

Name: Giang Tran

Homework #1

2.2.1

a) ~~Attributes~~ Attributes of Accounts: acctNo, type, Balance.

Attributes of Customers: firstName, lastName, idNo, account.

b) Tuples of Accounts: (12345, savings, 12000)

(23456, checking, 1000)

(34567, savings, 25)

Tuples of Customers: (Robbie, Banks, 901-222, 12345)

(Lena, Hand, 805-333, 12345)

(Lena, Hand, 805-333, 23456)

c) Components of first tuple: 12345, savings and 12000 of Accounts.

Components of second tuple: Lena, Hand, 805-333, 12345 of Customers.

d) Relation schema for Accounts

Accounts (acctNo, type, balance)

Relation schema for Customers:

Customers (firstName, lastName, idNo, account)

e) Database schema:

Accounts (

acctNo: ~~integer~~ integer,

type: string,

balance: ~~integer~~ float

)

Customers (

firstName: string,

lastName: string,

idNo: ~~integer~~ string,

account: ~~integer~~ string,

Suitable domains for attributes:

f) acctNo: integer

type: string

balance: integer float

firstName: string

lastName: string

idNo: integer string

accounts: integer

g) Relation Accounts:

type	acctNo	balance
savings	12345	12000
checking	23456	1000
savings	34567	25

Relation Customers:

idNo	account	firstName	lastName
901-222	12345	Robbie	Banks
805-333	23456	Lena	Hand
805-333	12345	Lena	Hand

2.3.1

a) CREATE TABLE Product (

maker VARCHAR(30)

model INT, PRIMARY KEY,

type VARCHAR(20)

)

b) CREATE TABLE PC (

model INT, PRIMARY KEY,

Speed FLOAT,

ram FLOAT,

hd FLOAT,

price INT

);

c) CREATE TABLE ~~Printer~~ Laptop (

model INT, PRIMARY KEY,

speed FLOAT,

ram FLOAT,

hd FLOAT,

screen FLOAT,

price INT

);

d) CREATE TABLE ~~Printer~~ (

model INT, PRIMARY KEY,

color BOOLEAN,

* type VARCHAR(20),

price INT

);

e) ALTER TABLE ~~Printer~~ DROP color;

f) ALTER TABLE Laptop ADD od VARCHAR(20) DEFAULT 'none'

Name: Giang Tran

2/2

Homework #3

2.4.3

Classes (class, type, country, numGuns, bore, displacement)

Ships (name, class, launched)

Battles (name, date)

Outcomes (ship, battle, result)

a) $R_1 := \Pi_{\text{class, country}} (\text{bore} \geq 16 \text{ (Classes)})$

Result:

class	country
Iowa	USA
North Carolina	USA
Yamato	Japan

b) $R_1 := \Pi_{\text{name}} (\text{launched} < 1921 \text{ (Ships)})$

Result:

name

Haruna

Hiei

Kirishima

Kongo

Ramillies

Renown

Repulse

Resolution

Revenge

Royal Oak

Royal Sovereign

Tennessee

G.T

c) $R_1 := \Pi_{\text{ship}} (O_{\text{battle}} = \text{'Denmark Strait'} \wedge O_{\text{result}} = \text{'sunk'})$ (Outcomes)

ship

Bismarck

Hood

d) $R_1 := \Pi_{\text{name}} (O_{\text{displacement}} > 35000)$ (Ships \bowtie Classes)

Result:

name

Iowa

Missouri

Musashi

New Jersey

North Carolina

Washington

Wisconsin

Yamato

e) $R_1 := \rho_{R_1(\text{name}, \text{battle}, \text{result})} (O_{\text{Outcomes}})$

$R_2 := \Pi_{\text{name}, \text{displacement}, \text{numGuns}} (O_{\text{battle}} = \text{'Guadalcanal'})$ ($R_1 \bowtie \text{Ships} \bowtie \text{Classes}$)

Result:

<u>name</u>	<u>displacement</u>	<u>numGuns</u>
Kirishima	32000	8
South Dakota		
Washington	37000	9

f) $R_3 := P_{R_1 \text{ (capital ship)}}(\text{Name (Ships)}) \bowtie P_{R_2 \text{ (capital ship)}}(\text{Ship (Outcomes)})$

$R_3 := \text{capital ship}(R_3) \bowtie_{R_1 \text{ (capital ship)}}(R_3)$

Result:

$R_1 \text{ (capital ship)}$

Arizona

Bismarck

California

Duke of York

Fuso

Hood

King George V

Kirishima

Prince of Wales

Rodney

Scharnhorst

South Dakota

Tenes Tennessee

Washington

West Virginia

Yamashiro

Haruna

Hiei

Iowa

Kongo

Missouri

Musashi

New Jersey

North Carolina

Ramillies

Renown

Repulse

Resolution

Revenge

Royal Oak

Royal Sovereign

Wisconsin

Yamato

G.T

g) $R_1 := \pi_{R_1}(\pi_{name, class}(Ships))$

$R_2 := \pi_{R_2}(\pi_{name, class}(Ships))$

$R_3 := \pi_{class}(Classes) - \pi_{class}(R_1 \bowtie_{(R_1.class = R_2.class \wedge R_1.name \neq R_2.name)} R_2)$

Result: class
Bismarck

h) $R_1 := \pi_{country}(\sigma_{type = 'BB'}(Classes)) \cup \pi_{country}(\sigma_{type = 'BC'}(Classes))$

Result: country
Japan
Gt. Britain

i) $R_1 := \pi_{R_1(Ship, battle, result, date)}(\text{Battles} \bowtie_{(name = battle)} \text{Outcomes})$

$R_2 := \pi_{R_2(Ship, battle, result, date)}(\text{Battles} \bowtie_{(name = battle)} \text{Outcomes})$

$R_3 := \pi_{Ship}(R_1 \bowtie_{(R_1.Ship = R_2.Ship \wedge R_1.Result = 'damaged' \wedge R_1.Date < R_2.Date)} R_2)$

Result: Ship
Prince of Wales
South Dakota

Name: Giang Tran

Homework #5

6.4.6

Product (maker, model, type)

PC (model, speed, ram, hd, price)

Laptop (model, speed, ram, hd, screen, price)

Printer (model, color, type, price)

a) select avg (speed) from PC;

b) select avg (speed) from Laptop where price > 1000;

c) select avg (price) from PC, Product where PC.model = Product.model AND Product.maker = 'A';

d) select avg (price) from (select avg (price) from PC, Product where PC.model = Product.model AND maker = 'D')

UNION

select avg (price) from Laptop, Product where Laptop.model = Product.model AND maker = 'D');

e) select speed, avg (price) from PC group by speed;

f) select maker, avg (screen) from Laptop L, Product P where L.model = P.model group by maker;

g) select maker from Product P, PC where P.model = PC.model group by maker having count (distinct (P.model)) >= 3;

h) select maker from Product P, PC where P.model = PC.model AND price >= ALL (select price from PC) group by maker;

i) select speed, avg (price) from PC where speed > 2.0 group by speed;

G.T

j) select avg(hd) from PC, Product P where
PC.model = P.model AND maker IN (select
maker from P, Printer Pr where P.model = Pr.model);

Name: Giang Tran

2/2

Homework #6

6.5.1

Product (maker, model, type)

PC (model, speed, ram, hd, price)

Laptop (model, speed, ram, hd, screen, price)

Printer (model, color, type, price)

a) insert into Product values ('c', '1100', ~~'pc'~~ 'pc');

insert into PC values ('1100', 3.2, 1024, 180, 2499);

b) insert into Product (maker, model, type)

(select maker, model + 1100, 'laptop' from Product
where type = 'pc');

insert into Laptop (model, speed, ram, hd, screen, price)

(select model + 1100, speed, ram, hd, 17, price + 500 from
PC);c) ~~(*)~~ delete from PC where hd < 100;d) delete from Laptop where model IN (select model from
Product where maker IN (select NOT IN (select maker
from Product where type = 'printer')));

distinct

e) update Product set maker = 'A' where maker = 'B';

f) Update PC set ram = ram * 2, hd = hd + 60;

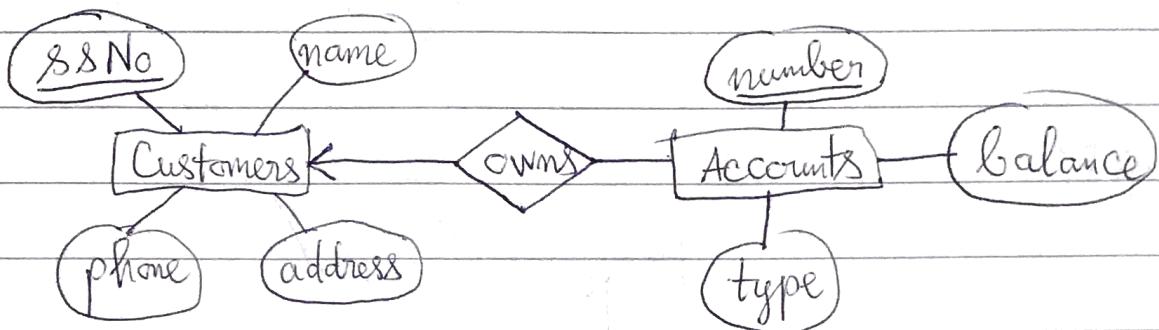
g) ~~(*)~~ update Laptop set screen = screen + 1, price = price - 100
where model IN (select model from Product where maker = 'B');

Name : Giang Tran

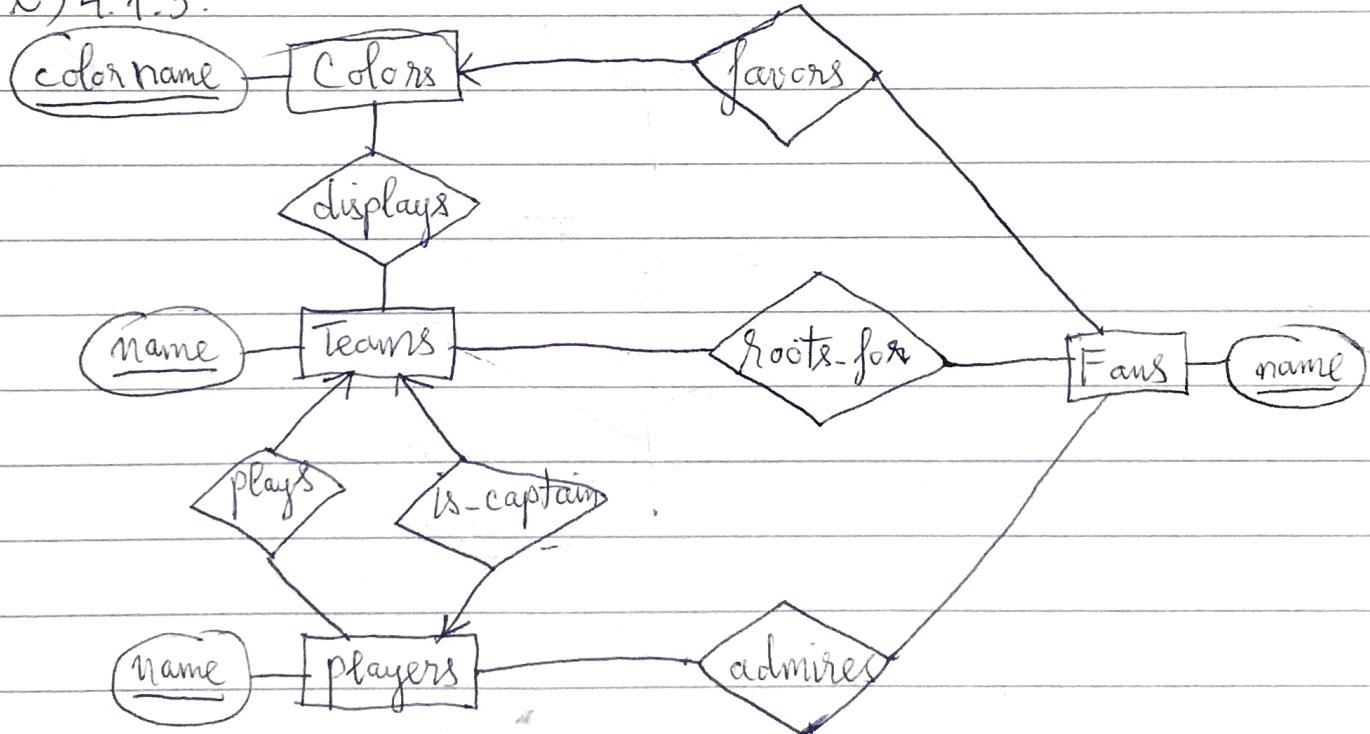
Homework #8

4.3.1

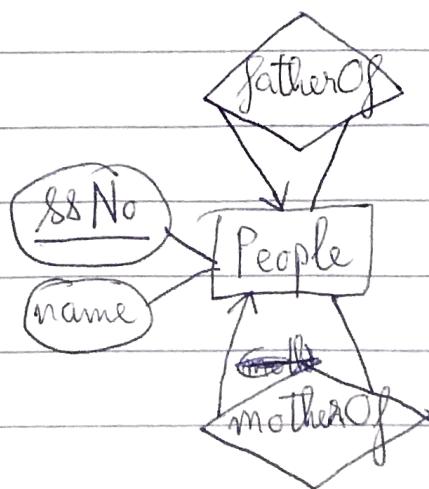
a) 4.1.1:



b) 4.1.3:



c) 4.1.6:



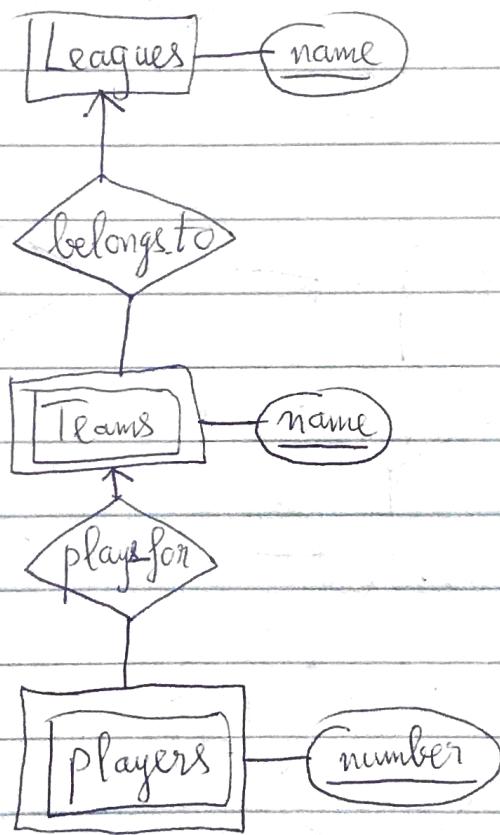
G.T

4.4.4

a)



b)



Name: Giang Tran

2/2

Homework #9

7.1.3

Product table has model as primary key foreign
PC table has model as primary key and program key
Laptop table has model as primary key and foreign key
Printer table has model as primary key and foreign key

CREATE TABLE Product (

maker char(255) char(20),

model int, PRIMARY KEY,

type char(20)

);

CREATE TABLE PC (

model int, PRIMARY

speed real,

ram int,

hd int

price int,

FOREIGN KEY (model) REFERENCES Product (model)

);

G.T

```
CREATE TABLE Laptop(
    model int,
    Speed real,
    ram int,
    hd int,
    screen real,
    price int,
    FOREIGN KEY (model) REFERENCES Product (model)
);
```

```
CREATE TABLE Printer(
    model int,
    color boolean,
    type char(20),
    price int,
    FOREIGN KEY (model) REFERENCES Product (model)
);
```

7.2.2

a) CREATE TABLE Laptop(
 model int PRIMARY KEY,
 speed real CHECK (speed >= 2.0),
 ram int,
 hd int,
 screen real,
 price ~~real~~ int
);

b) CREATE TABLE Printer(

model int PRIMARY KEY,

color boolean,

~~(type)~~ type char(20) CHECK (type IN ('laser', 'ink-jet',
'bubble-jet')),

price int,

~~(model)~~ FOREIGN KEY (model) REFERENCES Product(model)

);

c) CREATE TABLE Product(

maker char(20),

model int PRIMARY KEY,

type char(20) CHECK (type IN ('pc', 'laptop', 'printer'))

);

(d)

d) CREATE TABLE Product(

maker char(20),

model ~~(char)~~ int CHECK (model IN (select model from PC))

UNION (select model from Laptop)

UNION (select model from Printer)),

type char(20)

);

G.T

7.4.1

a) CREATE ASSERTION CHECK

(NOT EXISTS

((select maker from Product JOIN PC)

INTERSECT

(select maker from Product JOIN Laptop))

);

7.5.2

b) CREATE TRIGGER NewPrinterTrigger

AFTER INSERT ON Printer

REFERENCING NEW ROW AS NewRow,

NEW TABLE AS NewStuff

FOR EACH ROW

WHEN (NOT EXISTS (select * from Product where

Product.model = NewRow.model))

DELETE FROM Printer

WHERE (model, color, type, price) IN NewStuff;