Lab Assignment 6

CIS430/CIS530
Dr. Sunnie S. Chung

Trigger and Stored Procedure:

1. Write Triggers that implement Referential Integrity for FK Dno of Employee that is named as Constraint EMPDEPTFK in Table Employee as specified in Figure 4.2 in the textbook in the following DDL statements below for Company database that you created for Lab2. (You may keep the all the new tuples inserted in Lab3 and Lab4)

> **1) Before creating any trigger for this lab, Alter Table to Drop all the PK, FK, Unique Constraints, Cascade, Check options from the Tables Employee and Department for this lab to avoid any possible conflict with a system trigger or any table mutating problem.**
>
> **2) Write(Create) triggers to implement Constraint EMPDEPTFK in Table Employee based on the following rules as defined in DDL for Employee as in Figure 4.2:**
> **FK Dno of Employee On Delete SET DEFAULT (= 1) and**
> **FK Dno of Employee On Update CASCADE of Dnumber PK of Department**
>
> **3) Write (Create) Stored Procedure *SP_Audit_Dept* that inserts all the history of the data of changes by the trigger you created in 1) above into a table *Audit_Dept_Table*. See for the more specific instructions that are given in 2 below.**
>
> **4) Call the Stored procedure *SP_Audit_Dept* at the end of your Trigger to record all the history of the changes by the trigger.**

2. Write Stored Procedure *SP_Audit_Dept* as follow. Whenever data for a department in DEPARTMENT table are updated, deleted , both the previous and new values are recorded in an audit table *Audit_Dept_Table* to allow tracing the history of changes.

> **1) On Update of the trigger, Insert the new record into a table named *Audit_ Dept_Table* as follow:**

(date_of_change, old_Dname, new_Dname, old_Dnumber, new_Dnumber, old_Mgrssn, new_Mgrssn)

2) On Delete of the trigger, Insert the changes into *Audit_ Dept_Table* table as well. Since there is no new record for delete, so insert NULL for the new record columns new_Dname, new_Dnumber, new_Mgrssn of *Audit_ Dept_Table*.

```
CREATE TABLE EMPLOYEE
    ( . . . ,
      Dno        INT            NOT NULL        DEFAULT 1,
    CONSTRAINT EMPPK
      PRIMARY KEY (Ssn),
    CONSTRAINT EMPSUPERFK
      FOREIGN KEY (Super_ssn) REFERENCES EMPLOYEE(Ssn)
                  ON DELETE SET NULL          ON UPDATE CASCADE,
    CONSTRAINT EMPDEPTFK
      FOREIGN KEY(Dno) REFERENCES DEPARTMENT(Dnumber)
                  ON DELETE SET DEFAULT       ON UPDATE CASCADE);
CREATE TABLE DEPARTMENT
    ( . . . ,
      Mgr_ssn    CHAR(9)        NOT NULL        DEFAULT '888665555',
      . . . ,
    CONSTRAINT DEPTPK
      PRIMARY KEY(Dnumber),
    CONSTRAINT DEPTSK
      UNIQUE (Dname),
    CONSTRAINT DEPTMGRFK
      FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn)
                  ON DELETE SET DEFAULT   ON UPDATE CASCADE);
CREATE TABLE DEPT_LOCATIONS
    ( . . . ,
    PRIMARY KEY (Dnumber, Dlocation),
    FOREIGN KEY (Dnumber) REFERENCES DEPARTMENT(Dnumber)
                  ON DELETE CASCADE           ON UPDATE CASCADE);
```

Fig
Exa
hov
valu
inte
acti
in S

3. Let the Events happen to get each trigger you created fired by executing the following DML on Department table One at a time:

UPDATE	Department	Dnumber	4	to	99
DELETE	Department	Dnumber	5		

4. Show each trigger you create was fired correctly with Select statements on affected columns and tables Before and After the Trigger is fired .

5. Do not make all the DML events above in task 3 at once to fire all the triggers. Make one event per a Trigger creation per one transaction to test to avoid any possible table mutation problem. This means that your script or a query execution sequence should look like in the following order:

Login //if needed -- optional

1. **Create or replace Trigger_One on Update**
2. **Select statement to show the original table before the Trigger_One fired**
3. **One Update statement to trigger(fire) the Trigger_One**
4. **Select statement to show the tables affected by Trigger_One**
5. **Drop Trigger_One //optional**

Logout //optional

Login //optional

1. **Create or replace Trigger_Two on Delete**
2. **Select statement to show the original table before the Trigger_Two fired**
3. **One Delete statement to trigger the Trigger_Two**
4. **Select statement to show the table affected by Trigger_Two**
5. **Drop Trigger_Two //optional**

Logout //optional

5. Submit Screenshots of each trigger creation, triggering event, and Select on affected columns and tables to show that your triggers all get correctly fired

as well as the data inserted in *Audit_ Dept_Table* by the Stored Procedure called at the end of each trigger.

For your Lab5 Output,
For each Select statement to show table changes by each Trigger action,
Explain specifically which column values in which Table are changed by which trigger action why... so on
If no explain on any data change, credits will be taken off

COMPANY DATABASE

EMPLOYEE

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
John	B	Smith	123456789	09-Jan-55	731 Fondren, Houston, TX	M	30000	987654321	5
Franklin	T	Wong	333445555	08-Dec-45	638 Voss, Houston, TX	M	40000	888665555	5
Joyce	A	English	453453453	31-Jul-62	5631 Rice, Houston, TX	F	25000	333445555	5
Ramesh	K	Narayan	666884444	15-Sep-52	975 Fire Oak, Humble, TX	M	38000	333445555	5
James	E	Borg	888665555	10-Nov-27	450 Stone, Houston, TX	M	55000		1
Jennifer	S	Wallace	987654321	20-Jun-31	291 Berry, Bellaire, TX	F	43000	888665555	4
Ahmad	V	Jabbar	987987987	29-Mar-59	980 Dallas, Houston, TX	M	25000	987654321	4
Alicia	J	Zelaya	999887777	19-Jul-58	3321 Castle, SPring, TX	F	25000	987654321	4

DEPARTMENT

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE
Headquarters	1	888665555	19-Jun-71
Administration	4	987654321	01-Jan-85
Research	5	333445555	22-May-78
Automation	7	123456789	06-Oct-05

DEPENDENT

ESSN	DEPENDENT_NAME	SEX	BDATE	RELATIONSHIP
123456789	Alice	F	31-Dec-78	Daughter
123456789	Elizabeth	F	05-May-57	Spouse
123456789	Michael	M	01-Jan-78	Son
333445555	Alice	F	05-Apr-76	Daughter
333445555	Joy	F	03-May-48	Spouse
333445555	Theodore	M	25-Oct-73	Son
987654321	Abner	M	29-Feb-32	Spouse

DEPT_LOCATIONS

DNUMBER	DLOCATION
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

WORKS_ON

ESSN	PNO	Hours
123456789	1	32.5
123456789	2	7.5
333445555	2	10
333445555	3	10
333445555	10	10
333445555	20	10
453453453	1	20
453453453	2	20
666884444	3	40
888665555	20	
987654321	20	15
987654321	30	20
987987987	10	35
987987987	30	5
999887777	10	10
999887777	30	30