

Assignment 2: Normalization

Introduction

The Assignment requires to normalize given data by converting to all three Normal Forms. In simpler terms, data has to be processed to be in organized form. By sticking to rules, new tables and attributes should be created to eliminate redundancy and inconsistency.

Explanation of Data

Provided data includes information about textbooks that are used in various courses. It includes the following details:

CRN: ID number of course

ISBN: ID of book

Title: Title of book.

Authors: Authors of book.

Edition: Edition of book.

Publisher: Publisher of book.

Publisher Address: Address of Publisher.

Pages: Number of pages in book.

Year: Year the book published.

Course Name: Name of the course for which the book is learned at.

ER-Diagram of Unnormalized Data



Organization

- Introduction
- Explanation of Data
- ER-Diagram of Unnormalized data
- Organization
- 1st Normal Form
 - Rule
 - Problem and Solution
 - ER-Diagram
- 2nd Normal Form
 - Rule
 - Problem and Solution
 - ER-Diagram
- 3rd Normal Form
 - Rule
 - Problem and Solution
 - ER-Diagram
- ChatGPT Solution
- Comparison

1st Normal Form

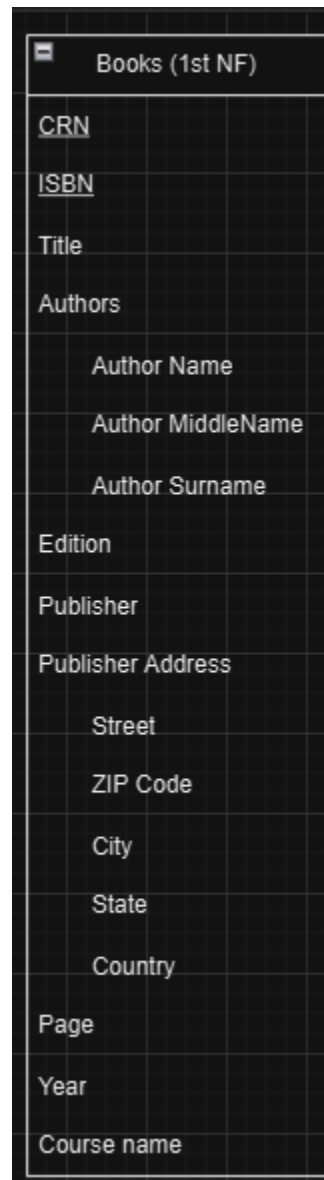
Rule

According to 1st Normal Form there should be neither atomicity nor redundancy in table.

Problem and Solution

Authors attribute of given table violates the 1st Normal Form by having several authors in the same row. The solution to this issue is to separate each author into a different row. Additional to that, to make data more organized and practical to manipulate, converted Author and Publisher Address attributes into Composite attribute. Author is divided into Author Name, Author MiddleName, and Author Surname attributes. Author Name and Author Surname are not null as they cannot be empty, while Author MiddleName accepts null values, owing to not all authors having middle names. Publisher Address is divided into Street, Zip code, City, State, Country attributes. Street, City, and Country are not null as they are mandatory fields to define address. State can accept null as there are countries without states, and ZIP code accepts null due to some addresses not containing it.

ER-Diagram



2nd Normal Form

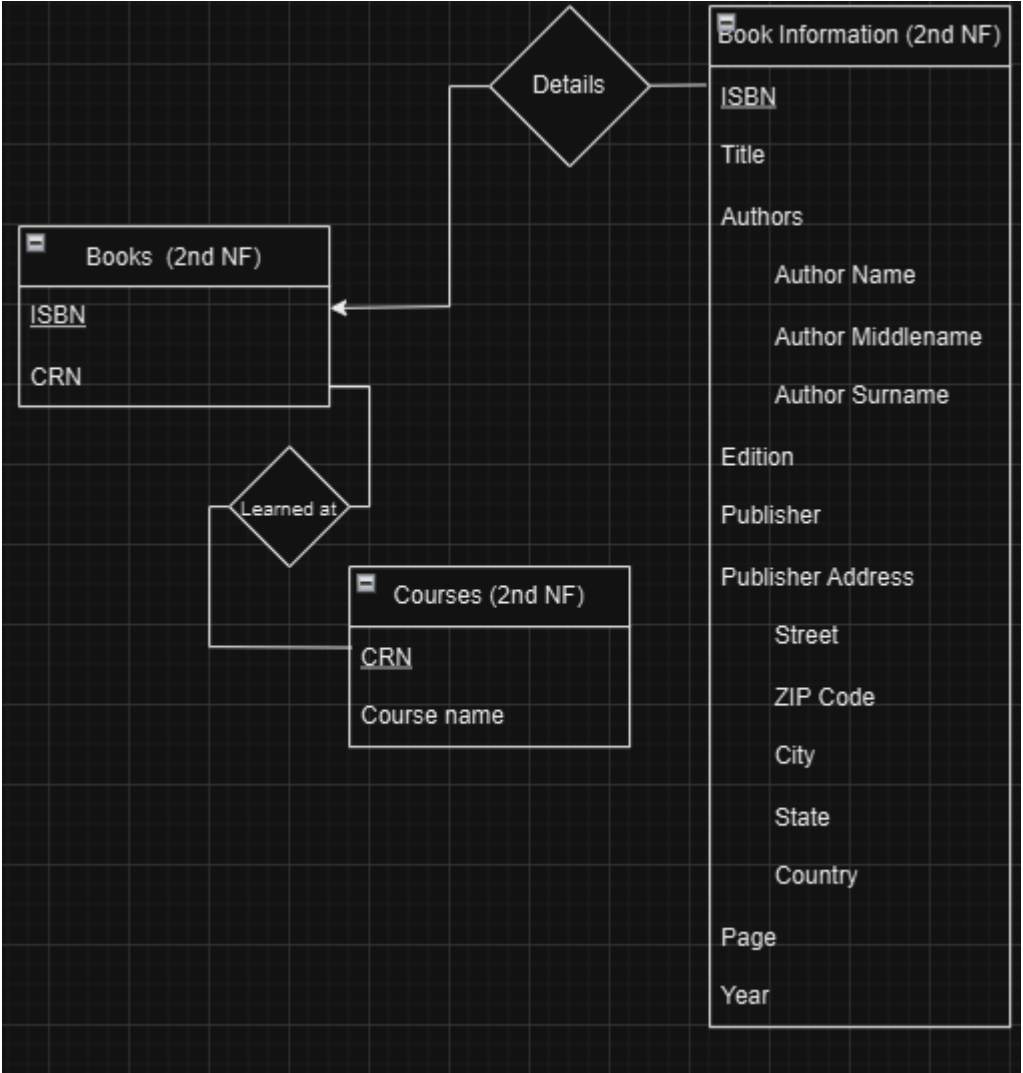
Rule

According to 2nd Normal Form, 1st Normal Form should be satisfied, and partial dependencies should be eliminated.

Problem and Solution

In the 1st Normal Form ISBN and CRN are primary keys. Non-key attributes not being dependent on both means partial dependency. Course Name attribute is only dependent on CRN. Therefore, it is moved to a new table called courses. All the rest of the non-key attributes are dependent on ISBN. Thus, all of them are relocated to the new table called Book Information.

ER-Diagram



3rd Normal Form

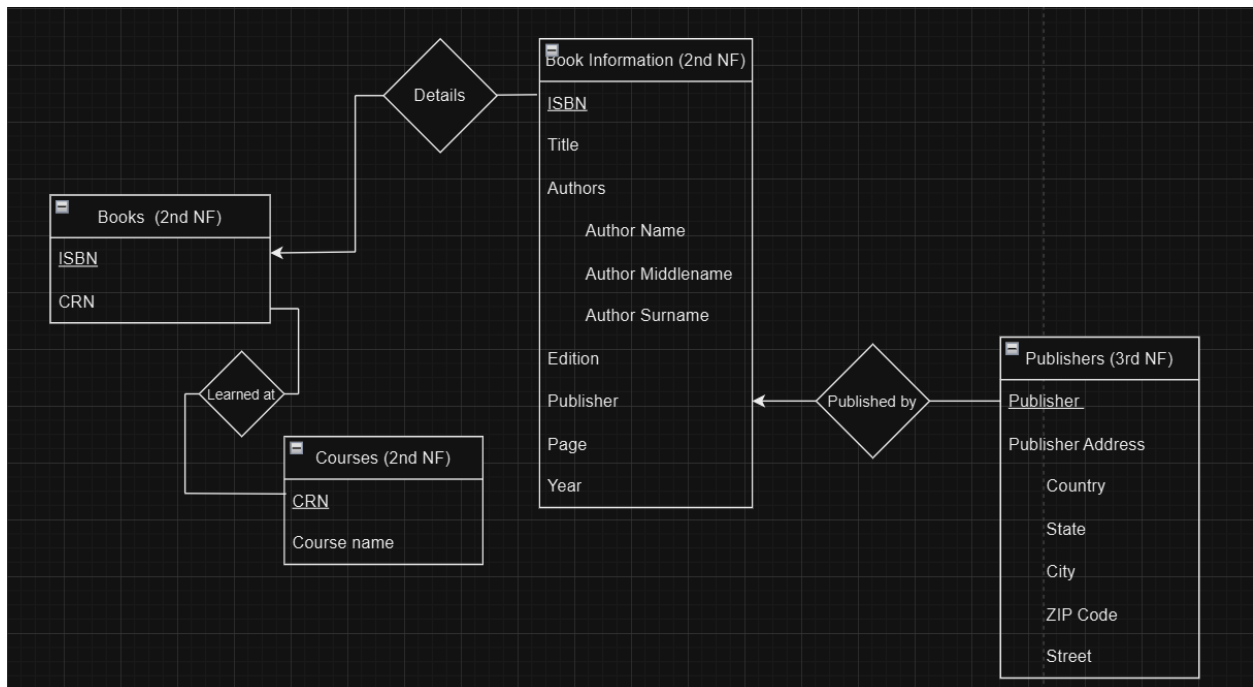
Rule

According to 3rd Normal Form, 2nd Normal Form should be satisfied, and transitive dependencies should be eliminated.

Problem and Solution

Transitive dependency occurs when a non-key attribute is dependent on another attribute that is dependent on primary key. The only case violating this rule is Publisher Address. Hence, it is transferred with its child attributes to the new table called Publishers.

ER-Diagram



ChatGPT Solution



Unnormalized1.xlsx
Spreadsheet

Normalize the given data. Show in each normal form. Give ER diagram of each normal form



I'll first examine the data in your file to understand its structure. Then, I'll guide you through the process of normalizing it into each normal form and creating an ER diagram for each.

The dataset contains the following columns:

- **CRN (PK1):** Course Registration Number (primary key)
- **ISBN (PK2):** International Standard Book Number (secondary primary key)
- **Title:** Title of the book
- **Authors:** List of authors
- **Edition:** Edition of the book
- **Publisher:** Publisher of the book
- **Publisher address:** Address of the publisher
- **Pages:** Number of pages
- **Year:** Publication year
- **Course name:** Name of the course

Approach to Normalization

1. **Unnormalized Form (UNF):** The current table has repeating groups in the "Authors" column, which will be divided into separate columns in the next steps.
2. **First Normal Form (1NF):** Each attribute will contain atomic (indivisible) values.
3. **Second Normal Form (2NF):** Remove partial dependencies; this applies only if the table has a composite primary key.
4. **Third Normal Form (3NF):** Remove transitive dependencies to ensure non-key columns are dependent only on the primary key.
5. **Boyce-Codd Normal Form (BCNF):** Ensure that every determinant is a candidate key.

I'll start by transforming this data into 1NF, then proceed through each normal form. Let's start with 1NF by making sure each cell has atomic values. [-]

The data is now in **First Normal Form (1NF)**, with each author represented as an atomic value in a separate row for each book.

Moving to Second Normal Form (2NF)

To achieve 2NF, we must remove partial dependencies. Here, the composite primary key includes both `CRN` (Course Registration Number) and `ISBN`. Attributes like `Course name` depend only on `CRN`, not the combination of `CRN` and `ISBN`. Therefore, we need to split the table into separate entities:

1. **Course Table:** Contains information specific to courses.
2. **Book Table:** Contains information specific to books.
3. **Course-Book Association Table:** Links courses and books together based on `CRN` and `ISBN`.

I'll create these tables to move the data into 2NF. [↗]

The data is now in **Second Normal Form (2NF)**:

1. **Course Table:** Contains unique values for each course, keyed by `CRN`.
2. **Book Table:** Contains details for each book, keyed by `ISBN` and listing each author in a separate row.
3. **Course-Book Association Table:** Links courses and books with unique pairs of `CRN` and `ISBN`.

Moving to Third Normal Form (3NF)

To achieve 3NF, we need to remove transitive dependencies. In the **Book Table**, the `Publisher address` depends on `Publisher`, not directly on `ISBN`. We'll create a separate **Publisher Table** for `Publisher` and `Publisher address`, and link it to the **Book Table** using `Publisher`.

Let's proceed to create the 3NF tables. [↗]

It seems like I can't do more advanced data analysis right now. Please try again later.

However, here's an overview of the next steps you can follow to achieve 3NF:

1. **Publisher Table:** Create a separate table with unique values of `Publisher` and `Publisher address`.
2. **Update Book Table:** Remove the `Publisher address` column from the Book Table and reference the Publisher Table.

If you want, I can guide you on how to proceed with these steps in a spreadsheet or a database tool! Let me know if you'd like more details or if you'd like me to help with anything else. [↗]

Comparison

Across the normalization stages, I compare my approach to that of ChatGPT's solution and find that there are several differences. For the 1st Normal Form, ChatGPT just recommends adding rows to handle atomicity violations as they occur; adding rows is all you need here to handle multiple authors per book at all. On the other hand, I separated the "Authors" attribute into separate attributes for first, middle and last names, which improved the data structure by making that data fully atomic and more easily queried on a specific component of the name. I also split out "Publisher Address" into individual attributes (Street, Zip Code, City, State, Country) for more fine grained address information, which ChatGPT neglected.

The 2nd Normal Form is the one in which both solutions achieve the needed structure by breaking any partial dependency by splitting the "Course Name" integral to a new 'Courses' table. Both implementations here are basically the same, since it's twice moving non-key attributes based on just a single portion of the composite primary key to different tables.

The 3rd Normal Form is handled almost equally in the both solutions, with each binding the transitive dependency of 'Publisher Address' to a new 'Publishers' table. Where I differ is that I defined more specific attributes in the address fields, which gives this more granular data. While the 2nd Normal Form and 1st Normal Form stages are similar on both solutions, the alterations made by my method in 1st Normal Form provide a structured data model that is better suited to future data management needs than ChatGPT's alterations.