Date
4-04-2020

Computer Science 2nd Year

Subject - 'DBMS'

Second Normal form (2NF)

For a table to be in the second Normal form it must satisfy two condition.

1) The table should be in the 'first Normal form.

2) There should be no partial Dependency.

Dependency :

Let's take an example of a student table with columns Student_Id, name, reg.no (registration number), branch and address (Student's home address).

Student Id	name	reg-no	branch	address
10	Akash	256	CS	DEF
20	Akash	678	EC	GHI
30	Chandan	921	EC	JKL

In this table Student-Id is the primary key and will be unique for every row, hence we can use Student-Id to fetch any row of data from this table.

Even for a case, where student names are same, if we know the student-Id we can easily fetch the correct record.

Student-Id	name	reg:no	Branch	address
10	Akash	258	CS	DBR
20	Akash	678	Ec	GME
.				

Hence we can say a primary key for a table is the column or a group of column (Composite Key) which can uniquely Identify each record in the table.

I can ask from branch name of student with Student-Id 10; and I can get it. Similarly, If I ask for name of student with Student-Id 10 and 20, I will get it. So all I need is Student-Id and every other Column depends on it, or

Can be fetched using it.
This is dependency and we also call it functional dependency.

Partial dependency :

Now that we know what dependency is, Now we understand what partial dependency is.

For a simple table like Student, a single column like Student-Id can uniquely Identify all the records in a table.

But this is not true all the time. So now let's extend our example to see if more than 1 Column together can act as a primary key.

Let create another table for Subject, which will have Subject_Id and Subject-name fields and Subject-Id will be the primary key.

Subject Id	Subject Name
1	Java
2	C++
3	PHP

Now we have a Student table with Student Information and another table Subject for storing subject Information.

Let's create another table ~~Score~~ Score, to store the marks obtained by Student in the respective subject. We will also be saving name of the teacher who ~~teacher~~ teaches that subject along with marks.

Score Id	Student Id	Subject Id	mark	teacher
1	10	1	70	Java teacher
2	10	2	75	C++ teacher
3	11	1	80	Java Teacher

In the Score table we are saving the Student_Id to know which student's marks are these and ~~Student_Id~~ Subject_Id to know for which subject the marks are for

Together, | Student_Id + Subject_Id | form

a candidate key for this table, which can be the primary key.

Confused, How this combination can be a primary key.

See if I ask you to get me marks of student with student_Id 10, and your get it from this table? No, because you don't know for which subject. and if I give you Subject-Id you would not know for which student Hence we need

Student_Id + Subject_Id

to uniquely Identify any row.

But where is partial Dependency

Now if you look at the score table, we have a column names teacher which is only dependent on the subject for Java its Java teacher and for C++ its C++ teacher and so on

Now as we just discussed that the primary key for this table is a composite of two columns which is Student_Id and Subject_Id

but the teacher name only depends on subject hence the Subject-Id and has nothing to do with Student-Id.

This is partial dependency, where an attribute in a table depends on only a part of the primary key and not on the whole key.

How to remove partial dependency?

The objective is to remove teacher's name from Score table.

The simplest solution is to remove column teacher from Score table and add it to the Subject table. Hence the subject table will become

Subject table

Subject-Id	Subject_name	teacher
1	Java	Java teacher
2	C++	C++ teacher
3	Php	Php teacher

and our Score table is now in the 2NF with no partial dependency.

Score_Id	Student_Id	Subject_Id	Marks
1	10	1	70
2	10	2	75
3	11	1	80

Summary (2NF)

① For a table to be in the 2NF it should be in the First Normal form and it should not have partial dependency.

② Partial dependency exists, when for a composite primary key, any attribute in the table depends only on a part of the primary key and not on the complete primary key.

③ to remove partial dependency, we can divide the table, remove the attribute which is causing partial dependency and move it to some other table where it fits in well.

Shardha

4/04/2020

Shardha Vaish