Normalization Example

Below is an extract from a company report, which describes projects being worked upon by employees. This report is to be "normalized".

Project Management Report

Project Code:	PC010			
Project Title:	Pensions System			
Project Manager:	Tae Hyung			Project Budget: $24,500
Emp No	**Emp Name**	**Dept No**	**Dept Name**	**Hourly Rate**
S1001	YoonA	L004	IT	$22.00
S1003	Y S Ho	L023	Business	$20.50
S2101	K S Hyun	L004	IT	$21.00
Total Staff on Project: 3			**Average Hourly Rate:**	**$21.17**

Step 1: Un-normalized Form (UNF)

- Select the data source and convert into an un-normalized table (UNF)
- The process is as follows:
  - Create column headings for each data item on the report (**ignoring any calculated fields**).
  - Enter sample data into table.
  - Identify a **key** for table (and underline it).
  - Remove duplicate data (in this example, for the chosen key of Project Code, the values for Project Code, Project Title, Project Manager and Project Budget are duplicated if there are two or more employees working on the same project).

UNF: Un-normalized Table

Step 2: First Normal Form (1NF)

- Transform a table of un-normalized data into first normal form (1NF).
- **Rule: remove any repeating attributes to a new table**.
- The process is as follows:
  - Identify repeating attributes.
  - Remove these repeating attributes to a new table together with a **copy** of the key from the UNF table.
  - Assign a key to the new table (and underline it). The key from the original un-normalized table **always** becomes **part** of the key of the new table. A **compound key** is created. The value for this key must be unique for each entity occurrence.

- After removing the duplicate data, the repeating attributes are easily identified.
- In the previous table the Emp No, Emp Name, Dept No, Dept Name and Hourly Rate attributes are repeating. That is, there is potential for more than one occurrence of these attributes for each project code. These are the repeating attributes and have been to a new table together with a copy of the original key (i.e. Project Code).
- A key of Project Code and Emp No has been defined for this new table. This combination is unique for each row in the table.

1NF with Repeating Attributes Removed

Step 3: Second Normal Form (2NF)

- Transform the data in first normal form (1NF) into second normal form (2NF).
- **Rule: remove any non-key attributes that only depend on part of the table key to a new table**.

- Ignore tables with (a) a simple key or (b) with only key attributes (these go straight to 2NF with no conversion).
- The process is as follows:
  - Take each non-key attribute in turn and ask the question: is this attribute dependent on one part of the key?
  - If yes, remove attribute to new table with a **copy** of the **part** of the key it is dependent upon. The key it is dependent upon becomes the key in the new table. Underline the key in this new table.
  - If no, check against other part of the key and repeat above process.
  - If still no, i.e. not dependent on either part of key, keep attribute in current table.

2NF: Partial Key Dependencies Removed

Step 4: Third Normal Form (3NF)

- Transform the data in second normal form (2NF) into third normal form (3NF).
- **Rule: remove to a new table any non-key attributes that are more dependent on other non-key attributes than the table key**.
- Ignore tables with zero or only one non-key attribute (these go straight to 3NF with no conversion).
- The process is as follows:
- If a non-key attribute is more dependent on another non-key attribute than the table key:
  - Move the **dependent** attribute, together with a **copy** of the non-key attribute upon which it is dependent, to a new table.
  - Make the non-key attribute, upon which it is dependent, the key in the new table.

3NF: Non-Key Dependencies Removed