Normalization of Database Tables

By

Oluwabunmi Olanrewaju

COMP 123

[Intro to Business Computing](https://online.saskpolytech.ca/d2l/home/104310)

Table of Contents

**Introduction2**

Database Normalization1

Update anomaly1

Insertion anomaly 1

Deletion anomaly1

**Type chapter title (level 1)4**

Type chapter title (level 2)5

Type chapter title (level 3)6

1. **Introduction**

In developing plan for database, one of the most important aspect to be taken into account is to ensure that the duplication is minimized. This is done for the following reasons:

1. Reducing the amount of storage needed to store the data.
2. Eliminating redundant data.
3. Ensuring data dependencies make sense that is data is logically stored

**1.1 Database Normalization**

Database Normalization is a technique that helps in designing database in an optimal manner. It is a systematic approach of decomposing tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anomalies. The main idea of database normalization is to divide the tables into smaller sub-tables and store pointers to data rather than replicating it, making it easier to handle and update the database without facing data loss.

**1.1.1 Update anomaly:** When the same information can be expressed on multiple rows, therefore updates to the relation may result in logical inconsistencies. For example, each record in an "Employee information table" relation contain an Employee ID, Employee Address, pay period etc. Thus a change of address for a particular employee may need to be applied to multiple records (one for each pay period). If the update is only partially successful – the employee's address is updated on some records but not others – then the relation is left in an inconsistent state. Specifically, the relation provides conflicting answers to the question of what this particular employee's address is. This phenomenon is known as an update anomaly.

**1.1.2 Insertion anomaly:** There are circumstances in which certain facts cannot be recorded at all. For example, we cannot record the details of a new department because no employee has been assign to the department, and we cannot also record a newly hired employee who has been assigned to this new department, except by setting the job Code to null. This phenomenon is known as an insertion anomaly.

**1.1.3 Deletion anomaly:** A Delete Anomaly exists when certain attributes are lost because of the deletion of other attributes. For example, deleting Wong Jodie from our database, will cause us not to only lose the fact that Wong Jodie is an employee, but also the fact that we have a department called cleaner as he is the only employee under the cleaner department.

**First Normal Form (1NF)**

The First normal form endures that each cell of a table should contain exactly one value. A table is in first normal form if:

1. There are no repeating groups
2. All data values are atomic
3. Each field has a unique name
4. It has a primary key

|  |  |
| --- | --- |
| UNF | |
|  | Employee Id |
|  | SIN |
|  | Employee Last Name |
|  | Employee First Name |
|  | Street |
|  | City |
|  | Province |
|  | Postal |
|  | Job Code |
|  | Position |
|  | Pay rate |
|  | Income Tax |
|  | Birth Date |
|  | Hire Date |
|  | Job Code Date |
|  | Pay Week End Date |
|  | Days Available |
|  | Hours |
|  | Over-Time |
|  | Person Hours Worked |
|  | Supervisor |
|  | Supervisor Cell# |
|  |  |
|  | Committee Id |
|  | Committee Name |
|  | Meeting Night |
|  |  |
|  | OH&S Committee |
|  | Party Committee |
|  | Social Res. Committee |

|  |  |
| --- | --- |
| 1NF | |
| PK | Employee Id |
|  | SIN |
|  | Employee Last Name |
|  | Employee First Name |
|  | Street |
|  | City |
|  | Province |
|  | Postal |
|  | Job Code |
|  | Position |
|  | Pay rate |
|  | Income Tax |
|  | Birth Date |
|  | Hire Date |
|  | Job Code Date |
|  | Supervisor First Name |
|  | Supervisor Last Name |
|  | Supervisor Cell# |
|  |  |
| PK | Employee Id |
| PK | Pay Week End Date |
|  | Days Available |
|  | Hours |
|  | Over-Time |
|  | Person Hours Worked |
|  |  |
| PK | Employee Id |
| PK | Committee Id |
|  | Committee Name |
|  | Meeting Night |

From the Un-Normalized Form database tables provided in excel, there were lots of repeating groups as some of the employee worked under the two pay periods and some are in more than one committee. The first step to normalized this is to remove the repeating group by taking the set of data causing the repetition and take it as a new table.

When removing fields form a table to form another table, we must always take a copy of the primary key (identifier) from the table we removed them from. The primary key in the UNF table in this assignment is the Employee ID.

Atomic data are data elements in their lowest level of details. So non-atomic fields should be split such as the supervisor name should be further broken down into supervisor first name and supervisor last name. in deciding this, the following must be put into consideration:

1. Do all values in the field look like they belong in two or more fields instead?
2. Are you able to name what the new fields would be called from looking at the values in the field?
3. Is it of use to the database system or does it make the data clearer to have the values separated?

Each column should have unique name to identify each of them. Also each table should have a primary key which is the identifier for each record and it cannot be repeated. However, we can have a composite primary key that is a primary key composed of multiple columns used to identify a record uniquely in our database.

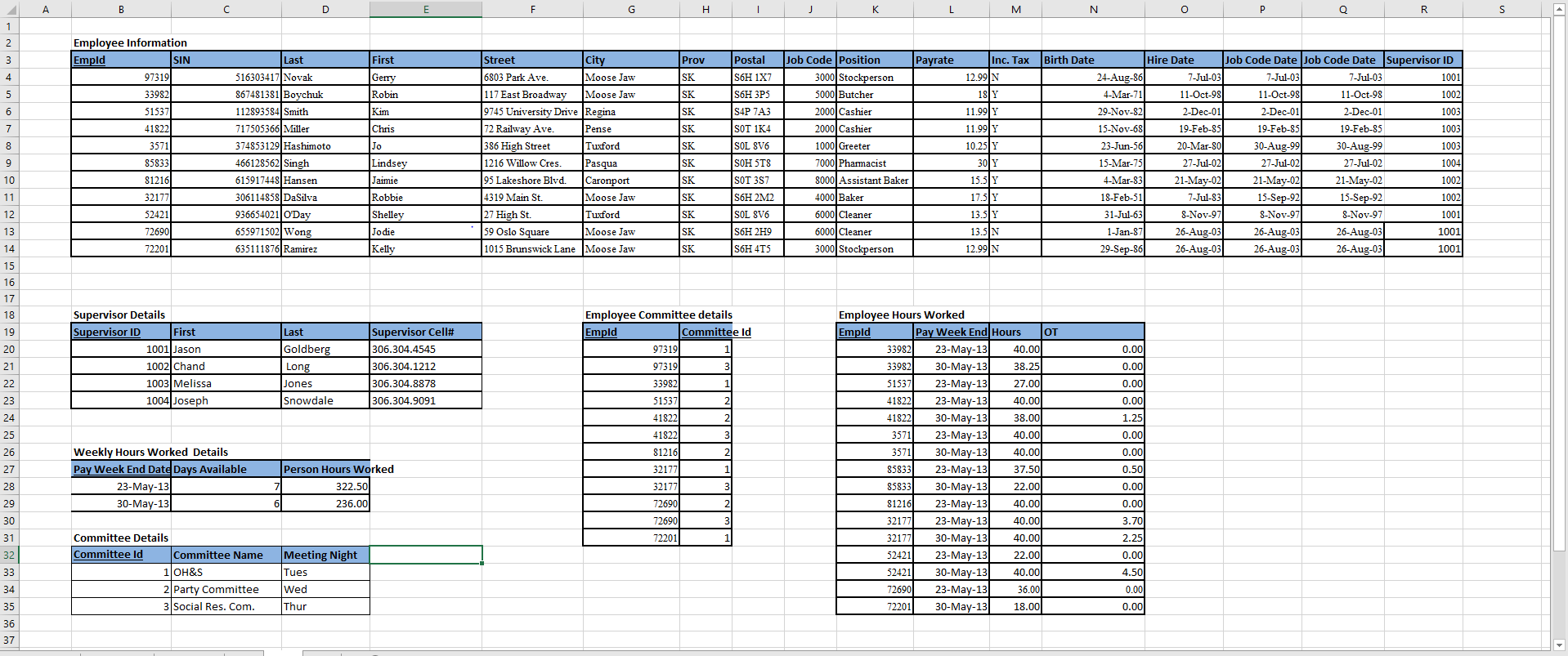
**Second Normal Form (2NF)**

A table is in second normal form if:

1. It is first in First Normal Form (1NF)
2. All non - key attributes are dependent on all parts of the primary key
3. Each field has a unique name
4. It has a primary key

There was a bit introduction of INF because, it is necessary in order to prepare the second normal form. Here, we will discuss in details each table in our 2NF and what inform our decision to have those tables.

Figure 1.1(2NF Relations)



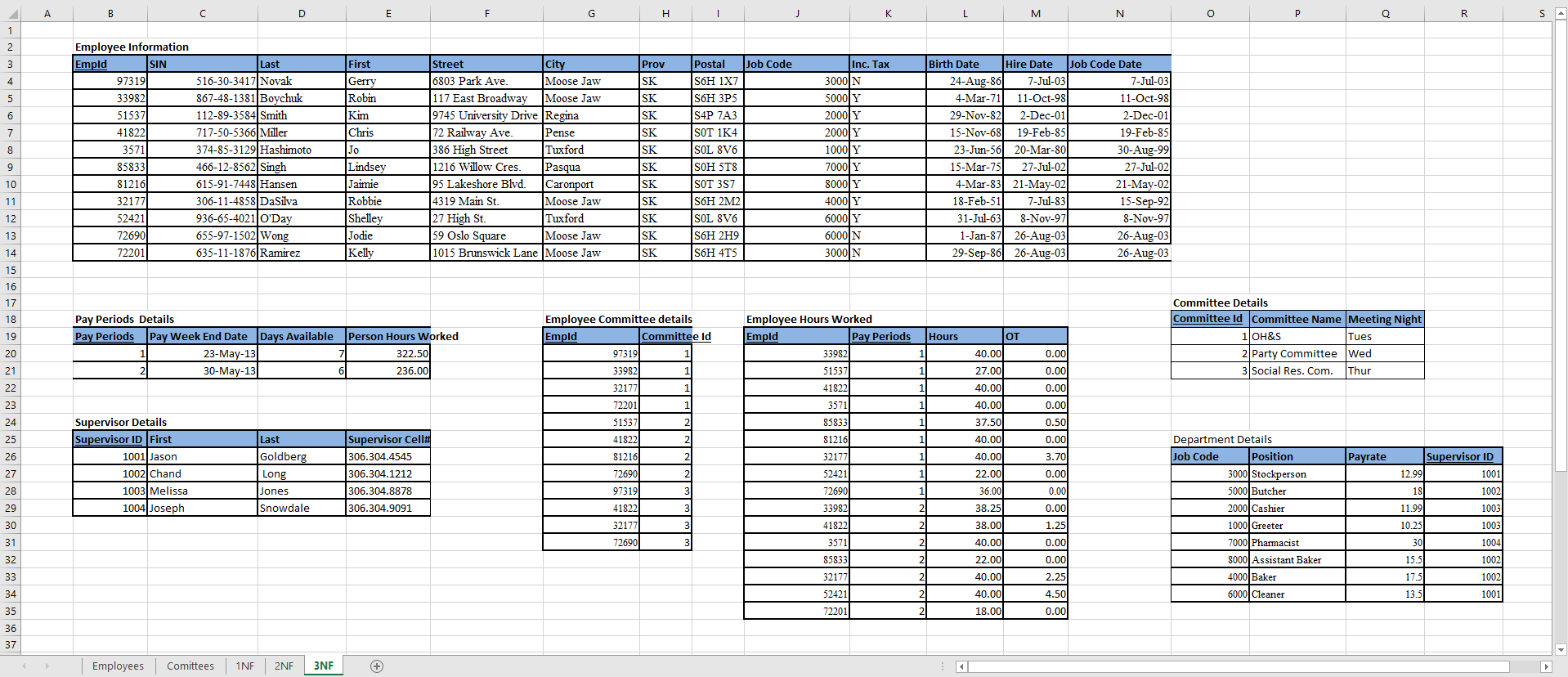
A transitive functional dependency is when changing a non-key column, might cause any of the other non-key columns to change.

3NF (Third Normal Form) Rules

Rule 1- Be in 2NF

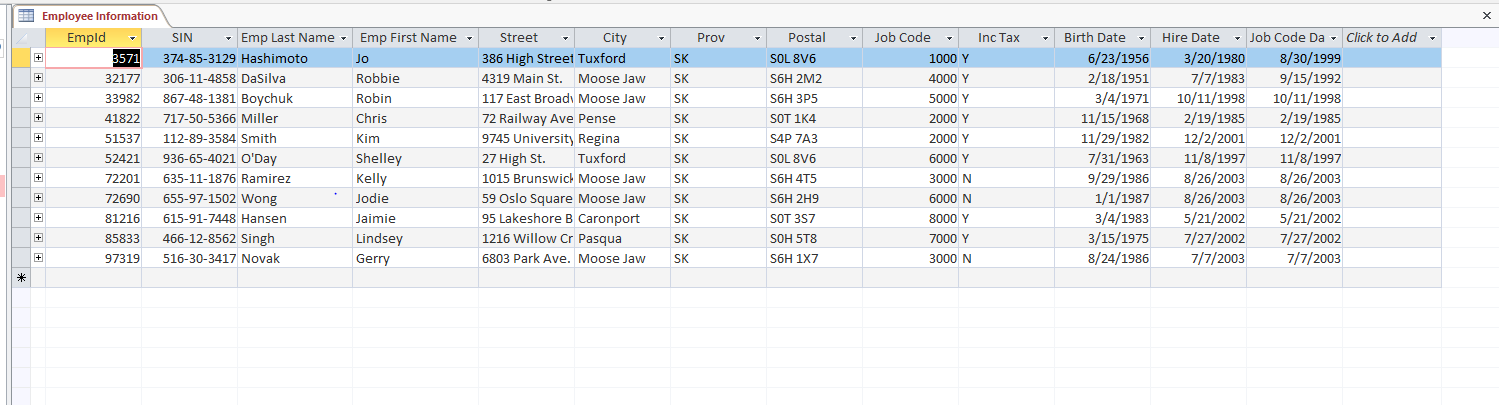
Rule 2- Has no transitive functional dependencies

Figure 1.2(3NF Relations)

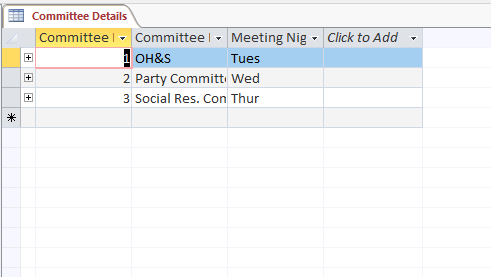


MS Access Relations

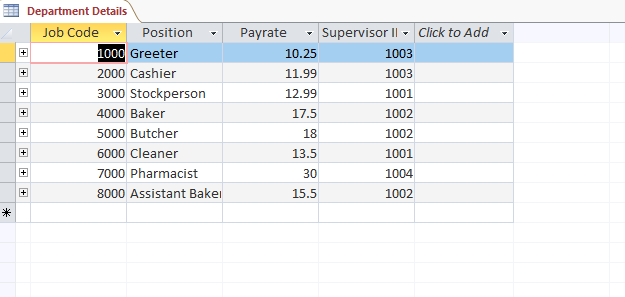
Employee Information



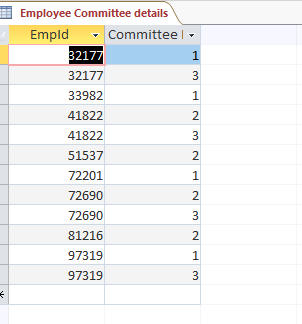
Committee Details



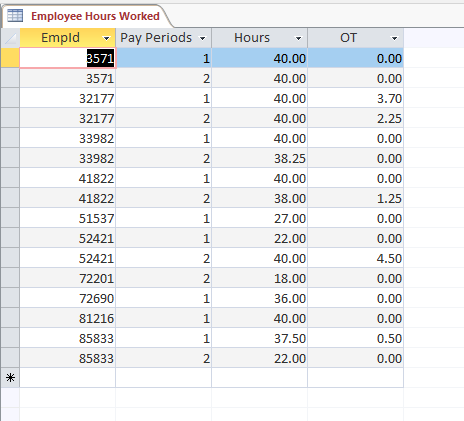
Department Details



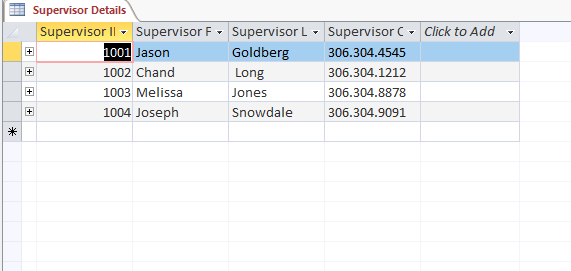
Employee Committee Details



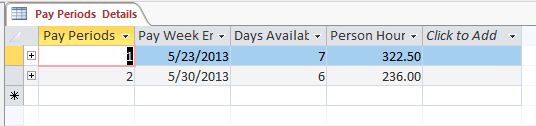
Employee Hours Worked



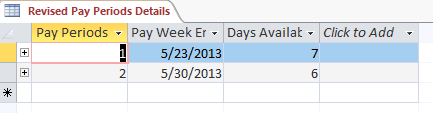
Supervisor Details



Pay Periods Details



Revised Pay Periods Details



Database design decisions

Database relationships diagram