



Course Number and Name: CSE 4308 Database Management Systems Lab	
Student Name: Nafisa Maliyat	Student ID: 200042133
Report Submission Date: 15 October, 2022	Name of Lab Instructor: Md. Bakhtiar Hasan, Lecturer, CSE Zannatun Naim Sristy, Lecturer, CSE

Lab 6: Entity Relationship (ER) Data Model

Overview:

This lab provided us with a given scenario and system requirements had to be extracted. A ER-Diagram was created based on the requirements and consecutive DDL statements had to be written based on the diagram.

On the next pages, I have mentioned the following :

- the scenario,
- the ER-Diagram of the given scenario
- part by part problem analysis,
- problems faced (if any) during solution of the tasks,
- the queries and their results on the SQL command line.

Scenario:

We are Booked (WB) is the legal depository of all new books and other printed materials in Bangladesh. Previously they stored all their information on paper. Recently they have decided to use a database.

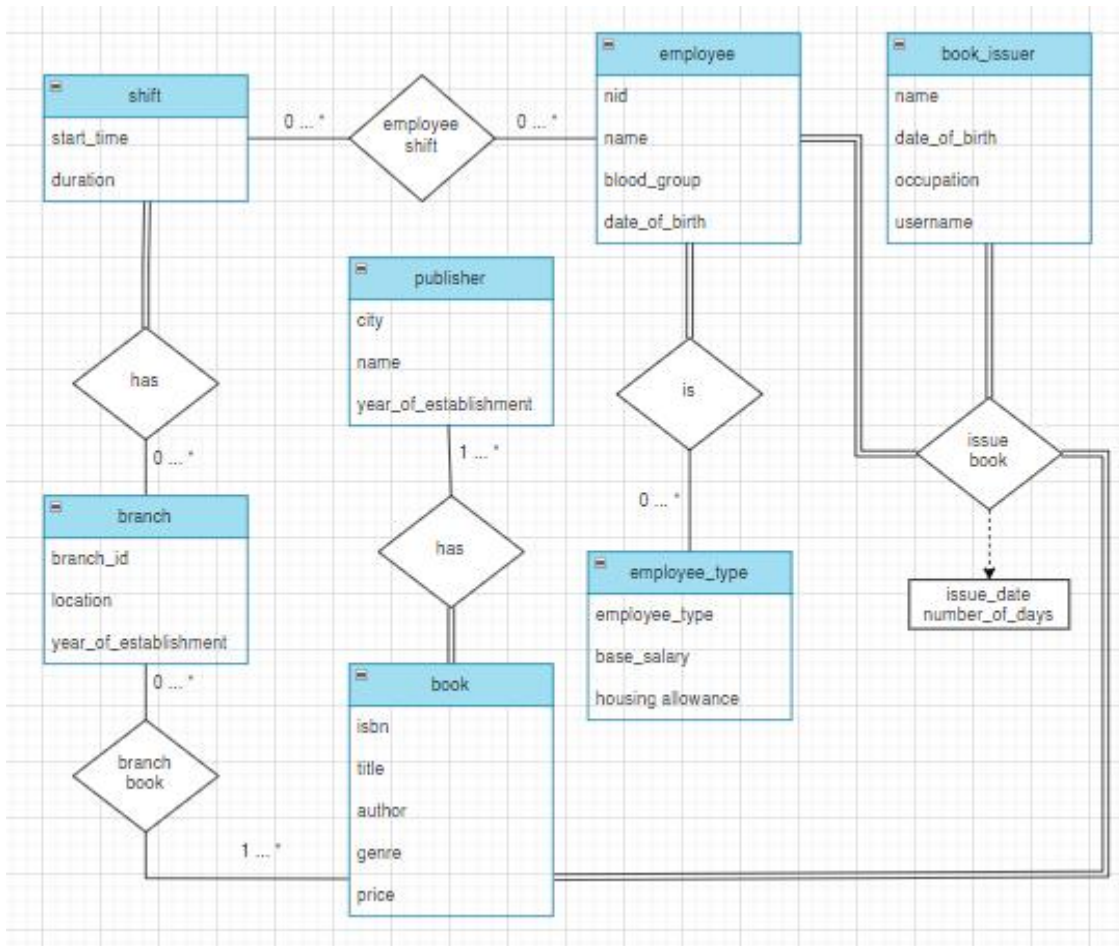
They have come up with the following requirements:

- There are many branches of WB in different locations in the country. Each branch has its unique branch ID, location, and year of establishment.
- Every branch is maintained by some employees. During the recruitment process, the National ID (NID), name, blood group, and birth date are stored. WB has three types of employees, namely Admin, Librarian, and Maintenance. Each has a separate base salary and 40% housing allowance based on the base salary.
- The employees work in different shifts in different branches. Each shift starts at a certain time on a specific day of the week. It also has a fixed duration.
- Of course the branches house many books. Each book can be identified by a 13-digit ISBN number. The name, author, genre, and price of the books are also stored. The number of copies of a book for each branch should be also tracked efficiently.
- Each book comes from different publishers. The publishers have their name, city, and establishment year. Note that, the same city will not have more than one publisher with the same name.
- To issue a book from any branch, a user has to create an account, providing their unique username. In addition to that, the user's name, date of birth, hometown, and occupation are also stored.
- When a book is issued by a user, the employee sets the issue date and number of days the user can keep the book. It might happen that sometimes the employee forgets to put the duration. In that case, the book should be returned within 15 days. Additionally, information about the user, book, and employee involved needs to be tracked for future purposes.

Task-1:

Problem Statement:

Draw an ER Diagram, without any data redundancy, specifying the cardinality explicitly. You may add additional attributes only if it is needed.



Task 1:

Part 1:

There are many branches of WB in different locations in the country. Each branch has its unique branch ID, location, and year of establishment.

Analysis of the problem:

This requires a table – branch which stores ID, location and year of establishment. Since ID is unique for each branch, ID is the primary key.

Any problems faced and how it was solved:

There were no problems faced since the query was straightforward and simple.

Results:

```
create table branch
(
  branch_id varchar2(8),
  location varchar2(20),
  year_of_establishment int,
  constraint pk_branch primary key(branch_id)
);
```

Table created.

Part 2:

Every branch is maintained by some employees. During the recruitment process, the National ID (NID), name, blood group, and birth date are stored. WB has three types of employees, namely Admin, Librarian, and Maintenance. Each has a separate base salary and 40% housing allowance based on the base salary.

Analysis of the problem:

Another table is required - employee that stores the National ID (NID), name, blood group, and birth date, where NID is the primary key. In addition there is an attribute called employee_type.

Another table called employee_type has the attributes employee_type, base salary and housing allowance. The employee_type attribute in employee table is a foreign key referencing the primary key attribute employee_type in employee_type table.

Any problems faced and how it was solved:

It was difficult to think of a way of including base salary and housing allowance in employee table and make sure every admin employee has the same base salary. The solution was to make a different table called employee_type since storing the same base salary and house allowance values will introduce redundancy and lead to inconsistency.

Results:

```
create table employee_type
(
    employee_type varchar2(10),
    base_salary numeric(6,2),
    house_allowance numeric(6,2) as (0.4 * base_salary),
    constraint pk_employee_type primary key(employee_type)
);

create table employee
(
    NID varchar2(13),
    name varchar2(30),
    blood_group varchar2(5),
    date_of_birth date,
    employee_type varchar2(12),
```

```
constraint pk_employee primary key(NID),  
constraint fk_employee_employee_type foreign key(employee_type)  
references employee_type(employee_type)  
);
```

Table created.

Table created.

Part 3:

The employees work in different shifts in different branches. Each shift starts at a certain time on a specific day of the week. It also has a fixed duration.

Analysis of the problem:

This introduces a new table called shift that is connected to branch to store what branch the shift is in.

There is a relationship between the tables shift and employees. Since the relationship is many-to-many, a junction table employee_shift has to be created. This will contain foreign key references to both tables employee and branch.

Since start time and day of week is unique for each shift, they will be additional attributes as well as primary key. The attribute duration will also be needed.

Any problems faced and how it was solved:

The relationships were rather confusing but identifying the required tables, attributes and relationships between them solved the issue.

Results:

```
create table shift
(
    start_time varchar2(10),
    duration varchar2(10),
    day_of_week varchar2(10),
    branch_id varchar2(8),
    constraint pk_shift primary key(day_of_week, start_time)
);

create table employee_shift
(
    day_of_week varchar2(10),
    start_time varchar2(10),
    NID varchar2(13),
    constraint fk_employee_employee_shift foreign key (NID) references
employee(NID),
    constraint fk_shift_employee_shift foreign key (day_of_week,
start_time) references shift(day_of_week, start_time)
);
```

Table created.

Table created.

Part 4:

Of course the branches house many books. Each book can be identified by a 13-digit ISBN number. The name, author, genre, and price of the books are also stored. The number of copies of a book for each branch should be also tracked efficiently.

Each book comes from different publishers. The publishers have their name, city, and establishment year. Note that, the same city will not have more than one publisher with the same name.

Analysis of the problem:

A new table called book is needed to hold the required attributes where ISBN is the primary key. The number of books can be tracked using a view that has a query to calculate the number of copies.

Another table called publisher table has to be created first since book will hold a foreign key referencing publisher table. Publisher table will have primary key name and city and an additional attribute establishment year.

Since a book can be in many branches and a branch can have many books, a junction table will be required that will not hold any extra attribute.

Any problems faced and how it was solved:

If number of copies were considered as an attribute, it would be difficult to track. Instead a view would be a perfect solution since it keeps updated information.

Results:

```
create table publisher
(
    city varchar2(15),
    name varchar2(30),
    year_of_establishment int,
    constraint pk_publisher primary key(city, name)
);

create table book
(
    isbn varchar2(13),
    title varchar2(30),
    author varchar2(30),
    genre varchar2(20),
    price numeric(6,2),
    publisher_city varchar2(15),
    publisher_name varchar2(30),
```

```

        constraint fk_publisher_book foreign key (publisher_city,
publisher_name) references publisher(city,name),
        constraint pk_book primary key(isbn)
    );

create table book_branches
(
    isbn varchar2(13),
    branch_id varchar2(8),
    constraint fk_book_branches_book foreign key (isbn) references
book(isbn),
    constraint fk_book_branches_branch foreign key(branch_id) references
branch(branch_id)
);

create or replace view book_count as
    select count(isbn) as book_count
    from book_branches
    group by isbn, branch_id;

```

Table created.

Table created.

Table created.

View created.

Part 5:

To issue a book from any branch, a user has to create an account, providing their unique username. In addition to that, the user's name, date of birth, hometown, and occupation are also stored.

Analysis of the problem:

An user table has to be created containing the required attribute and the unique username will be the primary key.

Any problems faced and how it was solved:

The name user was not allowed in SQL since it was a keyword. Thus the table was named book_issuer as it also reflects the role of the user table (to borrow a book).

Results:

```
create table book_issuer
(
    name varchar2(30),
    date_of_birth date,
    hometown varchar2(20),
    occupation varchar2(20),
    username varchar2(30),
    constraint pk_user primary key(username)
);
```

Table created.

Part 6:

When a book is issued by a user, the employee sets the issue date and number of days the user can keep the book. It might happen that sometimes the employee forgets to put the duration. In that case, the book should be returned within 15 days. Additionally, information about the user, book, and employee involved needs to be tracked for future purposes.

Analysis of the problem:

This states a relationship between user, employee and book. A junction table is required to hold the foreign keys referencing those tables plus additional attributes issue_date and number_of_days (set to default value 15).

Any problems faced and how it was solved:

There were no problems faced since the statement was straightforward and simple.

Results:

```
create table issue_book
(
    issue_date date,
    number_of_days int default 15,
    username varchar2(30),
    NID varchar2(13),
    isbn varchar2(13),
    constraint pk_employee_issue_book foreign key (NID) references
employee(NID),
    constraint pk_book_issue_book foreign key (isbn) references book(isbn),
    constraint pk_book_issuer_issue_book foreign key (username) references
book_issuer(username)
);
```

Table created.