**Database Modeling Project**

By

Oluwabunmi Olanrewaju

COMP 123

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

# Summary

This report addresses data redundancy through normalization process. This is a complete process that started with three separate un-normalized table (Employee information table, committee table and a table showing committee members), which was completed with seven normalised tables in 3NF. By the end of this report, you will notice that duplication has been removed (apart from the keys needed to establish the links between those tables).

The following normal forms were discussed in this section:

1. **First normal form:** A table is in the first normal form if it contains no repeating columns.
2. **Second normal form:** A table is in the second normal form if it is in the first normal form and contains only columns that are dependent on the whole (primary) key.
3. **Third normal form:** A table is in the third normal form if it is in the second normal form and all the non-key columns are dependent only on the primary key. If the value of a non-key column is dependent on the value of another non-key column more than the primary key, we have a situation known as transitive dependency. This was resolved by removing the columns dependent on non-key items to another table.

Table of Contents

[Summary 2](#_Toc531259758)

[Introduction 5](#_Toc531259759)

[Database Normalization 5](#_Toc531259760)

[Definition of Key Words: 5](#_Toc531259761)

[First Normal Form (1NF) 7](#_Toc531259762)

[Figure 2.1(Breakdown into 1NF from UNF) 7](#_Toc531259763)

[Second Normal Form (2NF) 9](#_Toc531259764)

[Figure 3.1 (2NF Relationship) 10](#_Toc531259765)

[Supervisor Details Table 10](#_Toc531259766)

[Weekly Hours Worked Details Table 11](#_Toc531259767)

[Employee Committee details 11](#_Toc531259768)

[Figure 3.2 (2NF Excel Tables) 12](#_Toc531259769)

[Third Normal Form (3NF) 13](#_Toc531259770)

[Employee Information Table 13](#_Toc531259771)

[Other Tables in 2NF 13](#_Toc531259772)

[Figure 4.1 (3NF Relationship) 14](#_Toc531259773)

[Figure 4.2 (3NF Tables) 15](#_Toc531259774)

[MS Access Relations 17](#_Toc531259775)

[Employee Information 17](#_Toc531259776)

[Committee Details 17](#_Toc531259777)

[Department Details 17](#_Toc531259778)

[Employee Committee Details 18](#_Toc531259779)

[Employee Hours Worked 18](#_Toc531259780)

[Supervisor Details 19](#_Toc531259781)

[Pay Periods Details 19](#_Toc531259782)

[Revised Pay Periods Details 19](#_Toc531259783)

[Database design decisions 20](#_Toc531259784)

[Database relationships diagram 20](#_Toc531259785)

[Committee table considerations 20](#_Toc531259786)

[Key field issue resolution 20](#_Toc531259787)

[Calculated query field solution 21](#_Toc531259788)

[Query Results 22](#_Toc531259789)

[Query 5.1 (Employees that worked more than 35 hours in any pay period) 22](#_Toc531259790)

[Query 5.2 (Employees that worked overtime hours in the 30-May-13 pay period) 22](#_Toc531259791)

[Query 5.3 (Employees job code, their manager’s name and their manager’s cell number Given an Employee ID) 22](#_Toc531259792)

[Query 5.4 (Skill sets by city) – Using Moose-Jaw as a Sample 23](#_Toc531259793)

[Query 5.5 (Employees that worked on certain Committee and their Job Description) 23](#_Toc531259794)

[Query 5.6 (List of employees being supervised and their job code given a supervisor last name) 23](#_Toc531259795)

[Query 5.7 (Query Showing Person Hours Worked for Individual Pay Period based on the Accountant concerns) 23](#_Toc531259796)

[Appendix A 24](#_Toc531259797)

[Sample Reports 24](#_Toc531259798)

[Report 6.1 - Employees that worked more than 35 hours in any pay period 24](#_Toc531259799)

[Report 6.2 – (Skill sets by city) – Using Moose-Jaw as a Sample 24](#_Toc531259800)

[Report 6.3 – (Employees that worked on certain Committee and their Job Description) – Using OH&S Committee as a sample: 25](#_Toc531259801)

[Report 6.3 – (List of employees being supervised and their job code given a supervisor last name) – using Goldberg as a sample 25](#_Toc531259802)

# Introduction

Data modeling is the analysis of data [objects](https://www.webopedia.com/TERM/O/object.html) and their relationships to other data objects. Data modeling is often the first step in [database](https://www.webopedia.com/TERM/D/database.html). it involves a progression from conceptual model to logical model to physical schema.

In the design of a data model, normalization is the process of adjusting table and [relations](https://gerardnico.com/data/modeling/relationship) to:

* eliminate certain types of data (redundancy | replication) to improve [consistency](https://gerardnico.com/data/property/consistency),
* produce a clearer and readable data model.
* provide maximum flexibility to meet future information needs by keeping tables corresponding to object types in their simplified forms.
* avoid update anomalies

## 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 complex tables into simpler smaller sub-tables, making it easier to handle and update the database without facing data loss.

### Definition of Key Words:

**Data Redundancy:** This means that the duplication of data should be kept to a minimum for several reasons. For example, it is unnecessary to store an employee's home address in more than one table.

**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 "un-normalized 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.

**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.

**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 un-normalized 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.

**Dependencies:** Structural problems within tables that causes database anomalies. The types of Dependencies we have include:

1. **Repeating Group:**  The existence of multiple values for a particular attribute in a specific record
2. **Partial Dependencies:** Occurs when one or more non-key attributes are dependent on only part of the primary key, rather than the whole key.
3. **Transitive Dependencies**: Occurs in a table where non-key attributes are dependent on another non-key attribute and independent of the table’s primary key.

# 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

## Figure 2.1(Breakdown into 1NF from UNF)

|  |  |
| --- | --- |
| 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 employees 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

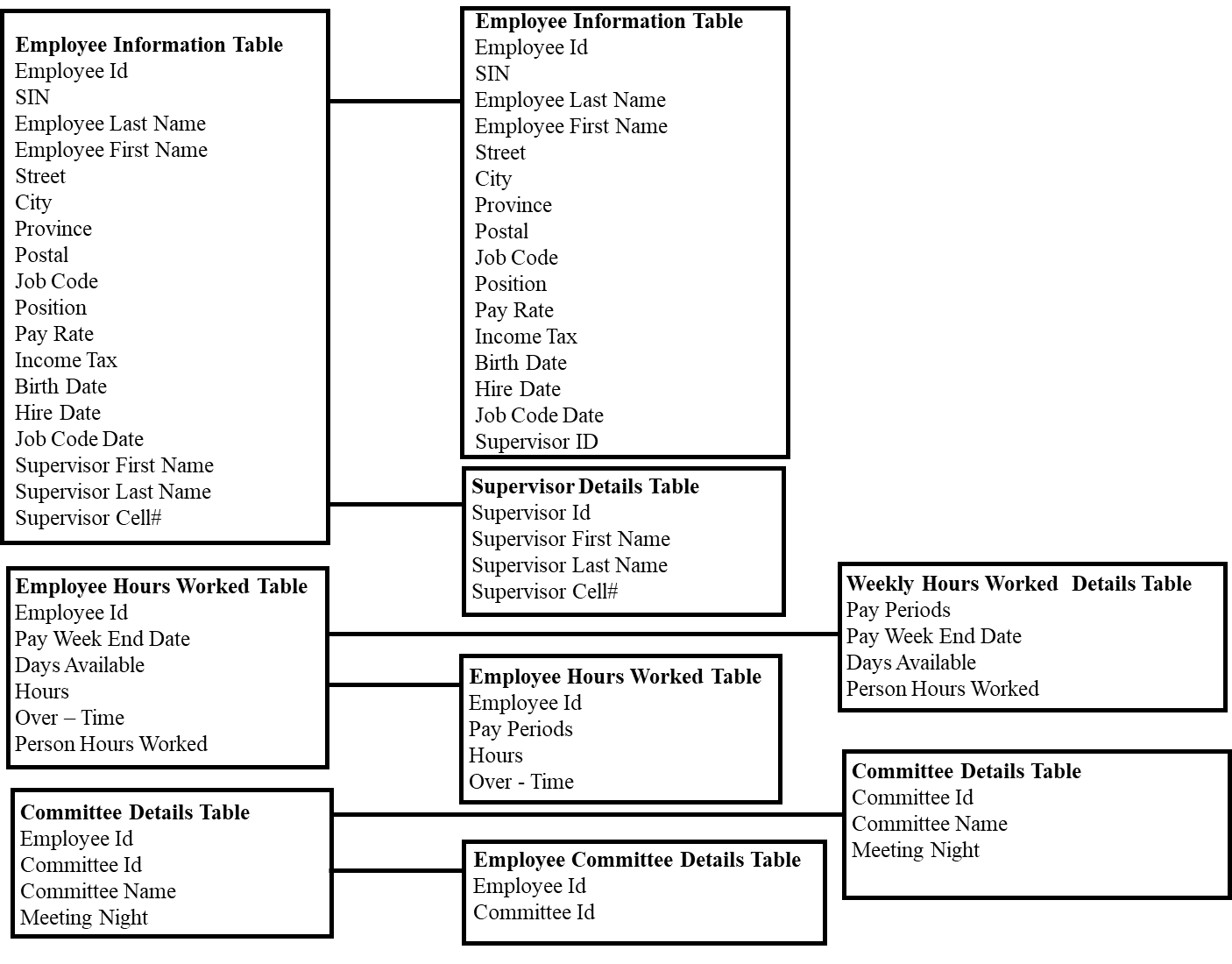
Here, each table in our 2NF will be discussed in details and what inform the decision to have those tables.

All attributes of 2NF in 1NF have been discussed except for “All non - key attributes are depende9nt on all parts of the primary key”. To ensure this, each non-key attribute in turn should be checked and the question to be asked: is this attribute dependent on **one part** of the primary key?

* If yes, the attribute to a new table with a **copy** of the **part** of the primary key it is dependent upon should be removed. The key it is dependent upon becomes the primary key in the new table.
* If no, other part of the key should be checked and repeat above process
* If still no, keep attribute in current table.

Base on the above note on 2NF, individual table in 2NF from the project will be discussed in details and reasons behind the decision to create them.

## Figure 3.1 (2NF Relationship)



The First set of tables (employee information table, employee hours worked table and committee table) are from 1NF. Recall the main essence of 1NF is to avoid date repetition. Hence, reason why the **Employee Hours Worked Table** was taken out from employee information table. Since an employee can work under the two pay week period (23-May-13 and 30-May-13). Also, we have a situation where an employee can be in more than one committee (for example Gerry Novak is in OH&S Committee and Social Res. Committee) and if this is reflected in the employee information table, some data such as name, address, birth date etc will be repeated, this is the reason we **Committee Details Table**.

Now that we have our data in 1NF (check attributes of 1NF above), next is to check if there are any non - key attributes not dependent on all parts of the primary key to achieve 2NF.

## Supervisor Details Table

From the employee information table in 1NF, the supervisor name, supervisor last name and supervisor cell are not dependent on employee ID as a supervisor can supervise more than one employee depending on their departments for example Chad Long is managing butchers and bakers. So these three columns were taken off to form a new table called Supervisor Details Table, introducing a primary key to the table using supervisor ID. The supervisor ID will be a foreign key in the Employee Information table so as to keep a relationship with the supervisor details table.

## Weekly Hours Worked Details Table

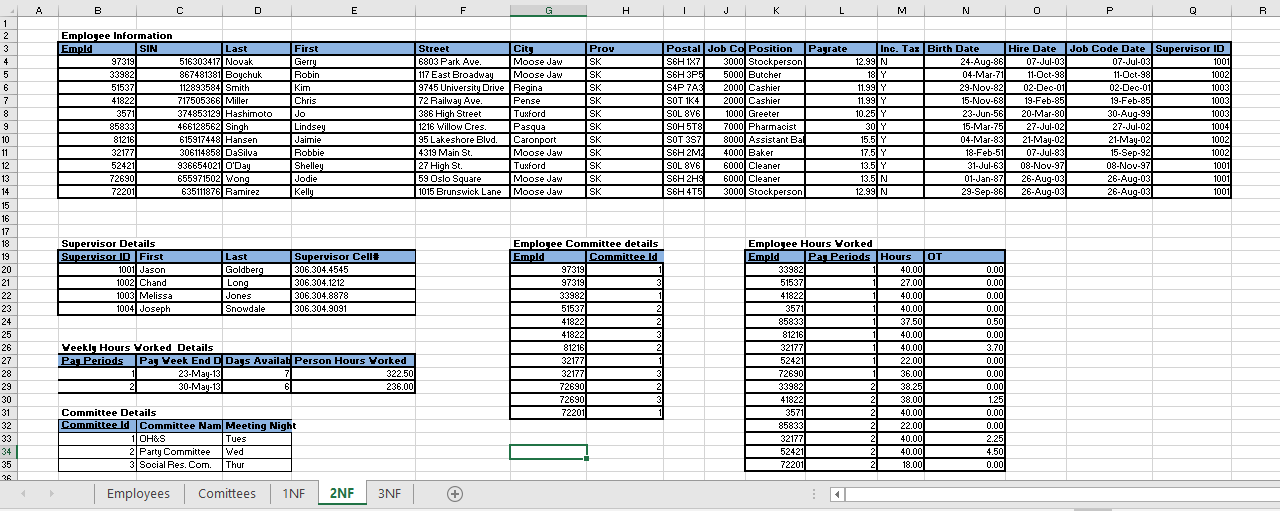
Under **Employee Hours Worked Table** in 1NF, there was a composite primary key that is multiple columns used to identify a record uniquely. These columns were employee ID and Pay Week End Date. However, days available and person’s hours worked were only dependent on a column (Pay Week End Date) and not on employee ID, so it was taken out to form a new table called **Weekly Hours Worked Details Table.** In the new table, the primary key was changed to pay periods (a new column) because the senior developer does not like to use dates as key fields. The pay periods can be seen on **Employee Hours Worked Table** as a foreign key in order to keep the relationship with the new table **Weekly Hours Worked Details Table.**

## Employee Committee details

From **Committee Details Table** in 1NF, we had some columns not dependent on the whole primary key. The committee name and meeting night columns are associated to **Committee ID** and not on **Employee ID** as employee isn’t a determinant of name of the committee neither the meeting night. So the composite primary key was taken off to retain just the **Committee ID** on the table as other non-primary key column depends on it. The column taken off **Committee ID** and **Employee ID** form a new table called **Employee Committee details** to retain the relationship between the **Employee Information Table** and **Committee Details Table.** Note, the committee ID is not taken to the **Employee Information Table,** as this will cause data repetition since an employee can be in more than one committee.

The screenshot of 2NF relations can be seen below:

## Figure 3.2 (2NF Excel Tables)



# Third Normal Form (3NF)

A table is in third normal form if:

* 1. Be in 2NF
  2. Has no transitive functional dependencies
  3. Each field has a unique name
  4. It has a primary key

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

The process in 3NF 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 primary key in the new table.
* **Leave** the non-key attribute, upon which it is dependent, in the original table and it becomes a **foreign key** in the original table.

Checking individual table from transitive functional dependencies:

## Employee Information Table

There are three non-key columns (position, Pay-rate and supervisor ID) dependent on another non-key column (Job Code) in that when the job code of a particular employee is change, this inevitable affect the three columns mentioned earlier. For instance, if Novak Gerry is being move from a Job code 2000 to 3000, it will be necessary to update the position, pay-rate and the current supervisor of Novak to reflect this change.

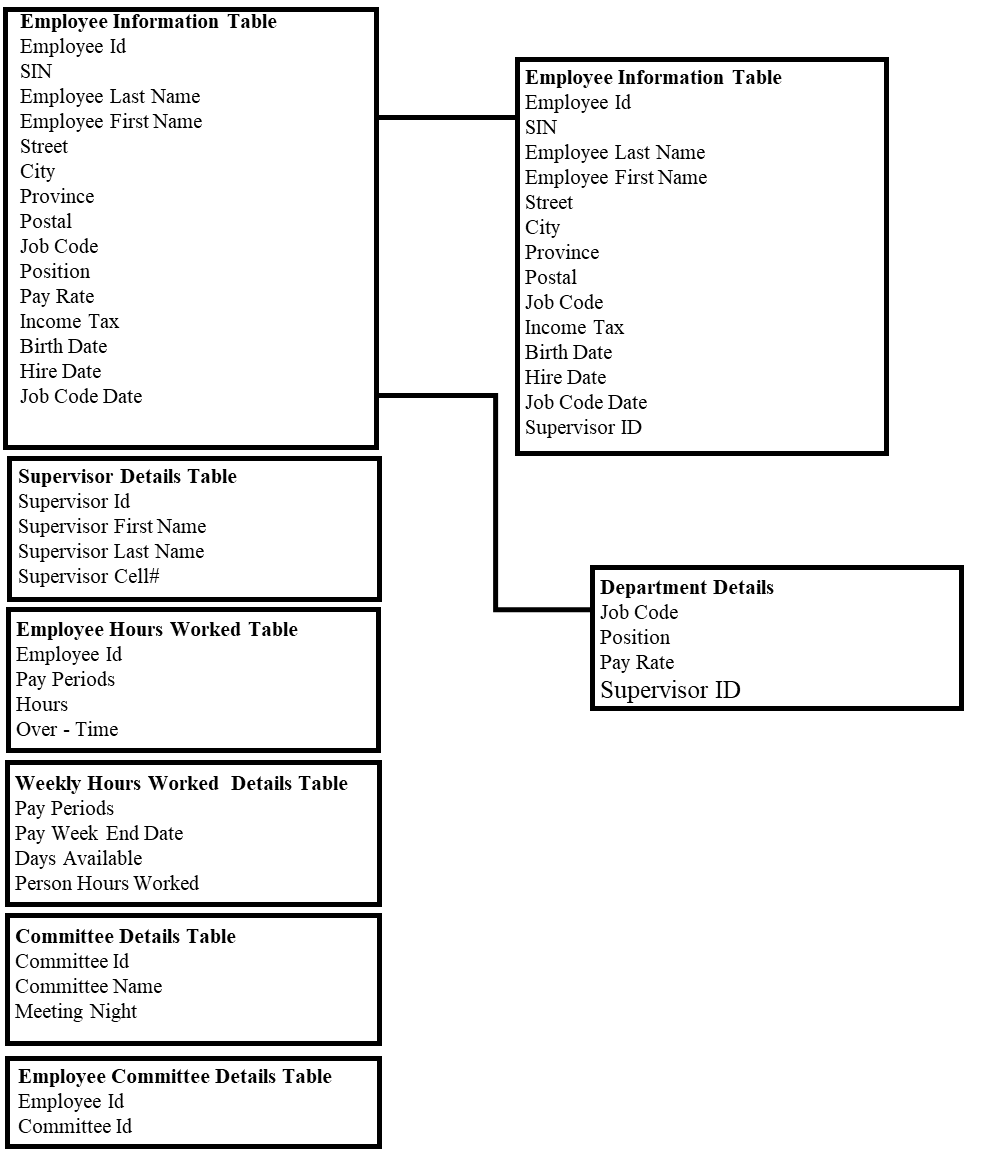
These columns (position, Pay-rate and supervisor ID) with Job code is taken out from **Employee Information Table** to form a new table called **Department Details Table** but the job code column is retained in the **Employee Information Table** to keep the relationship.

## Other Tables in 2NF

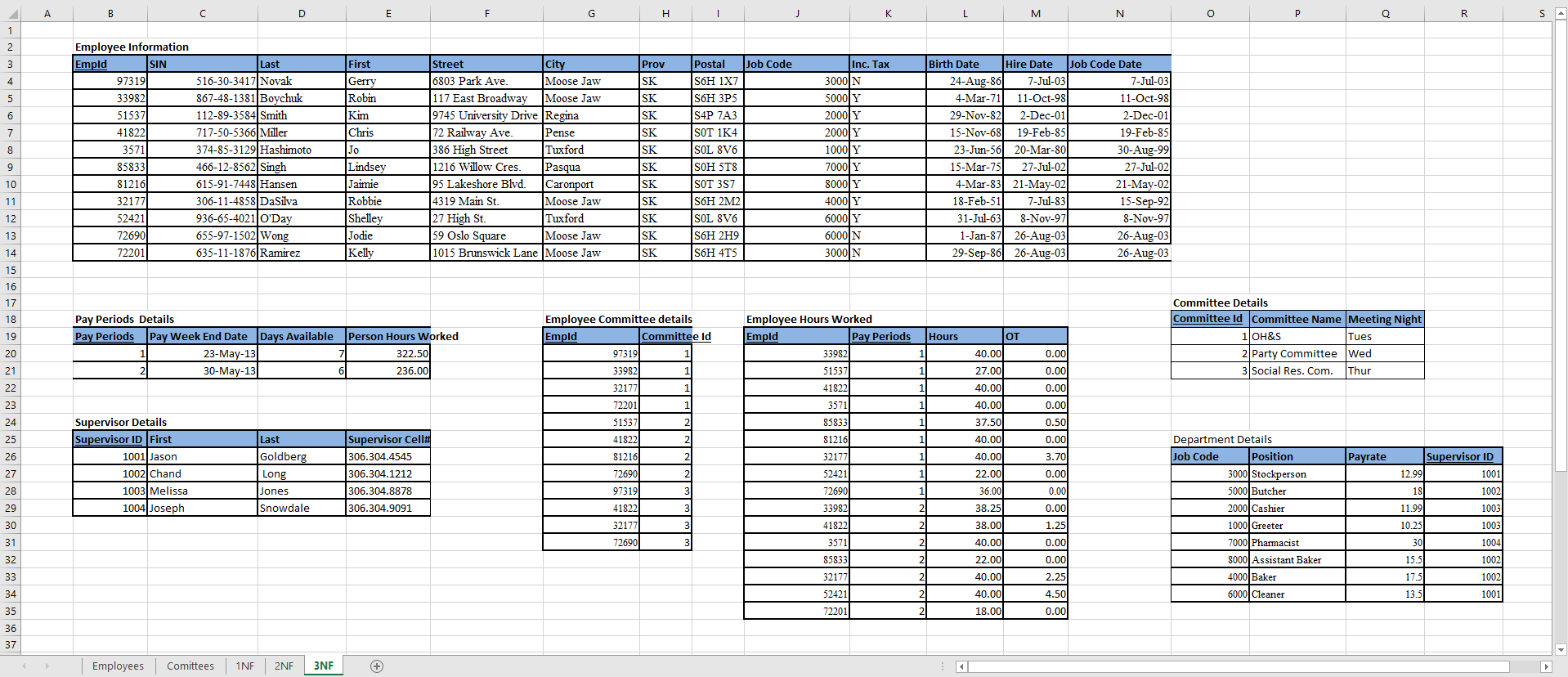
None of the non key columns depends on another non key column in any of the other table. They solely depend on their respective primary key.

Below are two diagram; first showing the transmission from 2NF to 3NF as Employee Information Table is further broken down while the other tables retain their form. The second diagram is the 3NF Relations

## Figure 4.1 (3NF Relationship)



## Figure 4.2 (3NF Tables)



Earlier, I said the essence of data normalization was to decompose tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anomalies. Since the 3NF is completed, the next step is to check if the 3NF relations have been able to solve the issue of data redundancy and anomalies.

From the tables, data redundancy has been remove since there are no repetition of data except for keys needed to establish the links between the various tables.

Anomaly:

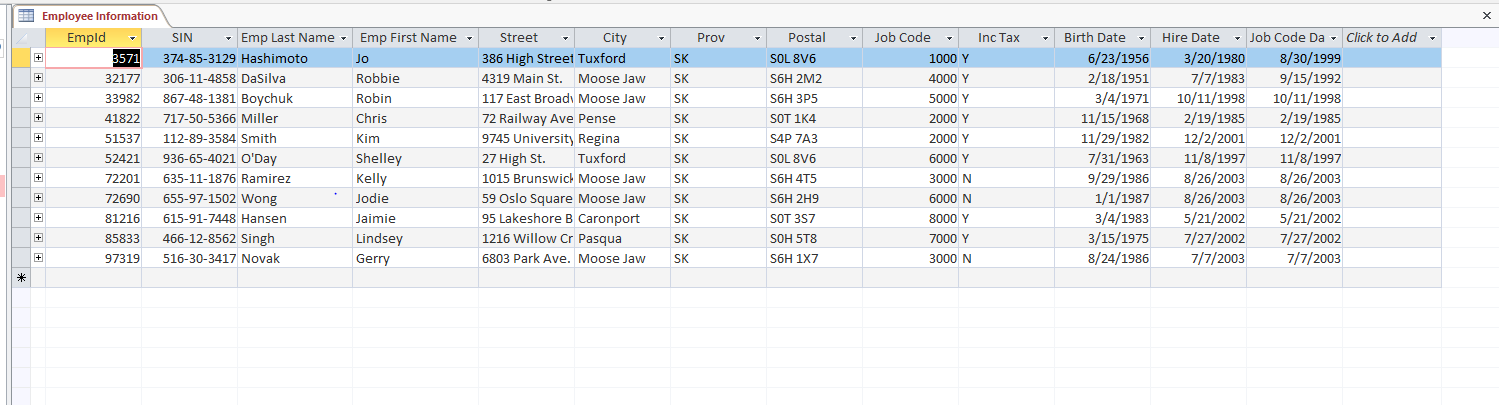
Deletion Anomaly: Information about a particular employee can now be deleted without affecting the department information.

Insertion Anomaly: A new department can now be created into the database without necessarily transferring some employees immediately.

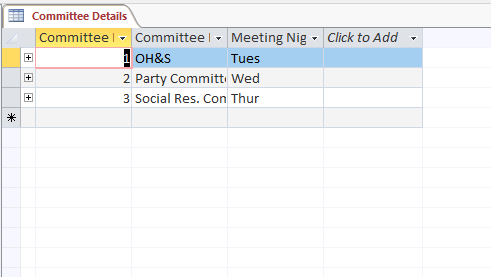
Update Anomaly: Since information are not expressed in multiple lines, issue of partial update have been resolved. There is no longer need to adjust multiple lines just to update same information due to data repetition.

# MS Access Relations

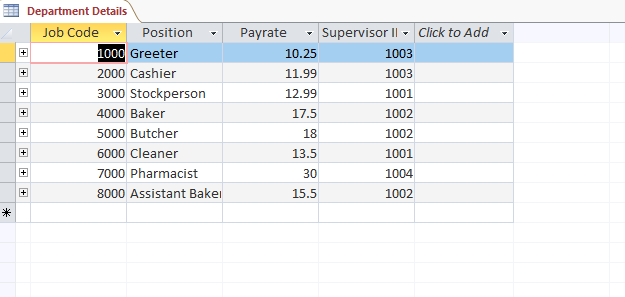
## Employee Information



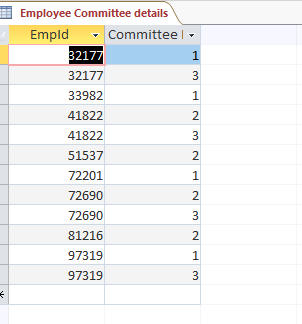
## Committee Details



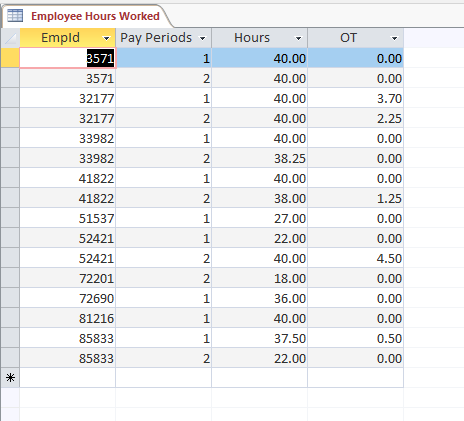
## Department Details



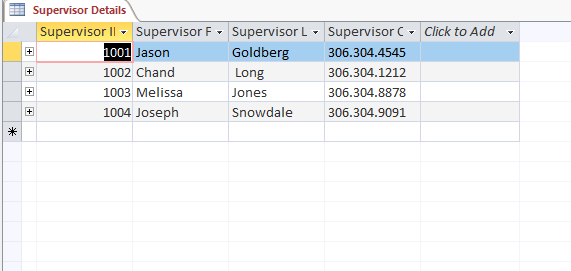
## Employee Committee Details



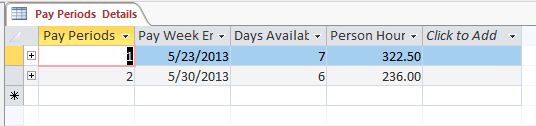
## Employee Hours Worked



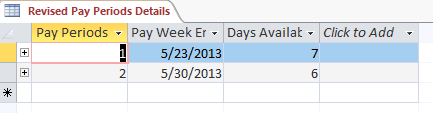
## Supervisor Details



## Pay Periods Details

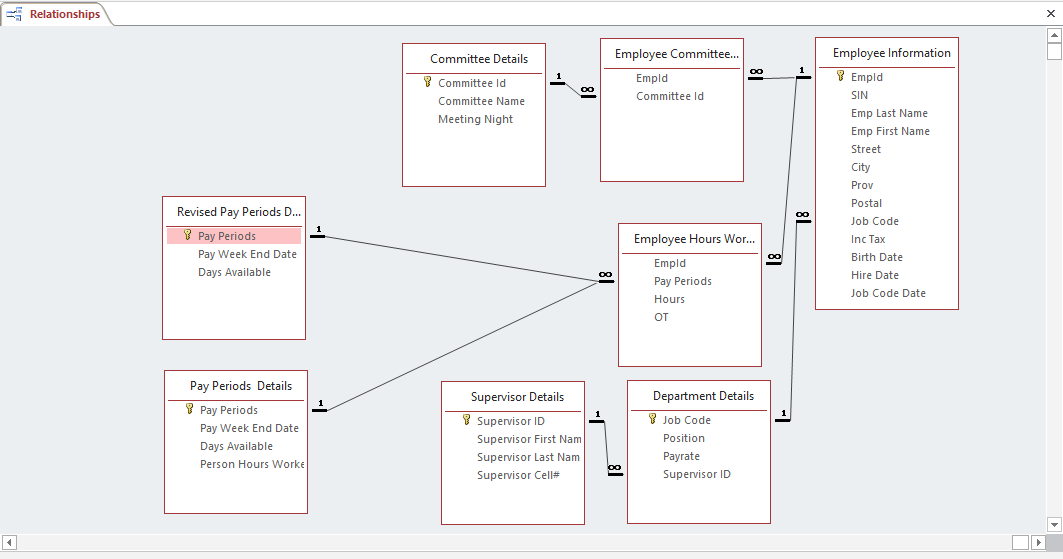


## Revised Pay Periods Details



# Database design decisions

## Database relationships diagram



I have two Pay Periods Details Table in that the accountant is concerned that total person hours currently do not appear to include the over- time and sometimes the number might be revised for an employee, and be not changed in the table. So I took   off the person hours worked in the revised to proof it is ok to take it off and still have the data available through a query. Apart from that, basically all similar column title from various table are linked together e.g. EMPID on Employee Information table is link to EMPID on employee committee table.

## Committee table considerations

I kept the committee details table showing the committee ID, Committee Name and Meeting night. However, a new table called Employee Committee Table that connect the committee details table and Employee Information Table. This table shows the committees the various employee serves on.

## Key field issue resolution

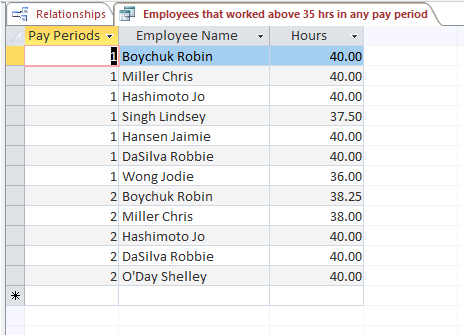
Since the senior developer does not like to use dates as key fields, Pay Periods Details Table was modified in the database to remove pay week end date as a primary field. This was done by creating a new field called pay periods. The pay period field represents the pay week end date for each week.

## Calculated query field solution

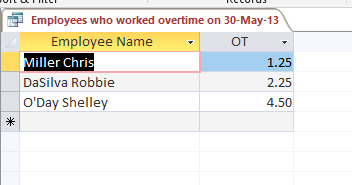
The only calculated query field is in query 5.7. This query calculate the person hours worked plus the overtime under each period. This was achieved by summing up the individual employee hour worked with their overtime. Then, this sum is further group together under each period and sum up to reflect the person hours worked under each period.

# Query Results

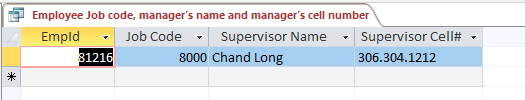
## Query 5.1 (Employees that worked more than 35 hours in any pay period)



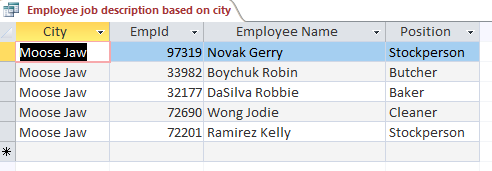
## Query 5.2 (Employees that worked overtime hours in the 30-May-13 pay period)



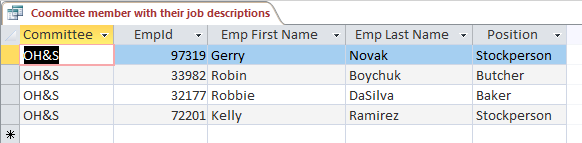
Query 5.3 (Employees job code, their manager’s name and their manager’s cell number Given an Employee ID) – This is just a sample of an employee



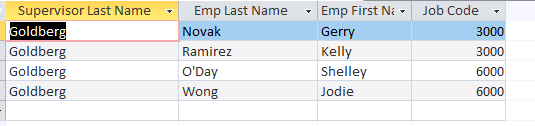
## Query 5.4 (Skill sets by city) – Using Moose-Jaw as a Sample

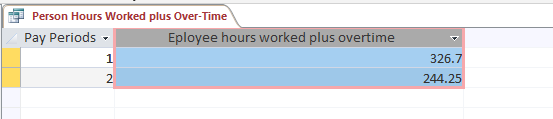


Query 5.5 (Employees that worked on certain Committee and their Job Description) – Using OH&S Committee as a sample:



Query 5.6 (List of employees being supervised and their job code given a supervisor last name) – using Goldberg as a sample

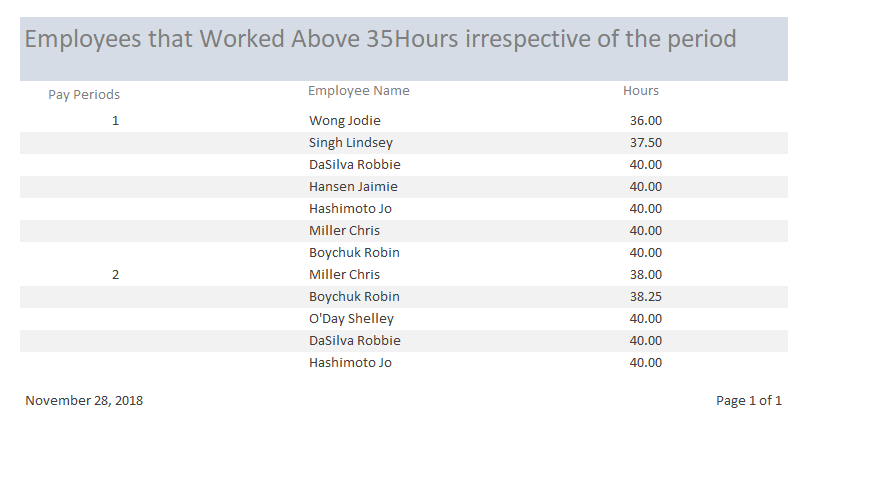


Query 5.7 (Query Showing Person Hours Worked for Individual Pay Period based on the Accountant concerns)

# Appendix A

## Sample Reports

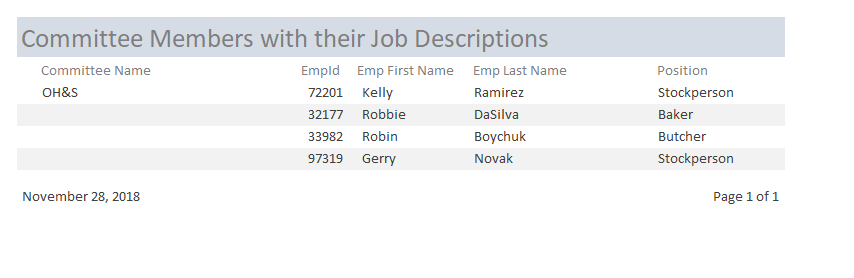
### Report 6.1 - Employees that worked more than 35 hours in any pay period



### Report 6.2 – (Skill sets by city) – Using Moose-Jaw as a Sample



### Report 6.3 – (Employees that worked on certain Committee and their Job Description) – Using OH&S Committee as a sample:



### Report 6.3 – (List of employees being supervised and their job code given a supervisor last name) – using Goldberg as a sample

