

Database system

Theoretical lectures

The Sixth lecture

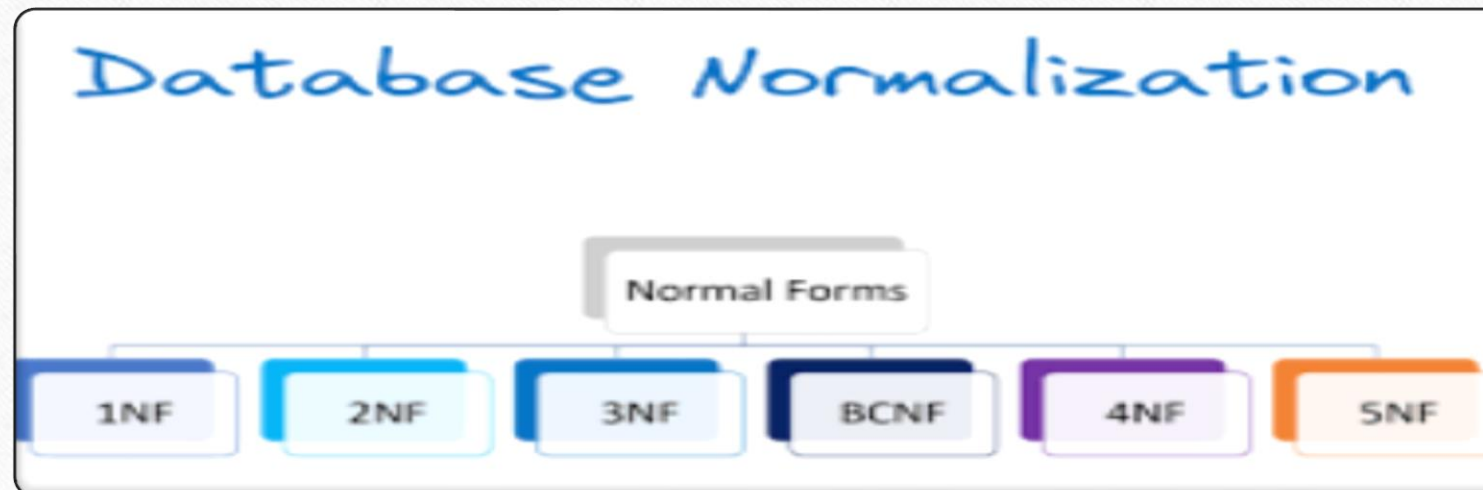
Assistant teacher Shaymaa Abdulkarem



Types of Normalization:

There are types of normalization

- ✓ 1. First Normal Form (1NF)
- ✓ 2. Second Normal Form (2NF)
- ✓ 3. Third Normal Form (3NF)



Types of Normalization

There are four basic rules that a tables should follow to be in the first normal form.

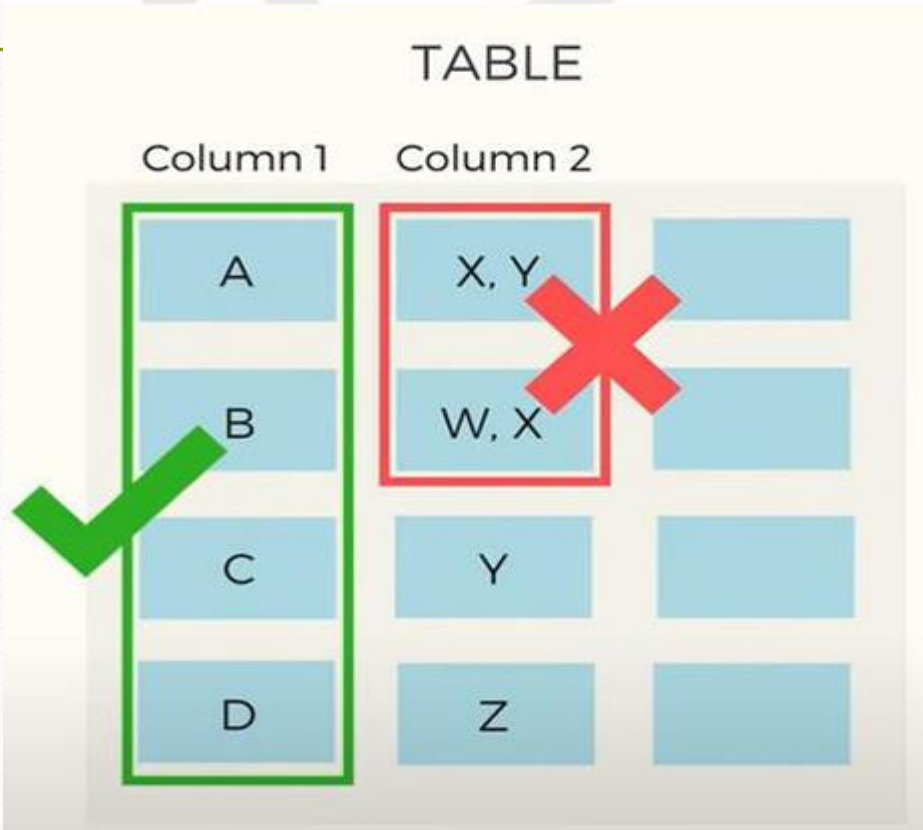
Rule 1 :

Each column should contain atomic values.

For example:

Entries like **X,Y** and **W,X** violate this rule.

“Each attribute must contain only a single value from its pre-defined domain”



Column 1	Column 2
A	X, Y
B	W, X
C	Y
D	Z

Types of Normalization

Rule 2 :

– A column should contain values that are of the same type.

For example:

Entries like **R** in **DOB** and 8-9-86 violate this rule.

“don’t inter-mix different types of values in any column.”

TABLE		
DOB	Name	
26-10-89	A	
13-2-92	SK	
16-11-65	SA	
R	8-9-86	



Types of Normalization

Rule 3 :

- Each column should have a unique name.

For example:

F_name and L_name different name.

“same name leads to confusion at the time of data retravel.”

TABLE

DOB	F_Name	L_Name
26-10-89	A	A
13-2-92	S	K



TABLE

DOB	Name	Name
26-10-89	A	A
13-2-92	S	K



Types of Normalization

Rule 4 :

- Order in which data is saved doesn't matter.

For example:

3, 4, 1 and 2 are not order in Roll_no

“using SQL query, you can easily fetch data in any order from a table.”

TABLE		
Roll_no	F_Name	L_Name
3	A	A
4	S	K
1	S	A
2	R	A

Types of Normalization

Example:

How to solve this ?

STUDENTS TABLE		
rollno	name	subject
101	Akon	OS, CN
103	Ckon	JAVA
102	Bkon	C, C++

Violation of 1 NF

Break the values



STUDENTS TABLE		
rollno	name	subject
101	Akon	OS ✨ CN
103	Ckon	JAVA
102	Bkon	C ✨ C++

Types of Normalization

STUDENTS TABLE		
rollno	name	subject
101	Akon	OS
101	Akon	CN
103	Ckon	JAVA
102	Bkon	C
102	Bkon	C++

Types of Normalization

2. Second Normal

Form For a table a table to be in the second normal form there are two conditions:

1. it should be in the first normal form
 2. it should not have any partial dependencies
-
- ❖ **Prime attribute** – An attribute, which is a part of the prime-key, is known as a prime attribute.
 - ❖ **Non-prime attribute** – An attribute, which is not a part of the prime-key, is said to be a non prime attribute.

Example

Dependency is ?



student_id	name	reg_no	branch	address
1	Akon	CSE-18	CSE	TN
2	Akon	IT-18	IT	AP
3	Bkon	CSE-18	CSE	HR
4	Ckon	CSE-18	CSE	MH

As the **student_id** in this table will be unique, it can be used easily to fetch any data.

Branch name of student with **student_id=1**

Name of student with **student_id=1**

student_id	name	reg_no	branch	address
------------	------	--------	--------	---------

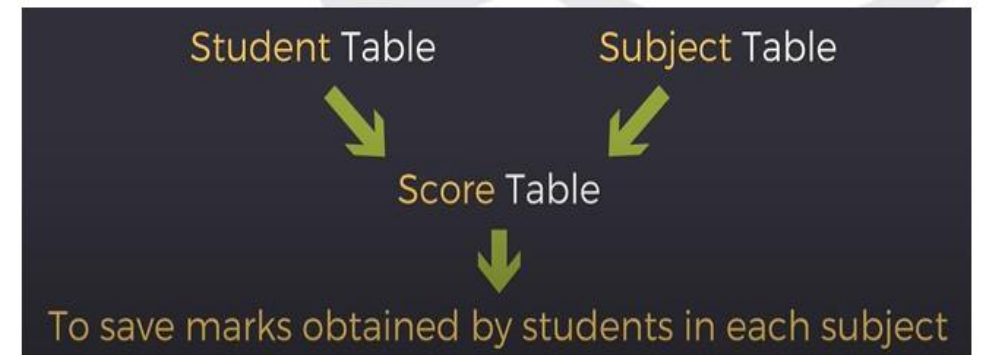
A Primary Key can be used to fetch data from any column in the table.

This is Dependency or Functional Dependency

Example

Partial Dependency is ?

SUBJECT TABLE	
subject_id	subject_name



Example

Primary key should be Score_id



The student_id + Subject_id together makes a more meaningful primary key

SCORE TABLE				
score_id	student_id	subject_id	marks	teacher
1	10	1	82	Mr. J
2	10	2	77	Mr. C++
3	11	1	85	Mr. J
4	11	2	82	Mr. C++
5	11	4	95	Mr. P

subject_id = 1 is for Java

subject_id = 2 is for C++

Example

If I ask to get the student_id 10 mark. Can or not find it ?

SCORE TABLE				
score_id	student_id	subject_id	marks	teacher
1	10	1	82	Mr. J
2	10	2	77	Mr. C++
3	11	1	85	Mr. J
4	11	2	82	Mr. C++
5	11	4	95	Mr. P

SCORE TABLE				
score_id	student_id	subject_id	marks	teacher
1	10?	1	82	Mr. J
2	10	2	77	Mr. C++
3	11	1	85	Mr. J
4	11	2	82	Mr. C++
5	11	4	95	Mr. P

So, we need to have student_id and subject_id.
Can uniquely identify any row of data in Score table

We have many to many relationship

Types of Normalization

Many to many relationship means

One Student can opt for more than **one Subject**

&

One Subject can be opted by more than one **Student**

Example

This is partial dependency

score_id	student_id	subject_id	marks	teacher
1	10	1	82	Mr. J
2	10	2	77	Mr. C++
3	11	1	85	Mr. J
4	11	2	82	Mr. C++
5	11	4	95	Mr. P



Teacher column only depends on subject and not on student

Example

How to remove the partial dependency ? Our objective is to remove teacher column from score table

SCORE TABLE						
score_id	student_id	subject_id	marks	teacher		
1	10	1	82	Mr. J		
2	10	2	77	Mr. C++		
3	11	1	85	Mr. J		
4	11	2	82	Mr. C++		
5	11	4	95	Mr. P		



SUBJECT TABLE	
subject_id	subject_name
1	Java
2	C++
3	C#
4	Php

Example

How to remove the partial dependency ? Our objective is to remove teacher column from score table

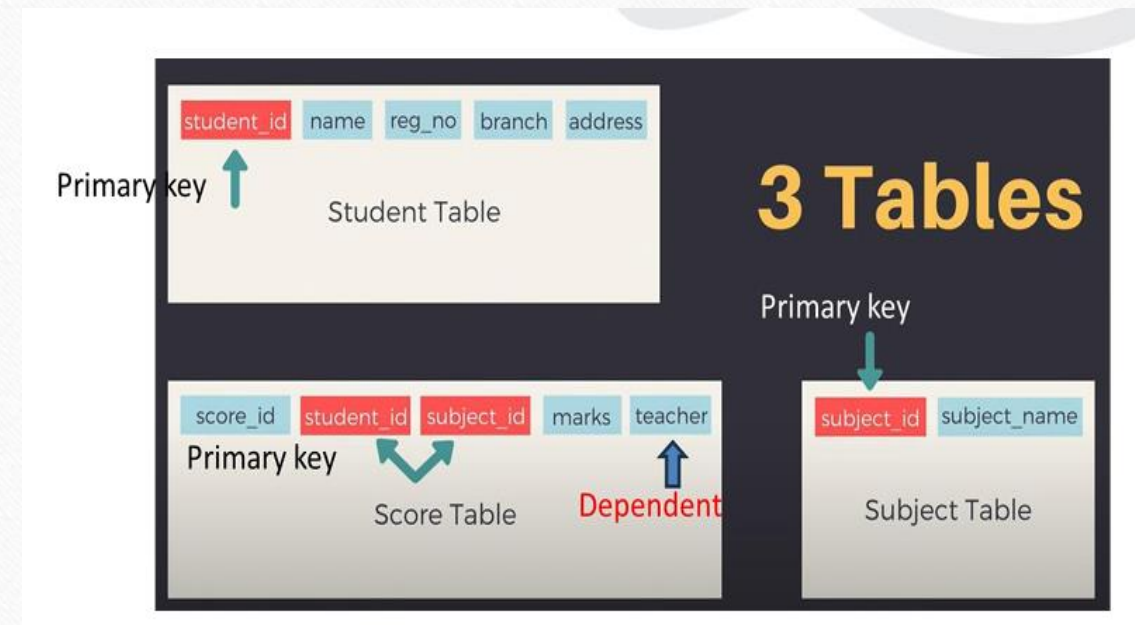
Teacher TABLE	
teacher_id	teacher_name
1	Mr. J
2	Mr. C++
3	Mr. C#
4	Mr. P
Can even add more info. related to teachers like date of joining, salary etc.	

Types of Normalization

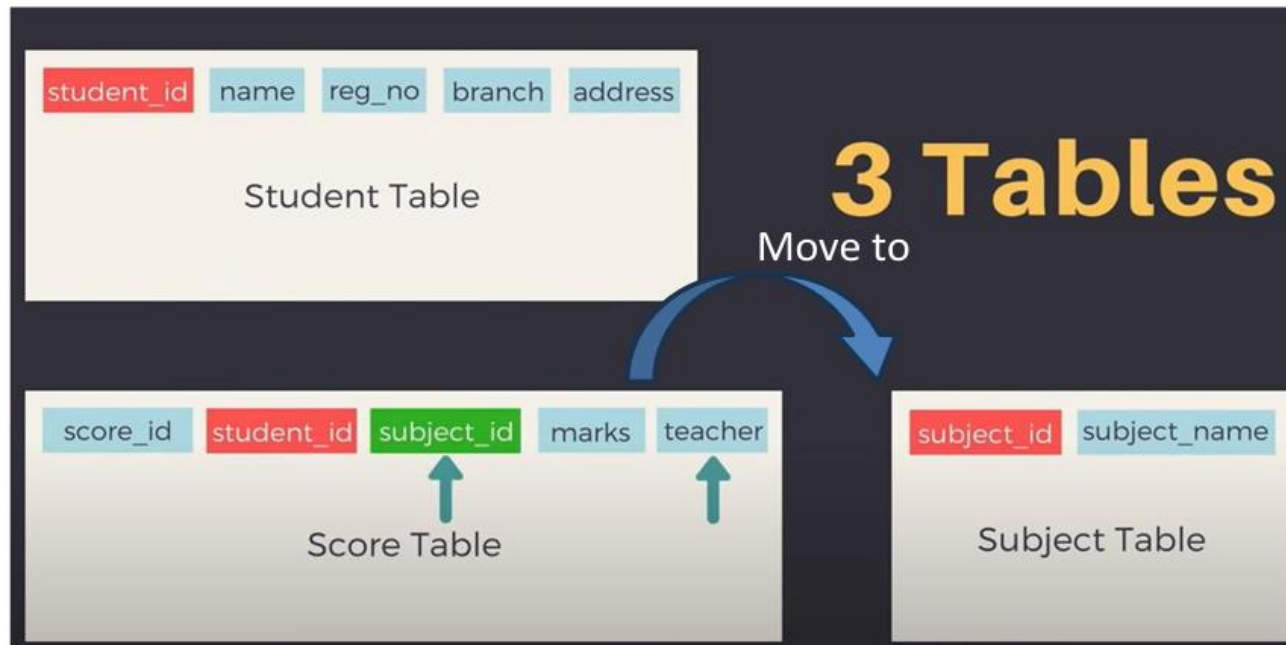
3. Third Normal

Form For a relation to be in Third Normal Form,

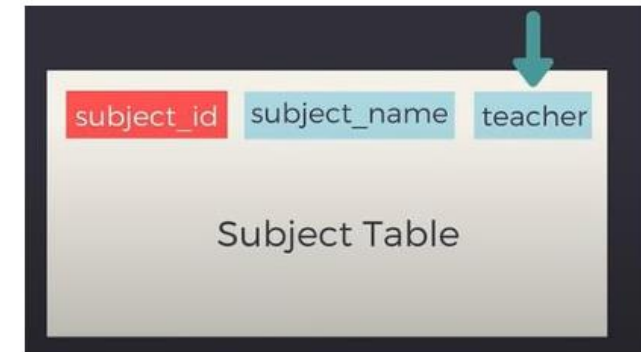
- It must be in Second Normal form and the following must satisfy
- It should not have trans



Types of Normalization



Second normal form



In 2nd normal form

Types of Normalization

If we want add more information in our score table
Exam Name and Total Mark

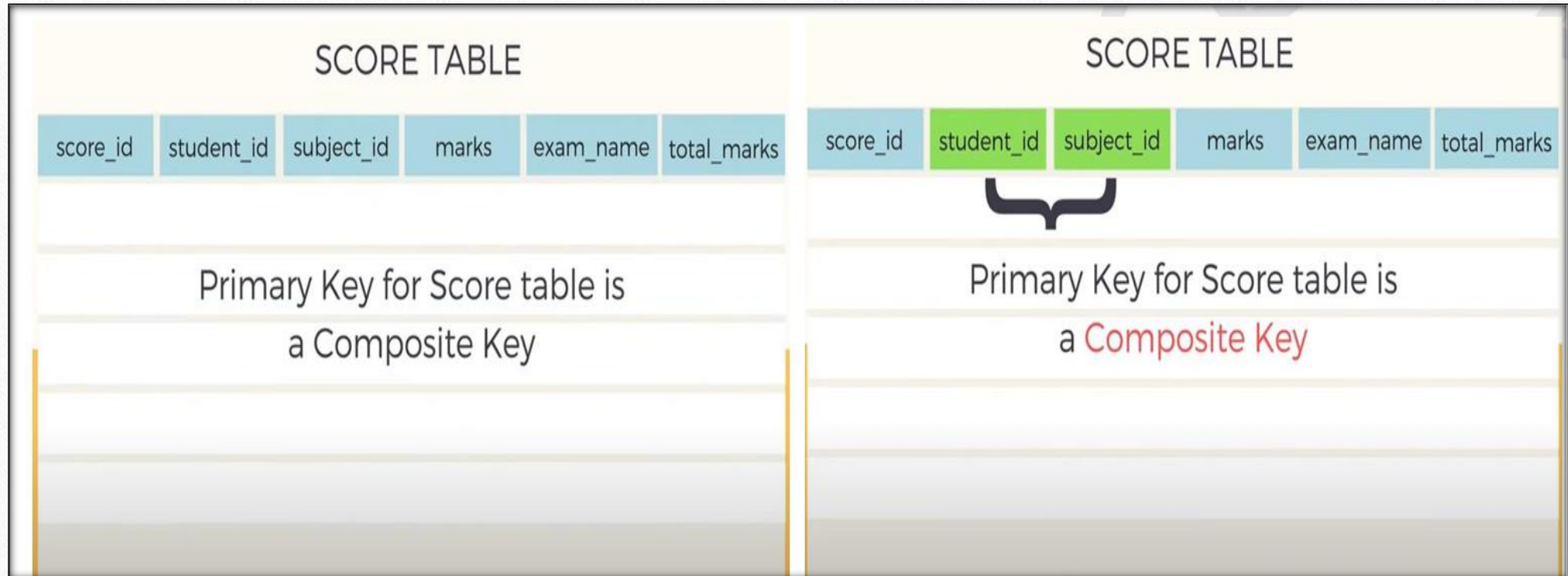
SCORE TABLE					
score_id	student_id	subject_id	marks	exam_name	total_marks

2 new columns

Types of Normalization

What is Transitive dependency ?

With add the 2 column



Types of Normalization

Example



A Mechanical Student will have
Workshop exam but IT student won't

Exam depends on Student



For Java, we have Practical exam but
for Engineering Maths, we don't.

Exam depends on Subject

Exam depends on Student

and

Exam depends on Subject

Types of Normalization

For `total_marks`

`Total_mark` depends on `exam_name`



Practicals are of 30-40 marks



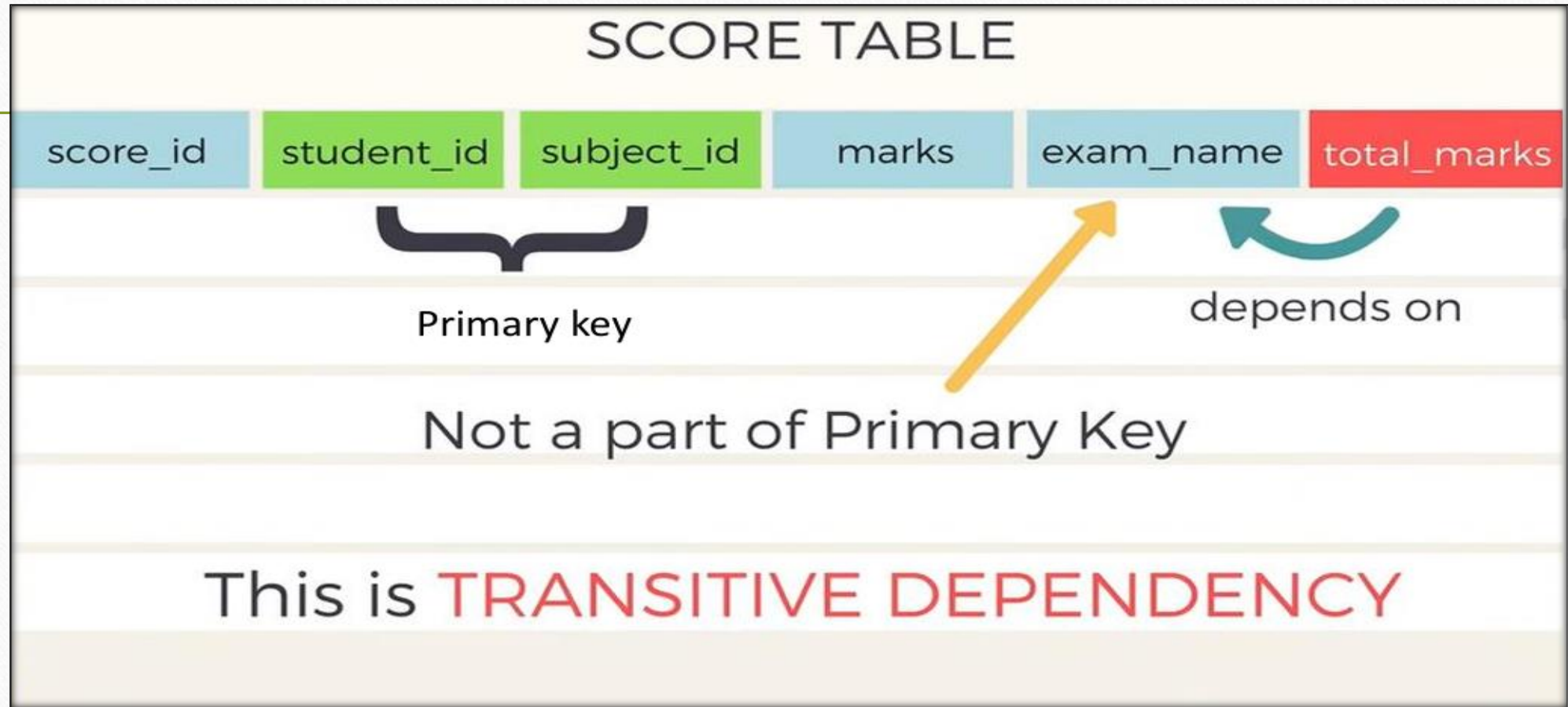
Main Exams are of 60-70 marks



Sessionals are of 40-50 marks

So the value of `total_marks` changes with Exams

Types of Normalization



Types of Normalization

