**PL/SQL:**

DROP TABLE CSURGERY;

DROP TABLE CPET;

DROP TABLE COWNER;

DROP TABLE CCOWNER;

DROP TABLE CKIND;

DROP TABLE CSPEC;

DROP TABLE CVET\_DOCTOR;

CREATE TABLE CSURGERY AS

SELECT \* FROM SURGERY;

CREATE TABLE CPET AS

SELECT \* FROM PET;

CREATE TABLE COWNER AS

SELECT \* FROM OWNER;

drop table ccowner;

CREATE TABLE CCOWNER AS

SELECT \* FROM COWNER;

CREATE TABLE CKIND AS

SELECT \* FROM KIND;

CREATE TABLE CSPEC AS

SELECT \* FROM SPEC;

CREATE TABLE CPRICE\_CHANGE AS

SELECT \* FROM PRICE\_CHANGE;

CREATE TABLE CVET\_DOCTOR AS

SELECT \* FROM VET\_DOCTOR;

DROP TABLE VET\_DOCTOR;

DROP TABLE PET;

DROP TABLE OWNER;

DROP TABLE KIND;

DROP TABLE SPEC;

DROP TABLE PRICE\_CHANGE;

CREATE TABLE KIND

(idKind integer,

KindName varchar2(50),

PRIMARY KEY (idKind)

);

CREATE TABLE OWNER

(idOwner integer,

OName varchar2(50),

OSurname varchar2(50),

PRIMARY KEY (idOwner));

CREATE TABLE SPEC

(idSpec integer,

Specialization varchar2(50),

PRIMARY KEY (idSpec)

);

CREATE TABLE VET\_DOCTOR

(idDoctor integer,

DName varchar2(50),

DSurname varchar2(50),

Salary integer,

spec integer,

PRIMARY KEY (idDoctor),

FOREIGN KEY (spec) REFERENCES spec(idSpec));

CREATE TABLE PET

(idPet integer,

PetName varchar2(50),

dateOfBirth date,

owner integer,

kind integer,

PRIMARY KEY (idPet),

FOREIGN KEY (kind) REFERENCES kind(idKind)

);

CREATE TABLE SURGERY

(idSurgery integer,

pet integer,

doctor integer,

Price integer,

Issue varchar2(50),

Duration integer,

dateOfSurgery date,

PRIMARY KEY (idSurgery),

FOREIGN KEY (pet) REFERENCES Pet(idPet),

FOREIGN KEY (doctor) REFERENCES Vet\_doctor(idDoctor));

INSERT INTO KIND VALUES (1, 'Cat');

INSERT INTO KIND VALUES (2, 'Dog');

INSERT INTO KIND VALUES (3, 'Parrot');

INSERT INTO KIND VALUES (4, 'Snake');

INSERT INTO KIND VALUES (5, 'Hamster');

INSERT INTO KIND VALUES (6, 'Mouse');

INSERT INTO KIND VALUES (7, 'Turtle');

INSERT INTO KIND VALUES (8, 'Rabbit');

INSERT INTO KIND VALUES (9, 'Ferret');

INSERT INTO KIND VALUES (11, 'Spider');

INSERT INTO OWNER VALUES ( 133, 'Azamat', 'Maraimbekov');

INSERT INTO OWNER VALUES ( 154, 'Viktor', 'Tsoy');

INSERT INTO OWNER VALUES ( 178, 'Evgeniy', 'Ponasenkov');

INSERT INTO OWNER VALUES ( 141, 'Harry', 'Smith');

INSERT INTO OWNER VALUES ( 122, 'Madara', 'Uchiha');

INSERT INTO OWNER VALUES ( 100, 'Lana', 'Roads');

INSERT INTO OWNER VALUES ( 199, 'Richard', 'Waterson');

INSERT INTO OWNER VALUES ( 168, 'Dmitry', 'Gordon');

INSERT INTO OWNER VALUES ( 145, 'Ben', 'Ten');

INSERT INTO OWNER VALUES ( 131, 'Ragnar', 'Lothbrok');

INSERT INTO OWNER VALUES ( 172, 'Thomas', 'Shelby');

INSERT INTO OWNER VALUES ( 150, 'Thor', 'Odinsson');

INSERT INTO OWNER VALUES ( 162, 'Adolf', 'Miller');

INSERT INTO SPEC VALUES ( 10, 'Felinology');

INSERT INTO SPEC VALUES ( 20, 'Kinology');

INSERT INTO SPEC VALUES ( 30, 'Ornitology');

INSERT INTO SPEC VALUES ( 40, 'Herpetology');

INSERT INTO SPEC VALUES ( 50, 'Ratology');

INSERT INTO SPEC VALUES ( 60, NULL);

INSERT INTO VET\_DOCTOR VALUES ( 900, 'Maria', 'Vans', 2000, 10);

INSERT INTO VET\_DOCTOR VALUES ( 901, 'Paul', 'Dickinson', 1500, 10);

INSERT INTO VET\_DOCTOR VALUES ( 909, 'Neil', 'Tyson', 300, 50);

INSERT INTO VET\_DOCTOR VALUES ( 987, 'Katarina', 'Ten', 2000, 20);

INSERT INTO VET\_DOCTOR VALUES ( 965, 'Joseph', 'Peaks', 2500, 10);

INSERT INTO VET\_DOCTOR VALUES ( 942, 'Danil', 'Danilov',4000, 60);

INSERT INTO VET\_DOCTOR VALUES ( 976, 'Abdullah', NULL, 2000, 20);

INSERT INTO VET\_DOCTOR VALUES ( 939, 'Daria', 'Dantsova', 5000, 40);

INSERT INTO VET\_DOCTOR VALUES ( 962, 'KaiSa', 'Kassadinovic', 3000, 30);

INSERT INTO PET VALUES ( 142 , 'Melania', '14/02/2017', 133, 1);

INSERT INTO PET VALUES ( 273, 'Pushinka', '04/10/2003', 133, 9);

INSERT INTO PET VALUES ( 821, 'Nasus', '22/12/2009', 133, 2);

INSERT INTO PET VALUES ( 309, 'Barker', '09/01/2011', 154, 3);

INSERT INTO PET VALUES ( 871, 'Chia', '22/03/2011', 100, 6);

INSERT INTO PET VALUES ( 652, 'Nelly', '30/09/2011', 178, 1);

INSERT INTO PET VALUES ( 333, 'Obito', '12/06/2017', 122, 9);

INSERT INTO PET VALUES ( 133, 'Silvy', NULL, 133, 5);

INSERT INTO PET VALUES ( 912, 'Gumball', '03/05/2011', 199, 1);

INSERT INTO PET VALUES ( 762, 'Annaise', '04/06/2018', 199, 8);

INSERT INTO PET VALUES ( 455, 'Joky', '27/04/2020', 168, 4);

INSERT INTO PET VALUES ( 671, 'Bjorn', '21/08/2015', 131, 2);

INSERT INTO PET VALUES ( 221, 'Loki', '23/07/2016', 150, 4);

INSERT INTO PET VALUES ( 699, 'Arthur', '19/12/2016', 172, 3);

INSERT INTO PET VALUES ( 148, 'Hund', '19/01/2021', 162, 2);

INSERT INTO PET VALUES ( 442, 'TenBen', '13/09/2014', 145, 1);

INSERT INTO PET VALUES ( 141, 'Zed', '26/11/2018', 141, 4);

INSERT INTO PET VALUES ( 555, 'Susan', NULL, 178, 7);

INSERT INTO SURGERY VALUES ( 5021, 142, 900, 150, 'Heart attack', 67, '01/02/2019');

INSERT INTO SURGERY VALUES ( 5133, 273, 942, 200, 'Rabies', 150, '26/08/2020');

INSERT INTO SURGERY VALUES ( 5948, 309, 962, 350, 'Swine influenza', 30, '25/01/2017');

INSERT INTO SURGERY VALUES ( 5869, 871, 909, 150, 'Heart attack', 60, '02/06/2019');

INSERT INTO SURGERY VALUES ( 5048, 652, 901, 250, 'Rabies', 167, '12/02/2021');

INSERT INTO SURGERY VALUES ( 5312, 333, 942, 50, 'Extreme hair loss', 40, '20/04/2020');

INSERT INTO SURGERY VALUES ( 5812, 133, 909, 140, 'Heart attack', 65, '05/08/2018');

INSERT INTO SURGERY VALUES ( 5321, 912, 901, 450, 'Brain damage', 207, '20/12/2018');

INSERT INTO SURGERY VALUES ( 5305, 762, 909, 200, 'Rabies', 142, '23/11/2018');

INSERT INTO SURGERY VALUES ( 5813, 455, 939, 140, 'Heart attack', 79, '15/09/2020');

INSERT INTO SURGERY VALUES ( 5272, 699, 962, 350, 'Swine influenza', 35, '08/11/2019');

INSERT INTO SURGERY VALUES ( 5442, 671, 987, 450, 'Brain damage', 220, '06/06/2020');

INSERT INTO SURGERY VALUES ( 5574, 221, 939, 90, 'Extreme hair loss', 42, '30/05/2020');

INSERT INTO SURGERY VALUES ( 5560, 148, 976, 150, 'Heart attack', 90, '22/07/2021');

INSERT INTO SURGERY VALUES ( 5552, 141, 939, 270, 'Rabies', 117, '11/10/2019');

INSERT INTO SURGERY VALUES ( 5760, 555, 939, 200, 'Rabies', 123, '08/06/2017');

INSERT INTO SURGERY VALUES ( 5694, 442, 900, 100, 'Teeth infection', 100, '13/01/2020');

INSERT INTO SURGERY VALUES ( 5412, 333, NULL, 450, 'Brain damage', 232, '05/08/2018');

INSERT INTO SURGERY VALUES ( 5550, 821, 976, 150, NULL, 88, '30/03/2019');

INSERT INTO SURGERY VALUES ( 5171, 821, 987, 280, 'Rabies', 188, '27/12/2019');

--Triggers

--1. Trigger to record all updates on owners ids

--in a separate table Records (new id, old id and date), with specifying date of update.

DROP TABLE Records;

CREATE TABLE Records(

idOwn INTEGER,

idNewOwn INTEGER,

dateofupdating DATE

);

DROP TRIGGER Recording;

CREATE OR REPLACE TRIGGER Recording

AFTER UPDATE

ON cowner

FOR EACH ROW

DECLARE

CURSOR cursor\_first IS

SELECT \*

FROM ccowner

WHERE idowner = :OLD.idowner;

cursor\_second cursor\_first%rowtype;

BEGIN

OPEN cursor\_first;

LOOP

FETCH cursor\_first INTO cursor\_second;

EXIT WHEN cursor\_first%notfound;

INSERT INTO Records VALUES(:NEW.idowner, :OLD.idowner, sysdate);

END LOOP;

CLOSE cursor\_first;

END;

UPDATE cowner

SET idowner = 101

WHERE idowner = 100;

SELECT \* FROM cowner;

SELECT \* FROM records;

--2. Trigger preventing from creating more than 3 surgeries for one pet.

CREATE OR REPLACE TRIGGER TooMuchSurg

BEFORE INSERT OR UPDATE ON csurgery

FOR EACH ROW

DECLARE

CURSOR petsur IS

SELECT COUNT(\*)sur\_count

FROM csurgery

WHERE pet = :new.pet;

toomuch EXCEPTION;

countSur INTEGER;

BEGIN

OPEN petsur;

LOOP

FETCH petsur into countSur;

EXIT WHEN petsur%notfound;

IF countSur > 2 THEN

RAISE toomuch;

ELSE

dbms\_output.put\_line('Surgery has been set.');

END IF;

END LOOP;

CLOSE petsur;

EXCEPTION WHEN toomuch THEN

raise\_application\_error(-20010, ' Sorry ');

END;

INSERT INTO CPET VALUES (228, '4irik', '12.12.1212', 133, 11);

INSERT INTO CSURGERY VALUES (999, 228, 976, 9, 'Very bad', 2, '30.03.0303');

INSERT INTO CSURGERY VALUES (1000, 228, 976, 9, 'Very bad', 2, '30.03.0303');

INSERT INTO CSURGERY VALUES (1001, 228, 976, 9, 'Very bad', 2, '30.03.0303');

INSERT INTO CSURGERY VALUES (1002, 228, 976, 9, 'Very bad', 2, '30.03.0303');

CREATE OR REPLACE TRIGGER YoungestPet

AFTER INSERT OR UPDATE OR DELETE

ON cpet

DECLARE

YouPet date;

CURSOR maxDob IS

SELECT dateofbirth

FROM cpet;

NoNew EXCEPTION;

dating DATE;

numb INTEGER;

BEGIN

OPEN maxdob;

LOOP

FETCH maxdob INTO dating;

EXIT WHEN maxdob%notfound;

END LOOP;

SELECT COUNT(idpet) INTO numb

FROM cpet;

IF (maxdob%rowcount = numb) THEN

SELECT MAX(dateofbirth) INTO YouPet

FROM cpet;

dbms\_output.put\_line('Youngest pet birthday is: ' || YouPet);

ELSE RAISE NoNew;

END IF;

CLOSE maxDob;

EXCEPTION WHEN NoNew THEN

dbms\_output.put\_line('Nothing new');

END;

INSERT INTO CPET VALUES(010, 'Kakadu', '20/12/2012', NULL,NULL);

--3.

--Procedures

--1. Procedure to raise salaries by 50% of vet doctors whose specialization is 30

CREATE OR REPLACE PROCEDURE OrniGift

AS

nodoctors EXCEPTION;

CURSOR doctor IS

SELECT idDoctor

FROM cvet\_doctor

WHERE spec = 30;

counts INTEGER;

iddoc INTEGER;

BEGIN

SELECT COUNT(iddoctor) INTO counts

FROM cvet\_doctor

WHERE spec = 30;

IF counts = 0 THEN RAISE nodoctors;

ELSE

OPEN doctor;

LOOP

FETCH doctor into iddoc;

EXIT WHEN doctor%notfound;

UPDATE CVET\_DOCTOR

SET salary = salary\*1.5

WHERE iddoctor = iddoc and spec = 30;

dbms\_output.put\_line('Ornitologists received the raise for salary! Yay!!');

END LOOP;

END IF;

EXCEPTION WHEN nodoctors THEN

dbms\_output.put\_line('Ornitologists do not exist.');

END;

EXEC OrniGift;

select \* from vet\_doctor;

select \* from cvet\_doctor;

--2. Procedure to show on the screen id of vet doctor and number of his surgeries. Make it for all doctors having a given specialization as a parameter.

CREATE OR REPLACE PROCEDURE Surgeries\_Done (special INTEGER)

AS

noSpec EXCEPTION;

noDoc EXCEPTION;

numSpec INTEGER;

numDoc INTEGER;

CURSOR vetdoc IS

SELECT iddoctor

FROM cvet\_doctor

WHERE spec = special;

unknown INTEGER;

numSurgeries INTEGER;

BEGIN

SELECT COUNT(idspec) INTO numSpec

FROM cspec

WHERE idspec = special;

IF numSpec = 0 THEN

RAISE noSpec;

ELSE

SELECT COUNT(iddoctor) INTO numDoc

FROM CVET\_DOCTOR

WHERE spec = special;

IF numDoc = 0 THEN

RAISE noDoc;

ELSE

OPEN vetdoc;

LOOP

FETCH vetdoc INTO unknown;

EXIT WHEN vetdoc%notfound;

SELECT COUNT(idsurgery) INTO numSurgeries

FROM CSURGERY

WHERE doctor = unknown;

dbms\_output.put\_line(unknown || ' has operated ' || numSurgeries || ' surgeries.');

END LOOP;

CLOSE vetdoc;

END IF;

END IF;

EXCEPTION WHEN noSpec THEN

dbms\_output.put\_line('No specialization!');

WHEN noDoc THEN

dbms\_output.put\_line('No doctors!');

END;

EXEC Surgeries\_Done(30);

--

CREATE OR REPLACE PROCEDURE Surg\_done (special INTEGER)

AS

noSpec EXCEPTION;

noDoc EXCEPTION;

numSpec INTEGER;

numDoc INTEGER;

CURSOR vetdoc IS

SELECT iddoctor

FROM cvet\_doctor

WHERE spec = special;

unknown INTEGER;

numSurgeries INTEGER;

BEGIN

SELECT COUNT(idspec) INTO numSpec

FROM cspec

WHERE idspec = special;

IF numSpec = 0 THEN

RAISE noSpec;

ELSE

SELECT COUNT(iddoctor) INTO numDoc

FROM CVET\_DOCTOR

WHERE spec = special;

IF numDoc = 0 THEN

RAISE noDoc;

ELSE

OPEN vetdoc;

LOOP

FETCH vetdoc INTO unknown;

EXIT WHEN vetdoc%notfound;

SELECT COUNT(idsurgery) INTO numSurgeries

FROM CSURGERY

WHERE doctor = unknown;

dbms\_output.put\_line(unknown || ' has operated ' || numSurgeries || ' surgeries.');

END LOOP;

CLOSE vetdoc;

END IF;

END IF;

EXCEPTION WHEN noSpec THEN

dbms\_output.put\_line('No specialization!');

WHEN noDoc THEN

dbms\_output.put\_line('No doctors!');

END;

--3. Procedure to change NULL speciaazliation to 'Undefined'

CREATE OR REPLACE PROCEDURE NullUndef

AS

NoSpec EXCEPTION;

num INTEGER;

idSpecial INTEGER;

Cursor special IS

select idspec

from cspec

where specialization is NULL;

Begin

select count(idspec) into num

from cspec

where specialization is NULL;

if num = 0 then raise NoSpec;

else

open special;

loop

fetch special into idSpecial;

exit when special%notfound;

update cspec

set specialization = 'Undefined'

where idspec = idspecial;

end loop;

end if;

close special;

exception when NoSpec then

dbms\_output.put\_line('No specializations');

end;

exec nullundef;

select \* FROM cspec;

**T-SQL:**

CREATE TABLE KIND

(idKind int,

KindName varchar(50),

PRIMARY KEY (idKind)

);

CREATE TABLE OWNER

(idOwner integer,

OName varchar(50),

OSurname varchar(50),

PRIMARY KEY (idOwner));

CREATE TABLE SPEC

(idSpec int,

Specialization varchar(50),

PRIMARY KEY (idSpec)

);

CREATE TABLE VET\_DOCTOR

(idDoctor int,

DName varchar(50),

DSurname varchar(50),

Salary int,

spec int,

PRIMARY KEY (idDoctor),

FOREIGN KEY (spec) REFERENCES spec(idSpec));

CREATE TABLE PET

(idPet int,

PetName varchar(50),

dateOfBirth date,

owner int,

kind int,

PRIMARY KEY (idPet),

FOREIGN KEY (kind) REFERENCES kind(idKind)

);

CREATE TABLE SURGERY

(idSurgery int,

pet int,

doctor int,

Price int,

Issue varchar(50),

Duration int,

dateOfSurgery date,

PRIMARY KEY (idSurgery),

FOREIGN KEY (pet) REFERENCES Pet(idPet),

FOREIGN KEY (doctor) REFERENCES Vet\_doctor(idDoctor));

INSERT INTO KIND VALUES (1, 'Cat');

INSERT INTO KIND VALUES (2, 'Dog');

INSERT INTO KIND VALUES (3, 'Parrot');

INSERT INTO KIND VALUES (4, 'Snake');

INSERT INTO KIND VALUES (5, 'Hamster');

INSERT INTO KIND VALUES (6, 'Mouse');

INSERT INTO KIND VALUES (7, 'Turtle');

INSERT INTO KIND VALUES (8, 'Rabbit');

INSERT INTO KIND VALUES (9, 'Ferret');

INSERT INTO KIND VALUES (11, 'Spider');

INSERT INTO OWNER VALUES ( 133, 'Azamat', 'Maraimbekov');

INSERT INTO OWNER VALUES ( 154, 'Viktor', 'Tsoy');

INSERT INTO OWNER VALUES ( 178, 'Evgeniy', 'Ponasenkov');

INSERT INTO OWNER VALUES ( 141, 'Harry', 'Smith');

INSERT INTO OWNER VALUES ( 122, 'Madara', 'Uchiha');

INSERT INTO OWNER VALUES ( 100, 'Lana', 'Roads');

INSERT INTO OWNER VALUES ( 199, 'Richard', 'Waterson');

INSERT INTO OWNER VALUES ( 168, 'Dmitry', 'Gordon');

INSERT INTO OWNER VALUES ( 145, 'Ben', 'Ten');

INSERT INTO OWNER VALUES ( 131, 'Ragnar', 'Lothbrok');

INSERT INTO OWNER VALUES ( 172, 'Thomas', 'Shelby');

INSERT INTO OWNER VALUES ( 150, 'Thor', 'Odinsson');

INSERT INTO OWNER VALUES ( 162, 'Adolf', 'Miller');

INSERT INTO SPEC VALUES ( 10, 'Felinology');

INSERT INTO SPEC VALUES ( 20, 'Kinology');

INSERT INTO SPEC VALUES ( 30, 'Ornitology');

INSERT INTO SPEC VALUES ( 40, 'Herpetology');

INSERT INTO SPEC VALUES ( 50, 'Ratology');

INSERT INTO SPEC VALUES ( 60, NULL);

INSERT INTO VET\_DOCTOR VALUES ( 900, 'Maria', 'Vans', 2000, 10);

INSERT INTO VET\_DOCTOR VALUES ( 901, 'Paul', 'Dickinson', 1500, 10);

INSERT INTO VET\_DOCTOR VALUES ( 909, 'Neil', 'Tyson', 300, 50);

INSERT INTO VET\_DOCTOR VALUES ( 987, 'Katarina', 'Ten', 2000, 20);

INSERT INTO VET\_DOCTOR VALUES ( 965, 'Joseph', 'Peaks', 2500, 10);

INSERT INTO VET\_DOCTOR VALUES ( 942, 'Danil', 'Danilov',4000, 60);

INSERT INTO VET\_DOCTOR VALUES ( 976, 'Abdullah', NULL, 2000, 20);

INSERT INTO VET\_DOCTOR VALUES ( 939, 'Daria', 'Dantsova', 5000, 40);

INSERT INTO VET\_DOCTOR VALUES ( 962, 'KaiSa', 'Kassadinovic', 3000, 30);

INSERT INTO PET VALUES ( 142 , 'Melania', '2017-02-14', 133, 1);

INSERT INTO PET VALUES ( 273, 'Pushinka', '2003-10-04', 133, 9);

INSERT INTO PET VALUES ( 821, 'Nasus', '2009-12-22', 133, 2);

INSERT INTO PET VALUES ( 309, 'Barker', '2011-01-09', 154, 3);

INSERT INTO PET VALUES ( 871, 'Chia', '2011-03-22', 100, 6);

INSERT INTO PET VALUES ( 652, 'Nelly', '2011-09-30', 178, 1);

INSERT INTO PET VALUES ( 333, 'Obito', '2017-06-12', 122, 9);

INSERT INTO PET VALUES ( 133, 'Silvy', NULL, 133, 5);

INSERT INTO PET VALUES ( 912, 'Gumball', '2011-05-03', 199, 1);

INSERT INTO PET VALUES ( 762, 'Annaise', '2018-06-04', 199, 8);

INSERT INTO PET VALUES ( 455, 'Joky', '2020-04-27', 168, 4);

INSERT INTO PET VALUES ( 671, 'Bjorn', '2015-8-21', 131, 2);

INSERT INTO PET VALUES ( 221, 'Loki', '2016-07-23', 150, 4);

INSERT INTO PET VALUES ( 699, 'Arthur', '2016-12-19', 172, 3);

INSERT INTO PET VALUES ( 148, 'Hund', '2021-01-19', 162, 2);

INSERT INTO PET VALUES ( 442, 'TenBen', '2014-09-13', 145, 1);

INSERT INTO PET VALUES ( 141, 'Zed', '2018-11-26', 141, 4);

INSERT INTO PET VALUES ( 555, 'Susan', NULL, 178, 7);

INSERT INTO SURGERY VALUES ( 5021, 142, 900, 150, 'Heart attack', 67, '2019-02-01');

INSERT INTO SURGERY VALUES ( 5133, 273, 942, 200, 'Rabies', 150, '2020-08-06');

INSERT INTO SURGERY VALUES ( 5948, 309, 962, 350, 'Swine influenza', 30, '2017-01-25');

INSERT INTO SURGERY VALUES ( 5869, 871, 909, 150, 'Heart attack', 60, '2019-06-02');

INSERT INTO SURGERY VALUES ( 5048, 652, 901, 250, 'Rabies', 167, '2021-02-12');

INSERT INTO SURGERY VALUES ( 5312, 333, 942, 50, 'Extreme hair loss', 40, '2020-04-20');

INSERT INTO SURGERY VALUES ( 5812, 133, 909, 140, 'Heart attack', 65, '2018-08-05');

INSERT INTO SURGERY VALUES ( 5321, 912, 901, 450, 'Brain damage', 207, '2018-12-20');

INSERT INTO SURGERY VALUES ( 5305, 762, 909, 200, 'Rabies', 142, '2018-11-23');

INSERT INTO SURGERY VALUES ( 5813, 455, 939, 140, 'Heart attack', 79, '2020-09-15');

INSERT INTO SURGERY VALUES ( 5272, 699, 962, 350, 'Swine influenza', 35, '2019-11-08');

INSERT INTO SURGERY VALUES ( 5442, 671, 987, 450, 'Brain damage', 220, '2020-06-06');

INSERT INTO SURGERY VALUES ( 5574, 221, 939, 90, 'Extreme hair loss', 42, '2020-05-30');

INSERT INTO SURGERY VALUES ( 5560, 148, 976, 150, 'Heart attack', 90, '2021-07-20');

INSERT INTO SURGERY VALUES ( 5552, 141, 939, 270, 'Rabies', 117, '2019-10-11');

INSERT INTO SURGERY VALUES ( 5760, 555, 939, 200, 'Rabies', 123, '2017-06-08');

INSERT INTO SURGERY VALUES ( 5694, 442, 900, 100, 'Teeth infection', 100, '2020-01-13');

INSERT INTO SURGERY VALUES ( 5412, 333, NULL, 450, 'Brain damage', 232, '2018-08-05');

INSERT INTO SURGERY VALUES ( 5550, 821, 976, 150, NULL, 88, '2019-09-30');

INSERT INTO SURGERY VALUES ( 5171, 821, 987, 280, 'Rabies', 188, '2019-12-27');

select \* from KIND;

select \* from OWNER;

select \* from PET;

select \* from SPEC;

select \* from SURGERY;

select \* from VET\_DOCTOR;

--Triggers:

--1. Trigger not allowing to insert pets with birthdate later than the current date

DROP trigger BirthReal;

CREATE TRIGGER BirthReal ON PET FOR INSERT

AS

IF EXISTS(SELECT 1 FROM inserted WHERE dateOfBirth > CONVERT(date, sysdatetime()))

BEGIN

DECLARE

dateCursor CURSOR FOR SELECT idPet FROM inserted

WHERE dateOfBirth > CONVERT(date, sysdatetime());

DECLARE

@idpet integer;

OPEN dateCursor

FETCH NEXT FROM dateCursor INTO @idpet;

WHILE @@FETCH\_STATUS = 0

BEGIN

DELETE FROM PET WHERE idPet = @idpet;

FETCH NEXT FROM dateCursor INTO @idpet;

end;

CLOSE dateCursor;

Deallocate dateCursor;

END;

INSERT INTO PET VALUES(9939, 'ff', '2025-11-09', NULL, NULL);

select \* from pet;

--2. Trigger that tracks all deleted changes in spec and writes them in a separate table, using a datetime of deleting

DROP TABLE Tracking;

CREATE TABLE Tracking(

delSpec INTEGER,

delDate DATETIME

);

DROP TRIGGER TrackSpec

CREATE TRIGGER TrackSpec ON SPEC FOR DELETE

AS

IF EXISTS(SELECT 1 FROM deleted)

BEGIN

DECLARE

specCursor CURSOR FOR SELECT idSpec FROM deleted;

DECLARE

@idSpec INTEGER;

OPEN specCursor;

FETCH NEXT FROM specCursor INTO @idSpec;

WHILE @@FETCH\_STATUS = 0

BEGIN

INSERT INTO Tracking VALUES(@idSpec, SYSDATETIME())

FETCH NEXT FROM specCursor INTO @idSpec;

end

close specCursor;

deallocate specCursor;

end

DELETE FROM SPEC WHERE idSpec = 2;

SELECT \* FROM Tracking;

select \* from spec;

select \* from deleted;

--3. Vet surgery has banned an owner Lucy Steinfield. This trigger does not allow to have her in the table Owner.

drop table Blacklist;

create table Blacklist(

idPerson integer,

OnameB VARCHAR(50),

OsurnameB VARCHAR(50),

PRIMARY KEY (idPerson)

);

INSERT INTO Blacklist VALUES (1, 'Lucy', 'Steinfield');

INSERT INTO Blacklist VALUES (2, 'Ryan', 'Laurince');

INSERT INTO Blacklist VALUES (3, 'Optimus', 'Prime');

INSERT INTO Blacklist VALUES (4, 'David', 'King')

DROP trigger BanPeople;

CREATE TRIGGER BanPeople ON OWNER FOR INSERT, UPDATE

AS

BEGIN

DECLARE

ownCursor CURSOR FOR SELECT idOwner FROM Owner, Blacklist

WHERE OName = OnameB and OSurname = OsurnameB;

DECLARE

@idown integer;

DECLARE

@errMsg VARCHAR(50);

OPEN ownCursor

FETCH NEXT FROM ownCursor INTO @idown;

WHILE @@FETCH\_STATUS = 0

BEGIN

DELETE FROM OWNER WHERE idOwner = @idown;

SELECT @errMsg = 'This person is in the blacklist.';

RAISERROR(@errMsg, 16, 1);

FETCH NEXT FROM ownCursor INTO @idown;

end;

CLOSE ownCursor;

Deallocate ownCursor;

END;

INSERT INTO OWNER VALUES(321321, 'Optimus', 'Prime');

INSERT INTO OWNER VALUES(229113, 'Mega', 'tron');

UPDATE OWNER SET Oname = 'Optimus', OSurname = 'Prime' WHERE idOwner = 229113;

select \* from owner;

--Procedures:

--1. Procedure to raise salaries by 50% of vet doctors whose specialization is 30

DROP PROCEDURE OrniGift;

CREATE PROCEDURE OrniGift

AS

BEGIN

DECLARE

@counts INTEGER;

DECLARE

@iddoct INTEGER;

DECLARE

imCursor CURSOR LOCAL FOR

SELECT idDoctor

FROM VET\_DOCTOR

WHERE SPEC = 30;

SET nocount on;

SELECT @counts = count(idDoctor)

FROM VET\_DOCTOR

WHERE SPEC = 30;

BEGIN

IF @counts = 0

BEGIN

RAISERROR('No doctors', 16, 1);

END

ELSE

BEGIN

open imCursor;

FETCH imCursor INTO @iddoct;

UPDATE VET\_DOCTOR

SET Salary = Salary\*1.5

WHERE idDoctor = @iddoct and spec = 30;

print('Salaries were raised for ornitologists');

end

close imCursor;

deallocate imCursor;

SELECT \* FROM VET\_DOCTOR;

END

END;

EXEC OrniGift;

--2 Procedure to show number of surgeries for the given id of a doctor.

DROP PROCEDURE Surgeries\_done;

CREATE PROCEDURE Surgeries\_done

@docId INTEGER,

@docSurg varchar(50) OUTPUT

AS

BEGIN

DECLARE

@numsurg INTEGER;

DECLARE

docCursor CURSOR LOCAL FOR

select doctor

from SURGERY

where doctor = @docId;

DECLARE

@x INTEGER;

SET nocount ON;

SELECT @numsurg = count(idSurgery)

FROM SURGERY

WHERE doctor = @docId;

BEGIN

OPEN docCursor;

FETCH docCursor into @x;

SELECT @numsurg = count(idSurgery)

FROM SURGERY

WHERE doctor = @docId;

SELECT @docSurg = 'Doctor ' + cast(@docId as VARCHAR(50)) + ' has ' + cast(@numsurg as VARCHAR(50)) + ' surgeries';

close docCursor;

deallocate docCursor;

end

end;

DECLARE @new\_docSurg VARCHAR(50);

EXECUTE Surgeries\_done 981, @new\_docSurg OUTPUT;

PRINT @new\_docSurg;

--3 Procedure to change NULL specialization to 'Undefined'

DROP PROCEDURE NullUndef;

CREATE PROCEDURE NullUndef

AS

BEGIN

DECLARE @numspec INTEGER;

DECLARE @idSp INTEGER;

DECLARE specCursor CURSOR LOCAL FOR

SELECT idSpec

FROM SPEC

WHERE Specialization IS NULL;

BEGIN

select @numspec = count(idSpec)

FROM SPEC

WHERE Specialization is NULL;

if @numspec = 0

BEGIN

RAISERROR ('No specializations',16,1);

end

ELSE

OPEN specCursor;

FETCH specCursor INTO @idSp;

UPDATE SPEC

set Specialization = 'Undefined'

where idSpec = @idSp;

close specCursor;

deallocate specCursor;

RETURN @numspec;

end

end;

DECLARE @x INT;

EXECUTE @x = NullUndef;

PRINT CAST(@x AS VARCHAR(50)) + ' NULL values were changed to Undefined';

select \* from SPEC;

INSERT INTO SPEC VALUES(03, NULL);