Minh Ngo

CS157A

SJSU ID: 010980398

HW3

Q1.

a.

CREATE TABLE Movies (

title CHAR(100),

year INT,

length INT,

genre CHAR(10),

studioName CHAR(30),

producerC# INT,

PRIMARY KEY (title, year),

FOREIGN KEY (producerC#) REFERENCES MovieExec(cert#)

);

b.

CREATE TABLE Movies (

title CHAR(100),

year INT,

length INT,

genre CHAR(10),

studioName CHAR(30),

producerC# INT REFERENCES MovieExec(cert#)

ON DELETE SET NULL

ON UPDATE SET NULL,

PRIMARY KEY (title, year)

);

c.

CREATE TABLE Movies (

title CHAR(100),

year INT,

length INT,

genre CHAR(10),

studioName CHAR(30),

producerC# INT REFERENCES MovieExec(cert#)

ON DELETE CASCADE

ON UPDATE CASCADE,

PRIMARY KEY (title, year)

);

d.

CREATE TABLE StarsIn (

movieTitle CHAR(100) REFERENCES Movie(title),

movieYear INT,

starName CHAR(30),

PRIMARY KEY (movieTItle, movieYear, starName)

);

Q2 (7.4.1)

a.

No manufacturer of PC’s may also make laptops.

CREATE ASSERTION CHECK

(NOT EXISTS

(

(SELECT maker FROM Product NATURAL JOIN PC)

INTERSECT

(SELECT maker FROM Product NATURAL JOIN Laptop)

)

);

b.

A manufacturer of a PC must also make a laptop with at least as great a processor speed?

CREATE ASSERTION LaptopAtLeastAsFast CHECK

(NOT EXISTS

(SELECT \*

FROM PC P, Product Prod1

WHERE P.model = Prod1.model AND NOT EXISTS

(SELECT \*

FROM Laptop L, Product Prod2

WHERE L.model = Prod2.model AND Prod1.maker = Prod2.maker

AND L.speed >= P.speed))

);

c.

If a laptop has larger main memory than a PC, then the laptop must also have a higher price than the PC?

CREATE ASSERTION LaptopMoreMemory CHECK

(NOT EXISTS

(SELECT \*

FROM PC P, Laptop L

WHERE P.price < L.price AND P.ram < L.ram)

);

d.

If a relation Product mentions a model and its type, then this model must appear in the relation appropriate to that type?

CREATE ASSERTION ProductRelation CHECK

(

(SELECT type

FROM Product

WHERE type = ‘pc’ AND model IN (SELECT model FROM PC))

OR (type = ‘laptop’ AND model IN (SELECT model FROM Laptop))

OR (type = ‘printer’ AND model IN (SELECT model FROM Printer))

);

Q3. (7.5.2)

a.

When updating the price of a PC, check that there is no lower priced PC with the same speed.

CREATE TRIGGER LowPricePCTrigger

AFTER UPDATE OF price ON PC

REFERENCING

OLD ROW AS OldRow,

OLD TABLE AS OldStuff,

NEW ROW AS NewRow,

NEW TABLE AS NewStuff

FOR EACH ROW

WHEN (NewRow.price < ALL

(SELECT PC.price FROM PC

WHERE PC.speed = NewRow.speed))

BEGIN

DELETE FROM PC

WHERE (model, speed, ram, hd, price) IN NewStuff;

INSERT INTO PC

(SELECT \* FROM OldStuff);

END;

b.

When inserting a new printer, check that the model number exists in Product.

CREATE ASSERTION modelPresent

AFTER INSERT ON Printer

CHECK (EXISTS

(Select Product.model

FROM Product

WHERE model = Product.model));

END;

c.

When making any modification to the Laptop relation, check that the average price of laptops for each manufacturer is at least $1500.

CREATE TRIGGER AvgPriceTrigger

AFTER UPDATE OF price ON Laptop

REFERENCING

OLD TABLE AS OldStuff,

NEW TABLE AS NewStuff

FOR EACH STATEMENT

WHEN (1500 > (SELECT AVG(price) FROM Laptop))

BEGIN

DELETE FROM Laptop

WHERE (model, speed, ram, hd, screen, price) IN NewStuff;

INSERT INTO Laptop

(SELECT \* FROM OldStuff);

END;

d.

When updating the RAM or Hard Disk of any PC check that the updated PC has at least 100 times as much hard disk as RAM?

CREATE TRIGGER HardDiskTrigger

AFTER UPDATE OF hd, ram ON PC

REFERENCING

OLD ROW AS OldRow,

OLD TABLE AS OldStuff,

NEW ROW AS NewRow,

NEW TABLE AS NewStuff

FOR EACH ROW

WHEN (NewRow.hd < NewRow.ram \* 100)

BEGIN

DELETE FROM PC

WHERE (model, speed, ram, hd, price) IN NewStuff;

INSERT INTO PC

(SELECT \* FROM OldStuff);

END;

Q4. (8.1.1)

a) A view RichExec giving the name, address, certificate number, and net worth of all executives with a net worth of at least $10,000,000?

CREATE VIEW RichExec AS

SELECT \*

FROM MovieExec

WHERE netWorth >= 10000000;

b) A view StudioPress giving the name, address, and certificate number of all executives who are studio presidents?

CREATE VIEW StudioPres (name, address, cert#) AS

SELECT MovieExec.name, MovieExec.address, MovieExec.cert#

FROM MovieExec, Studio

WHERE MovieExec.cert# = Studio.presC#;

c) A view ExecutiveStar giving the name, address, gender, birth date, certificate number, and net worth of all individuals who are both executives and stars?

CREATE VIEW ExecutiveStar (name, address, gender, birthdate, cert#, netWorth) AS

SELECT star.name, star.address, star.gender, star.birthdate, exec.cert#, exec.netWorth

FROM MovieStar star, MovieExec exec

WHERE star.name = exec.name AND star.address = exec.address;

Q5. (8.2.3)

a)  Is this view updatable?

No, the view is not updatable since it is constructed from two different relations.

b)  Write an instead-of trigger to handle an insertion into this view.

CREATE TRIGGER NewPCInsert

INSTEAD OF INSERT ON NewPC

REFERENCING NEW ROW AS NewRow

FOR EACH ROW

(INSERT INTO Product VALUES(NewRow.maker, NewRow.model, ‘pc’))

(INSERT INTO PC VALUES(NewRow.model, NewRow.speed, NewRow.ram, NewRow.hd, NewRow.price));

c)  Write an instead-of trigger to handle an update of the price.

CREATE TRIGGER NewPCUpdate

INSTEAD OF UPDATE ON NewPC

REFERENCING NEW ROW AS NewRow

FOR EACH ROW

UPDATE PC SET price = NewPC.price where model = NewPC.model;

d)  Write an instead-of trigger to handle a deletion of a specified tuple from this view.

CREATE TRIGGER NewPCDelete

INSTEAD OF DELETE ON NeePC

REFERENCING OLD ROW AS OldRow

FOR EACH ROW

(DELETE FROM Product WHERE model = OldRow.model)

(DELETE FROM PC where model = OldRow.model);