Normalisation

Databsase Design

**Normalisation** is the process of organizing the columns (attributes) and tables (relations) of a relational database to minimize data redundancy.

Being able to do data analysis more easily is reason enough for an organization to engage in data normalization. There are, however, many more reasons to perform this process, all of them highly beneficial. One of the most notable is the fact that data normalization means databases take up less space[1].

**A primary key** is a key in a relational database that is unique for each record. It is a unique identifier, such as a driver license number, telephone number or employee ID. It cannot contain null values. A relational database must always have one and only one primary key. Primary keys typically appear as columns in relational database tables.

**A foreign key** is a field or collection of fields in one table that uniquely identifies a row of another table. It is a column or group of columns in relational database table that provides a link between data in two tables. If there is more than one table, they are linked by foreign keys. The foreign key constraint is used to prevent actions that would destroy links between tables and prevent invalid data from being inserted into the foreign key column, because it has to be one of the values contained in the table it points to[2][3].

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EmployeeID\* | EmployeeName | Salary | DeptName | DeptLocation | DeptBudget |
| 100 | Sean | 35,000 | Sales | Dublin | 750,000 |
| 101 | Mary | 36,000 | Sales | Dublin | 750,000 |
| 102 | John | 40,000 | Sales | Dublin | 750,000 |
| 104 | Albert | 55,000 | R&D | Galway | 1,500,000 |
| 105 | Conor | 52,000 | R&D | Galway | 1,500,000 |
| 106 | Meave | 50,000 | R&D | Galway | 1,500,000 |
| 107 | Tom | 50,000 | R&D | Galway | 1,500,000 |
| 108 | Alice | 44,500 | HR | Limerick | 250,000 |

**Proposed Employee table**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| EmployeeID\* | EmployeeName | EmployeeSurname | BirthDate | Gender | HireDate | PhoneNumber | DeptNo |
| 100 | Sean | Moran | 20/05/1975 | M | 15/02/2019 | 52136987 | 100 |
| 101 | Mary | King | 15/02/1980 | F | 14/05/2018 | 58932145 | 100 |
| 102 | John | Long | 22/06/1990 | M | 22/05/2018 | 21587624 | 100 |
| 104 | Albert | Higgins | 30/11/1978 | M | 01/06/2017 | 15978963 | 200 |
| 105 | Conor | Arkins | 05/12/1988 | M | 09/08/2017 | 45698732 | 200 |
| 106 | Meave | O’Connor | 30/06/1981 | F | 25/08/2015 | 85469875 | 200 |
| 107 | Tom | Hanks | 07/01/1979 | M | 02/04/2016 | 12914569 | 200 |
| 108 | Alice | White | 19/07/1977 | F | 05/09/2017 | 45879365 | 300 |

Primary Key – EmpolyeeID. Each record needs to be unique and it is true in this database. This field should not be null and changed.

The “DeptNo” column in the Employee table is a FOREIGN KEY in the Employee table.

Thedata which is entered in the Employee table would be constrained by this foreign key reference. “DeptNo” that exist in the Department table may be referenced in the employee table.

**Proposed Department table**

|  |  |  |  |
| --- | --- | --- | --- |
| DeptName | DeptNo\* | DeptLocation | DeptBudget |
| Sales | 100 | Dublin | 750,000 |
| R&D | 200 | Galway | 1,500,000 |
| HR | 300 | Limerick | 250,000 |

The “DeptNo” column in the Depatrment table is the PRIMARY KEY in the Department table.

The Department table could be linked to Employee table to see what employees belong to what departments and Employees to Department to see what is the name, location and budget of the certain department.

**Proposed Salaries table**

|  |  |  |  |
| --- | --- | --- | --- |
| EmployeeID | Salary | From\_date | To\_date |
| 100 | 35,000 | 15/02/2019 | 02/08/2020 |
| 101 | 36,000 | 14/05/2018 | 02/08/2020 |
| 102 | 35,000 | 22/05/2018 | 15/09/2019 |
| 102 | 40,000 | 16/09/2019 | 08/08/2020 |
| 104 | 50,000 | 01/06/2017 | 31/05/2018 |
| 104 | 52,000 | 01/06/2018 | 31/05/2019 |
| 104 | 55,000 | 01/06/2019 | 02/08/2020 |
| 105 | 50,000 | 09/08/2017 | 31/08/2018 |
| 105 | 52,000 | 01/09/2018 | 02/08/2020 |
| 106 | 45,000 | 25/08/2015 | 31/08/2017 |
| 106 | 48,000 | 01/09/2017 | 31/08/2019 |
| 106 | 50,000 | 01/09/2019 | 02/08/2020 |
| 107 | 45,000 | 01/04/2016 | 31/03/2018 |
| 107 | 48,000 | 01/04/2018 | 31/03/2019 |
| 107 | 50,000 | 01/04/2019 | 02/08/2020 |
| 108 | 43,000 | 05/09/2017 | 30/09/2019 |
| 108 | 44,500 | 01/10/2019 | 02/08/2020 |

The Salary table was created to link Employee table and it would be done by “EmployeeID” as a foreign key. It could also be linked to Department table using “DeptNo” as a foreign key but I think salaries/wages are more attributed to the employees than departments. Salary table shows history of individual salaries of each employee and whoever analyzes the data from Salary table can easily get to dates of employments and increases of wages. Very useful data but should be confidential and not seen by everyone.

First Normal Form (1NF) rules in this database are met.

Second Normal Form (2NF) is also met as there is a primary key – EmployeeID.

3NF(Third Normal Form) rules not met as there are transitive functional dependencies. A transitive dependency exists when there is an intermediate functional dependency[4].

# References

[1] <https://www.import.io/post/what-is-data-normalization-and-why-is-it-important/> (02/08/2020)

[2] <https://www.geeksforgeeks.org/difference-between-primary-key-and-foreign-key/> (02/08/2020)

[3] <https://www.w3schools.com/sql/sql_foreignkey.asp> (02/08/2020)

[4] <https://en.wikipedia.org/wiki/Third_normal_form> (02/08/2020)

The lecture slides presented in the course – Module: Applied Databases, Gerard Harrison – Higher Diploma in Data Analytics, 2020