Project Phase-3
Library

Team: Up To Data

Naimeesh Narayan Tiwari - 2020101074
Soveet Kumar Nayak - 2020101086
Khushi Agarwal - 2020101092

- ER to Relational model

We followed the steps for ER-to-Relational Mapping and got the above diagram.
The steps were as follows:

Step 1: Mapping of Regular Entity Types

Step 2: Mapping of Weak Entity Types

Step 3: Mapping of Binary 1:1 Relationship Types [we had no Binary 1:1 hence skipped]

Step 4: Mapping of Binary 1:N Relationship Types

Step 5: Mapping of Binary M:N Relationship Types [we had no Binary M:N hence skipped]

Step 6: Mapping of Multivalued Attributes

Step 7: Mapping of N-ary Relationship Types

Step 8: Mapping Specialization or Generalization

here we followed the below method :

[screenshot from the prescribed textbook about our approach]

- **Option 8C: Single relation with one type attribute.** Create a single relation L with attributes $\text{Attrs}(L) = \{k, a_1, \ldots, a_n\} \cup \{\text{attributes of } S_1\} \cup \ldots \cup \{\text{attributes of } S_m\} \cup \{t\}$ and $\text{PK}(L) = k$. The attribute t is called a **type** (or **discriminating**) attribute whose value indicates the subclass to which each tuple belongs, if any. This option works only for a specialization whose subclasses are *disjoint*, and has the potential for generating many NULL values if many specific (local) attributes exist in the subclasses.

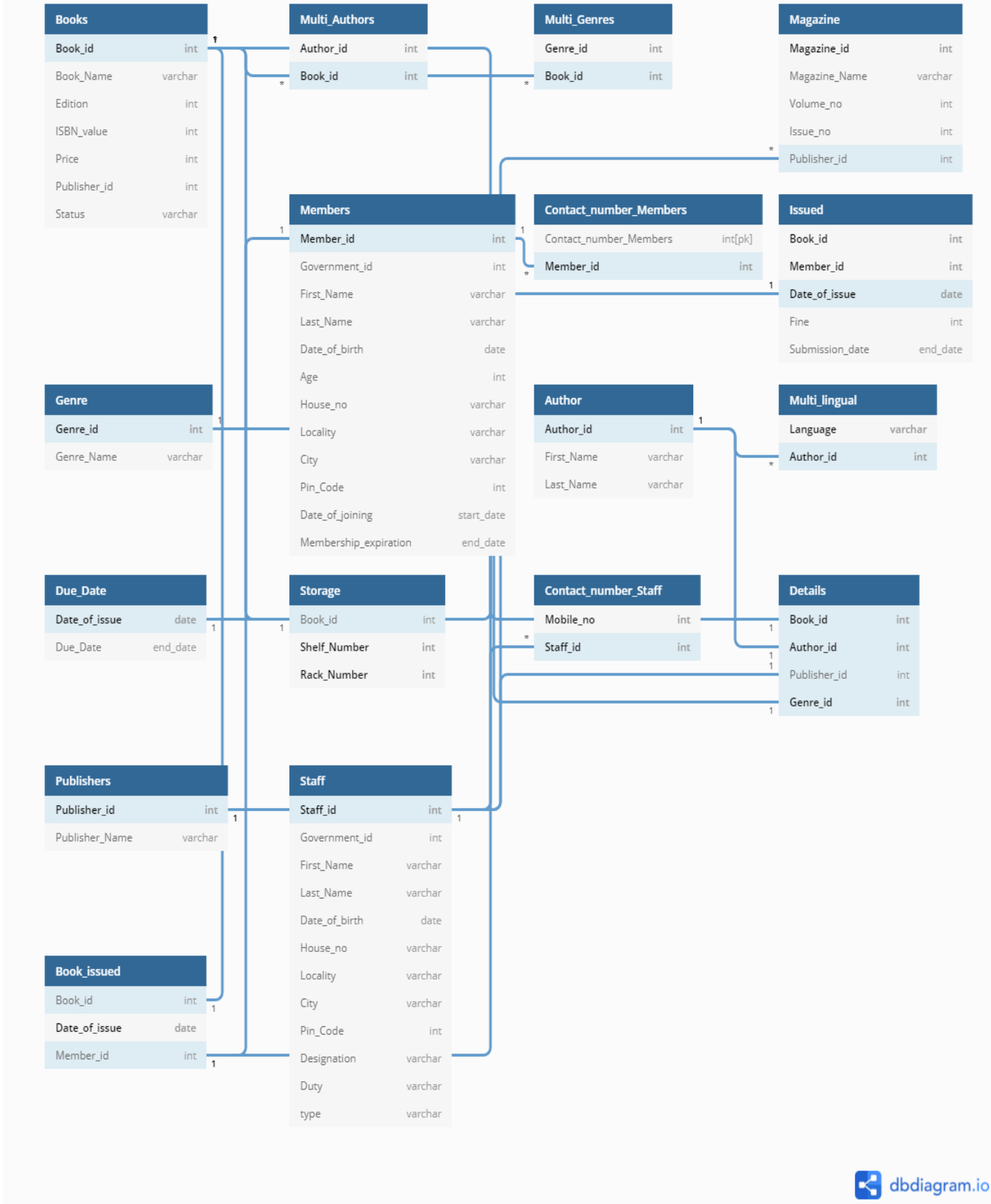
Conversion to 1NF, 2NF, 3NF

- Relational model after conversion to 1NF
    - A relation will be 1NF if it contains an atomic value.
    - It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attributes.
    - The first normal form disallows the multi-valued attribute, composite attribute, and their combinations.

As can be seen from the data types in the preceding Relational diagram, every given attribute can only take atomic values, implying that the criteria for the First Normal Form are met.

- Relational model after conversion to 2NF
    - In the 2NF, the relation must be in 1NF.
    - In the second normal form, all non-key attributes are fully functional dependent on the primary key.

In the table **Issues** (1NF Form); *Fine*, *Submission Date* can be identified by *Book_id, Member_id* and *Date_of_issue* whereas we can individually determine the *Due_date* with the help of the *Date_of_Issue.* This violates the 2NF form and has been dealt appropriately.


Books
Book_id int
Book_Name varchar
Edition int
ISBN_value int
Price int
Publisher_id int
Status varchar
Multi_Authors
Author_id int
Book_id int
Multi_Genres
Genre_id int
Book_id int
Magazine
Magazine_id int
Magazine_Name varchar
Volume_no int
Issue_no int
Publisher_id int
Members
Member_id int
Government_id int
First_Name varchar
Last_Name varchar
Date_of_birth date
Age int
House_no varchar
Locality varchar
City varchar
Pin_Code int
Date_of_joining start_date
Membership_expiration end_date
Contact_number_Members
Contact_number_Members int[pk]
Member_id int
Issued
Book_id int
Member_id int
Date_of_issue date
Fine int
Submission_date end_date
Genre
Genre_id int
Genre_Name varchar
Author
Author_id int
First_Name varchar
Last_Name varchar
Multi_lingual
Language varchar
Author_id int
Due_Date
Date_of_issue date
Due_Date end_date
Storage
Book_id int
Shelf_Number int
Rack_Number int
Contact_number_Staff
Mobile_no int
Staff_id int
Details
Book_id int
Author_id int
Publisher_id int
Genre_id int
Publishers
Publisher_id int
Publisher_Name varchar
Staff
Staff_id int
Government_id int
First_Name varchar
Last_Name varchar
Date_of_birth date
House_no varchar
Locality varchar
City varchar
Pin_Code int
Designation varchar
Duty varchar
type varchar
Book_issued
Book_id int
Date_of_issue date
Member_id int
dbdiagram.io

- Relational model after conversion to 3NF
    - A relation will be in 3NF if it is in 2NF and does not contain any transitive partial dependency.
    - A relation is in 3NF if at least one of the following conditions holds in every non-trivial functional dependency X -> Y:
        - X is a super key.
        - Y is a prime attribute(each element of Y is part of some candidate key).

Here, we found transitive dependency in the following tables:

1. Table Books [ISBN_value]

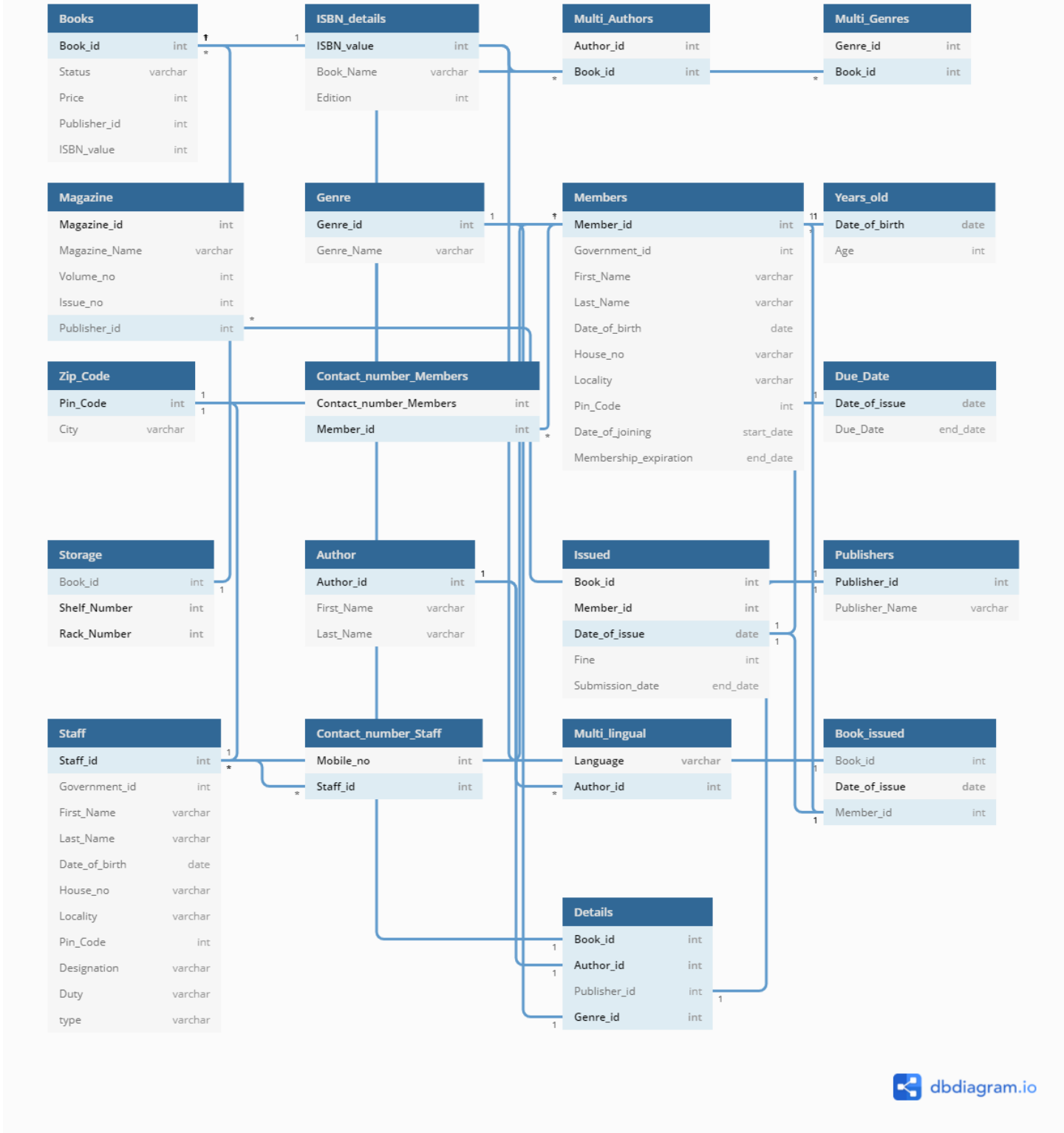
   The new table formed is Table ISBN_details.

2. Table Members [Date_of_birth, Pin_Code]

   The new tables formed are : Table Years_old and Table Zip_Code

3. Table Staff [Date_of_birth, Pin_Code]

   The new tables formed are : Table Years_old and Table Zip_Code .


Books
Book_id int
Status varchar
Price int
Publisher_id int
ISBN_value int
ISBN_details
ISBN_value int
Book_Name varchar
Edition int
Multi_Authors
Author_id int
Book_id int
Multi_Genres
Genre_id int
Book_id int
Magazine
Magazine_id int
Magazine_Name varchar
Volume_no int
Issue_no int
Publisher_id int
Genre
Genre_id int
Genre_Name varchar
Members
Member_id int
Government_id int
First_Name varchar
Last_Name varchar
Date_of_birth date
House_no varchar
Locality varchar
Pin_Code int
Date_of_joining start_date
Membership_expiration end_date
Years_old
Date_of_birth date
Age int
Zip_Code
Pin_Code int
City varchar
Contact_number_Members
Contact_number_Members int
Member_id int
Due_Date
Date_of_issue date
Due_Date end_date
Storage
Book_id int
Shelf_Number int
Rack_Number int
Author
Author_id int
First_Name varchar
Last_Name varchar
Issued
Book_id int
Member_id int
Date_of_issue date
Fine int
Submission_date end_date
Publishers
Publisher_id int
Publisher_Name varchar
Staff
Staff_id int
Government_id int
First_Name varchar
Last_Name varchar
Date_of_birth date
House_no varchar
Locality varchar
Pin_Code int
Designation varchar
Duty varchar
type varchar
Contact_number_Staff
Mobile_no int
Staff_id int
Multi_lingual
Language varchar
Author_id int
Book_issued
Book_id int
Date_of_issue date
Member_id int
Details
Book_id int
Author_id int
Publisher_id int
Genre_id int
dbdiagram.io

Changes:

- Removed Synonyms attribute from Authors.