Analysis, Design and Implementation of a Relational Database System PRAC1

Kyle Cuthbert – st20249341

Table of Contents

[Normalization 3](#_Toc121853451)

[Data Specified 3](#_Toc121853452)

[Separate Normalization 4](#_Toc121853453)

[Revised Normalization 8](#_Toc121853454)

[Schema 11](#_Toc121853455)

[Entity Relation Diagram 13](#_Toc121853456)

[Data dictionary 14](#_Toc121853457)

[Creation Script 16](#_Toc121853458)

[Insert Data Script 17](#_Toc121853459)

[Various Query Examples 18](#_Toc121853460)

# Normalization

## Data Specified

For the assessment brief we have been given the below information which represents the requirements of the users:

* For students
  + Name
  + Place of birth
  + Date of birth
  + Home School
  + City
  + Country
  + Contact number
  + Courses studied and grades (at home school not from course list)
  + Recommendation letter *(Yes/No Field, to indicate if it is included)*
  + Invitation from the host *(Yes/No Field, to indicate if it is included)*
* For researchers:
  + Name
  + Place of birth
  + Date of birth
  + Home university
  + City
  + Country
  + Contact number
  + Required fund (*Amount required*)
  + Proposal (*One proposal only):*
    - Intro
    - Background
    - Risks
    - Methods

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Course title** | **Course code** | **Host university** | **Duration** | **Type** |
| Computer Security | CS1022 | Cardiff Met University | 36 months | BSc |
| Data Science | DS2221 | Swansea University | 12 months | MSc |
| Bioinformatics | BI0011 | MIT | 36 months | PhD |

|  |  |  |
| --- | --- | --- |
| **Grant title** | **Available fund** | **Duration** |
| Short term | 20K | 6 months |
| ESRC | 250K | 24 months |
| Welsh Government | 150K | 12 months |

## Separate Normalization

The first step that we must take in the normalization process is to put the related data together to compose a ‘bad’ table and populate it with example test data. This ‘bad’ table will then be decomposed into several smaller ‘good’ tables that will be the process of normalization.

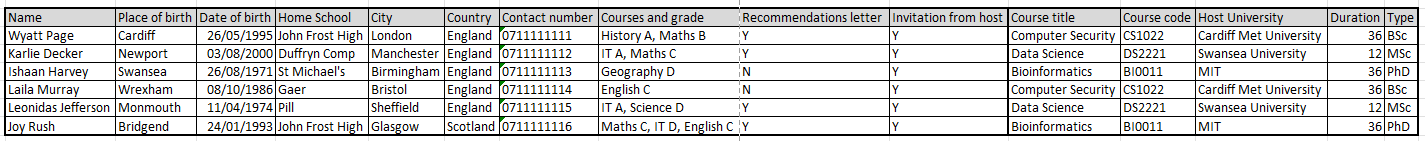
The stages of normalization are called normal forms with the higher the number of normal forms representing a better design.

Having interpretated the information provided I initially decided that two separate normalization processes will be carried out, one for the student data and one for the researcher data. However, after completing the steps of normalization I noticed that I could link the two separate sets of data with a table for common personal data.

Below I have explained the steps taken for both normalizations.

Test data has been added to represent the un-normalized data:

**Student Data**



**Researcher**



The next stage that we must take is normalizing the data to 1NF. For data to meet 1NF a primary key must be included.

As part of 1NF columns need to be unique, i.e., each cell can only contain one value and cannot have mixed types of data (there are no examples of mixed data types here). As a result, the ‘Courses and grades’ column will need to be separated into two separate columns.

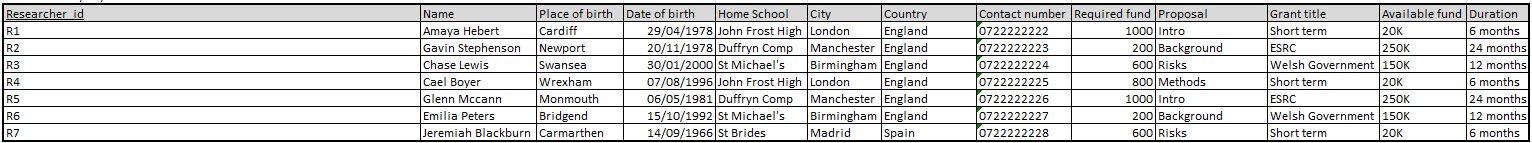
The next stage of making the data conform to 1NF will be to remove the repeating attributes. The Course Code will remain in the student table as the foreign key.

**Student**

Table

Description automatically generated

**Researcher**



The next step to be taken as part of the normalization process is to take the data from 1NF to 2NF. Separate tables should be created for sets of values that apply to multiple records. These must be related with a foreign key. Please note that the student grades have a composite key.

**Student**

Graphical user interface, application, table

Description automatically generated

**Researcher**

**Graphical user interface

Description automatically generated with medium confidence**

The next stage of normalization is to take the form of 3NF which is to eliminate any transitive dependencies where a non-key attribute does not depend on another non-key attribute. In the student data, we do not have any transitive dependencies, but we do in the researchers.

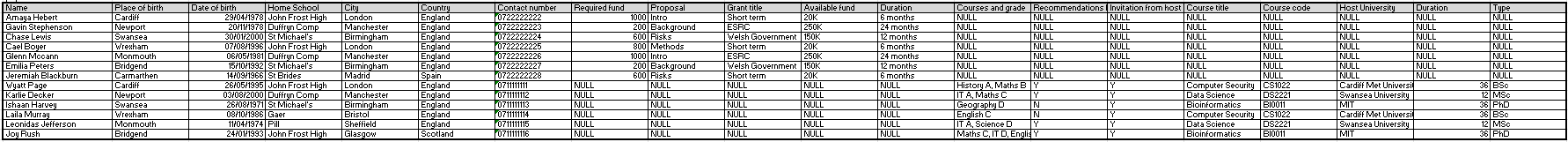
**Graphical user interface

Description automatically generated**

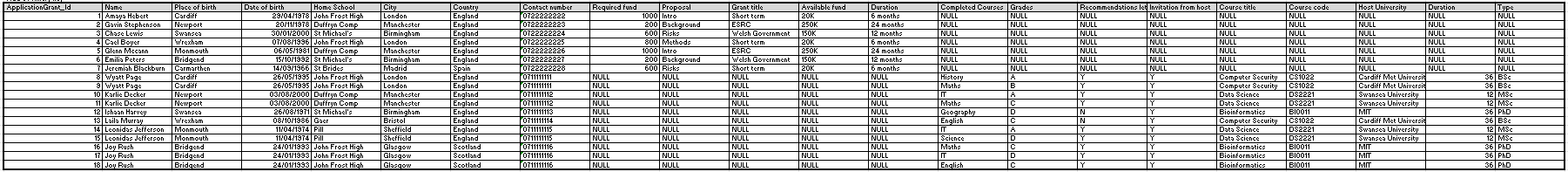
## Revised Normalization

As mentioned above I noticed that the was common data between the two separate sets of data and yet there was no link between the two. This meant that there could never be any cross over e.g., a researcher that was also a student or the ability to add another ‘role’. I then decided to perform the steps of normalization again which can be seen below and in the **‘Joint’** tab of the **‘Normalisation.xlsx’** spreadsheet:

**Un-normalized**



**1NF**



**2NF**

**Table

Description automatically generated**

**3NF**

**Table

Description automatically generated**

# Schema

After the first attempt at normalization, I created a schema which highlighted the disconnect between the two sets of data:

Graphical user interface, diagram

Description automatically generated

Once the normalization had been improved, I created a new schema which can be found in the file named **‘UniversitySchema.png’**.

Diagram

Description automatically generated

# Entity Relation Diagram

The entity relation diagram can be opened using diagrams.net(<https://www.diagrams.net/>) and can be found in the file named **‘ERD-EntityRelationshipDiagram.drawio’**.

Diagram

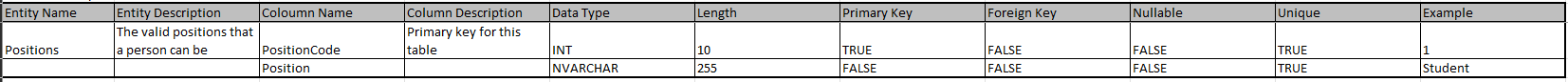
Description automatically generated

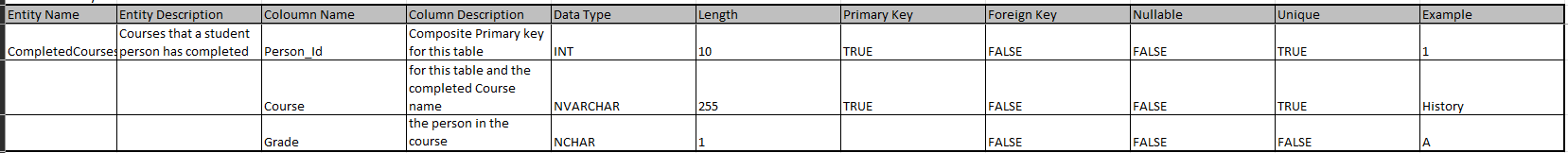
# Data dictionary

To assist with the database design a data dictionary has been created and is named **‘Data Dictionary.xlsx’**.

A screenshot of a computer

Description automatically generated



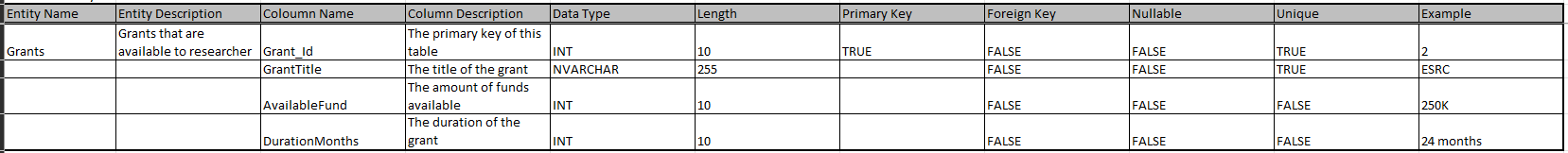


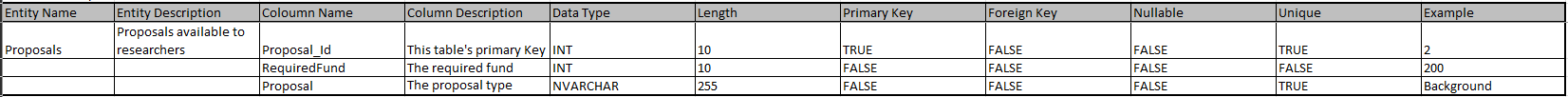
Graphical user interface

Description automatically generated with medium confidence

Calendar

Description automatically generated





Graphical user interface, text, application

Description automatically generated

# Creation Script

The database creation script is named **‘CreateTables.sql’**.

Run the ‘CREATE DATABASE’ in isolation and then comment the command out.

Remove the comment characters for the ‘USE’ command and execute the query.

Graphical user interface, text, application

Description automatically generated

# Insert Data Script

The script to insert test data is named **‘InsertDataIntoTables.sql’**.

Graphical user interface, text, application, email

Description automatically generated

Text, letter

Description automatically generated

A picture containing text

Description automatically generated

# Various Query Examples

To test various queries and how the tables are connected I created a file containing various queries. This file is called **‘VariousQueries.sql’.**

**Text

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

**Text

Description automatically generated with low confidence**

**Text

Description automatically generated**

Text

Description automatically generated