

### 1) What is Normalization:

- It is a process for evaluating and correcting table structures to minimize data redundancies and therefore reduce the likelihood of data anomalies.

### 2) When is a table in 1NF?

- A relation is said to be in the first normal form if and only if the domain of each attribute contains only atomic(indivisible) values, and the value of each attribute contains only a single value from that domain.

### 3) When is a table in 2NF?

- A relation will be in the second normal form (2NF) if it is in 1NF and all non-key attributes are fully functional dependent on the primary key

### 4) When is a table in 3NF?

- A relation will be in Third Normal form(3NF) if it is in 3NF and no transition dependency exists.

### 5) Dependency Diagram

**Desirable dependencies:**

**(INV\_NUM, PROD\_NUM) =>**

SALE\_DATE, PROD\_LABEL, VEND\_CODE, VEND\_NAME, QUANT\_SOLD, PROD\_PRICE

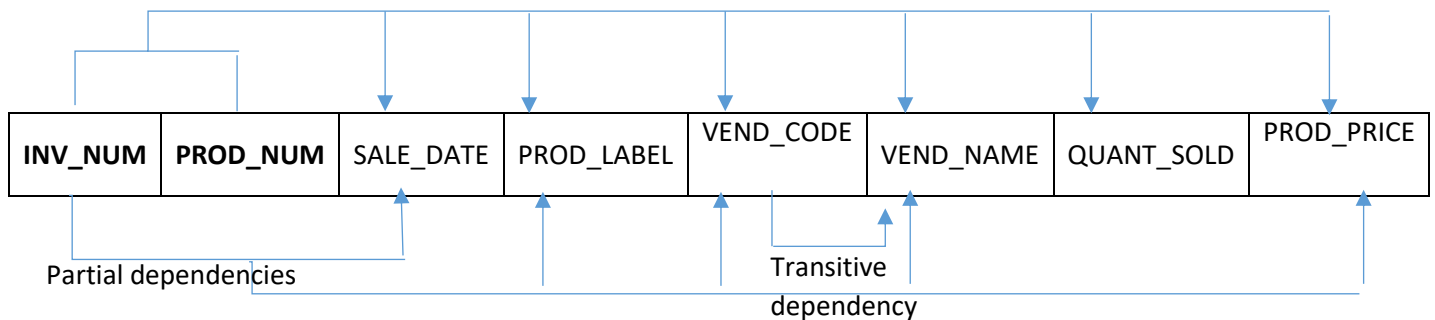
**Partial dependencies:**

INV\_NUM -> SALE\_DATE

PROD\_NUM -> PROD\_LABEL, VEND\_CODE, VEND\_NAME, PROD\_PRICE

**Transitive dependency:**

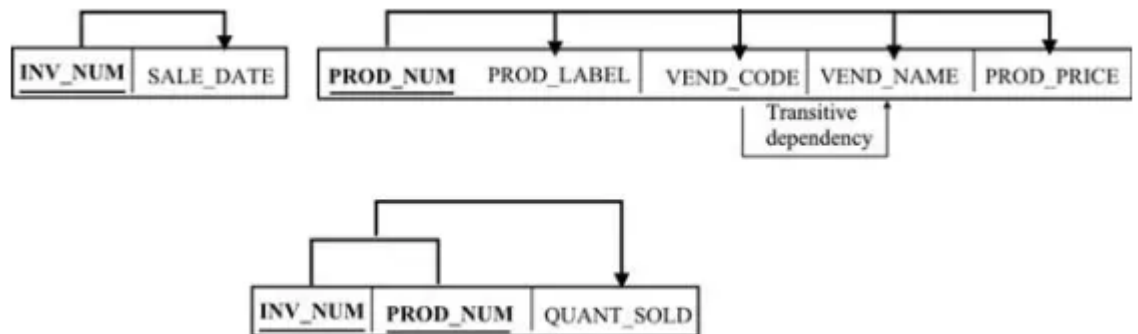
PROD\_NUM -> VEND\_NAME (i.e. PROD\_NUM -> VEND\_CODE, VEND\_CODE -> VEND\_NAME)



Partial dependencies

6) To remove all partial dependencies, we have to create the following new tables:

- (INV\_NUM, SALE\_DATE) – **Third Normal Form (3NF)**
- (PROD\_NUM, PROD\_LABEL, VEND\_CODE, VEND\_NAME, PROD\_PRICE) – **SECOND NORMAL FORM**
- (INV\_NUM, PROD\_NUM, QUANT\_SOLD) – **3NF**



7) To remove transitive dependencies, we do the following:

- (PROD\_NUM, PROD\_LABEL, VEND\_CODE, PROD\_PRICE) – **3NF**
- (VEND\_CODE, VEND\_NAME) – **3NF**

