Chapter 8 Review
(Relational Database Design)

8장에서 우리가 살펴본 것

- 함수의 종속성 (Functional Dependencies)
- 정규화 (Normalization)
  - 3NF
  - BCNF
- 무손실 분해 (Lossless-Join Decomposition)
- 종속성 보존 (Dependency Preservation)

함수의 종속성 (Functional Dependencies)

함수의 종속성(**FD**)의 개념

A → B

- **어떤 속성 A의 값을 알면 다른 속성 B의 값이 유일하게 정해지는 의존 관계**
- **속성 A는 속성 B를 결정함 (즉 A는 결정자, B는 종속자 임)**
- **릴레이션에서 FD는 아래의 공식으로 구함**

$t_1[\alpha] = t_2[\alpha] \Rightarrow t_1[\beta] = t_2[\beta]$ t1, t2에 대해서 속성 α의 값이 서로 같다면, 속성 β의 값도 서로 같다.

$t_1[\alpha] \neq t_2[\alpha] \vee t_1[\beta] = t_2[\beta]$ t1, t2에 대해서 속성 α의 값이 서로 다르다면, 속성 β의 값은 서로 같거나 다르다.

함수의 종속성 법칙 **(Armstrong Axiom)**

위에서 5개까지는 무조건 알아둬야함

주어진 함수의 종속성 F의 **Closure**(폐포)인 **F ⁺**를 구하기 위한 법칙

재귀법칙 (Reflexivity)	: If Y ⊆ X, then X → Y
증가법칙 (Augmentation)	: If X → Y, then XZ → YZ
이행법칙 (Transitivity)	: If X → Y and Y → Z, then X → Z
결합법칙 (Union)	: If X → Y and X → Z, then X → YZ
분해법칙 (Decomposition)	: If X → YZ, then X → Y and X → Z
가이행법칙 (Pseudotransitivity)	: If X → Y and WY → Z, then WX → Z

함수의 종속성 (Functional Dependencies)

문제: **FD**를 이용하여 **A**라는 속성이 릴레이션 **R**에서 **Superkey**임을 또는 **Superkey**가 아님을 보여라 (강의노트 **8.12**)

attribute 페포와 (전용알고리즘으로 구함) 함수종속성 페포 (암스트롱으로구함)를 구하는 것은 다름

K라는 속성이 릴레이션 R의 Superkey이면 K → R라는 함수의 종속성 성립

- R(A,B,C)에서 A가 만약 Superkey라면 A → R, 즉 A→ ABC라는 함수의 종속성 성립

***R* = (*A*, *B*, *C*) *F* = {*A* → *B*, *B* → *C*}** 일 때 **A**가 **Superkey**임을 보여라?

1가지 이상이 나올수 있음

2가지방법 사용 가능
- ✓ 주어진 FD를 이용해서 Armstrong Axiom을 사용하여 보이는 방법
- ✓ A의 Closure(페포) A^+를 구해서 보이는 방법. **<- 알고리즘가지고 풀어야함**

Armstrong Axiom 사용

1. A → B, B → C는 이행 규칙을 사용하게 되면 A → C로 추론가능
2. A → C에 B를 양쪽에 대입하는 증가법칙을 사용하게 되면 AB → BC 임
3. A → B에 A를 양쪽에 대입하는 증가법칙을 사용하게 되면 A → AB이기 때문에 2번의 결과가 이행규칙으로 A → BC가 되는 것을 알 수 있음
4. A → BC에 A를 양쪽에 대입하는 증가법칙을 사용하게 되면 A → ABC 이기 때문에 A는 Superkey임

함수의 종속성 (Functional Dependencies)

문제: 함수의 종속성을 이용하여 **A**라는 속성의 **Closure(**폐포**)**인 **A** $^+$을 모두 구하라 **(**강의노트 **8.31)**

A $^+$라는 것은 릴레이션 R에서 A가 결정자로 사용된 함수의 종속성들의 집합을 의미함

- **Attribute의 폐포를 구하는 알고리즘 이용**

```
result := α;
while (changes to result) do
      for each β → γ in F do
         begin
            if β ⊆ result then  result := result ∪ γ
         end
```

주의: Attribute의 폐포와 함수의 종속성 F의 폐포는 다르게 구함!!

***R* = (*A*, *B*, *C*, *D*) *F* = {*A* → *B*, *AB* → *C*, *BC* → *D*)}** 일 때 **A** $^+$ 을 구하여라**?**

1. *Result* = *A*
2. *Result* = *AB* ($A \rightarrow B$)
3. *Result* = *ABC* ($AB \rightarrow C$)
4. *Result* = *ABCD* ($BC \rightarrow D$)

A $^+$ 는 $A \rightarrow A$

$A \rightarrow AB$

$A \rightarrow ABC$

$A \rightarrow ABCD$

왼쪽은 A의 모든 폐포

A는 릴레이션 R에서 Superkey인가? Yes!!!

→ 즉 폐포를 구하는 알고리즘을 통해서도 Superkey 확인 가능

함수의 종속성 (Functional Dependencies)

배점높은 문제로 비슷한 유형이 나올수 있음

문제: FD를 이용하여 릴레이션 **R**의 **Candidate Key**를 구하고 제시된 **F**의 **Canonical Cover**를 구하여라 (강의노트 **8.32 & 8.34)**

Candidate Key는 유일성과 최소성을 모두 만족하는 속성임

Canonical Cover는 주어진 함수의 종속성에서 중복이 제거된 minimal set임

R = (A, B, C) F = {A → B, A → C, B → C, AC → B, AB → C)} 일 때

Candidate Key와 **F**의 **Canonical Cover**를 구하여라**?**

사실 이것은 이전에 나온 학생수강등록문제를 응용

→ Candidate Key를 구하기 위해서는 함수의 종속성에서 결정자로 사용된 Attribute들의 폐포들을 구해서 먼저 superkey인지 확인해야함

1. A^+, B^+, AC^+, AB^+ 구하기 (A, AB, AC가 superkey임)
2. A, AB, AC 중 Candidate key를 찾아야함 (최소성 만족) **즉 ,A만 후보키임**

   * A가 Candidate Key인가 ?

   : A → B, A → C라는 2개의 FD를 이용해서 결합법칙을 이용하면
   A → BC가 됨 따라서 후보키임

   * AB가 Candidate Key인가 ? **즉, A->C를 통해 AB->C를 만들수 있기 때문에 아님**

   : **<u>AB → C</u>**<u>라는</u> **<u>FD</u>**<u>는</u> **<u>A → C</u>**<u>만 이용해서 표현가능</u> 따라서 후보키 아님

   * AC가 Candidate Key인가 ?

   : **<u>AC → B</u>**<u>라는</u> **<u>FD</u>**<u>는</u> **<u>A → B</u>**<u>만 이용해서 표현가능</u> 따라서 후보키 아님

→ Canonical Cover는 ***{A → B, B → C}***임

유도되지 않는 더이상 건들수없는 최소의 관계

정규화(Normalization)

규준커버나 후보키구하는것중 둘중에하나 나옴

BCNF 개념

릴레이션 R에서 함수의 종속성 $\alpha \rightarrow \beta$가 성립할 때

- $\alpha \rightarrow \beta$ 가 trivial한 관계이거나 (i.e., $\beta \subseteq \alpha$)
- α 가 릴레이션 *R*의 Superkey여 야함

BCNF는 Dependency Preservation(종속성 보존)을 보장하지 않음

무손실 분해는 보존함

3NF 개념

릴레이션 R에서 함수의 종속성 $\alpha \rightarrow \beta$가 성립할 때

- $\alpha \rightarrow \beta$ 가 trivial한 관계이거나 (i.e., $\beta \subseteq \alpha$)
- α 가 릴레이션 *R*의 Superkey이거나
- $\beta - \alpha$ 의 속성 A는 릴레이션 R의 Candidate Key를 포함하고 있어야함

2개중 하나를 잘 작성해야함

3NF는 릴레이션 R의 FD가 비이행적(non-transitive)으로 종속될 때 임

- 이행적 종속인 $\alpha \rightarrow \beta$, $\beta \rightarrow \gamma$ 가 성립할 때 $\alpha \rightarrow \gamma$가 성립되면 3NF가 아님
- 단 *β 가 Superkey가 아니고* $\alpha \rightarrow \beta$, $\beta \rightarrow \gamma$ 를 통해 추론된 $\alpha \rightarrow \gamma$에서 $\gamma \subseteq \alpha$ 이면 3NF가 될 수 있음 **함정이므로 주의**

3NF는 Dependency Preservation(종속성 보존)을 보장 (3번째 조건 때문!!!)

정규화(Normalization)

문제: Simplified Test를 이용하여 릴레이션 R이 BCNF인지 Test하여라 (강의노트 8.42)

Simplified Test는 릴레이션 R의 주어진 FD의 폐포를 구해서 BCNF인지 Test하는 것이 아니라 주어진 FD만 가지고 Test해보는 방법

- 분해된 릴레이션에 대해서는 Simplified Test 적용할 수 없음

***R* = (*A, B, C, D*), *F* = { $A \rightarrow B$, $B \rightarrow CD$} 일 때 R은 BCNF인지 Test하여라**

→ Simplified Test이기 때문에 F의 폐포인 $A \rightarrow CD$는 이용하지 않음

1. 주어진 F에서 trivial한 관계 찾기

   → 만약 없다면 결정자로 사용된 Attribute들의 폐포들을 구해서 superkey인지 확인
2. A^+, B^+ 구하기 (A만 superkey임) → 따라서 BCNF가 아님 **B가 슈퍼키가 아니기때문**

문제: FD를 이용하여 릴레이션 R을 BCNF로 분해하여라 (강의노트 8.17, 8.44~8.46)

이 경우는 결정자가 릴레이션 R에서 Superkey가 아닌 함수의 종속성 이용 (즉 BCNF를 위반하는 FD부터 찾아라!!!)

Superkey 또는 Candidate key가 주어질 수도 있고 안 주어질 수도 있음
→위반되는 FD를 찾기 위해 superkey를 먼저 찾아야할 수도 있음

R = ($\alpha \cup \beta$)와 R =($R - (\beta - \alpha)$) 이용

정규화(Normalization)

R= (*A, B, C, D, E, F, G, H, I, J, K*)이고 **F ={*A* → *BCD, HI* → *J, AEFG* → *HIK*}**
일 때 **BCNF**로 분해하여라 **(Candidate Key**는 **{A,E,F,G}**임**)**

→ 1. 먼저 BCNF를 위반하는 함수의 종속성을 찾음
: Candidate Key가 AEFG라면, ***A* → *BCD, HI* → *J***는 ***BCNF***를 위반하는 **FD**임

2. ***A* → *BCD***를 이용하여 분해
: α = A
β = *BCD*
(α U β) = (A,B,C,D)
(*R* - (β - α)) = (A, *E, F, G, H, I, J, K*) **J 가 결정자가 아니라 BCNF위반**

3. 분해된 2개의 릴레이션이 BCNF를 만족하는지 검사
: (A,B,C,D) ***A* → *BCD***로 인해 **BCNF**
(A, *E, F, G, H, I, J, K*) *HI* → *J* 로 인해 BCNF가 아님 (*HI*가 superkey 아님)

4. (A, *E, F, G, H, I, J, K*) 라는 릴레이션을 ***HI* → *J*** 를 이용하여 분해
(α U β) = (H, I, J)
(*R* - (β - α)) = (A, *E, F, G, H, I, K*)

5. 분해된 2개의 릴레이션이 BCNF를 만족하는지 검사
: (H,I,J) ***HI* → *J***로 인해 **BCNF**
(A, *E, F, G, H, I, K*) ***AEFG* → *HIK***로 인해 BCNF

R1 =(A,B,C,D) R2 = (H,I,J) R3= (A,E,F,G,H,I,K)로 분해 가능

정규화(Normalization)

문제: 릴레이션 R과 FD가 주어졌을 때 릴레이션 R이 3NF임을 보여라
(강의노트 8.48~8.49) **R의 모든 속성이 슈퍼키이면 BCNF, 3NF둘다 만족함**

R= (A, B, C) F={$AB \rightarrow C$, $C \rightarrow B$} 일 때 3NF임을 보여라 **좀더 긴 릴레이션을 문제로 냄**

→ 주어진 F에서 trivial한 관계가 없다면, 결정자로 사용된 Attribute들의 폐포들을 구해서 superkey인지 확인하고 candidate key를 구함

1. AB^+, C^+ 구하기 (AB만 superkey이며, candidate key임)
   → 따라서 **R**은 **BCNF**가 아님
2. 이행적 종속성이 없는지 확인
   → **$AB \rightarrow C$, $C \rightarrow B$** 임으로 **$AB \rightarrow B$**이기 때문에 이행적 종속성 있음
3. 이행적 종속성이 있다면 추론된 FD가 Trivial한지 검사
   → **$AB \rightarrow B$**는 $B \subseteq AB$이기 때문에 Trivial

$\therefore$ **R= (A, B, C)**는 **3NF**임

OR

2. $\beta - \alpha$ 의 속성 A가 릴레이션 R의 Candidate Key를 포함하고 있는지 확인
   → **$AB \rightarrow C$** 는 AB가 Superkey이기때문에 $\beta - \alpha$ 를 하지 않아도 3NF만족
   → **$C \rightarrow B$의** $\beta - \alpha$ 는 B이며, B는 Candidate key에 포함

$\therefore$ **R= (A, B, C)**는 **3NF**임

무손실 분해(Lossless-Join Decomposition)

무손실 분해 개념

릴레이션 R을 R1과 R2로 분해하였을 때 R1과 R2를 다시 Join한 경우 원래 가지고 있던 정보에서 어떠한 데이터 손실도 없는 경우 (즉 릴레이션 R과 동일한 경우)

분해한 R1과 R2는 다음과 같은 FD중 최소한 하나가 F^+ 에 속하면 릴레이션 R에 대한 무손실 분해임

$$\left.\begin{aligned} R_1 \cap R_2 \rightarrow R_1 \\ R_1 \cap R_2 \rightarrow R_2 \end{aligned}\right\}$$

결국 $R_1 \cap R_2$ 가 R_1 혹은 R_2 의 superkey를 구성

BCNF와 3NF로 분해된 릴레이션은 무손실 분해임

문제: 릴레이션 R이 R1과 R2로 분해되었을 때 무손실 분해임을 보여라 (강의노트 8.33 ~ 8.34)

(AB)->(AB)C
AB->AB
AB->C

R= (A, B, C), F={$AB \rightarrow C$, $C \rightarrow B$}이며 **R1=(A, B), R2=(B,C)**로 분해되었을 때 무손실 분해임을 보여라

C->AC
C->BC
C->B
CC->CB
C->BC

$AB \cap BC \rightarrow AB$ 즉 $B \rightarrow AB$

$AB \cap BC \rightarrow BC$ 즉 $B \rightarrow BC$

B는 Superkey가 아니며 F^+ 에 속하지 않음으로 무손실 분해가 아님!!!

R1=(A, C), R2=(B,C)는 ?

종속성 보존(Dependency Preservation)

종속성 보존 개념

릴레이션 R을 R1과 R2로 분해하였을 때, 릴레이션 R에 정의된 FD가 R1과 R2에 대해서 그대로 유지되면 종속성 보존(Dependency Preservation)임

- ✓ 분해된 R1과 R2의 FD들의 폐포가 분해 전 릴레이션 R의 FD의 폐포와 같다면 종속성 보존

$$(F_1 \cup F_2 \cup \ldots \cup F_n)^+ = F^+$$

폐포의 폐포를 구했는데 다르면 종속성보존이 안 됨이거나 한번폐포를 구했는데 만족하지 못하면 종속성만족을 안함

- ✓ 3NF는 종속성 보존을 보장함
- ✓ BCNF는 종속성 보존을 보장하지 않음 (중요!!: 분해를 어떻게 하느냐에 따라 종속성이 보장된 분해가 될 수도 있고 아닐 수도 있음)

순차적으로 풀어가는것이 좋음

- 주어진 FD의 순서대로 BCNF를 위반하는 FD를 이용하여 분해하는 경우
- 주어진 FD의 역순으로 BCNF를 위반하는 FD를 이용하여 분해하는 경우
- 주어진 FD의 폐포 중 BCNF를 위반하는 FD를 이용하여 분해하는 경우

문제: 릴레이션 **R**이 **(<u>BCNF 분해를 통해</u>) R1**과 **R2**로 분해되었을 때 종속성 보존임을 보여라 **(**강의노트 **8.33 ~ 8.34)**

$R = (A, B, C)$ $F = \{A \rightarrow B, A \rightarrow C\}$ Key = $\{A\}$일 때

$R_1 = (A, B)$, $R_2 = (A, C)$으로 분해 된 경우 R_1과 R_2는 종속성 보존인가?

→ 종속성 보존!!! $(A \rightarrow B \cup A \rightarrow C)^+ = A \rightarrow BC$ (*결합법칙*) $F^+ = A \rightarrow BC$

분해법칙으로 A->B와 A->C로 분해해 만족함