Terminology

• R (matric_no, firstname, surname, tutor_no, tutor_name)

tutor_no -> tutor_name

– A given tutor_no uniquely identifies (AKA functionally determines) a tutor_name.

– Tutor_name is dependent on tutor_no

– Tutor_no is the determinant

– An implied determinant (underlined) is also present in R:

• matrix_no -> firstname, surname, tutor_no, tutor_name

First Normal Form (Atomicity)

- A relation is in 1NF if, and only if, it contains no repeating attributes or groups of attributes (must be atomic values).
- A table with repeating groups is not in 1NF
  - it is an `un-normalized table'.
- To remove repeating groups, either:
  - flatten the table and extend the key, or
  - decompose (split) the relation

Example:

- A relation is in 1NF if it contains no repeating groups
- Remember to put the primary key from the original relation into both new relations.

a is Primary Key for the whole relation

R (**a** ,b, (**c** ,d)) becomes

R(**a** ,b)

R1(**a**, c, d)

Second Normal Form

KeyPart1 + KeyPart2 => attribute1, attribute2, …

- A relation is in 2NF if, and only if, it is in 1NF and every non-key attribute is fully functionally dependent on the whole key.
- Thus all non-key attributes must depend on the whole key. Another way of saying this is that there must be no partial key dependencies (PKDs).
- Problems arise only when there is a compound key

Example:

- A relation is in 2NF if it is in 1NF and has no partial key functional dependencies
- NOTE: A relation in 1NF with a single key field must (inevitably) be in 2NF
- DECOMPOSE:
  - One relation for the attributes that are fully dependent upon the key.
  - One relation for each part of the key that has partially dependent attributes

R (**a** , **b** ,c,d)

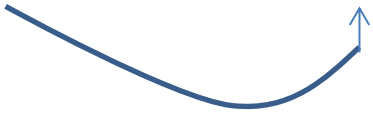
a->c becomes

R (**a** , **b** ,d)

R1 (**a** , c)

Third Normal Form

Key -> non-key attribute -> non-key attribute



Transitivity

- 3NF removes virtually all the redundant data
- A relation is in 3NF if, and only if,
  - it is in 2NF and
  - there are no transitive functional dependencies
- A transitive functional dependency can only occur if there is
- more than one non-key field
- A non-key field must provide a fact about the key, the whole
- key (2NF) and nothing but the key (3NF).

Example:

- A relation is in 3NF if it is in 2NF and has no transitive functional dependencies
- NOTE: A relation in 2NF with only one non-key attribute must (inevitably) be in 3NF
- DECOMPOSE To remove transitive functional dependencies, remove the attributes involved in the transitive dependency to a new relation

R(**a** , **b** , c , d)

c -> d Becomes

R(**a** , **b** , c)

R1(**c** , d)

Example

Repeating Groups

matric_no	Name	date_of_birth	subject	grade
960100	Smith, J	14/11/1977	Databases Soft_Dev ISDE	C A D
960105	White, A	10/05/1975	Soft_Dev ISDE	B B
960120	Moore, T	11/03/1970	Databases Soft_Dev Workshop	A B C
960145	Smith, J	09/01/1972	Databases	B
960150	Black, D	21/08/1973	Databases Soft_Dev ISDE Workshop	B D C D

Student(matric_no, name, date_of_birth, (subject, grade))
name, date_of_birth -> matric_no

---

1 NF

Student(matric_no, name, date_of_birth, subject, grade)

Flatten table and extend primary key

UNIVERSITY

STUDENT #2

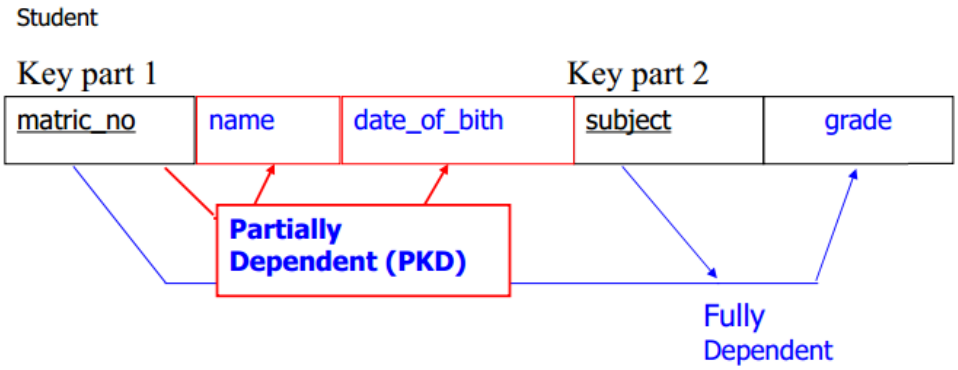
Redundancy

matric_no	name	date_of_birth	Subject	grade
960100	Smith, J	14/11/1977	Databases	C
960100	Smith, J	14/11/1977	Soft_Dev	A
960100	Smith, J	14/11/1977	ISDE	D
960105	White, A	10/05/1975	Soft_Dev	B
960105	White, A	10/05/1975	ISDE	B
960120	Moore, T	11/03/1970	Databases	A
960120	Moore, T	11/03/1970	Soft_Dev	B
960120	Moore, T	11/03/1970	Workshop	C
960145	Smith, J	09/01/1972	Databases	B
960150	Black, D	21/08/1973	Databases	B
960150	Black, D	21/08/1973	Soft_Dev	D
960150	Black, D	21/08/1973	ISDE	C
960150	Black, D	21/08/1973	Workshop	B

2 NF

Dependency Diagram

- A dependency diagram is used to show how non-key attributes relate to each part or combination of parts in the primary key.



Student (matric_no, name, date_of_birth)

Record (matric_no, subject, grade)

- the primary key from the original relation is included in both of the new relations ! !

Record

matric_no	subject	grade
960100	Databases	C
960100	Soft_Dev	A
960100	ISDE	D
960105	Soft_Dev	B
960105	ISDE	B
...	...	...
960150	Workshop	B

Student

matric_no	name	date_of_birth
960100	Smith,J	14/11/1977
960105	White,A	10/05/1975
960120	Moore,T	11/03/1970
960145	Smith,J	09/01/1972
960150	Black,D	21/08/1973

3 NF

UNIVERSITY

Example

Key field — Non-Key fields

Project_no	Manager	Address
p1	Black,B	32 High Street
p2	Smith,J	11 New Street
p3	Black,B	32 High Street
p4	Black,B	32 High Street

Project has more than one non-key field so we must check for transitive dependencies

Problem

- Data redundancy arises from this situation:
  - we will duplicate address if a manager is in charge of more than one project
  - this causes problems if we have to change the address – it requires changing several entries, and this can lead to errors.

Fix

- Eliminate the transitive functional dependency by splitting
- (decomposing) the table
  - create two relations - one with the transitive dependency in it, and another for all of the remaining attributes.
  - split Project into Project and Manager.
- the determinant attribute becomes the primary key in thenew relation i.e., manager becomes the primary key to the Manager relation
- the original key is the primary key to the remaining nontransitive attributes - in this case, project_no remains the key to the new Projects table.

Result : 3NF

- So now we need to store the address only once
- If we need to know a manager's address we can look it up in the Manager relation
- The manager attribute is the link between the two tables -- in the Projects table, manager is now a foreign key.
- These relations are now in third normal form.

Project

Project_no	Manager
p1	Black,B
p2	Smith,J
p3	Black,B
p4	Black,B

Manager

Manager	Address
Black,B	32 High Street
Smith,J	11 New Street