

First normal form

- 1 An attribute (column) of a table cannot store multiple values. It should store only atomic values
- 2 There are no repeating groups of columns
e.g. [client-name1] [client-name2] [client-name3]

Second normal form

- 1 The table should be in the first Normal Form
- 2 There should be no Partial Dependency.

Third normal form

- 1) Table has to be in the second Normal Form
- 2) Table should not contain transitive dependency on the primary key

- As all non-primary-key fields depend on a primary key, you have to search do any non-primary-key fields depend on another non-primary-key

First Normal Form

- 1 An attribute (column) of a table cannot store multiple values. It should store only atomic values
- Any column of your table should not contain multiple values. Non-key attributes must be functionally dependent on the primary key.

→ If column B depend on column A and column C depend on column B, Therefore, column C has a transitive dependency on A

Database Systems

Normalisation

- Data with duplicate values leads to anomalies, but when information is stored in one place and one place only, the possibility of inconsistent data is reduced.
- Properly normalized database will:
 - require less storage
 - be easier to handle and update, without facing data loss.
 - be more informative to users.

UNF	1NF	2NF	3NF
<u>invoice_id</u> client_id client_name client_phone product_id product_name product_price quantity amount total amount	<u>invoice_id</u> Product_id * client_id client_name client_phone product_price quantity amount total amount <u>Product_id</u> Product_name	<u>invoice_id</u> product_id * client_id * product_price quantity amount Total amount <u>Product_id</u> Product_name <u>Client_id</u> client_name Client_phone	<u>invoice_id</u> product_id * client_id * Total amount Invoice_id * } Composite product_id * } PK product_price quantity amount <u>Product_id</u> Product_name <u>Client_id</u> client_name client_phone