2nd Normal Form

The **second normal form** removes partial functional dependencies in the table. For a table to be in 2NF, it must be in 1NF.

Let’s summarise what you have learnt in this segment.

Suppose two columns, Column B and Column C, form a composite key for a table. The value of Column B can determine the value of Column D. The value of column C can determine the values of Columns E and F. The values of Columns B and C together can determine the values of Columns A, D, E and F.

* (B, C) -> A, D, E, F
* B -> D
* C -> E, F

B and C are the prime attributes here. The values of columns D, E and F depend on a prime attribute and not on the entire composite key. This is a case of partial dependency on the composite key. The value of column A is dependent on both B and C. This is a case of full functional dependency on the composite key.

To remove partial dependencies, you need to separate the partial dependencies into new tables. In this case, the following three tables will be formed:

1. Table 1: <A, B, C>. B and C together will act as a composite key, and A is fully functionally dependent on the composite key.
2. Table 2: <B, D>. B will act as a primary key, and D is dependent on this primary key.
3. Table 3: <C, E, F>. C will act as a primary key. E and F will be dependent on this primary key.