

Database- design and programming Exercises

Kristoffer Klokke

2022

Contents

6	Week	6
6.1	Indicate the following according to figure 1.	6
6.1.a	The attributes of each realtion	6
6.1.b	The tuples of each realtion	6
6.1.c	The components of one tube of each realtion	7
6.1.d	The relation schema of each realtion	7
6.1.e	The database schema	7
6.1.f	A suitable domain of each attribute	7
6.1.g	Another equivalent way to present each relation.	7
6.2	In a table with the following attributes which are valid example of keys	8
6.3	How many ways can relation be represented if it has:	8
6.3.a	Four attributes and five tuples	8
6.3.b	n attributes and m tuples	8
6.4	Write a database schema of the following relations	8
6.4.a	Write a schema for <i>Product</i>	8
6.4.b	Write a schema for <i>PC</i>	8
6.4.c	Write a schema for <i>Printer</i>	9
6.4.d	Write an alternation for Printer and delete the attribute color	9
6.4.e	Add an <i>od</i> attribute for PC, which defaults to none an otherwise can be cd or dvd	9
8	Week	9
8.1	Working with linear notation	9
8.1.a	PC models which have speed of at least 3.00?	9
8.1.b	PC manufacturers which makes PC with a hdd with at least 100GB	9
8.1.c	Find model and price of all products made by manufacturer <i>B</i>	10
8.1.d	Find model numbers of all color laser printers	10
8.1.e	Find manufactures that sell Laptops but not PC	10
8.1.f	Find hd size which accour in two or more PC's	10
8.1.g	Find PC models which have the same speed and RAM, a pair should only be listed once	10
8.1.h	Find PC models which have the same speed and RAM, a pair should only be listed once	11
8.1.i	Find manufactore(s) of the fastest PC or laptop	11
8.1.j	Find manufactores who sellf at lest three PC's	11

8.1.k	Find manufactores who sell atleast 3 different speed PC's	11
8.1.l	Find manufactores who sell exactly three PC's	12
8.2	In the following data, what is the result of $\pi_{speed}(PC)$ when treated as a bag and set	13
9	Week	14
9.1	In the query SELECT A B is b an attribute or alias	14
9.2	Write the following queries based on the following tables	14
9.2.a	Find the address of MGM studios	14
9.2.b	Find Sandra Bullock's birthday	14
9.2.c	Find all the stars that appeared either in a movie made in 1980 or a movie with Love in the title	14
9.2.d	Find all executive worth at least \$10,000,000	14
9.2.e	Find all the stars who either are male or live in Malibu	14
9.2.f	Who were the male stars in <i>Titanic</i>	15
9.2.g	Which stars appeared in movies produced by MGM in 1995	15
9.2.h	Who is the presiden of MGM Studios	15
9.2.i	Which movies are longer than <i>GonewiththeWind</i>	15
9.2.j	Which executive are worth more than Merv Griffin	15
9.3	Describe possible values for a and b in the following conditions to be true	15
9.3.a	a = 10 OR b = 20	15
9.3.b	a = 10 AND b = 20	15
9.3.c	a < 10 OR a >= 10	15
9.3.d	a = b	16
9.3.e	a <= b	16
9.4	Write the following queries based on the following tables	16
9.4.a	Find the makers of PC's with a speed of at least 3.0	16
9.4.b	Find the printers with the highest price	16
9.4.c	Find the laptops whose speed is slower than that of any PC	16
9.4.d	Find the model number of the item (PC, Laptop, Printer) with the highest price	16
9.4.e	Find the maker of the color printer with lowest price	17
10	Week	17
10.1	Write SQL queries for the following database	17
10.1.a	What PC models have a speed greater than 3	17
10.1.b	Find model and price for all products made by manufacturer B	17

10.1.c	Find manufactureres of at least two different computers with speed grather than 2.8	17
10.1.d	Find the average speed of PC's	17
10.1.e	Find the average speed of laptops costing over \$1000	18
10.1.f	Find the average price of PC's and Laptops made by manufacturer "D"	18
10.1.g	Find for each different speed, the average price of a PC	18
10.1.h	Find for each manufacturer the average screen size of its laptops	18
10.1.i	Find the manufacturers that make at least three different model of PC	18
10.1.j	Find for each manufacturer who selss PC's the maximum price of a PC.	18
10.1.k	Find for each speed of PC above 2.0 the average price	18
11	Week	18
11.1	What are the expexted FD's in the following database and what key would it have	18
11.2	Consider the relation with schema $R(A, B, C, D)$ and FD's $AB \rightarrow C, C \rightarrow D, D \rightarrow A$	19
11.2.a	What are all the nontrivial FD's that follow from the given FD's? You should restrict yourself to FD's with single attributes on the righth side	19
11.2.b	What are all the keys of R	19
11.2.c	What are all the superkeys for R that is not keys?	19
11.3	Find BCNF violations and decompose the schema	20
11.3.a	$R(A, B, C, D)$ with $AB \rightarrow C, C \rightarrow D$, and $D \rightarrow A$	20
11.3.b	$R(A, B, C, D, E)$ and $AB \rightarrow C, DE \rightarrow C, B \rightarrow D$	20
11.4	Perform the chase method on the relations $R(A, B, C), R(B, C, D), R(A, C, E)$ using the given FD's	20
11.4.a	$B \rightarrow E$ and $CE \rightarrow A$	20
11.4.b	$A \rightarrow D, D \rightarrow E$ and $B \rightarrow D$	20
11.5	The following exercise is on the relation $Courses(C, T, H, R, S, G)$	21
11.5.a	What are all the keys for $Courses$	21
11.5.b	Verify that the given FD's are their own minimal basis	21
11.5.c	Make the relation into a 3NF and check if any BCNF violation accour	21
12	Week	22
12.1	Create a ER diagram from the given information	22
12.2	Create a ER diagram from the given information	22

12.3	Create a ER diagram from the given information	22
12.4	Create a relation schema from the given diagram	23
12.5	Create a relation using the different methods, from the given diagram	23
12.5.a	Straigt E/R model	24
12.5.b	The object oriented	24
12.5.c	The null method	24
13	Week	24
13.1	Create the constraint for the following database	24
13.1.a	The producer of a movie must be someone mentioned in MovieExec. Modifications to MovieExec that violate this constraint are rejected.	24
13.1.b	Repeat (a), but violations result in the producerC# in Movie being set to NULL	24
13.1.c	Repeat (a), but violations result in the deletion or up- date of the offending Movie tuple	25
13.1.d	A movie that appears in StarsIn must also appear in Movie. Handle violations by rejecting the modification.	25
13.1.e	A star appearing in StarsIn must also appear in Movi- eStar. Handle violations by deleting violating tuples.	25
13.1.f	The movies can not contain a movie before 1915	25
13.1.g	The movies can not contain a movie shorter than 60 and longer than 250	25
13.1.h	The movies can only be from Disney, Fox, MGM, or Paramount.	25
13.2	Write the following trigger analoge for insert and delete	26

6 Week

6.1 Indicate the following according to figure 1.

<i>acctNo</i>	<i>type</i>	<i>balance</i>
12345	savings	12000
23456	checking	1000
34567	savings	25

The relation **Accounts**

<i>firstName</i>	<i>lastName</i>	<i>idNo</i>	<i>account</i>
Robbie	Banks	901-222	12345
Lena	Hand	805-333	12345
Lena	Hand	805-333	23456

The relation **Customers**

Figure 1: Two relations of a banking database

6.1.a The attributes of each relation

Accounts: *acctNo*, *type*, *balance*

Customers: *firstName*, *lastName*, *idNo*, *account*

6.1.b The tuples of each relation

- 12345, *savings*, 12000
- 23456, *checking*, 1000
- 34567, *savings*, 25

- *Robbie, Banks*, 901 – 222, 12345
- *Lena, Hand*, 805 – 333, 12345
- *Lena, Hand*, 805 – 333, 23456

6.1.c The components of one tuple of each relation

12000

Banks

6.1.d The relation schema of each relation

Accounts(*acctNo*, *type*, *balance*)

Customers(*firstName*, *lastName*, *idNo*, *account*)

6.1.e The database schema

Accounts, *Customers*

6.1.f A suitable domain of each attribute

- *acctNo* - *INT*
- *type* - *VARCHAR*[20]
- *balance* - *INT*
- *firstName* - *VARCHAR*[20]
- *lastName* - *VARCHAR*[20]
- *idNo* - *CHAR*[7]
- *account* - *INT*

6.1.g Another equivalent way to present each relation.

The attributes could simply just be in a different order.

6.2 In a table with the following attributes which are valid example of keys

title, year, length, genre, studioName, producerC#

- title, year
- title, year, studioName
- title, length
- length, genre, studioName, year

6.3 How many ways can relation be represented if it has:

6.3.a Four attributes and five tuples

$$4! \cdot 5! = 2880$$

6.3.b n attributes and m tuples

$$n! \cdot m!$$

6.4 Write a database schema of the following relations

The database schema includes

Product(make, model, type)

PC(model, speed, ramhd, price)

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

Printer(model, color, type, price)

6.4.a Write a schema for *Product*

```
CREATE TABLE Product(VARCHAR[20] maker, INT model, INT type)
```

The type is here an int where 0 is PC, 1 is laptop and 2 is printer. There is no foreign keys due to it being the lookup table for the other relations

6.4.b Write a schema for *PC*

```
CREATE TABLE PC(INT model, FLOAT speed, INT ram, BOOLEAN hd,
FLOAT prize, FOREIGN KEY(Products) REFERENCES Products(model))
```

Here the model is a reference to products, speed is gigahertz of CPU

6.4.c Write a schema for *Printer*

```
CREATE TABLE Printer(INT model, BOOLEAN color, VARCHAR[20]  
type, FLOAT price, FOREIGN KEY(Products) REFERENCES Products(model))
```

6.4.d Write an alternation for *Printer* and delete the attribute *color*

```
ALTER TABKE Printer DROP color
```

6.4.e Add an *od* attribute for *PC*, which defaults to none an otherwise can be cd or dvd

```
ALTER TABLE PC ADD VARCHAR[20] od DEFAULT 'none'
```

8 Week

8.1 Working with linear notation

The following exercises uses the following schema:

Product(*maker*, *model*, *type*)

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

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

Printer(*model*, *color*, *type*, *price*)

8.1.a PC models which have speed of at least 3.00?

$\pi_{model}(\sigma_{speed > 3.00}(PC))$

8.1.b PC manufacturers which makes PC with a hdd with at least 100GB

$\pi_{maker}(Product \bowtie \sigma_{hd \geq 100}(PC))$

8.1.c Find model and price of all products made by manufacturer B

$$\begin{aligned}
man &:= \sigma_{maker=B}(Product) \\
PCModelPrice &:= \pi_{model,price}(man \bowtie PC) \\
LaptopModelPrice &:= \pi_{model,price}(man \bowtie Laptop) \\
PrinterModelPrice &:= \pi_{model,price}(man \bowtie Printer) \\
modPrice &:= PCModelPrice \cup LaptopModelPrice \cup PrinterModelPrice
\end{aligned}$$

8.1.d Find model numbers of all color laser printers

$$\pi_{model}(\sigma_{color=1 \wedge Dtype=laser}(Printer))$$

8.1.e Find manufactures that sell Laptops but not PC

Due to algebra not including a method for group by I have answered in form of SQL queries.

SELECT (SELECT maker FROM LAPTOP NATURAL JOIN Product GROUP BY maker) - (SELECT maker FROM PC NATURAL JOIN Product GROUP BY maker)

8.1.f Find hd size which accour in two or more PC's

$$\begin{aligned}
PC &= \pi_{model,hd}(PC) \\
PC2(model2, hd) &= \pi_{model,hd}(PC) \\
hd &= \pi_{hd}(\sigma_{model \neq model2}(PC \bowtie PC2))
\end{aligned}$$

8.1.g Find PC models which have the same speed and RAM, a pair should only be listed once

$$\begin{aligned}
Model1 &= \pi_{model,speed,RAM}(PC) \\
Model2(model2, speed, RAM) &= \pi_{model,speed,RAM}(PC) \\
models &= \pi_{model,model2}(\sigma_{model \neq model2}(PC \bowtie PC2))
\end{aligned}$$

8.1.h Find PC models which have the same speed and RAM, a pair should only be listed once

$$\begin{aligned}
ModelPC &= \pi_{model} \sigma_{speed \geq 2.8}(PC) \\
ModelLaptop &= \pi_{model} \sigma_{speed \geq 2.8}(Laptop) \\
models &= ModelPC \cup ModelLaptop \\
mans &= \pi_{maker, model}(models \bowtie Product) \\
twoProduct &= \sigma_{maker2=maker \text{ and } model2 \neq model}(\rho_{maker2, model2} mans \times mans)
\end{aligned}$$

8.1.i Find manufacture(s) of the fastest PC or laptop

$$\begin{aligned}
computers &= \pi_{model, speed} Laptop \cup PC \\
computers2(model2, speed2) &= computers \\
slowSpeed &= \pi_{speed}(computers \bowtie_{speed < speed2} computers2) \\
fastSpeed(fast) &= \pi_{speed}(computers) - slowSpeed \\
fastModels &= computers \bowtie_{speed=fast} fastSpeed \\
mans &= \pi_{maker}(fastModels \bowtie Product)
\end{aligned}$$

8.1.j Find manufactores who sell at lest three PC's

$$\begin{aligned}
model &= \rho_{model} PC \\
computers &= model \times model(model2) \times model(model3) \\
models &= \sigma_{model=model2 \text{ and } model2=model3}(computers) \\
mans &= \pi_{maker}(models \bowtie Product)
\end{aligned}$$

8.1.k Find manufactores who sell atleast 3 different speed PC's

$$\begin{aligned}
model &= \rho_{model} PC \\
computers &= model \times model(model2) \times model(model3) \\
models &= \sigma_{model=model2 \text{ and } model2=model3}(computers) \\
mans &= \pi_{maker}((models \bowtie Product)
\end{aligned}$$

8.1.1 Find manufactores who sell exactly three PC's

$$\begin{aligned} model &= \rho_{model} PC \\ computers &= model \times model(model2) \times model(model3) \\ models &= \sigma_{model=model2 \text{ and } model2=model3}(computers) \\ TooManycomputers &= model \times model(model2) \times model(model3) \times model(model4) \\ models &= \sigma_{model=model2 \text{ and } model2=model3}(computers) \\ TooManymodels &= \sigma_{model=model2 \text{ and } model2=model3 \text{ and } model3=model4}(computers) \\ mans &= \pi_{maker}((models \bowtie Product) \end{aligned}$$

8.2 In the following data, what is the result of $\pi_{speed}(PC)$ when treated as a bag and set

model	speed	ram	hd	price
1001	2.66	1024	250	2114
1002	2.10	512	250	995
1003	1.42	512	80	478
1004	2.80	1024	250	649
1005	3.20	512	250	630
1006	3.20	1024	320	1049
1007	2.20	1024	200	510
1008	2.20	2048	250	770
1009	2.00	1024	250	650
1010	2.80	2048	300	770
1011	1.86	2048	160	959
1012	2.80	1024	160	649
1013	3.06	512	80	529

Bag

speed
2.66
2.10
1.42
2.80
3.20
3.20
2.20
2.20
2.00
2.80
1.86
2.80
3.06

Set

speed
2.66
2.10
1.42
2.80
3.20
2.20
2.00
2.80
1.86
3.06

9 Week

9.1 In the query `SELECT A B` is `b` an attribute or alias

`B` will be an alias for it to be an attribute `A` and `B` has to be comma seperated.

9.2 Write the following queries based on the following tables

`Movies(title , year, length, genre, studioName, producerC#)`
`StarsIn(movie Title , movieYear, starName)`
`MovieStar(name, address, gender, birthdate)`
`MovieExec(name, address, cert# , netWorth)`
`Studio(name, address , presC#)`

9.2.a Find the address of MGM studios

```
SELECT address FROM Studio WHERE name = 'MGM'
```

9.2.b Find Sandra Bullock's birthday

```
SELECT birthdate FROM MovieStar WHERE name = 'Sandra Bullock'
```

9.2.c Find all the stars that appeared either in a movie made in 1980 or a movie with Love in the title

```
SELECT Star.name FROM MovieStar Star WHERE Star.name IN (SELECT starName FROM StarsIn, Movies WHERE MovieTitle = title AND (Movies.title LIKE '%Love%' OR Movies.year = 1980))
```

9.2.d Find all executive worth at least \$10,000,000

```
SELECT name FROM MovieExec WHERE netWorth > 10000000
```

9.2.e Find all the stars who either are male or live in Malibu

```
SELECT name FROM MovieStar WHERE gender = 'male' OR address LIKE '%Malibu%'
```

9.2.f Who were the male stars in *Titanic*

```
SELECT name FROM Movies, StarsIn, MovieStar WHERE title = 'Titanic'
AND title = movieTitle AND starName = name AND gender = male
```

9.2.g Which stars appeared in movies produced by MGM in 1995

```
SELECT starName FROM Movies NATURAL JOIN StarsIn WHERE stu-
dioName = 'MGM' AND title = movieTitle
```

9.2.h Who is the president of MGM Studios

```
SELECT MovieExec.name FROM Studio NATURAL JOIN MovieExec WHERE
Studio.name = 'MGM' AND presC# = cert#
```

9.2.i Which movies are longer than *GonewiththeWind*

```
SELECT title FROM Movies WHERE length > (SELECT length FROM
MOVIES WHERE title = 'Gone with the Wind')
```

9.2.j Which executive are worth more than Merv Griffin

```
SELECT name FROM MovieExec WHERE netWorth > (SELECT net-
Worth FROM MovieExec WHERE name = 'Merv Griffin')
```

**9.3 Describe possible values for a and b in the follow-
ing conditions to be true**

9.3.a $a = 10$ OR $b = 20$

a is 10, b can be anything including null, and the otherway around

9.3.b $a = 10$ AND $b = 20$

Both a has to be 10 and b has to be 20

9.3.c $a < 10$ OR $a \geq 10$

One of the statements has to be true where the other can both be UN-
KNOWN by being null or just FALSE by being 11 or higher or just TRUE.

9.3.d a = b

This will only be true if both have the same value, in case of null it will return UNKNOWN

9.3.e a <= b

If b is higher than a or equal to a it will return true. If one them is null it will be UNKNOWN an otherwise it will be false.

9.4 Write the following queries based on the following tables

Product(maker, model, type)
PC(model, speed, ram, hd, price)
Laptop(model, speed, ram, hd, screen, price)
Printer(model, color, type, price)

9.4.a Find the makers of PC's with a speed of at least 3.0

SELECT maker FROM Product WHERE model EXISTS (SELECT model FROM PC WHERE speed > 3.0)

9.4.b Find the printers with the highest price

SELECT model FROM Printer WHERE price = (SELECT MAX(price) FROM Printer)

9.4.c Find the laptops whose speed is slower than that of any PC

SELECT model FROM Laptop WHERE speed < (SELECT MIN(speed) FROM PC)

9.4.d Find the model number of the item (PC, Laptop, Printer) with the highest price

SELECT model FROM
(SELECT model, price FROM PC UNION SELECT model,price FROM Laptop UNION SELECT model,price FROM Printer)
WHERE price = SELECT MAX(price) FROM
(SELECT MAX(price) FROM PC) UNION (SELECT MAX(price) FROM Laptop) UNION (SELECT MAX(price) FROM Printer))

9.4.e Find the maker of the color printer with lowest price

```
SELECT maker FROM Products WHERE model EXISTS (SELECT model
FROM Printer WHERE type=color AND price = (SELECT MIN(price)
FROM Printer))
```

10 Week

10.1 Write SQL queries for the following database

```
Product(maker, model, type)
PC(model, speed, ram, hd, price)
Laptop(model, speed, ram, hd, screen, price)
Printer(model, color, type, price)
```

10.1.a What PC models have a speed greater than 3

```
SELECT * FROM PC WHERE speed > 3
```

10.1.b Find model and price for all products made by manufacturer B

```
SELECT price, model FROM
(SELECT price, model FROM PC) UNION
(SELECT price, model FROM Printer) UNION
(SELECT price, model FROM Laptop)
Where model EXISTS (SELECT model FROM product WHERE maker =
B)
```

10.1.c Find manufactureres of at least two different computers with speed grather than 2.8

```
SELECT maker FROM (SELECT maker FROM Product WHERE model
EXIST
((SELECT models FROM PC) UNION (SELECT models FROM Laptop)
Where speed > 2.8)) WHERE COUNT(maker) >= 2
```

10.1.d Find the average speed of PC's

```
SELECT AVG(speed) FROM PC
```

10.1.e Find the average speed of laptops costing over \$1000

SELECT AVG(speed) FROM Laptops WHERE price > 1000

10.1.f Find the average price of PC's and Laptops made by manufacturer "D"

SELECT AVG(price) FROM (SELECT model, price FROM Laptop) UNION
(SELECT model, price FROM PC) WHERE model EXIST (SELECT model
FROM Product WHERE maker = 'D')

10.1.g Find for each different speed, the average price of a PC

SELECT AVG(price) FROM PC GROUP BY speed

10.1.h Find for each manufacturer the average screen size of its laptops

SELECT maker, AVG(screen) FROM PC NATURAL JOIN Product GROUP
BY maker

10.1.i Find the manufacturers that make at least three different model of PC

SELECT maker FROM product WHERE type='pc' GROUP BY maker
HAVING COUNT(*) > 2

10.1.j Find for each manufacturer who selss PC's the maximum price of a PC.

SELECT MAX(price) FROM PC NATURAL JOIN Product GROUP BY
maker

10.1.k Find for each speed of PC above 2.0 the average price

SELECT AVG(price) FROM PC GROUP BY speed HAVING speed > 2

11 Week

11.1 What are the expexted FD's in the following database and what key would it have

- name

- Social Security number
- street address
- city
- state
- ZIP code
- area code
- phone number

Social Security number \rightarrow name, street address, city, state, ZIP code, area code

phone number \rightarrow name key: Social security number, phone number

11.2 Consider the relation with schema $R(A, B, C, D)$ and FD's $AB \rightarrow C, C \rightarrow D, D \rightarrow A$

11.2.a What are all the nontrivial FD's that follow from the given FD's? You should restrict yourself to FD's with single attributes on the right side

$$AB \rightarrow C$$

$$C \rightarrow D$$

$$D \rightarrow A$$

$$AB \rightarrow D$$

$$C \rightarrow A$$

11.2.b What are all the keys of R

$$AB^+ = \{C, D\}$$

$$C^+ = \{A, D\}$$

$$D^+ = \{A\}$$

$$BC^+ = \{D, A\}$$

$$BD^+ = \{A, C\}$$

$$AB, BC, DB$$

11.2.c What are all the superkeys for R that is not keys?

$$BC, BD, ABC, ABD, ABCD, BCD$$

11.3 Find BCNF violations and decompose the schema

11.3.a $R(A, B, C, D)$ with $AB \rightarrow C$, $C \rightarrow D$, and $D \rightarrow A$

$AB^+ = \{C, D, A\}$ - violation.

$R1 = R - AB^+ + AB = R(A, B)$

$R2 = AB^+ = R2(A, C, D)$

11.3.b $R(A, B, C, D, E)$ and $AB \rightarrow C$, $DE \rightarrow C$, $B \rightarrow D$

Starts with the original relation

$R(A, B, C, D, E)$

The table violates $AB \rightarrow C$

$AB^+ = \{C, D\}$

$R1 = R - AB^+ = R(A, B, E)$

$R2 = AB^+ + AB = R(A, B, C, D)$

But $R2$ violates $B \rightarrow D$

$B^+ = \{D\}$

$R3 = R2 - B^+ = R(A, B, C)$

$R4 = B^+ + B = R(B, D)$

Therefore the new relations are $R1, R3, R4$, since none now violates BCNF.

11.4 Perform the chase method on the relations

$R(A, B, C), R(B, C, D), R(A, C, E)$ using the given FD's

11.4.a $B \rightarrow E$ and $CE \rightarrow A$

A, B, C, D_1, E_1

A_2, B, C, D, E_2

A, B_3, C, D_3, E

$E_1 = E_2 - B \rightarrow E$

$A_2 = A - CE_1 \rightarrow A$

$E_1 = E - CE \rightarrow A$

Therefore on line two is now

A, B, C, D, E

11.4.b $A \rightarrow D, D \rightarrow E$ and $B \rightarrow D$

A, B, C, D_1, E_1

A_2, B, C, D, E_2

A, B_3, C, D_3, E

$D_1 = D - B \rightarrow D$

$$A_2 = A - A \rightarrow D$$

$$D_3 = D - A \rightarrow D$$

$$E_1 = E - D \rightarrow E$$

11.5 The following exercise is on the relation $Courses(C, T, H, R, S, G)$

The relation has the following FD's

$$C \rightarrow T$$

$$HR \rightarrow C$$

$$HT \rightarrow R$$

$$HS \rightarrow R$$

$$CS \rightarrow G$$

11.5.a What are all the keys for $Courses$

The candidate key are:

$$\{HS\}^+ = \{H, S, R, C, T, G\}$$

11.5.b Verify that the given FD's are their own minimal basis

$$C^+ = \{T\}$$

$$HR^+ = \{C, T\}$$

$$HT^+ = \{R, C\}$$

$$HS^+ = \{R, C, T\}$$

$$CS^+ = \{G, T\}$$

As it can be seen no one of the FD's closure result in another FD, therefore making it minimal.

11.5.c Make the relation into a 3NF and check if any BCNF violation accour

$$R1(C, T)$$

$$R2(H, R, C)$$

$$R3(H, T, R)$$

$$R4(H, S, R)$$

$$R5(C, S, G)$$

All if the schemas are in BCNF

12 Week

12.1 Create a ER diagram from the given information

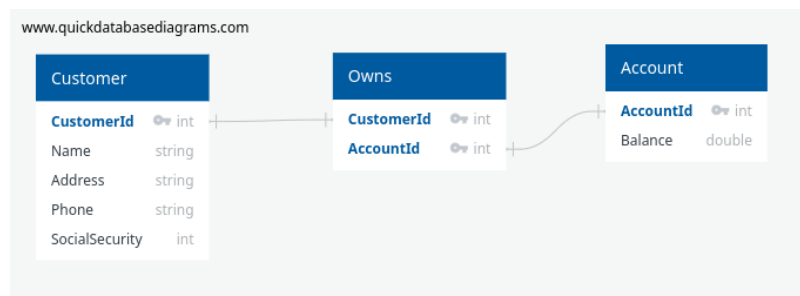


Figure 2: ER diagram describing a bank scenario

12.2 Create a ER diagram from the given information

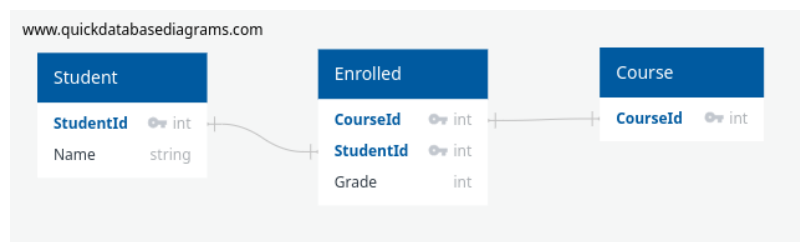


Figure 3: ER diagram describing a course and student enrollment

12.3 Create a ER diagram from the given information

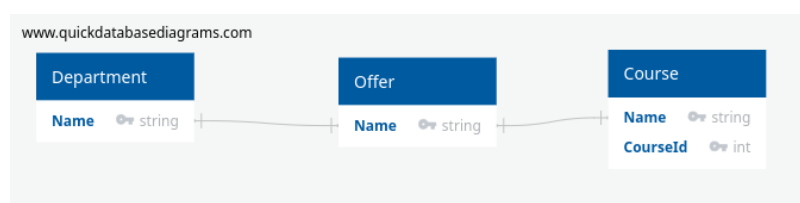
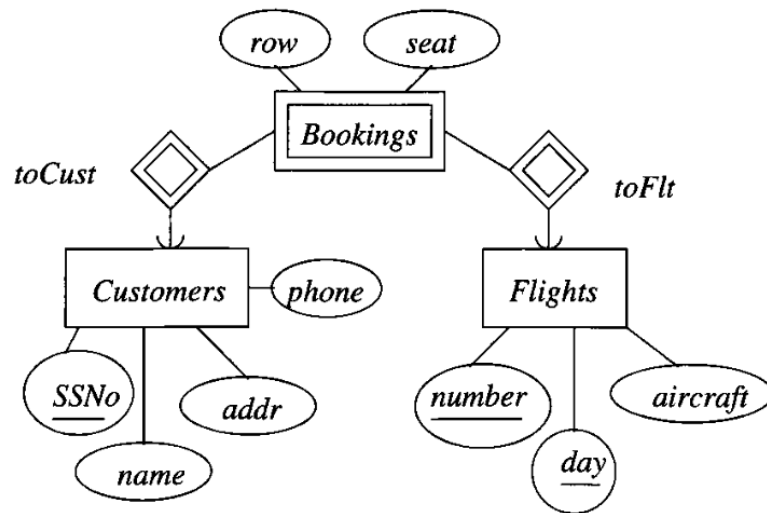


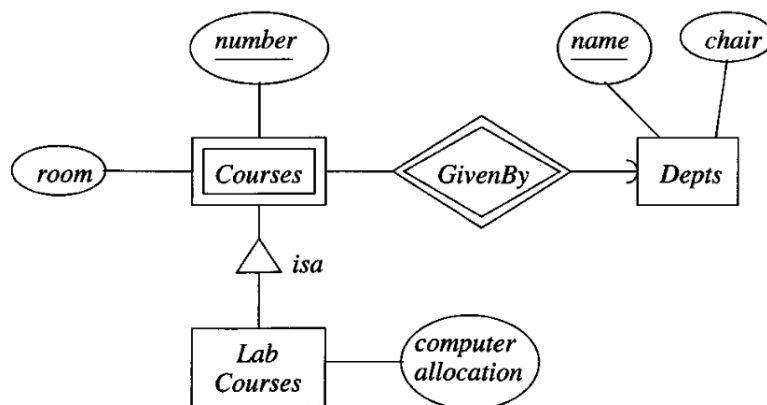
Figure 4: ER diagram describing a department and course situation with a weak key

12.4 Create a relation schema from the given diagram



Customer(SSNO primary,phone,addr,name)
 Flights(number primary,day primary, aircraft)
 Bookings(SSNo primary, number primary, day primary, row, seat)

12.5 Create a relation using the different methods, from the given diagram



12.5.a Straight E/R model

Courses(room,number primary)

Depts(name primary, chair, depName primary)

GivenBy(number primary, name primary, depName primary)

LabCourses(number primary, computerAllo)

12.5.b The object oriented

The same as Depts, GivenBy and Courses but...

LabCourses(number primary, room, ComputerAllo, depName primary)

12.5.c The null method

The same Depts and Given by but...

LabCourses(number primary, room, computerAllo, depName primary)

13 Week

13.1 Create the constraint for the following database

Movies(title , year, length, genre, studioName, producerC#)

StarsIn(movie Title , movieYear, starName)

MovieStar(name, address, gender, birthdate)

MovieExec(name, address, cert# , netWorth)

Studio(name, address , presC#)

13.1.a The producer of a movie must be someone mentioned in MovieExec. Modifications to MovieExec that violate this constraint are rejected.

Movies(title , year, length, genre, studioName, producerC# REFERENCES
MovieExec.cert#)

13.1.b Repeat (a), but violations result in the producerC# in Movie being set to NULL

Movies(title , year, length, genre, studioName, producerC# REFERENCES
MovieExec.cert# DEFERRABLE INITIALLY DEFERRED)

13.1.c Repeat (a), but violations result in the deletion or update of the offending Movie tuple

```
Movies(title , year, length, genre, studioName, producerC# REFERENCES
MovieExec.cert#
ON DELETE CASCADE
ON UPDATE CASCADE )
```

13.1.d A movie that appears in StarsIn must also appear in Movie. Handle violations by rejecting the modification.

```
StarsIn(movieTitle REFERENCES Movies.title , movieYear, starName)
```

13.1.e A star appearing in StarsIn must also appear in MovieStar. Handle violations by deleting violating tuples.

```
StarsIn(movieTitle , movieYear, starName REFERENCES MovieStar.name
ON DELETE CASCADE )
```

13.1.f The movies can not contain a movie before 1915

```
Movies(title , year CHECK (year >= 1915), length, genre, studioName,
producerC#)
```

13.1.g The movies can not contain a movie shorter than 60 and longer than 250

```
Movies(title , year, length CHECK (length > 60 AND 250 > length), genre,
studioName, producerC#)
```

13.1.h The movies can only be from Disney, Fox, MGM, or Paramount.

```
Movies(title , year, length, genre, studioName CHECK IN('Dinsey', 'MGM',
'Paramount'), producerC#)
```

13.2 Write the following trigger analoge for insert and delete

```
01 | CREATE TRIGGER AvgNetWorthTrigger
02 | AFTER UPDATE OF netWorthON MovieExec
03 | REFERENCING
04 |     OLD TABLE AS OldStuff,
05 |     NEW TABLE AS NewStuff
06 | FOR EACH ROW
07 | WHEN (500000 > (SELECT AVG(netWorth) FROM MovieExec))
08 | BEGIN
09 |     DELETE FROM MovieExec
10 |     WHERE (name, address , cert# , netWorth) IN NewStuff
11 |     ;
12 |     INSERT INTO MovieExec
13 |     (SELECT * FROM OldStuff);
14 | END;
```

```
01 | CREATE TRIGGER AvgNetWorthTrigger
02 | AFTER INSERT OF netWorthON MovieExec
03 | REFERENCING NEW ROW AS NewTuple
04 | FOR EACH ROW
05 | WHEN (500000 > (SELECT AVG(netWorth) FROM MovieExec))
06 | BEGIN
07 |     DELETE FROM MovieExec
08 |     WHERE (name, address , cert# , netWorth) IN NewTuple
09 |     ;
10 | END;
```

```
01 | CREATE TRIGGER AvgNetWorthTrigger
02 | AFTER DELETE OF netWorthON MovieExec
03 | REFERENCING OLD ROW AS DeletedTuple
04 | FOR EACH ROW
05 | WHEN (500000 > (SELECT AVG(netWorth) FROM MovieExec))
06 | BEGIN
07 |     INSERT INTO MovieExec
08 |     DeletedTuple
09 | END;
10 |
```