

1. 다음을 간단히 설명하라

- (가) Semijoin
- (나) Foreign Key
- (다) Nonloss Decomposition[2007]
- (라) Dependency Preservation[2007]
- (마) DW(Data warehouse) [1998]
- (바) star schema/ snow schema 비교설명[1998]

2. Health정리를 기술하고 증명하라. 이 정리의 역도 성립하는가를 밝히시오.[1998]

3. 주어진 릴레이션 스키마 R에 대한 FD집합 S가 주어졌을 때 임의의 FD $X \rightarrow Y$ 가 S로부터 유도될 수 있는지를 검사하는 적절한 절차를 설명하라. [1998]

3. 릴레이션스키마 R(A,B,C,D,E)를 가정한다. 이것을 두개의 스키마 R1(A,B,C)와 R2(A,D,E)로 분해한다. R에서 FD는 ($A \rightarrow BC$, $CD \rightarrow E$, $B \rightarrow D$, $E \rightarrow A$)이다. [1998]

(가) 이분해가 nonloss 분해인지 밝혀라.

(나) R을 nonloss이며, dependency-preserve하는 SNP형태로 고치시오.

5. 암스트롱의 공리들이 SOUND 하고 COMPLETE 하다는 것은 무엇을 의미 하는가, 다음의 규칙이 이러한 공리들로부터 추론됨을 증명하라.[2007,1999]

“if $A \rightarrow B$ and $C \rightarrow D$, then $A \cup (C - B) \rightarrow BD$ ”

연합규칙: UNION RULE : $A \rightarrow B$ 이고 $A \rightarrow C$ 이면, $A \rightarrow BC$ 임을 만족

분해규칙: DECOMPOSITION RULE : $A \rightarrow BC$ 이면 $A \rightarrow B$ 와 $A \rightarrow C$ 를 만족

가이행 규칙: PSEUDOTRANSITIVITY RULE : $A \rightarrow B$ 이고 $CB \rightarrow D$ 이면 $AC \rightarrow D$ 를 만족

“SOUND : 이 규칙들은 옳지 않은 함수적 종속을 생성하지 않으므로 건전하다고 한다
또 이 규칙들을 사용하여 주어진 함수적 종속의 집합F에 대해 F^+ 를 모두 찾을 수 있기에
이 규칙들은 완전하다고 말한다.”

“ IF $A \rightarrow B$ AND $C \rightarrow D$, THEN $A \cup (C - B) \rightarrow BD$ “임을 증명 하라

IF $A \rightarrow B$ AND $C \rightarrow D$ THEN $AC \rightarrow BD$ 가 된다.

$AC \rightarrow B$ 와 $AC \rightarrow D$ 가 되며,이—이 사실이 성립하면 $B \subseteq AC$ 이므로 $AB \subseteq AC$ 가 된다.

즉, 이는 $AC \rightarrow BD$ 가 됨을 만족 한다. 그러므로 이 식은 성립한다.

3. $SP(S\#,P\#,QTY)$ 에서 만족되는 모든 FD 들을 기술하라.[1999]

“ $S\# \rightarrow QTY, P\# \rightarrow QTY, S\#P\# \rightarrow QTY$ ”

실제적으로 적용 가능한 FD는 $\{S\#,P\# \} \rightarrow QTY$ 이다

3. $S(S\#, STATUS, CITY)$ 에서 FD 는 $\{S\# \rightarrow CITY, CITY \rightarrow STATUS\}$ 이다. S 를 다음 3 가지 DECOMPOSITION 으로 분할할때 발생하는 문제점을 각각 설명하고 차이점을 비교 설명하라.
[2007,1999]

(가) $SC(S\#,CITY), CS(CITY, STATUS)$

이 두 PROJECTION은 독립적이다 UPDATE가 다른 FD의 영향없이 이루어진다.

UPDATE후 JOIN을 하면 항상 VALID하다. 이는 BCNF의 형태이지만 JOIN시에 추가적인 ROWS들이 생성 된다

(나) $SC(S\#,CITY), SS(S\#, STATUS)$

두개의 PROJECTION은 비 독립적이다. UPDATE가 이루어지려면 FD $CITY \rightarrow STATUS$ 가 VIOLATE되는지 검사후 이루어진다. 이것으로 (나)번이 (가)번 보다는 만족 스럽지 못 하다. 그러나 여전히 어떤 도시에 있는 S#가 없으면 특정한 상태를 갖는 도시를 INSERT 할 수 없다. 3NF의 형태이다

(다) $ST(S\#,STATUS), CS(CITY, STATUS)$

NONLOSS하다. 그러나 VALID 한 DECOMPOSITION이 아니다.

3. $R(A,B,C,D,E,G)$ 의 FD집합은 다음과 같다. [1998,1999]

$\{AB \rightarrow C, C \rightarrow A, BC \rightarrow D, ACD \rightarrow B, D \rightarrow EG, BE \rightarrow C, CG \rightarrow BD, CE \rightarrow AG\}$

(가) $\{B,D\}^+$ 는? $\{C,D,E,G\} \rightarrow$

(나) irreducible cover들을 구하라.

$AB \rightarrow C \quad B \rightarrow C$

$C \rightarrow A \quad C \rightarrow A$

$BC \rightarrow D \quad BC \rightarrow EG, \quad BC \rightarrow E, BC \rightarrow G$ 그럼 $B \rightarrow G, B \rightarrow E$

$ACD \rightarrow B \quad AD \rightarrow B,$

$D \rightarrow EG \quad D \rightarrow E, D \rightarrow G$

$BE \rightarrow C \quad BE \rightarrow A \quad C \rightarrow A$ 이므로 $AB \rightarrow C, \quad B \rightarrow C$ 가 된다.(A가 C에 포함됨)

$CG \rightarrow BD \quad CG \rightarrow B, \quad CG \rightarrow D$ 이면 $G \rightarrow B$

$CE \rightarrow AG \quad CE \rightarrow A, \quad CE \rightarrow G$

$\{ B \rightarrow C, C \rightarrow A, B \rightarrow E, D \rightarrow E, D \rightarrow G \}$

3. 다음 두 FD 집합 S와 T에서 각각의 irreducible cover 와 Key를 구하고, 두집합이 서로 equivalent 한가를 밝히시오.

[2007,1998]

S = { B→A, AC→BD, A→B }

T = { BC→D, B→A, A→B }

[1999]

S = { A→BC, B→C, AB→C, A→B, AC→D }

T = { BC→C, A→D, A→B, A→C }

[1999]

S: A → BC A → B, A → C

B → C B → C

AB → C A → C

A → B A → B

AC → D A → D

{ A → B, A → C, B → C, A → D } KEY { A, B }

T: BC → C

A → D

A → B

A → C

{ A → B, A → C, A → D } KEY { A }

EQUIVALENT 하지 않다

3. R이 튜플집합을 갖는 한 릴레이션이고, T는 이 릴레이션상의 튜플변수이며, f(T)는 자유변수 T를 갖는 WFF라고 가정한다면, EXISTS(Tf(T))는 WFF(가)와 동등하고, FORALL T(f(T))는 WFF (나)와 동등하다. (가)와 (나)를 기술하시오.

4. 다음 두 릴레이션 relvar은 고용인(employee)과 프로그래머(programmer)에 관한 정보를 포함하는 데이터베이스를 나타낸다 :

EMP(EMP#,...JOB..)

PGMR(EMP#, ..., LANG...)

모든 프로그래머는 고용인이다. 그러나 그 반대는 참이 아니다, 참조 다이어그램과 적당한 데이터베이스 정의를 작성하라.

4. 어떤 소매상에 대한 정보를 릴레이션 SIMD(S,I,M,D)으로 나타낸다. 속성 S는 가게(Store)를, 속성 I는 항목(품목:Item)을, 속성 D는 부서(Department)를, 속성M은 관리자(Manager)를 나타낸다. 데이터에 대한 의미적 제약사항을 다음과 같이 가정한다. 각 가게에서는 어떤 품목을 많아야 한 부서에서만 판매하며, 부서별로 한 사람의 관리자가 있다. [2007,1998]

(가) 관련된 속성들 사이의 FD 다이어그램을 그리고, 후보Key를 구하라.

(나) SIMD는 어떤 정규형(normal form)인가? Anomaly 가 발생하는가?

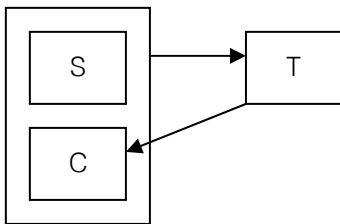
발생한다면 anomaly를 설명하고, 적절한 형태로 고쳐라.

4. 과목(C), 교수(T) 및 학생(S)에 대한 릴레이션 CLASS(C,T,S)를 가정한다. 각 과목별 학생은 한교수에게서 배우며, 각 교수는 단지 한 과목만 가르친다. 또한 각 과목은 여러 교수가 담당할 가르친다.

(가) CLASS에 대한 FD 다이어그램을 작성하라.

(나) CLASS에 어떤 정규형인가? Anomaly가 발생하는가?

발생한다면 그 이유를 설명하고 적절한 형태로 고쳐라. 또다른 문제는 발생하지 않는가?



=> 다이어그램은 위와 같이 됨 3NF임 BCNF는 아님 { S, T }, { S, C }

ANOMALY : UPDATE

만약 물리를 수강하는 'JONE'의 정보를 지우면 우리는 동시에 물리를 가르키는 'BROWN'교수의 정보도 함께 지워지게 된다. 교수는 CANDIDATE KEY가 아니다.

해결 방안 : ST{ S, T }, TC{ T, C } : BCNF의 형태로 가져 간다.

물론 이렇게 모든 문제가 해결 되는 것은 아니다. 여기서 문제점이 존재 하는데 이것은 릴레이션을 BCNF의 부분으로 나누는 것이 독립 부분으로 나누는 것과 충돌이 나타날 수 있다. 왜냐면 이 두개의 PROJECTION은 독립적으로 UPDATE될 수 없기 때문이다.

5. 암스트롱의 공리들이 SOUND 하고 COMPLETE 하다는 것은 무엇을 의미 하는가, 다음의 규칙이 이러한 공리들로부터 추론됨을 증명하라.[2007,1999]

“if $A \rightarrow B$ and $C \rightarrow D$, then $A \cup (C-B) \rightarrow BD$ ”

연합규칙: UNION RULE : $A \rightarrow B$ 이고 $A \rightarrow C$ 이면, $A \rightarrow BC$ 임을 만족

분해규칙: DECOMPOSITION RULE : $A \rightarrow BC$ 이면 $A \rightarrow B$ 와 $A \rightarrow C$ 를 만족

가이행 규칙: PSEUDOTRANSITIVITY RULE : $A \rightarrow B$ 이고 $CB \rightarrow D$ 이면 $AC \rightarrow D$ 를 만족

“SOUND : 이 규칙들은 옳지 않은 함수적 종속을 생성하지 않으므로 건전하다고 한다

또 이 규칙들을 사용하여 주어진 함수적 종속의 집합F에 대해 F^+ 를 모두 찾을 수 있기에 이 규칙들은 완전하다고 말한다.”

“ IF $A \rightarrow B$ AND $C \rightarrow D$, THEN $A \cup (C - B) \rightarrow BD$ “임을 증명 하라

IF $A \rightarrow B$ AND $C \rightarrow D$ THEN $AC \rightarrow BD$ 가 된다.

$AC \rightarrow B$ 와 $AC \rightarrow D$ 가 되며, 이—이 사실이 성립하면 $B \subseteq AC$ 이므로 $AB \subseteq AC$ 가 된다.

즉, 이는 $AC \rightarrow BD$ 가 됨을 만족 한다. 그러므로 이 식은 성립한다.

6. 관계형 대수(Relational Algebra)의 다음 연산에 동등한 SQL 문을 작성하라

(가) EXTEND P ADD (WEIGHT * 454) AS GMWT

```
=> SELECT P.P#, P.WEIGHT*454 AS GMWT
      FROM P;
```

(나) SUMMARIZE SP BY (P#) ADD SUM(QTY) AS TOTGY :

```
=> SELECT SP.P#, SUM(SP.QTY) AS TOTQY
      FROM SP
      GROUP BY SP.P#;
```

Or

```
SELECT SP.P#, (SELECT SUM(SP.QTY) FROM SP
                WHERE SP.P# = P.P#) AS TOTQTY
      FROM P;
```

99. 다음의 공급자-부품 데이터베이스에 대한 관계형스키마를 사용하여 답하시오.
[2007,1999]

S(S#, SNAME, STATUS, CITY)

P(P#, PNAME, COLOR, WEIGHT, CITY)

SP(S#, P#, QTY)

Q1-1: “Get supplier names for suppliers who supply at least one blue part”

Q1-2: “Get supplier numbers for suppliers who supply at least all those parts supplied by supplier S2”

Q2-1: “For each part, get the part number and the maximum quantity supplied of that part.”

Q2-2: “For each part supplied, get the part number and the total shipment quantity”

(가) 관계형 대수 표현식(Relational Algebra Expression)

Q1-1 : $((P \text{ WHERE } \text{COLOR} = \text{COLOR}('BLUE')) \{P\# \} \text{ JOIN } SP) \text{ JOIN } S \{SNAME\}$

Q1-2 : $S \{S\# \} \text{ DIVIDEBY } (SP \text{ WHERE } S\# = S\#('S2')) \{P\# \} \text{ PER } SP \{S\#, P\# \}$

(나) 튜플해석 표현식(tuple calculus expression)과

Q1-1 : $SX.SNAME \text{ WHERE EXISTS } SPX (SX.S\# = SPX.S\# \text{ AND}$

$\text{EXISTS } PX (PX.P\# = SPX.P\#$

$\text{AND } PX.COLOR = \text{COLOR}('BLUE'))$

prenex normal form, in which all quantifiers appear at the front of the WFF

SX.SNAME WHERE EXISTS SPX (EXISTS PX (SX.S# = SPX.S# AND
SPX.P# = PX.P# AND
PX.COLOR = COLOR('BLUE'))

Q1-2 : SX.S# WHERE FORALL SPY (SPY.S# \neq S#('S2') OR
EXISTS SPY (SPY.S# = SX.S# AND SPY.P# = SPX.P#))

Logical implication : IF p THEN q END IF (NOT p) OR q

SX.S# WHERE FORALL SPX (IF SPX.S# = S# ('S2') TEHN EXISTS SPY (SPY.S# = SX.S#
AND SPY.P# = SPX.P#) END IF)

(다) SQL

Q1-1 : SELECT DISTINCT S.SNAME FROM S
WHERE S.S# IN
(SELECT SP.S#
FROM SP
WHERE SP.P# IN
(SELECT P.P#
FROM P
WHERE P.COLOR = 'Red')) ;

(라) 도메인

Q1-1 : NAMEX WHERE EXISTS SX EXISTS PX
(S (S#:SX, SNAME:NAMEX)
AND SP (S#:SX, P#:PX) AND P (P#:PX, COLOR:COLOR('Red')))

Q1-2 : SX WHERE FORALL PX (IF SP (S#:S#('S2'), P#:PX)
THEN SP (S#:SX, P#:PX) END IF)

Q1: 적어도 공급자 S1이 공급하는 모든 부품을 공급하는 공급자의 번호를 구하라

Q2: 각 부품별 부품번호와 총 공급량을 구하라

Q3: 모든 부품을 공급하는 공급자에 대한 공급자의 이름을 구하라

(가) 관계형 대수로 표현 하라

Q1: S {S#} DIVIDED BY (SP WHERE S# = S# ('S1')) {P#} PER SP (S#, P#)

Q2: SUMMARIZE SP PER SP{P#} ADD SUM {QTY} AS TOTQY -- (마)

Q3: ((S {S#} DIVIDED BY P {P#} PER SP {S#,P#}) JOIN S) { SNAME }

(나) 튜플 해석 - TUPLE CALCULUS

RANGEVAR	SX	RANGES	OVER	S;
RANGEVAR	SY	RANGES	OVER	S;
RANGEVAR	SPX	RANGES	OVER	SP;
RANGEVAR	SPY	RANGES	OVER	SP;
RANGEVAR	PX	RANGES	OVER	P;

Q1: SX.S# WHERE FORALL SPY (IF SPX.S# = S# ('S1') THEN
EXIST SPY (SPY.S# = SX.S# AND SPY.P# = SPX.P#) END IF)

SX.S# WHERE FORALL SPY(SPY.S# =W= S#('S1') OR EXIST SPY(SPY.S# =
SX.S#
AND SPY.P# = SPX.P#)

Q2: (PX.P#, SUM(SPX WHERE SPX.P# = PX.P# , QTY) AS TOTQY)

Q3: SX.NAME WHERE FORALL PX (EXIST SPX (SPX.S# = SX.S# AND
SPX.P# = PX.P#))

(다) 도메인 해석

Q1: SX WHERE FORALL PX (IF SP (S#:S# ('S1'), P#:PX)
THEN SP (S#:SX, P#:PX) END IF)

Q2:

Q3: NAMEX WHERE EXIST SX (S (S#.SX, SANME, NAMEX) AND
FORALL (IF P (P#, PX) THEN
SP (S#.SX, P#.PX) END IF)

(라) SQL로 표현하면...

Q1:

SELECT S.S# FROM S
WHERE S.S# IN (SELECT SP.S# FROM SP

```
WHERE SP.P# IN ( SELECT P.P# FROM P
                WHERE P.P# = SP.P# AND P.P# = 'P1' ) );
```

Q2:

```
SELECT SP.P#, ( SELECT SUM(QTY)
                FROM SP
                WHERE P.P# = SP.P#) AS TOTQTY
FROM SP
```

Q3:

```
SELECT DISTINCT S.SNAME
FROM S
WHERE NOT EXIST ( SELECT * FROM P WHERE NOT EXIST
                  ( SELECT * FROM SP
                    WHERE SP.P# = P.P# AND SP.S# = S.S# ) );
```