREATE OR REPLACE TRIGGER emp\_uppercase

BEFORE INSERT OR UPDATE ON emp

FOR EACH ROW

DECLARE

v\_name VARCHAR2(10);

BEGIN

v\_name := :NEW.ename;

-- Check if input is character data

IF NOT REGEXP\_LIKE(v\_name, '^[a-zA-Z ]+$') THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Employee name should only contain characters and spaces');

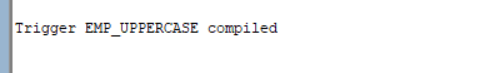
END IF;

-- Convert name to uppercase

:NEW.ename := UPPER(v\_name);

END;

/



2.Develop a trigger to store the old employee information into an OLD\_EMP table when an employee is resigned from the company.

CREATE TABLE old\_emp (

empno NUMBER(4),

ename VARCHAR2(10),

job VARCHAR2(9),

mgr NUMBER(4),

hiredate DATE,

sal NUMBER(7,2),

comm NUMBER(7,2),

deptno NUMBER(2)

);

CREATE TABLE OLD AS SELECT \* FROM emp WHERE 1=2;

CREATE OR REPLACE TRIGGER emp\_resignation\_trigger

AFTER UPDATE OF JOB ON emp

FOR EACH ROW

DECLARE

v\_old\_sal emp.sal%TYPE;

v\_old\_comm emp.comm%TYPE;

BEGIN

IF :OLD.job = 'RESIGNED' AND :NEW.job != 'RESIGNED' THEN

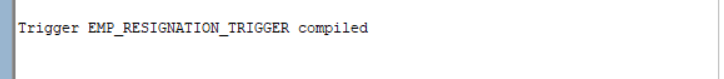
v\_old\_sal := :OLD.sal;

v\_old\_comm := :OLD.comm;

INSERT INTO old\_emp (empno, ename, job, mgr, hiredate, sal, comm, deptno)

VALUES (:OLD.empno, :OLD.ename, 'RESIGNED', :OLD.mgr, :OLD.hiredate, v\_old\_sal, v\_old\_comm, :OLD.deptno);

END IF;

END;

3.Develop a trigger to maintain backup of employees “EMP\_BACKUP” table when anew employee is joined to the company.

CREATE TABLE emp\_backup (

EMPNO number(4) not null,

ENAME varchar(10),

JOB varchar2(9),

MGR number(4),

HIREDATE date,

SAL number(7,2),

COMM number(7,2),

DEPTNO number(2)

);

CREATE OR REPLACE TRIGGER emp\_backup

AFTER INSERT ON emp

FOR EACH ROW

DECLARE

v\_ename emp.ename%TYPE;

BEGIN

v\_ename := :NEW.ename;

IF NOT REGEXP\_LIKE(v\_ename, '^[a-zA-Z ]+$') THEN

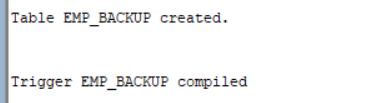
RAISE\_APPLICATION\_ERROR(-20001, 'Employee name should only contain characters and spaces');

END IF;

INSERT INTO emp\_backup (empno, ename, job, mgr, hiredate, sal, comm, deptno)

VALUES (:NEW.empno, :NEW.ename, :NEW.job, :NEW.mgr, :NEW.hiredate, :NEW.sal, :NEW.comm, :NEW.deptno);

END;



4.Develop a trigger to raise an error when an employee table is trying to update or insert or delete on either Saturday or Sunday.

CREATE OR REPLACE TRIGGER emp\_day\_check

BEFORE INSERT OR UPDATE OR DELETE ON emp

FOR EACH ROW

DECLARE

v\_day VARCHAR2(9);

BEGIN

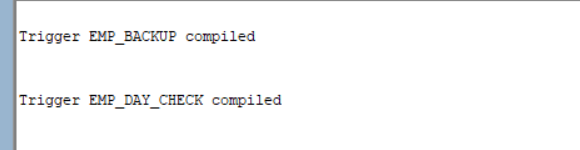
SELECT TO\_CHAR(SYSDATE, 'DAY') INTO v\_day FROM dual;

IF v\_day IN ('SATURDAY', 'SUNDAY') THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Cannot update, insert, or delete on Saturday or Sunday');

END IF;

END;



5.Develop a trigger to track DDL operations which are performed on the database and store them in a table DB\_AUDIT. Track the operation, database object name, user name, date and time.

CREATE TABLE log (

operation VARCHAR2(25),

owner VARCHAR2(25),

name VARCHAR2(25),

extra VARCHAR2(4000)

);

CREATE OR REPLACE TRIGGER ddl\_trigger

BEFORE CREATE OR ALTER OR DROP ON SCHEMA

DECLARE

l\_sysevent VARCHAR2(25);

BEGIN

SELECT ora\_sysevent INTO l\_sysevent FROM dual;

IF ( l\_sysevent IN ('DROP','CREATE') ) THEN

INSERT INTO log

SELECT ora\_sysevent, ora\_dict\_obj\_owner,

ora\_dict\_obj\_name, NULL

FROM dual;

ELSIF ( l\_sysevent = 'ALTER' ) THEN

INSERT INTO log

SELECT ora\_sysevent, ora\_dict\_obj\_owner,

ora\_dict\_obj\_name, sql\_text

FROM v$open\_cursor

WHERE upper(sql\_text) LIKE 'ALTER%';

END IF;

END;

/