

Discussion 05/04

Discussion 8-4

What is a *trivial functional dependency*? List all trivial FDs in the following schema and explain how the properties of trivial FDs hold.

Student(ID, Name, Address)

넘어감

Discussion 8-5

Derive additional FDs from your initial set using each of the three *Armstrong's Axioms*.

Student(ID, Name, Address, Sex, Age, Dept, Dept_office, Dept_Chair, College, Dean, AdvisorID, AdvName, AdvDept)

Dept \rightarrow Dept_office, Dept_Chair, College

College \rightarrow Dean

AdvisorID \rightarrow AdvName, AdvDept

Dept Dept_office \rightarrow Dept (by reflexivity)

College Dept \rightarrow College Dept_Chair (by augmentation)

Dept \rightarrow Dean (by transitivity)

Discussion 8-6

Show that the *Augmentation Rule* in Armstrong's Axioms is sound.

Augmentation Rule: $a \rightarrow b$ 일 때 $ra \rightarrow rb$.

$ra \rightarrow rb$ 에서 reflexivity 에 의해 $ra \rightarrow a$ 가 성립하고, $rb \rightarrow b$ 가 성립한다.

그 다음엔..?

\Rightarrow FD 의 정의에 의해 증명. $t_1(a) = t_2(a) \Rightarrow t_1(b) = t_2(b)$ 를 보여야 함.

Discussion 8-7

Is the schema in BCNF? If not decompose the relation into a set of relation schemas, each of which is in BCNF.

Student(ID, Name, Address, Sex, Age, Dept, Dept_office, Dept_Chair, College, Dean, AdvisorID, AdvName, AdvDept)

No.

R1 = {ID, Name, Address, Sex, Age, Dept, AdvisorID}

R2 = {Dept, Dept_Office, Dept_Chair, College}

R3 = {College, Dean}

R4 = {AdvisorID, AdvName, AdvDept}

Discussion 8-8

Show that the following is true for any relation $r(R)$ and its decomposition $\{R_1, R_2\}$.

$$r \subseteq \Pi_{R_1}(r) \bowtie \Pi_{R_2}(r)$$

Join 연산은 공유하는 attribute 가 없을 경우 cartesian product 가 되므로, 모든 경우의 수를 다 포함한다. 따라서 r 을 포함한다.

반대로 공유 attribute 가 있을 때는 공유하고 있는 attribute 에 대해서 모든 경우의 수를 생성하므로, r 을 포함한다.

따라서 두 경우 모두 join 의 결과가 r 을 포함하므로 위의 명제가 성립한다.

=> 교수님 설명: r 의 임의의 tuple t 가 join 연산에 존재한다는 것을 증명해야 함.

$$t(R_1) \bowtie t(R_2) \rightarrow t(R) = t.$$

Discussion 8-9

It has been emphasized that *redundancy creates problems*. Discuss where redundancy lies in the following relation.

<i>title</i>	<i>author</i>	<i>pub-name</i>	<i>pub-branch</i>	<i>keyword</i>
Compilers	Smith	McGraw-Hill	New York	parsing
Compilers	Jones	McGraw-Hill	New York	parsing
Compilers	Smith	McGraw-Hill	New York	analysis
Compilers	Jones	McGraw-Hill	New York	analysis
Networks	Jones	Oxford	London	Internet
Networks	Frick	Oxford	London	Internet
Networks	Jones	Oxford	London	Web
Networks	Frick	Oxford	London	Web

Title, author \rightarrow title, pub-name

Keyword \rightarrow title.

Pub-name \rightarrow pub-branch

책과 출판사가 같은 relation 에 들어 있으므로 pub-name 에만 dependent 한 pub-branch 가 반복해서 나오는 문제가 생겨 redundancy 가 생긴다. 또한 keyword 도 다른 relation 으로 분리 가능.

=> 교수님 설명: title ->> author, title ->> keyword. (MVD. Multi-valued dependency)

FD 가 모든 dependency 를 설명하는 건 아님. 하지만 이번 학기에는 MVD 는 공부 ㄴㄴ