

## Partial Dependency in DBMS

The functional dependency (FD) that a relation holds happens to be partial when the removal of one determining attribute gives a functional dependency that holds in the given relation. The FD that is not partial refers to being full.

### What is Partial Dependency in DBMS?

The FD (functional dependency)  $A \rightarrow B$  happens to be a partial dependency if B is functionally dependent on A, and also B can be determined by any other proper subset of A. For instance, we have a relationship like  $MO \rightarrow N$ ,  $M \rightarrow P$ , and  $P \rightarrow N$ . In this case, M is alone capable of determining N. It means that N is dependent partially on MO.

### When does Partial Dependency occur?

A partial dependency would occur whenever a non-prime attribute depends functionally on a part of the given candidate key. The 2NF (Second Normal Form) eliminates partial dependency. Let us take an example to understand this.

### Example

<Employee\_Task>

Employee_ID	Task_No	Employee_Name	Task_Name
C01	34	Mona	App Development
C02	58	Genine	UX/UI Designing

In the table given above, we have a partial dependency. Here is how:



Here, the prime key attributes are Employee\_ID and Task\_No, and also:

Employee\_ID = A unique ID of the employee

Employee\_Name = Name of the employee

Task\_No = A unique ID of the task

Task\_Name = The name of the task

As stated above, the non-prime attributes are Employee\_Name and Task\_Name. These must be dependent functionally on the part of the candidate key so as to be Partial Dependent.

The Employee\_Name can be determined using the Employee\_ID. It actually makes the relation Dependent Partially.

The Task\_Name can be determined using the Task\_No. It makes the relation Dependent Partially.

Thus, the <Employee\_Task> relation would violate the Second Normal Form in Normalization and is considered to be a bad database design.

We decompose the tables to remove Partial Dependency along with the violation on the second normal form:

<Employee\_Info>

Employee_ID	Task_No	Employee_Name
C01	34	Mona
C02	58	Genine



<Task\_Info>

Task_No	Task_Name
34	App Development
58	UX/UI Designing

Thus, the relation happens to be in the second normal form in the case of Database Normalization.

### Practice Questions on Partial Dependency in DBMS

1. The relation that does not consist of any partial dependency happens to be in which of these normal forms?

- A. Third
- B. Second
- C. First
- D. BCNF

**Answer: B. Second**

2. The functional dependency that is present between multiple non-key attributes is known as:

- A. Functional dependency
- B. Partial transitive dependency
- C. Transitive dependency
- D. Partial functional dependency

**Answer: C. Transitive dependency**



---

### **What is partial dependency in DBMS? Give an example.**

The FD (functional dependency)  $A \rightarrow B$  happens to be a partial dependency if B is functionally dependent on A, and also B can be determined by any other proper subset of A. For instance, we have a relationship like  $MO \rightarrow N$ ,  $M \rightarrow P$ , and  $P \rightarrow N$ . In this case, M is alone capable of determining N. It means that N is dependent partially on MO.

### **What is a partial and transitive dependency?**

Partial dependency occurs when one primary key determines some other attribute/attributes. On the other hand, transitive dependency occurs when some non-key attribute determines some other attribute.

### **What is full dependency in DBMS?**

A full FD is the state of normalization of the database that equates to the normalization standards of the 2NF. In simpler words, it means that this meets all the requirements of the First Normal Form, and all non-key attributes are functionally dependent fully on the primary key.

