

### First-Normal Form (1NF)

With our un-normalised relation now complete we are ready to start the normalisation process. First Normal form is probably the most important step in the normalisation process as it facilitates the breaking up of our data into its related data groups, with the following normalised forms fine tuning the relationships between and within the grouped data.

With First Normal Form we are looking to remove repeating groups. A repeating group is a domain or set of domains, directly relating to the key, that repeat data across tuples in order to cater for other domains where the data is different for each tuple.

Repeating Groups of Data

<u>StudentId</u>	StudentName	Year	Semester	UnitCode	UnitName
0023765	John Doe	2009	2	UG45783	Advance Database
0023765	John Doe	2009	2	UG45832	Network Systems
0023765	John Doe	2009	2	UG45734	Multi-User Operating Systems
0035643	Ann Smith	2009	2	UG45832	Network Systems
0035643	Ann Smith	2009	2	UG45951	Project
0061234	Peter Wolfe	2009	2	UG45783	Advance Database

In this example with Student ID as the primary key we see the three domains, StudentName, Year and Semester repeat themselves across the tuples for each of the different UnitCode and UnitName entries. Though workable it means our relation could potentially be huge with loads of repeating data taking up valuable space and costing valuable time to search through.

The rules of First Normal Form break this relation into two and relate them to each other so the information needed can be found without storing unneeded data. So from our example we would have one table with the student information and another with the Unit Information with the two relations linked by a domain common to both, in this case, the StudentId.

So the steps from UNF to 1NF are:

1. Identify repeating groups of data. Make sure your model data is of good quality to help identify the repeating groups and don't be afraid to move the domains around to help with the process.
2. Remove the domains of the repeating groups to a new relation leaving a copy of the primary key with the relation that is left.

3. The original primary key will not now be unique so assign a new primary key to the relation using the original primary key as part of a compound or composite key.
4. Underline the domains that make up the key to distinguish them from the other domains.

Taking our original example once we have followed these simple steps we have relations that looks like this:

<u>StudentId</u>	StudentName	Year	Semester
0023765	John Doe	2009	2
0035643	Ann Smith	2009	2
0061234	Pete		

  

<u>StudentId</u>	<u>UnitCode</u>	UnitName
0023765	UG45783	Advance Database
0023765	UG45832	Network Systems
0023765	UG45734	Multi-User Operating Systems
0035643	UG45832	Network Systems
0035643	UG45951	Project
0061234	UG45783	Advance Database

Compound Key

Tables in First Normal Form