ELDER CARE

Dedipya Kumar Jain

Introduction

Goal

The objective of this project is to create a relational database for keeping records of an Elder Care Agency. This relational database provides flexibility to maintain various complex data relationships for Elder Care System. The project is implemented using various features of DBMS like SQL, PL/SQL, Triggers and Stored Procedures to store complex relationships among data.

Requirements

Elder care agency has different branches in various parts of United States. So we need to maintain data records for all the branches.

1. We store details of all branches like branch id, contact, branch name and branch address.

2. Agency also needs to store details of its employees who are working in various branches. Agency stores Employee-SSN, Salary, Contact-number, Name, Address and Branch where employee is working.

3. Agency stores details of Elders who are registered with them. It includes name, age, Gender, Contact, Employee who has been assigned to this particular elder. It also stores the details of branch where elder is assigned.

4. .Agency stores the health record of each elder in case of medication. It keeps record for every elder who is registered. Health details include allergy, Dietary Requirements, Blood group and their Health Care policies. It also includes the contact of hospital or any clinic that is treating this elder. So that in case of emergency agency can give necessary care for Elders.

5. Agency keeps details of care they are providing to their customers. For example Cancer cares, Surgery care, Day care. It keeps track of each service, cost

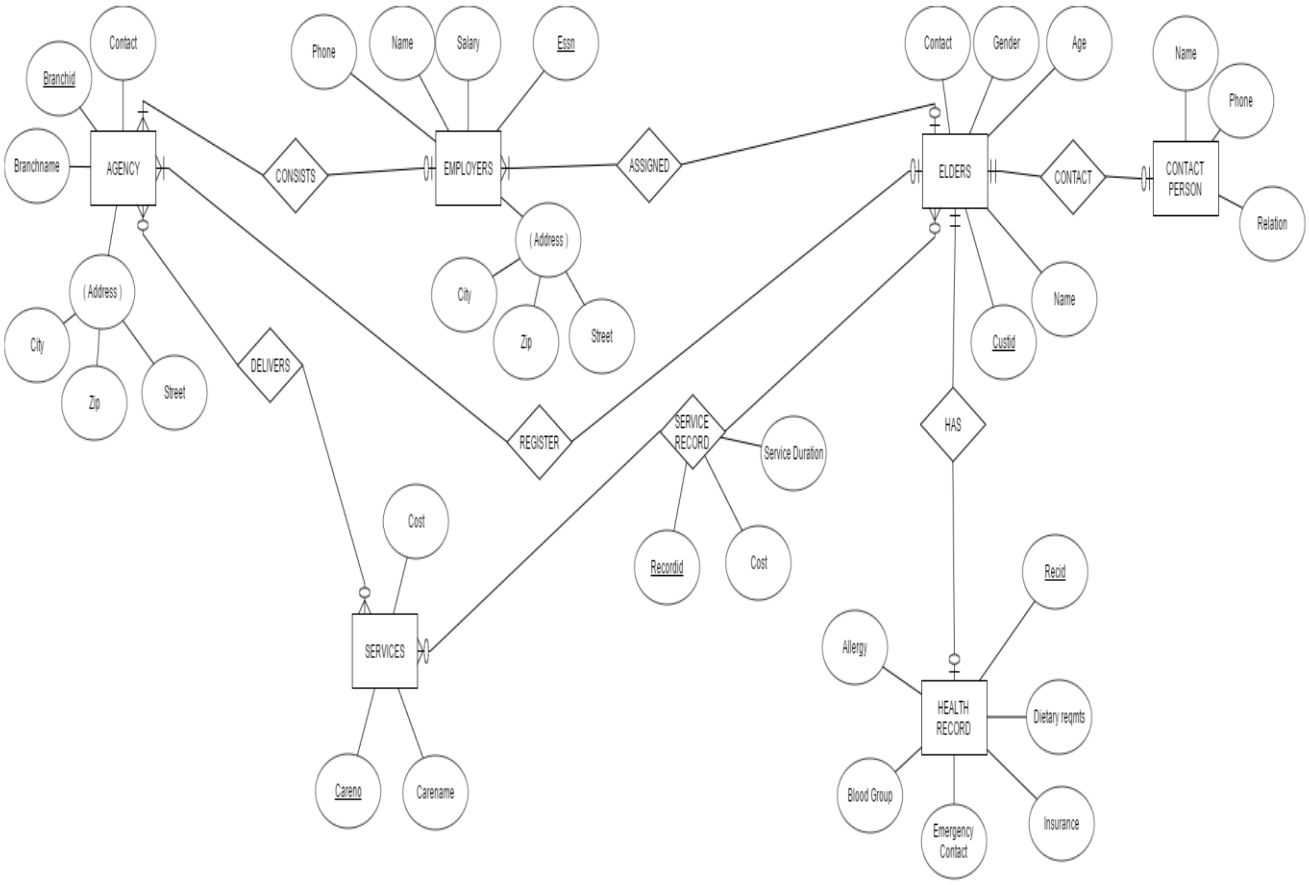
associated with these services.

6. .Agency also keeps track of services(Care) provided by different branches. All services may not be provided by one branch. So keeping track of this information is essential.

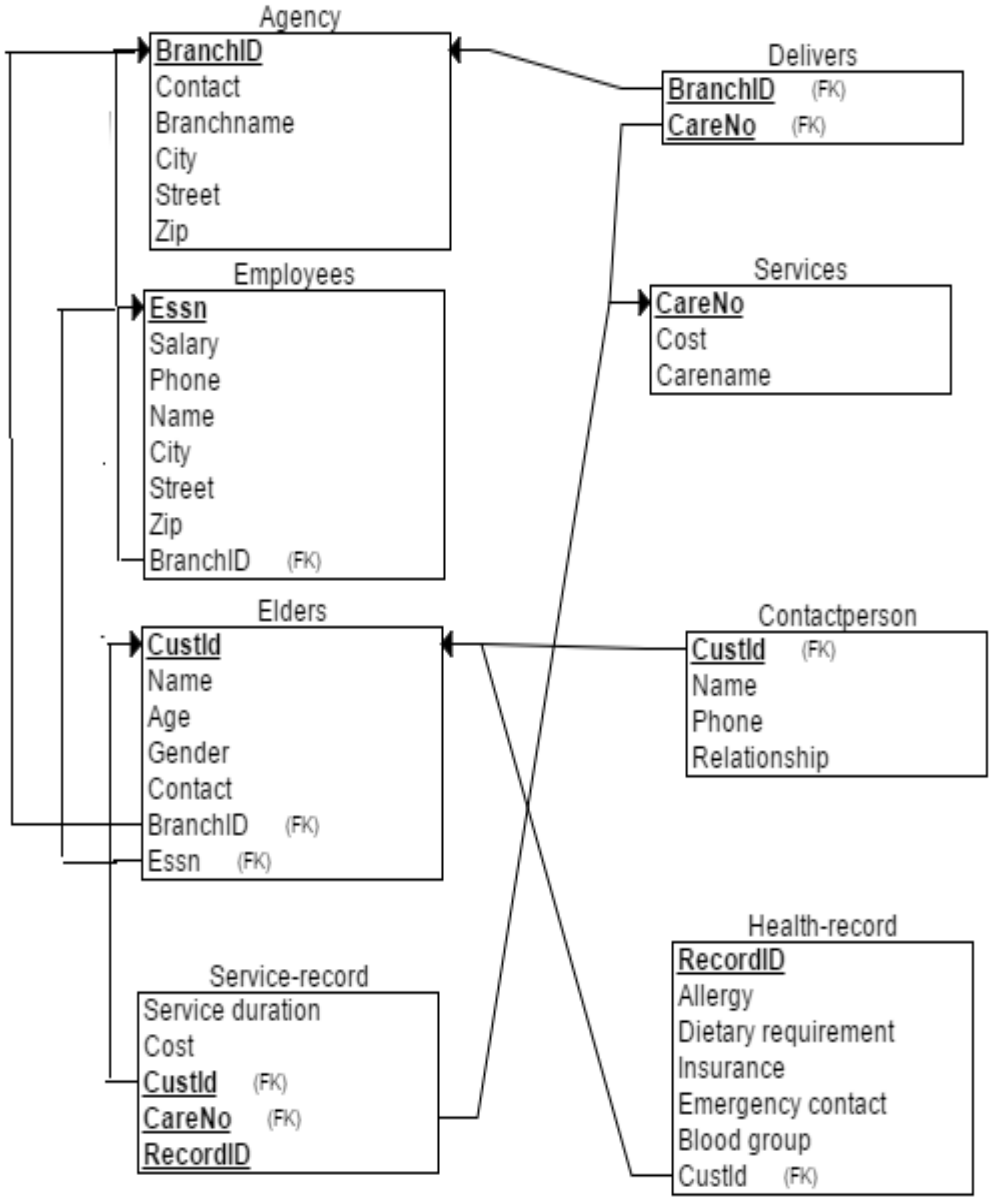
7. Agency keeps details of Service record which gives information about various services given to each customer, duration for this services and total cost that customer has to pay for the service offered.

.

ER Diagram



Relational schema


Agency
BranchID
Contact
Branchname
City
Street
Zip
Delivers
BranchID (FK)
CareNo (FK)
Employees
Essn
Salary
Phone
Name
City
Street
Zip
BranchID (FK)
Services
CareNo
Cost
Carename
Elders
CustId
Name
Age
Gender
Contact
BranchID (FK)
Essn (FK)
Contactperson
CustId (FK)
Name
Phone
Relationship
Health-record
RecordID
Allergy
Dietary requirement
Insurance
Emergency contact
Blood group
CustId (FK)
Service-record
Service duration
Cost
CustId (FK)
CareNo (FK)
RecordID

Normalization

When we map the ER diagram into relational schema they are already in 3NF form.

SQL Queries for creation and insertion of tables:

AGENCY

```
CREATE TABLE AGENCY(Branchid INT NOT NULL PRIMARY KEY,Contact INT
UNIQUE,Branchname VARCHAR(30),City VARCHAR(20),Street
VARCHAR(30),Zipcode INT);
```

EMPLOYEES

```
CREATE TABLE EMPLOYERS(Essn VARCHAR(15) NOT NULL PRIMARY KEY,Salary
INT NOT NULL CHECK(Salary>1000),Contact INT UNIQUE, Name
VARCHAR(20),City VARCHAR(20),Street VARCHAR(20),Zipcode INT, Branchid
INT NOT NULL,FOREIGN KEY(Branchid) REFERENCES AGENCY(Branchid) ON
DELETE CASCADE)
```

ELDERS

```
CREATE TABLE ELDERS(Custid INT NOT NULL PRIMARY KEY,Name
VARCHAR(15),Age INT,Gender CHAR,Contact INT UNIQUE,Branchid INT NOT
NULL,Ssn VARCHAR(15),FOREIGN KEY(Branchid) REFERENCES AGENCY(Branchid)
ON DELETE CASCADE,CONSTRAINT EMPCONS FOREIGN KEY(Ssn) REFERENCES
EMPLOYERS(Essn) ON DELETE SET NULL)
```

HEALTHRECORD

```
CREATE TABLE HEALTHRECORD(Recid INT NOT NULL PRIMARY KEY,Allergy
VARCHAR(50),Dietaryreq VARCHAR(50),Insurance
VARCHAR(40),Emergencycontact INT,Bloodgroup VARCHAR(5),Custid
INT,FOREIGN KEY(Custid) REFERENCES ELDERS(Custid) ON DELETE CASCADE)
```

SERVICES

```
CREATE TABLE SERVICES(Careno INT NOT NULL PRIMARY KEY,Cost INT
CHECK(Cost>2000),Carename VARCHAR(50))
```

DELIVERS

```
CREATE TABLE DELIVERS(Branchid INT NOT NULL,Careno INT NOT
NULL,PRIMARY KEY(Branchid,Careno),FOREIGN KEY(Branchid) REFERENCES
AGENCY(Branchid) ON DELETE CASCADE ,FOREIGN KEY(Careno) REFERENCES
SERVICES(Careno) ON DELETE CASCADE)
```

SERVICERECORD

```
CREATE TABLE SERVICERECORD(Careno INT NOT NULL,Custid INT NOT NULL
,Recid INT UNIQUE,Serviceduration VARCHAR(20),Cost INT,Carename
VARCHAR(50),PRIMARY KEY(Careno,Custid),FOREIGN KEY(Careno) REFERENCES
```

```
SERVICES(Careno) ON DELETE CASCADE, FOREIGN KEY(Careno) REFERENCES
SERVICES(Careno) ON DELETE CASCADE)
```

CONTACTPERSON

```
CREATE TABLE CONTACTPERSON(Custid INT NOT NULL PRIMARY KEY,Name
VARCHAR(50),Phone INT,Relation VARCHAR(50));
```

AGENCY TABLE INSERTION OF VALUES

```
INSERT INTO AGENCY VALUES(1,21432123,'ABCCaredallas','Richardson','xyz
renner crossing bldg',756432);
INSERT INTO AGENCY VALUES(2,21432432,'ABCCareohio','Ohio','asd walkman
crossing road',7521432);
INSERT INTO AGENCY VALUES(3,21343432,'ABCCareredmond','Seattle','aqw
capital crossing road',7521412);
INSERT INTO AGENCY
VALUES(4,2142232,'ABCCarecalifornia','california','asd trewqen
crossing road',7567432);
```

EMPLOYERS TABLE INSERTION OF VALUES

```
INSERT INTO EMPLOYERS
VALUES('SSN01078654321',10000,213345432,'Aby','Austin','crossing
street',756432,1)
INSERT INTO EMPLOYERS
VALUES('SSN017878654321',20000,2156345432,'Sunny','Chicago','swqe
street',756132,1)
INSERT INTO EMPLOYERS
VALUES('SSN10786543215',50000,213345132,'Sam','New York','ght
street',786432,2)
INSERT INTO EMPLOYERS
VALUES('SSN01578654321',15000,2133454892,'Rose','New Jersey','crossing
park',758732,2)
INSERT INTO EMPLOYERS
VALUES('SSN01078604321',8000,2143345432,'Charles','California','wood
park',762432,3)
INSERT INTO EMPLOYERS
VALUES('SSN01078659021',18000,213345732,'Maria','Arizona','walk
bridge',757832,3)
INSERT INTO EMPLOYERS
VALUES('SSN01074554321',19000,2453345432,'Rohan','Georgia','crossing
street',756467,4)
INSERT INTO EMPLOYERS
VALUES('SSN03128811321',14000,213212432,'George','Arlington','qwossing
street',756472,4)
```

ELDERS TABLE INSERTION OF VALUES

```
INSERT INTO ELDERS(Custid,Name,Age,Gender,Contact,Branchid)
VALUES(1,'JOHN',78,'M',213321456,1)
INSERT INTO ELDERS(Custid,Name,Age,Gender,Contact,Branchid)
VALUES(2,'RON',70,'M',223321456,2)
INSERT INTO ELDERS(Custid,Name,Age,Gender,Contact,Branchid)
VALUES(3,'MARIA',80,'F',215321456,3)
INSERT INTO ELDERS(Custid,Name,Age,Gender,Contact,Branchid)
VALUES(4,'CATHERINE',90,'F',224521456,4)
```

HEALTHRECORD TABLE INSERTION OF VALUES

```
INSERT INTO HEALTHRECORD VALUES(1,'Pollen','LOW CHOLESTROL FOOD','ABC
CORPORATION',21113344222,'A+',1);
INSERT INTO HEALTHRECORD VALUES(2,'Peanut Allergy','VEGETARIAN','AQW
CORPORATION',21112144222,'B+',2);
INSERT INTO HEALTHRECORD VALUES(3,'Egg Allergy','FOOD WITH NO
EGG','XYZ CORPORATION',21113344222,'A+',3);
INSERT INTO HEALTHRECORD VALUES(4,'Fish Allergy','NO FISH ITEMS','PWQ
CORPORATION',21113344222,'A+',4);
```

SERVICES TABLE INSERTION OF VALUES

```
INSERT INTO SERVICES VALUES(1,8000,'Alzheimer’s care')
INSERT INTO SERVICES VALUES(2,4000,'Stroke patients care')
INSERT INTO SERVICES VALUES(3,5000,'Day care')
INSERT INTO SERVICES VALUES(4,10000,'Cancer care')
```

DELIVERS TABLE INSERTION OF VALUES

```
INSERT INTO DELIVERS VALUES(1,1);
INSERT INTO DELIVERS VALUES(1,2);
INSERT INTO DELIVERS VALUES(2,3);
INSERT INTO DELIVERS VALUES(2,4);
INSERT INTO DELIVERS VALUES(3,1);
INSERT INTO DELIVERS VALUES(3,3);
INSERT INTO DELIVERS VALUES(4,2);
INSERT INTO DELIVERS VALUES(4,4);
```

CONTACTPERSON TABLE INSERTION OF VALUES

```
INSERT INTO CONTACTPERSON VALUES(1,'Rony',213321123,'SON');
INSERT INTO CONTACTPERSON VALUES(2,'Sony',212321123,'GRANDSON');
INSERT INTO CONTACTPERSON VALUES(3,'Jony',211321123,'SON');
INSERT INTO CONTACTPERSON VALUES(4,'Rose',215321123,'DAUGHTER');
```

Stored procedures

This procedure will displays agency branches along with care given, cost associated with the care which have been populated from three tables where the care number is given as input

```
set serveroutput on

CREATE OR REPLACE PROCEDURE branchcare(carenum IN SERVICES.Careno%TYPE) AS

bid AGENCY.Branchid%TYPE;

branchname AGENCY.Branchname%TYPE;

carenumber DELIVERS.Careno%TYPE;

carename SERVICES.Carename%TYPE;

cost SERVICES.Cost%TYPE;

careno SERVICES.Careno%TYPE;

CURSOR resultcare IS

SELECT A.Branchid,A.Branchname,D.Careno,S.Carename,S.Cost,S.Careno FROM AGENCY A
INNER JOIN DELIVERS D ON A.Branchid=D.Branchid INNER JOIN SERVICES S ON
D.Careno=S.Careno;

BEGIN

OPEN resultcare;

LOOP

FETCH resultcare INTO bid,branchname,carenumber,carename,cost,careno;

EXIT WHEN (resultcare%NOTFOUND);

if careno=carenum THEN

dbms_output.put_line(bid||' '||branchname||' '||carenumber||' '||carename||cost);

END IF;

END LOOP;

CLOSE resultcare;

END;
```

A sequence generator used to generate automatic record id while inserting into servicerecord table.

```
CREATE SEQUENCE seq_person

MINVALUE 1

START WITH 1

INCREMENT BY 1

CACHE 10
```

A stored procedure to insert record into Servicerecord when details like duration, customer id and careno is given. Sequence generator is used to populate unique Record id.

```
set serveroutput on

CREATE OR REPLACE PROCEDURE insertservice(duration IN
SERVICERECORD.Serviceduration%TYPE,cusid IN SERVICERECORD.Custid%TYPE,careno IN
SERVICES.Careno%TYPE) AS

totalcost INT;

carenumber INTEGER := careno;

cost SERVICES.Cost%TYPE;

carename SERVICES.Carename%TYPE;

CURSOR selserv is

SELECT Cost,Carename FROM SERVICES WHERE Careno=carenumber;

BEGIN

OPEN selserv;

FETCH selserv INTO cost,carename;

totalcost := cost*duration;

INSERT INTO SERVICERECORD VALUES(seq_person.nextval,duration,totalcost,carename,cusid);

CLOSE selserv;

END;
```

Triggers

Trigger fired when an update of carename in Services. This trigger will update the renamed carename in Servicerecord.

```
set serveroutput on

CREATE OR REPLACE TRIGGER careupdate

AFTER UPDATE ON SERVICES

FOR EACH ROW

DECLARE

carename SERVICES.Carename%TYPE;

recid SERVICERECORD.Recid%TYPE;

CURSOR serviceupdate is

SELECT Recid FROM SERVICERECORD;

BEGIN

open serviceupdate;

LOOP

FETCH serviceupdate INTO recid;

EXIT WHEN (serviceupdate%NOTFOUND);

UPDATE SERVICERECORD SET Carename=:new.Carename WHERE Carename=:old.Carename;

END LOOP;

CLOSE serviceupdate;

END careupdate;
```

This trigger is fired when an insertion occurs in Employee table. It will assign elders who has been registered with same branch as that of newly added employee. A row limit is given so only one row will updated based on FCFS and if assigned employee Column is NULL FOR Elder

```
create or replace trigger elderupdate
AFTER INSERT ON EMPLOYERS
FOR EACH ROW
DECLARE
branchid INT;
sn VARCHAR(50);
BEGIN
branchid:=:new.Branchid;
sn:=:new.Essn;
UPDATE ELDERS SET Ssn=sn where  Custid IN(SELECT Custid FROM ELDERS WHERE
Branchid=branchid AND Ssn IS NULL AND ROWNUM=1);
END elderupdate;
```

Rule implementation by using CHECK constraint

Check constraints are already implemented on ‘salary’ value for ‘employers’ table. Other check constraint is implemented on ‘cost’ value for services table.

Business Rules

Database design for Elder Care System

- An employee working in an agency should belong to at least one branch and the employee should not be part of more than one branch.
- Elders should be registered to at least one branch. They should not be registered to more than one branch.
- Employees will be assigned to elders who are registered with same branch in which Employee works.

- Elders are registered with one of these branches. Employees will be assigned to elders who are registered with same branch in which Employee works.
- Elder Care will offer various Services. A branch may not provide all Services and each Services have different Cost associated with it.
- Service Duration is stored in duration of months. While calculating total cost, duration is taken as months.

Constraints and Assumptions

- Minimum Salary of Employees should be 1000
- Minimum Cost of Service should be 2000
- An Employee can be assigned to more than one Elder.
- It is not mandatory that agency have to give all services.