Functional Dependencies and Normal Form

Normalization.

Benefit of Normalizing a Table

Std id	Name	Dept name	Building	school
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Priya	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	Computer Sc	SJT 602	SCOPE

- Lot of information is stored in one Table ?
- If we add School dean column. Then for all the students say if we have 1000 students then the same dean name is stored 1000 time.
- Redundancy : The major problem in relations
- Normalization : Will reduce the redundancy.
- Problems that come up due to redundancy can be reduced
- Mention syllabus, Quiz 1

Three main problems that can come up due to redundancy:

- Insertion anomaly
- Deletion anomaly
- Updation anomaly

Std id	**Name**	**Dept name**	**Building**	**school**
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Priya	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	Computer Sc	SJT 602	SCOPE
NULL	NULL	English	TT 432	SSL

For adding a new department in this table is a issue. Without adding a student you cannot add a department?

Insertion anomaly

If we want to update the change of building and school of a department, same has to be done for all the students.

In case one or two tuples are not updated then we have **updation anomaly**. Inconsistent data.

Std id	**Name**	**Dept name**	**Building**	**school**
1	Rohit	Mat	PRP 654	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Priya	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	English	SJT 602	SSL

If we want to delete the student with name Singh then English department gets deleted. Which we don't want to happen.

Deletion Anomaly

Std id	**Name**	**Dept name**	**Building**	**school**
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Priya	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	English	SJT 602	SSL

Std id	Name	Dept name	Building	school
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Priya	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	Computer Sc	SJT 602	SCOPE
7	Vijay	English	TT 432	SSL

Std id	Name	Dept name
1	Rohit	Mat
2	Sanju	Computer Sc
3	Priya	Mat
4	Praveen	Phy
5	Ronald	Phy
6	Singh	Computer Sc
7	Vijay	English

Dept name	Building	school
Mat	PRP 102	SAS
Computer Sc	SJT 602	SCOPE
Phy	TT 645	SAS
English	TT 432	SSL

Process of splitting large relations in to sub relations: Normalization
Free from update, delete, insert operations.

Recall : Superkey, Candidate key

Functional dependences

Std id	Name	Dept name	Building	school
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Priya	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	Computer Sc	SJT 602	SCOPE
7	Vijay	English	TT 432	SSL

$FD: X \rightarrow Y$

Let R be a relation schema and let X and Y be nonempty sets of attributes in R. We say that an instance r of R satisfies the FD X → Y, if the following holds for every pair of tuples t_1 and t_2 in R:

$$\text{If } t1.X = t2.X, \text{ then } t1.Y = t2.Y.$$

Functional dependences

Std id	Name	Dept name	Building	school
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Praveen	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	Computer Sc	SJT 602	SCOPE
7	Vijay	English	TT 432	SSL

Std id → Std id

Std id → Name

Std id → Dept name

Dept name → Building

Dept name, Building → School

Name → School

Dept name → Name

Name, Dept name → School

School, Building → School

School, Building → Dept name, Name

School, Building → Name

Trivial Functional dependences

Std id	Name	Dept name	Building	school
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Praveen	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	Computer Sc	SJT 602	SCOPE
7	Vijay	English	TT 432	SSL

Std id → Std id

School, Building → School

$FD: X \rightarrow X$

$FD: X \rightarrow Y, \text{with } Y \subseteq X$

Non -trivial Functional dependences

Std id	Name	Dept name	Building	school
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Praveen	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	Computer Sc	SJT 602	SCOPE
7	Vijay	English	TT 432	SSL

$FD: X \rightarrow Y$, with $Y \cap X = \emptyset$

$FD: X \rightarrow Y$, with Y not a proper subset of X

Armstrong's Axioms, Inference Rules

- **Reflexivity:** If $X \supseteq Y$, then $X \rightarrow Y$.
- **Augmentation:** If $X \rightarrow Y$, then $XZ \rightarrow YZ$ for any Z.
- **Transitivity:** If $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$.
- **Union:** If $X \rightarrow Y$ and $X \rightarrow Z$, then $X \rightarrow YZ$.
- **Decomposition:** If $X \rightarrow YZ$, then $X \rightarrow Y$ and $X \rightarrow Z$.
- Composition : If $X \rightarrow Y$ and $A \rightarrow B$ then $XA \rightarrow YB$

Std id	Name	Dept name	Building	school
1	Rohit	Mat	PRP 102	SAS
2	Sanju	Computer Sc	SJT 602	SCOPE
3	Praveen	Mat	PRP 102	SAS
4	Praveen	Phy	TT 645	SAS
5	Ronald	Phy	TT 645	SAS
6	Singh	Computer Sc	SJT 602	SCOPE
7	Vijay	English	TT 432	SSL

Attribute Closure: The set of all FDs implied by a given set F of FDs is called the closure of F, denoted as F^+.

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

$FD: \{A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow E, E \rightarrow F\}$

Now how to determine the attribute closure of A, denoted by A^+

Determine the attribute closure of AD

Determine the attribute closure of B

In a schema with attributes A, B, C, D and E following set of functional dependencies are given
{A -> B, A -> C, CD -> E, B -> D, E -> A}
Which of the following functional dependencies is NOT implied by the above set?

a. CD-> AC
b. BD-> CD
c. BC-> CD
d. AC-> BC

Note: To check whether an FD A->B can be derived from an FD set F,

1.Find (A)+ using FD set F.
2.If B is subset of (A)+, then A->B is true else not true.

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

$FD: \{A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow E, E \rightarrow F, CD \rightarrow A\}$

Determine the Superkey:

If the attribute closure of a set of attributes contain all attributes of the given table then its a superkey.

Is A, AD, C, CD, B, E a superkey?

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

$FD: \{A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow E, E \rightarrow F\}$

Are A and AD candidate keys.

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

$FD: \{A \rightarrow B, C \rightarrow D, D \rightarrow E\}$

Is ABCDE and ACDE a Superkey.

Is ACDE a Candidate key.

Find all candidate keys.

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

$FD: \{A \rightarrow B, C \rightarrow D\}$

Ans: ACE

Prime Attributes: A,C,E

Prime attributes on FD on right side, then more possible candidate keys are there.

Consider a relation scheme R = (A, B, C, D, E, H) on which the following functional dependencies hold: {A–>B, BC–> D, E–>C, D–>A}. What are the candidate keys of R.

(a) {A,E},{B,E}
(b) {A,E}, {B,E}, {D,E}
(c) {A,E,H}, {B,E,H}, {B,C,H}
(d) {A,E,H}, {B,E,H}, {D,E,H}

Consider a relation scheme R = (A, B, C, D, E, H) on which the following functional dependencies hold: {A–>B, BC–> D, E–>C, D–>A}. What are the candidate keys of R.

(a) {A,E},{B,E}
(b) {A,E}, {B,E}, {D,E}
(c) {A,E,H}, {B,E,H}, {B,C,H}
(d) {A,E,H}, {B,E,H}, {D,E,H}

And: D

Find all candidate keys.

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

$FD: \{A \rightarrow B, B \rightarrow C, C \rightarrow A\}$

Consider the relation scheme R = {E, F, G, H, I, J, K, L, M, N} and the set of functional dependencies {{E, F} -> {G}, {F} -> {I, J}, {E, H} -> {K, L}, K -> {M}, L -> {N} on R. What is the key for R?

A.{E,F}
B.{E,F,H}
C. {E,F,H,K,L}
D. {E}

Consider the relation scheme R = {E, F, G, H, I, J, K, L, M, N} and the set of functional dependencies {{E, F} -> {G}, {F} -> {I, J}, {E, H} -> {K, L}, K -> {M}, L -> {N} on R. What is the key for R?

A. {E,F}
B. {E,F,H}
C. {E,F,H,K,L}
D. {E}

Ans. (B).