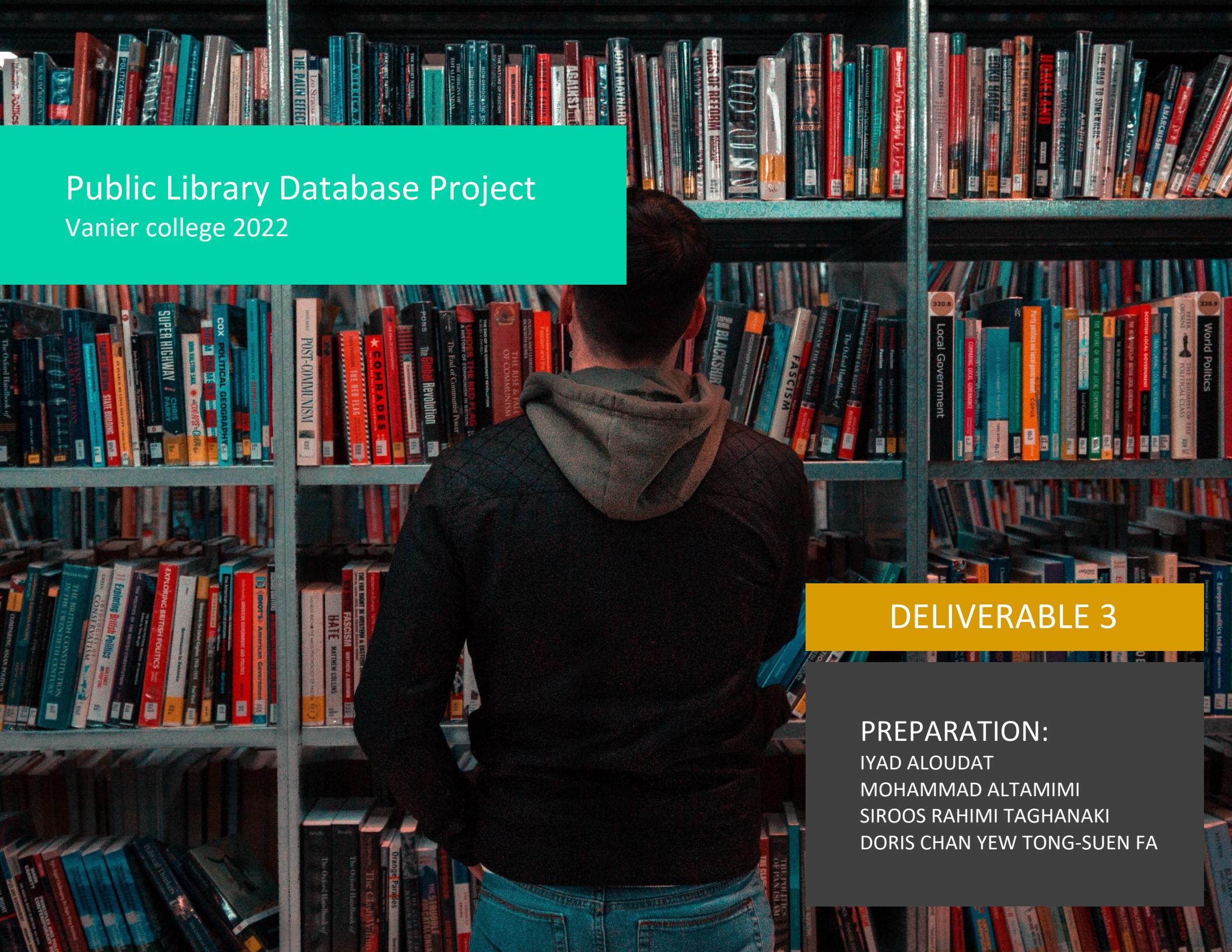


Public Library Database Project

Vanier college 2022



DELIVERABLE 3

PREPARATION:
IYAD ALOUDAT
MOHAMMAD ALTAMIMI
SIROOS RAHIMI TAGHANAKI
DORIS CHAN YEW TONG-SUEN FA

Deliverable

Done	Deliverable 1:	Page
Done	The conceptual database design using the Entity Relationship (E/R) diagram, shows entity types, relationships, attributes, key attributes, cardinalities, and participation constraints. Also, normalize the entities and include the steps of normalization process (to 3NF);	3 - 7
Done	The development of the logical design of the database (resulting by the mapping of an E/R diagram to a Relational model).	7 - 15
Done	Deliverable 2:	16-32
Done	Students must describe any revisions made to Deliverable 1 (based on feedback); Answer in Blue	4
Done	Create database schema using SQL DDL (Data Definition Language) statements, show how they enforced the constraints implied by their logical design, identify any constraints that are not captured by the ER diagram;	16-32
Done	Populate every relation with sufficient data using DML (Data Manipulation Language) statements. It should include the SQL scripts implemented and used for constructing the schema and loading the data in the database.	16-32
Done	Deliverable 3:	32-34
Done	Develop a set of SQL queries to query the database (there should be at least 5 queries). Include the statements of the queries and transactions scripts expressed in SQL along with the results returned;	32-34
Done	Apply advanced database concepts such as triggers to maintain data integrity, indexes, views, and procedures to improve system performance;	32-34
Done	There will be a project demonstration in the last week of the semester where students explain their work. Each group will have 15 minutes to demonstrate their database design	32-34

Introduction

The Database Library System is intended to Automate the library activities such as creating a new borrower, giving books to the borrowers, maintaining the details of all the item that were available in the books . This also helps the librarians by providing information such as total copies available each book, list of books that belong to a particular category (Short, Long Loan, Reference items, etc.).

Description of tables and columns

Customer Table		
Attribute	Attribute Name	Attributes Description
CardID [PK]	Card ID	The unique number for each customer
FirstName	First Name	First Name
LastName	Last Name	Last Name
Gender	Gender	Gender
DateOfBirth	Date of Birth	Date of Birth
Address	Full Address	Customer's full address
Phone	Phone Number	Phone or mobile Number
BranchID	Branch ID	The number of the branch in which the customer's account was created
NumBooksBorrowed	Num Books Borrowed	Number of books still borrowed by the customer

Borrow			
BorrowID	[PK]	Borrow ID	A unique number for each loan in the library, This ID is important to keep track record of the book (Borrowing record)
CardID	[FK]		The unique number for each customer
BookID	[FK]	Book ID	A unique number for each book in the library, even if it is a copy
BorrowDate		Borrow Date (Start)	Book borrow date (borrow Date)
RenewedNum		Renewed Number	The number of times the borrowing period has been extended
RenewDate		Renew Date (Start renew)	Borrowing period extension date (Renew Date)
ReturnDueDate		Return Due Date (End)	The due date for returning the borrowed book (Return Date)

LibraryBranch			
BranchID	[PK]	Branch ID	The number of the branch in which the customer's account was created
BranchName		Branch Name	The name of the branch
BranchAddress		Branch Address	The address of the Branch

Book			
BookID	[PK]	Book ID	A unique number for each book in the library, even if it is a copy
BookISBN	[FK]	Book ISBN	A unique numeric commercial book identifier
BranchID	[FK]	Branch ID	The number of the branch in which the customer's account was created
BookStatus		Book Status	Information about book Status: Borrowed, Available.

BookInfo			
BookISBN	[PK]	Book ISBN	A unique numeric commercial book identifier
BookTitle		Book Title	The title of the book
CategoryID	[FK]	Book Category	The category number of the book
PublisherID	[FK]	Publisher ID	The unique number of the publisher
NumberOfCopies		Number Of Copies	The number of copies the library has in stock
RackNum		Rack Number	The number of the rack

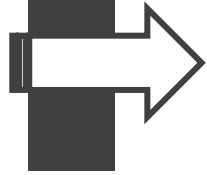
Categories			
CategoryID.	[PK]	Category ID	The category number of the book
CategoryName		Category Name	The category name of the book

Publisher			
PublisherID.	[PK]	Publisher ID	The unique number of the publisher
PublisherName		Publisher Name	The name of the publisher
PublisherAddress		Publisher Address	The address of the publisher
PublisherPhone		PublisherP hone	The phone number of the publisher

BookAuthors			
AuthorsID	[FK]	Authors ID	The number of the author
BookISBN	[FK]	Book ISBN	A unique numeric commercial book identifier

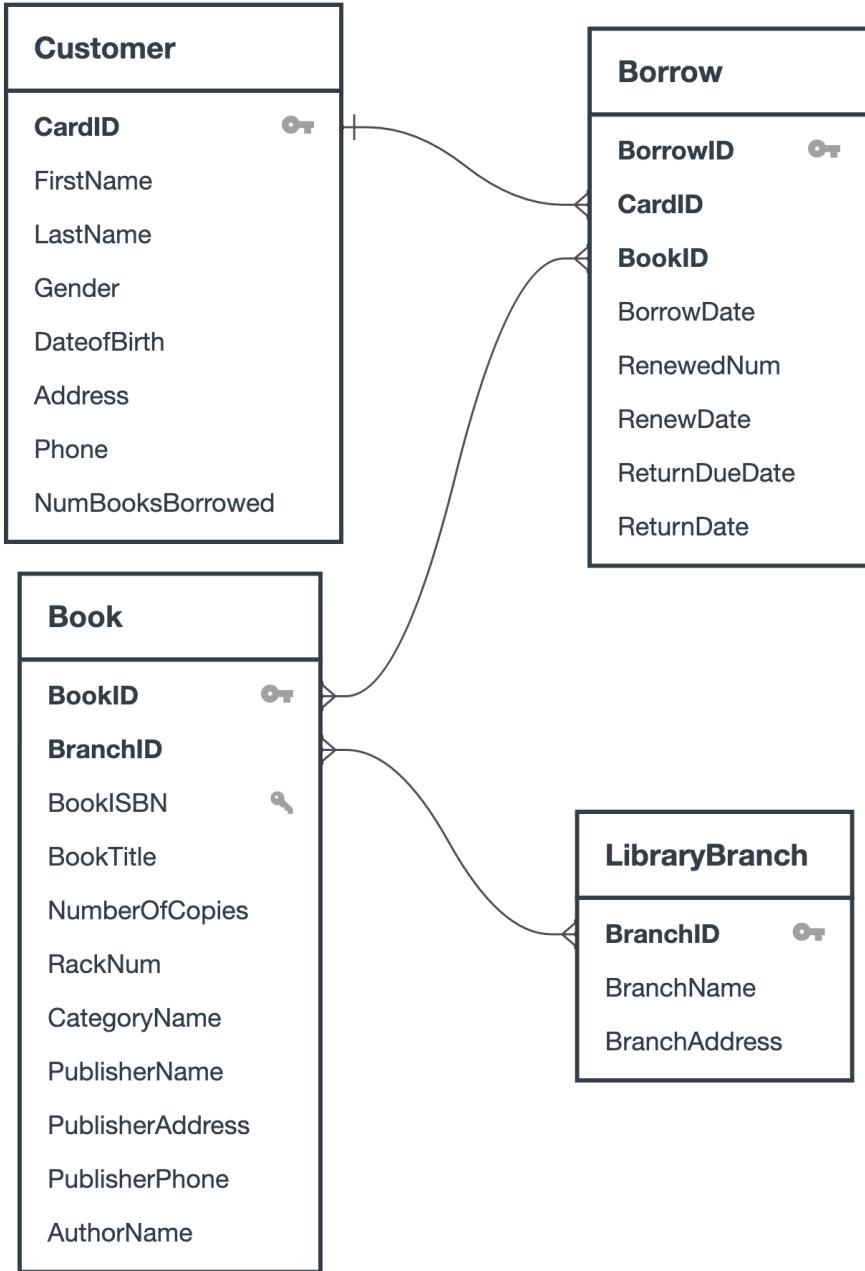
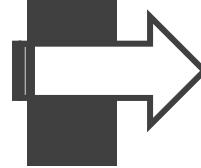
Authors			
AuthorID	[PK]	Author ID	The number of the author
AuthorName		Author Name	The name of the author

Normalization - 1NF

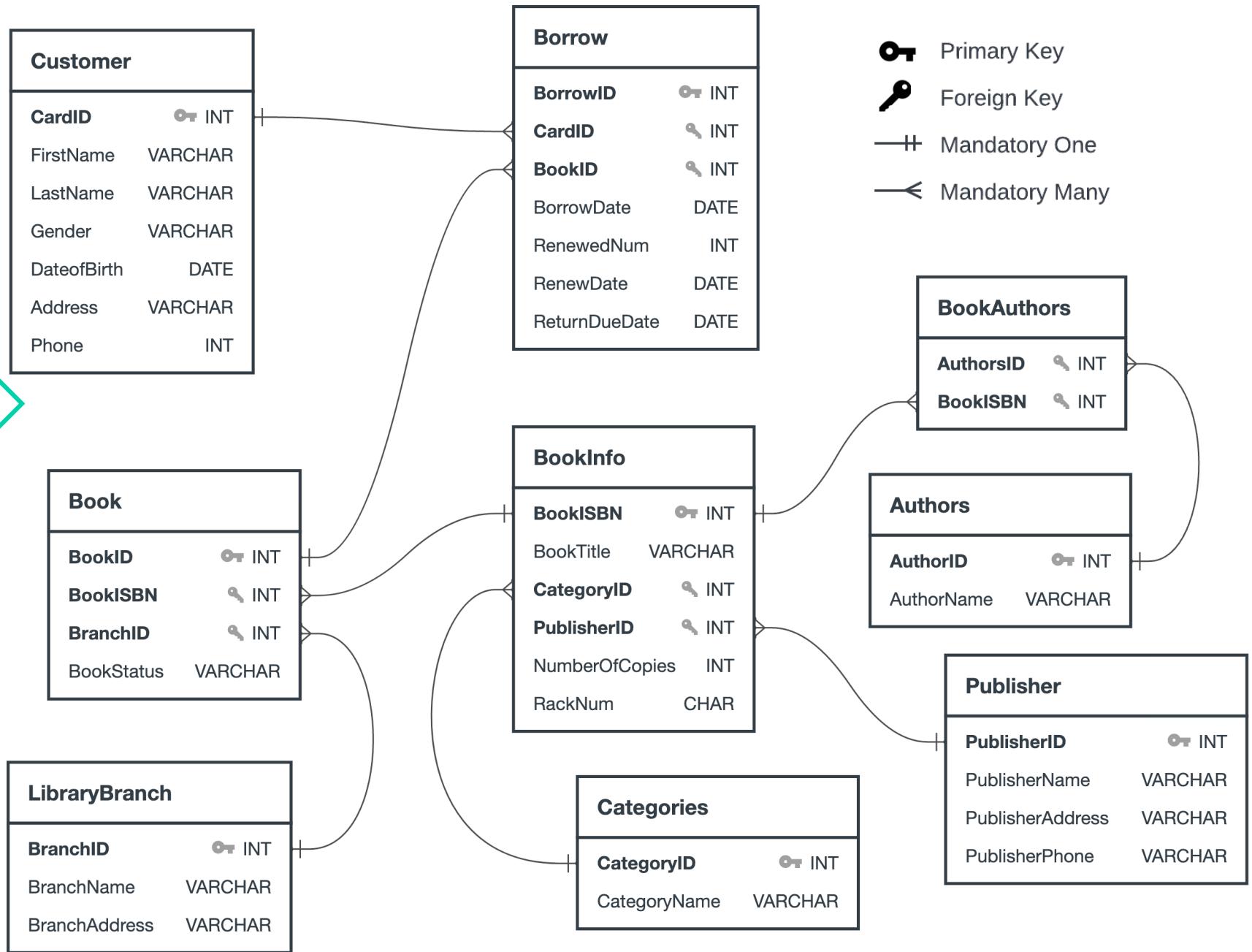


PublicLibrary	
CardID	key
FirstName	
LastName	
Gender	
DateOfBirth	
Address	
Phone	
NumBooksBorrowed	
BorrowDate	
RenewedNum	
RenewDate	
ReturnDueDate	
ReturnDate	
BranchName	
BranchAddress	
BookID	key
BookISBN	key
BookTitle	
NumberOfCopies	
RackNum	
CategoryName	
PublisherName	
PublisherAddress	
PublisherPhone	
AuthorName	

Normalization- 2NF



Normalization - 3NF

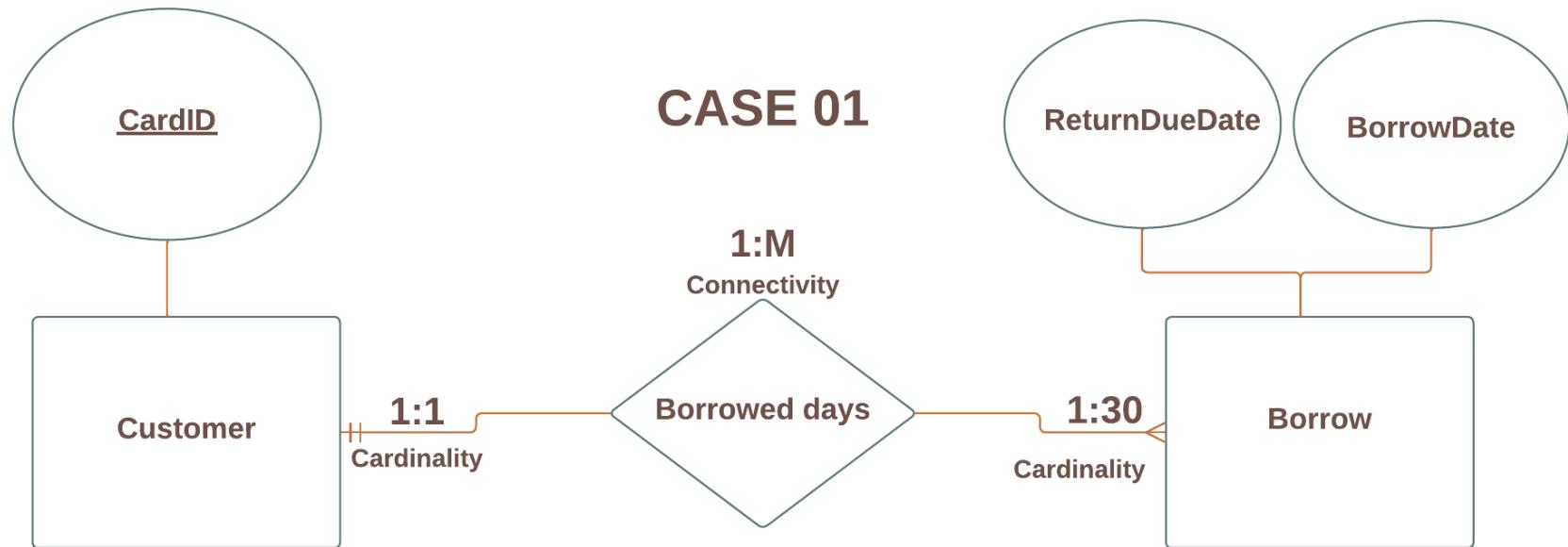


Case 01

The Maximum number of days allowed to borrow a book.

Business Rule

30 Days is the Maximum number of days allowed to borrow a book



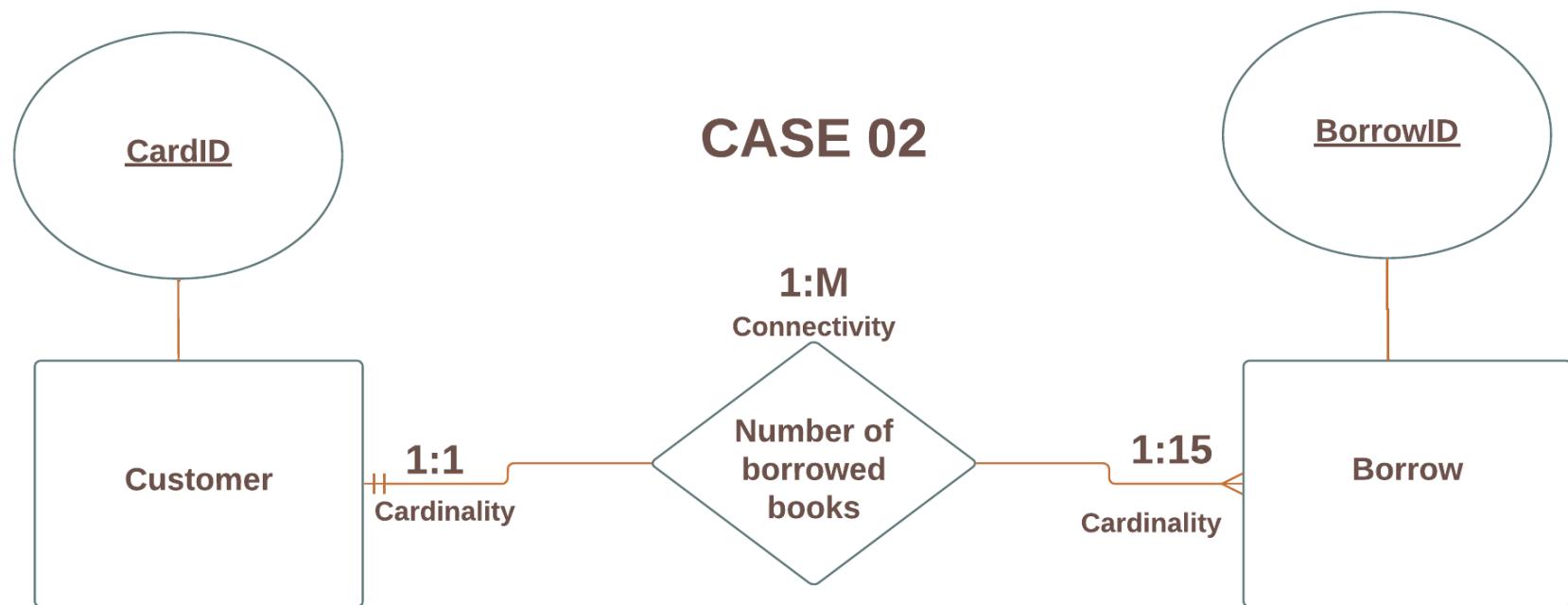
30 Days is the Maximum number of days allowed to borrow a book

Case 02

The maximum number of borrowed books.

Business Rule

15 books is The maximum number of borrowed books.



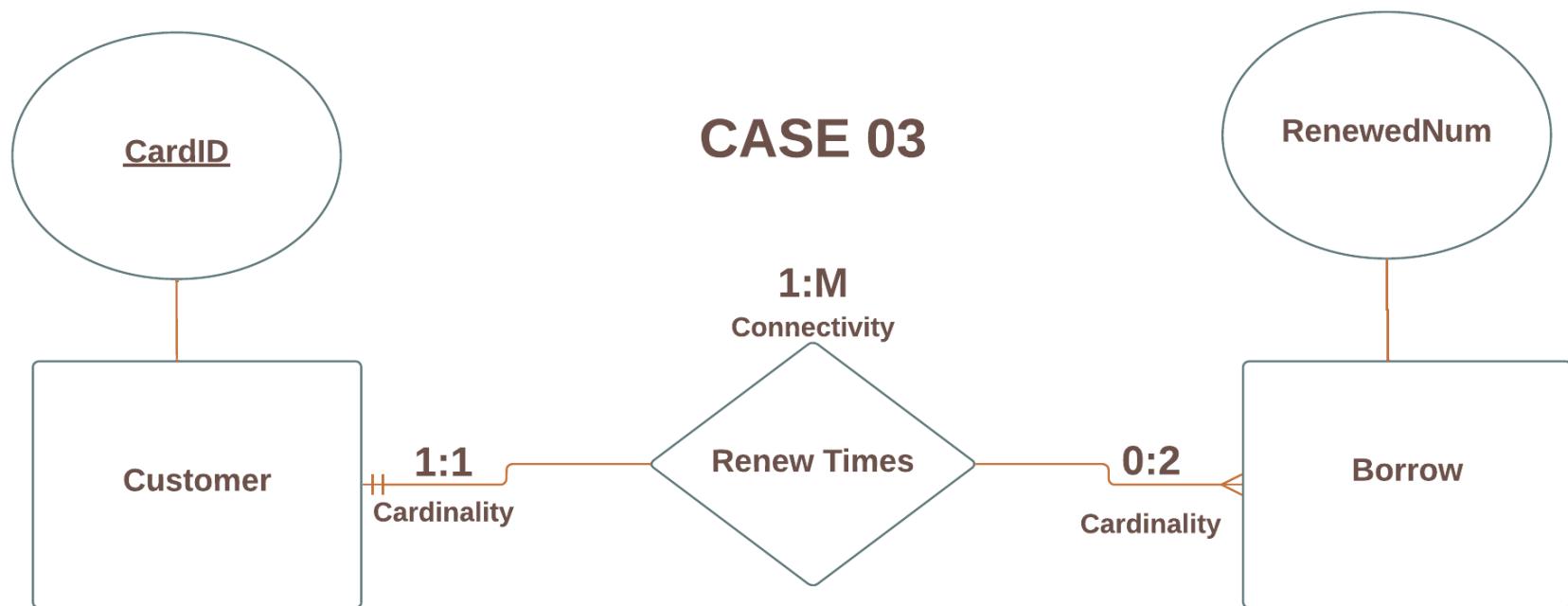
15 books is The maximum number of borrowed books.

Case 03

Renew the book return due date

Business Rule

Customers can renew the book return due date twice only



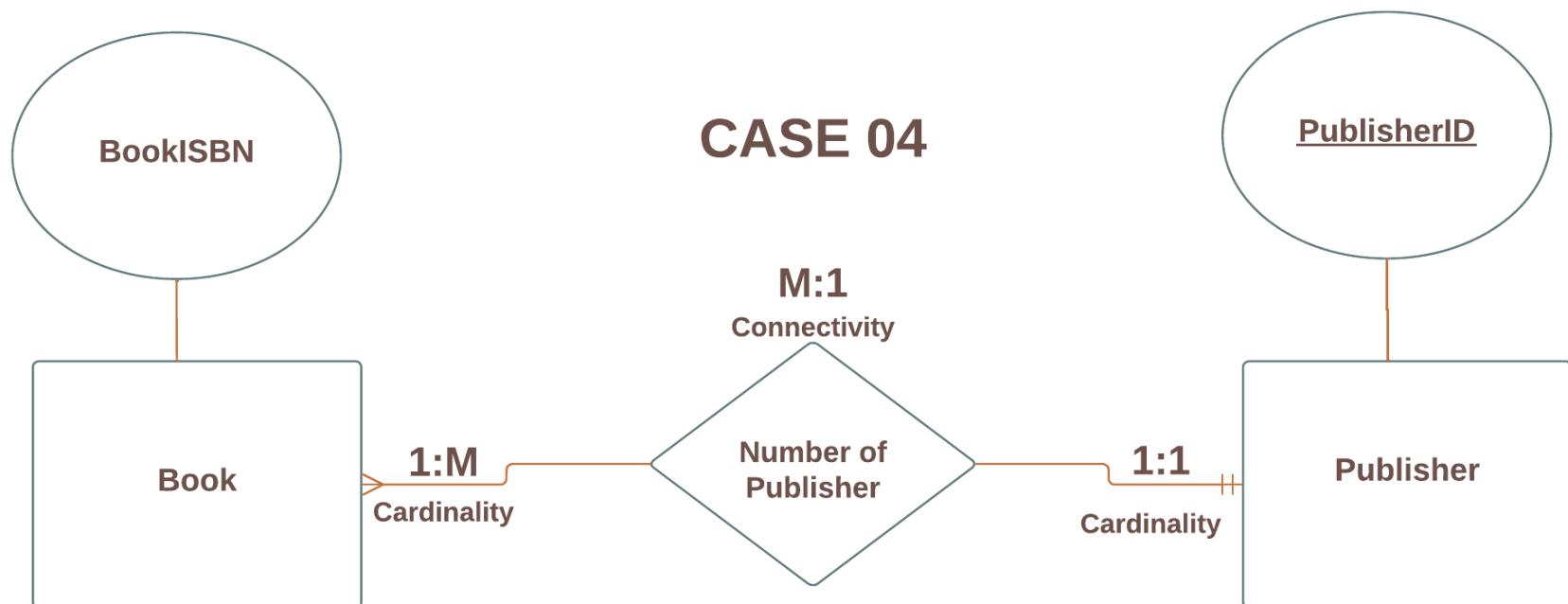
Customers can renew the book return due date twice only

Case 04

Number of Publisher for every Book

Business Rule

Every Book Should have only one Publisher



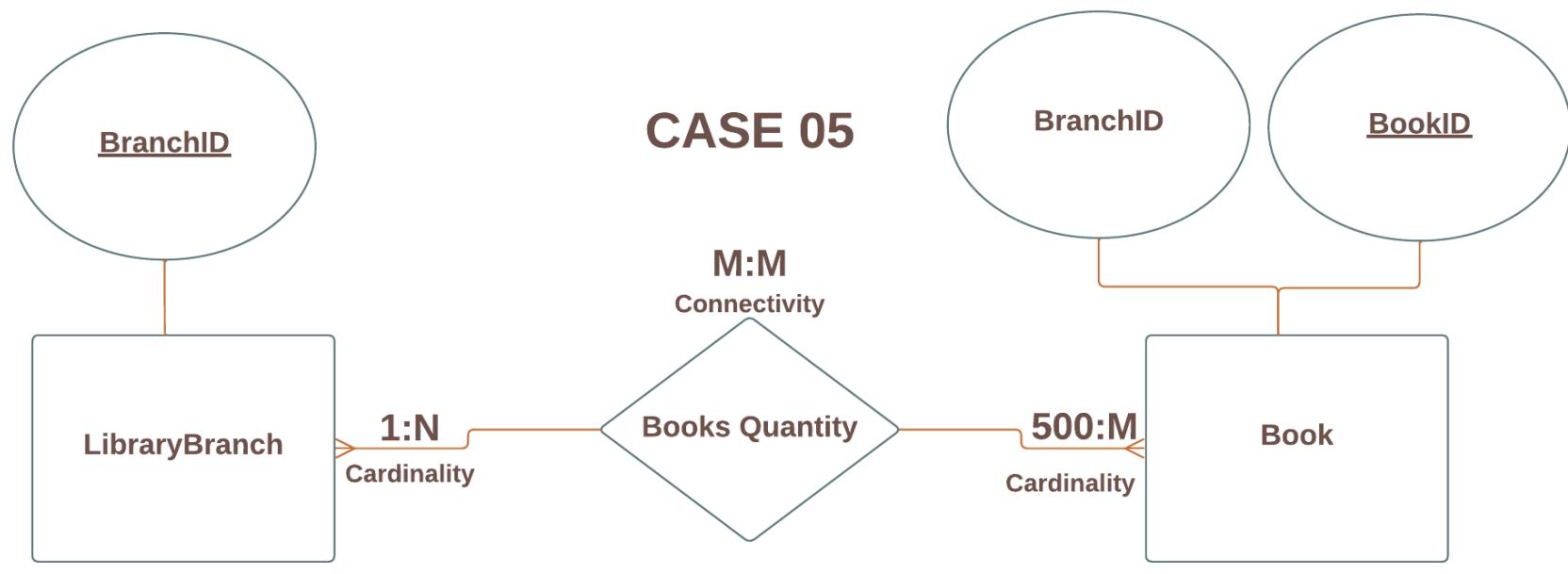
Every Book Should have only one Publisher

Case 05

Each branch must have enough books to borrow

Business Rule

Each Branch should have at least 500 Books



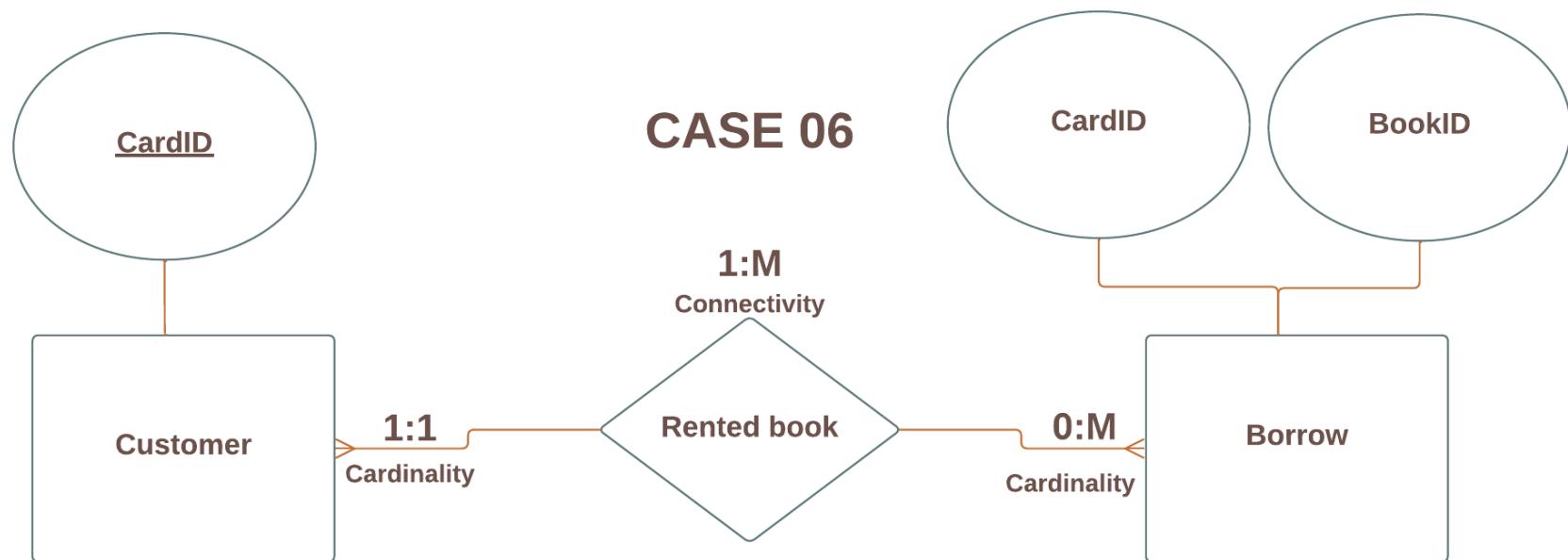
Each Branch should have at least 500 Books

Case 06

Book borrowing is available to all members

Business Rule

A book can be borrowed if available only.



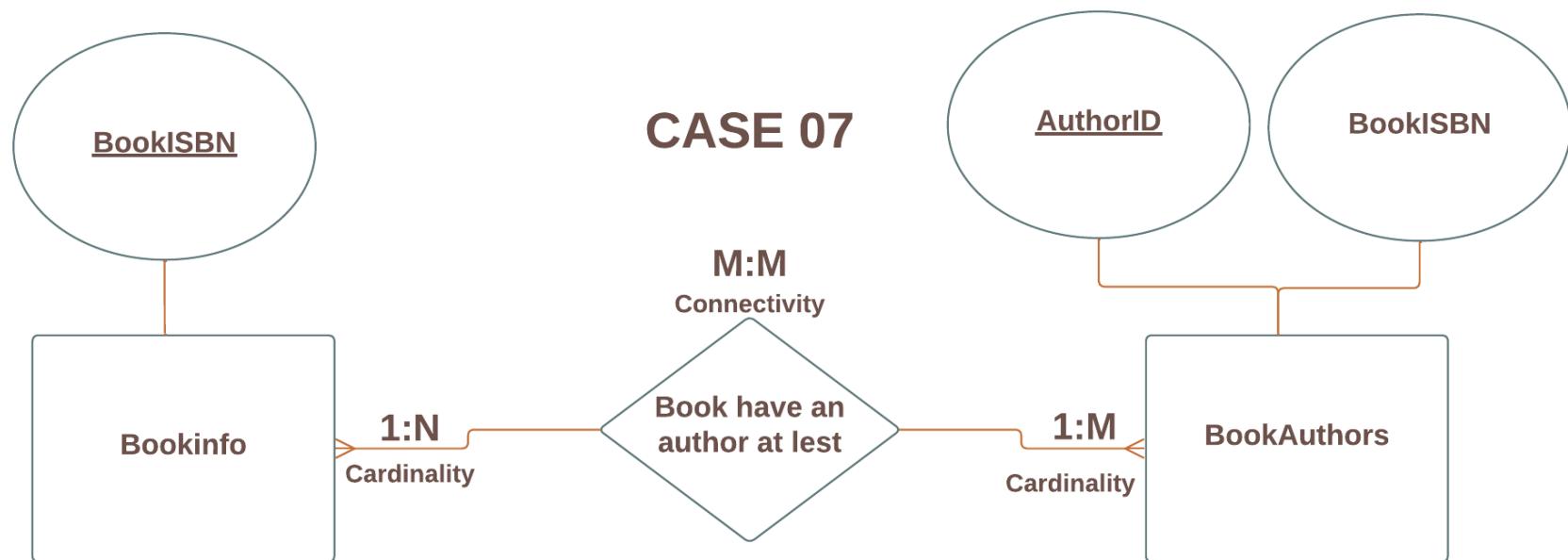
A bookID can be rented to one customer only.

Case 07

Every BookISBN should have one author at least in Book Authors Table

Business Rule

At Book Authors Table, AuthorsID and BookISB Should not be Null



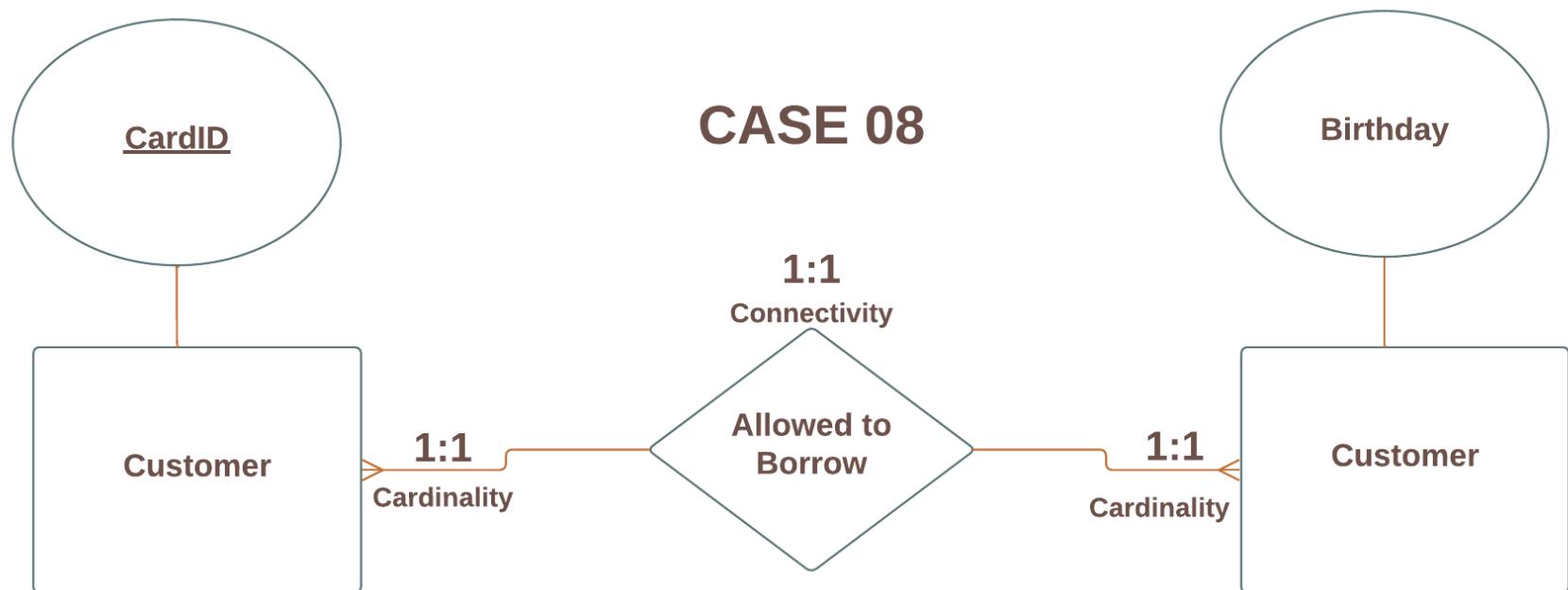
At Book Authors Table, Attributes for AuthorsID and BookISB Should not be Null

Case 08

Each one can be a customer

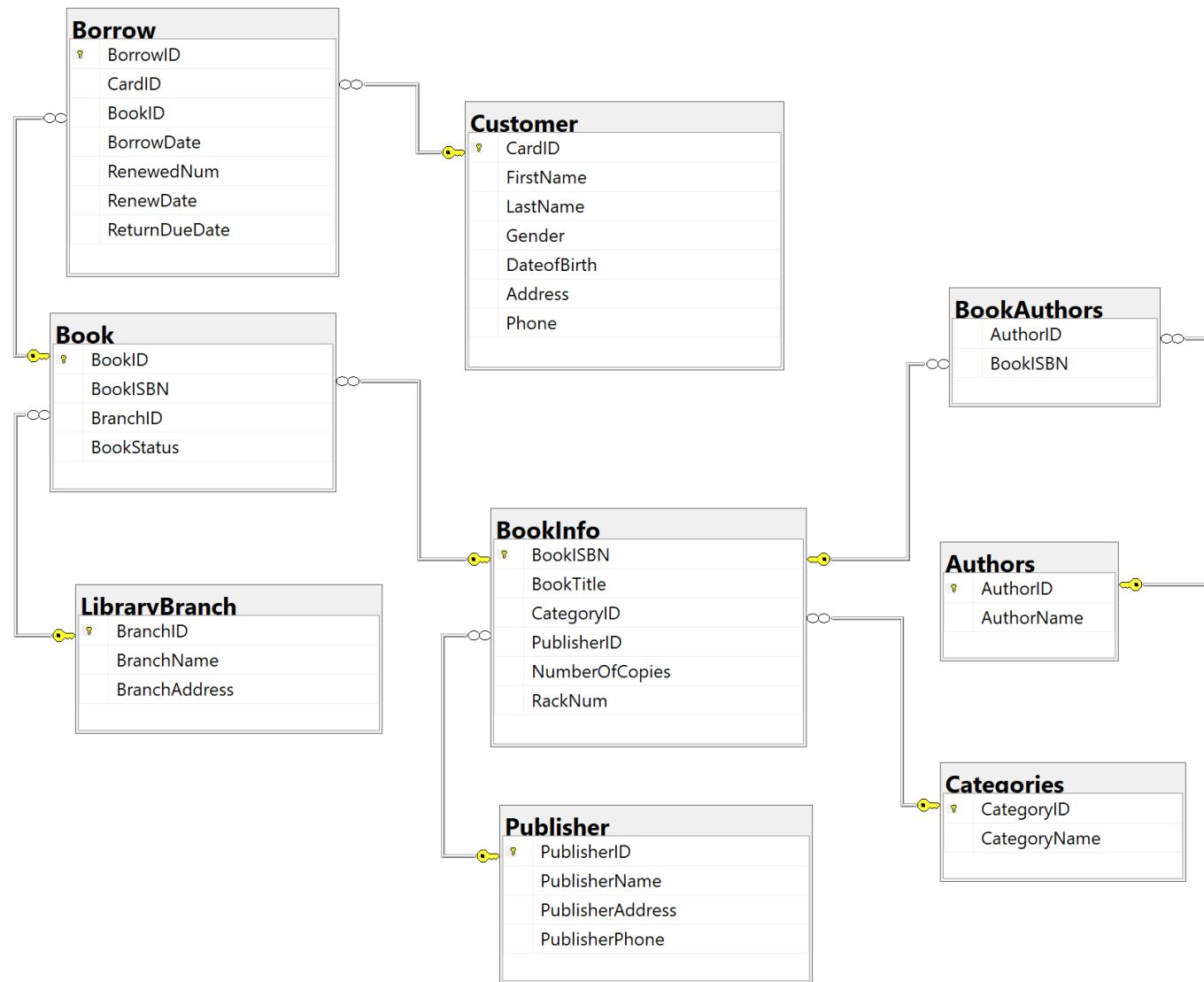
Business Rule

Each customer should be born before 2020 to be able to borrow a book.



Each customer should be born before 2020 to be able to borrow a book.

Deliverable 2:



SQLQuery1.sql - IYAD\lyad (59)*

1 | SELECT * from Customer;

2 |

110 %

Results Messages

	CardID	FirstName	LastName	Gender	DateofBirth	Address	Phone
1	1000000000	Danna	Luke	Female	1985-04-15	341 Derek Lane	2217294229
2	1000000001	Liv	Geffcock	Female	1999-10-13	64 Buell Lane	7745809382
3	1000000002	Sly	Marfe	Male	1972-05-28	05281 Stang Way	4104761045
4	1000000003	Allina	Palfreyman	Female	1979-08-16	56767 Artisan Center	9333497962
5	1000000004	Madonna	Gagie	Female	1979-06-16	173 Londonderry Avenue	8488846075
6	1000000005	Alyssa	Jenik	Female	1993-06-02	1 Anderson Plaza	2413295921
7	1000000006	Dorian	Klisch	Polygender	1970-02-26	090 Hauk Avenue	9386232035
8	1000000007	Martainn	De Blase	Male	1995-09-11	9538 Dahle Street	4922577375
9	1000000008	Orson	Tolcharde	Male	1986-02-15	17868 Blackbird Parkway	5434161715
10	1000000009	Jenna	Elsmere	Polygender	2009-08-25	41 Orin Point	6542360142
11	1000000010	Valerye	Parradine	Bigender	1985-02-05	1 Straubel Center	5638914664
12	1000000011	Ram	Simanek	Male	2001-08-09	1185 5th Parkway	5274913003
13	1000000012	Farlee	Beadles	Male	1982-01-21	9 Stephen Trail	1277032774
14	1000000013	Sibilla	Grichukhin	Female	1980-12-28	5536 Granby Court	1728681752
15	1000000014	Nicholas	Rolance	Male	1997-04-03	6 Oriole Way	1933453963
16	1000000015	Lynn	Widmore	Male	1980-03-08	7 Lunder Court	5887115678
17	1000000016	Phelia	Berthe	Female	1973-07-28	34568 Lindbergh Trail	7541070384
18	1000000017	Alexandros	Lattie	Male	1981-07-04	770 Pawling Street	9512714534
19	1000000018	Blakelee	Dankersley	Female	1984-08-13	652 Becker Pass	3465903191
20	1000000019	Cassandre	Gilkison	Female	1978-10-25	2 Ridgeway Street	5405875934
21	1000000020	Fidelio	Physic	Male	1978-03-27	4134 Mockingbird Circle	7696819784
22	1000000021	Darby	Gooke	Female	1983-07-03	229 Thompson Avenue	6549860299
23	1000000022	Rodolfo	Cicculini	Male	1977-04-11	73367 Stone Corner Hill	1408893320
24	1000000023	Terrence	Chinnery	Male	1974-02-01	870 McBride Crossing	3038466284
25	1000000024	Sawyere	Heber	Male	1979-07-01	88903 Express Street	5615716403

110 % ◀

Results Messages

	BookISBN	BookTitle	CategoryID	PublisherID	NumberOfCopies	RackNum
1	2783161484	Introduction to Algorithms	8	21	1	81
2	2921571537	The C Programming Language	8	16	1	5
3	3059981591	Big Data Big Challenges	8	16	1	52
4	3198391644	Designing Data	8	22	1	75
5	3336801698	Here Comes Peter Cottontail	10	8	1	29
6	3475211752	Thomas Crown Affair, The	4	19	1	47
7	3613621805	At Sword's Point	2	9	1	4
8	3752031859	Weary River	6	16	1	5
9	3890441912	Fatty Drives the Bus	9	4	1	7
10	4028851966	Young & Beautiful	10	9	1	26
11	4167262020	Bag of Hammers, A	8	11	1	8
12	4305672073	Animals United	2	14	1	32
13	4444082127	There Will Be Blood	10	10	1	42
14	4582492180	Ski Patrol	7	4	1	29
15	4720902234	Thousand Words, A	5	17	1	49
16	4859312288	Ragamuffin	7	20	1	37
17	4997722341	Hyde Park on Hudson	3	7	1	15
18	5136132395	Good Job: Stories of the FDNY, A	6	22	1	2
19	5274542448	Ishaqzaade	1	18	1	15
20	5412952502	Blind (Beul-la-in-deu)	4	10	1	6
21	5551362556	Taxi Blues	4	17	1	29
22	5689772609	Dead Silence	10	2	1	27
23	5828182663	Bat, The	6	11	1	48
24	5966592716	Choices	7	16	1	14
25	6105002770	Babysitter, The	2	6	1	9
26	6243412824	Alife	6	12	1	24
27	6381822877	Ordinary Decent Criminal	2	4	1	17
28	6520232931	Honeymoon	3	24	1	6
29	6658642984	388 Arletta Avenue	8	4	1	19
30	6797053038	Taira Clan Saga, The (Shin heike monogatari)	5	6	1	44
31	6935463092	Restraint	1	22	1	25
32	7073873145	Pajama Party	9	3	1	6
33	7212283199	Sport, Sport, Sport	7	24	1	2
34	7350693252	Beethoven's 5th	1	8	1	4
35	7489103306	Trial by Jury	2	5	1	33
36	7627513360	Stolen Seas	9	19	1	50
37	7765923413	Marathon Family, The (Maratonci Trce Poca...	4	9	1	30

```
/* ===== Introduction=====
```

```
Team:
```

```
IYAD ALOUDAT
```

```
MOHAMMAD ALTAMIMI
```

```
SIROOS RAHIMI TAGHANAKI
```

```
DORIS CHAN YEW TONG-SUEN FA
```

```
This File will Create DataBase, Table, Rules, and Data Records by Following the step below
```

- Create DataBase Name PublicLibrary
- Create Tables
- Create FOREIGN KEY
- Add Data Records
- Create Rules
- Procedure
- Trigger
- Test Trigger and Procedure

```
*/
```

```
/* ===== Database Library System Setup ===== */
```

```
CREATE DATABASE PublicLibrary;
```

```
GO
```

```
use PublicLibrary
```

```
GO
```

```
/* ===== TABLES ===== */
```

```
CREATE TABLE Customer (  
    CardID      Numeric(10,0)      NOT NULL IDENTITY(1000000000,1)  PRIMARY KEY,  
    FirstName   VARCHAR(50)        NOT NULL,  
    LastName    VARCHAR(50)        NOT NULL,  
    Gender      VARCHAR(10)       NOT NULL,  
    DateofBirth DATE             NOT NULL,  
    Address     VARCHAR(100)       NOT NULL,  
    Phone       Numeric(10,0)      NOT NULL
```

```
) ;
GO

CREATE TABLE LibraryBranch (
    BranchID      INT          NOT NULL IDENTITY(1,1) PRIMARY KEY,
    BranchName    VARCHAR(50)   NOT NULL,
    BranchAddress VARCHAR(50)   NOT NULL
);
GO

CREATE TABLE Categories (
    CategoryID    INT          NOT NULL IDENTITY(1,1) PRIMARY KEY,
    CategoryName  VARCHAR(50)   NOT NULL
);
GO

CREATE TABLE Publisher (
    PublisherID    INT          NOT NULL IDENTITY(1,1) PRIMARY KEY,
    PublisherName  VARCHAR(50)   NOT NULL,
    PublisherAddress VARCHAR(100) NOT NULL,
    PublisherPhone Numeric(10,0) NOT NULL
);
GO

CREATE TABLE BookInfo (
    BookISBN      Numeric(10,0) NOT NULL PRIMARY KEY,
    BookTitle     VARCHAR(50)   NOT NULL,
    CategoryID    INT          NOT NULL,
    PublisherID   INT          NOT NULL,
    NumberOfCopies INT          NOT NULL,
    RackNum       INT          NOT NULL
);
GO
```

```
CREATE TABLE Book (
    BookID      INT          NOT NULL  IDENTITY(1,1) PRIMARY KEY,
    BookISBN    Numeric(10,0)  NOT NULL,
    BranchID    INT          NOT NULL,
    BookStatus   VARCHAR(12)   NOT NULL
);
```

```
GO
```

```
CREATE TABLE Borrow (
    BorrowID     INT          NOT NULL  IDENTITY(1,1) PRIMARY KEY,
    CardID       Numeric(10,0) NOT NULL,
    BookID       INT          NOT NULL,
    BorrowDate   DATE         NOT NULL,
    RenewedNum   INT          NULL,
    RenewDate    DATE         NULL,
    ReturnDueDate DATE        NULL
);
```

```
GO
```

```
CREATE TABLE Authors (
    AuthorID     INT          NOT NULL  IDENTITY(1,1) PRIMARY KEY,
    AuthorName   VARCHAR(50)  NOT NULL
);
```

```
GO
```

```
CREATE TABLE BookAuthors (
    AuthorID     INT  NOT NULL,
    BookISBN    Numeric(10,0) NOT NULL
);
```

```
GO
```

```
/* ===== FOREIGN KEY : FK ===== */
```

```
ALTER TABLE BookInfo
ADD CONSTRAINT FK_CategoryID_BookInfo
FOREIGN KEY(CategoryID)
REFERENCES Categories(CategoryID);
GO
```

```
ALTER TABLE BookInfo
ADD CONSTRAINT FK_PublisherID_BookInfo
FOREIGN KEY(PublisherID)
REFERENCES Publisher(PublisherID);
GO
```

```
ALTER TABLE Book
ADD CONSTRAINT FK_BookISBN_Book
FOREIGN KEY(BookISBN)
REFERENCES BookInfo(BookISBN);
GO
```

```
ALTER TABLE Book
ADD CONSTRAINT FK_BranchID_Book
FOREIGN KEY(BranchID)
REFERENCES LibraryBranch(BranchID);
GO
```

```
ALTER TABLE Borrow
ADD CONSTRAINT FK_CardID_Borrow
FOREIGN KEY(CardID)
REFERENCES Customer(CardID);
GO
```

```
ALTER TABLE Borrow
ADD CONSTRAINT FK_BookID_Borrow
FOREIGN KEY(BookID)
```

```
REFERENCES Book(BookID);
```

```
GO
```

```
ALTER TABLE BookAuthors
```

```
ADD CONSTRAINT FK_BookISBN_BookAuthors
```

```
FOREIGN KEY(BookISBN)
```

```
REFERENCES BookInfo(BookISBN);
```

```
GO
```

```
ALTER TABLE BookAuthors
```

```
ADD CONSTRAINT FK_AuthorID_BookAuthors
```

```
FOREIGN KEY(AuthorID)
```

```
REFERENCES Authors(AuthorID);
```

```
GO
```

```
/* ===== Data Records ===== */
```

```
/* ## Customer Table ### */
```

```
insert into Customer (FirstName, LastName, gender, DateofBirth, Address, Phone)  
values
```

```
('Danna', 'Luke', 'Female', '1985-04-15', '341 Derek Lane', '2217294229'),  
('Liv', 'Geffcock', 'Female', '1999-10-13', '64 Buell Lane', '7745809382'),  
(('Sly', 'Marfe', 'Male', '1972-05-28', '05281 Stang Way', '4104761045'),  
(('Allina', 'Palfreyman', 'Female', '1979-08-16', '56767 Artisan Center', '9333497962'),  
(('Madonna', 'Gagie', 'Female', '1979-06-16', '173 Londonderry Avenue', '8488846075'),  
(('Alyssa', 'Jenik', 'Female', '1993-06-02', '1 Anderson Plaza', '2413295921'),  
(('Dorian', 'Klisch', 'Polygender', '1970-02-26', '090 Hauk Avenue', '9386232035'),  
(('Martainn', 'De Blase', 'Male', '1995-09-11', '9538 Dahle Street', '4922577375'),  
(('Orson', 'Tolcharde', 'Male', '1986-02-15', '17868 Blackbird Parkway', '5434161715'),  
(('Jenna', 'Elsmere', 'Polygender', '2009-08-25', '41 Orin Point', '6542360142'),  
(('Valerye', 'Parradine', 'Bigender', '1985-02-05', '1 Straubel Center', '5638914664'),  
(('Ram', 'Simanek', 'Male', '2001-08-09', '1185 5th Parkway', '5274913003'),  
(('Farlee', 'Beadles', 'Male', '1982-01-21', '9 Stephen Trail', '1277032774'),
```

```
('Sibilla', 'Grichukhin', 'Female', '1980-12-28', '5536 Granby Court', '1728681752'),  
('Nicholas', 'Rolance', 'Male', '1997-04-03', '6 Oriole Way', '1933453963'),  
('Lynn', 'Widmore', 'Male', '1980-03-08', '7 Lunder Court', '5887115678'),  
('Phelia', 'Berthe', 'Female', '1973-07-28', '34568 Lindbergh Trail', '7541070384'),  
('Alexandros', 'Lattie', 'Male', '1981-07-04', '770 Pawling Street', '9512714534'),  
('Blakelee', 'Dankersley', 'Female', '1984-08-13', '652 Becker Pass', '3465903191'),  
('Cassandre', 'Gilkison', 'Female', '1978-10-25', '2 Ridgeway Street', '5405875934'),  
('Fidelio', 'Physic', 'Male', '1978-03-27', '4134 Mockingbird Circle', '7696819784'),  
('Darby', 'Gooke', 'Female', '1983-07-03', '229 Thompson Avenue', '6549860299'),  
('Rodolfo', 'Cicculini', 'Male', '1977-04-11', '73367 Stone Corner Hill', '1408893320'),  
('Terrence', 'Chinnery', 'Male', '1974-02-01', '870 McBride Crossing', '3038466284'),  
('Sawyere', 'Heber', 'Male', '1979-07-01', '88903 Express Street', '5615716403');
```

GO

```
/* ## LibraryBranch ### */  
insert into LibraryBranch (BranchName, BranchAddress)  
values  
('Fixflex', '171 New Castle Crossing'),  
('Sonair', '9 Barby Road'),  
('Mat Lam Tam', '04095 Holy Cross Center'),  
('Asoka', '6 Welch Crossing'),  
('Fixflex', '347 Menomonie Crossing'),  
('Daltfresh', '8 Donald Hill'),  
('Y-find', '5 Declaration Junction'),  
('Toughjoyfax', '2 Tony Parkway'),  
('Flowdesk', '76 Prairie Rose Park'),  
('Prodder', '34 Basil Parkway');
```

GO

```
/* ## Categories ### */  
insert into Categories (CategoryName)  
values
```

```
('Action and Adventure'),
('Classics'),
('Detective and Mystery'),
('Fantasy'),
('History'),
('Horror'),
('Romance'),
('Science Fiction'),
('Short Stories'),
('Suspense and Thrillers');
```

```
GO
```

```
/* ## Publisher ### */
insert into Publisher (PublisherName, PublisherAddress, PublisherPhone)
values
('Hahn LLC', '9 Mariners Cove Road', '6645971159'),
('Metz, Hilll and Goldner', '133 Di Loreto Center', '3799469540'),
('Gleichner-Prosapco', '89497 Mayer Road', '7841788872'),
('Cruickshank-Windler', '75 High Crossing Road', '6715799599'),
('Predovic Inc', '79 Oneill Parkway', '2139437592'),
('Nolan, Cole and Schumm', '654 Paget Circle', '8259116693'),
('Von, Hegmann and Christiansen', '7825 Arizona Crossing', '2643801942'),
('Bartell-Tremblay', '93050 Towne Street', '8203939792'),
('Leannan, Bashirian and Murazik', '0503 Sugar Parkway', '9244176506'),
('Wunsch-Denesik', '7 Hallows Court', '6806396237'),
('Robel, Simonis and Friesen', '846 Drewry Pass', '2054176769'),
('Grant Inc', '200 Katie Hill', '2413807427'),
('West, Greenholt and Simonis', '51 Little Fleur Terrace', '5467642247'),
('Lemke-Predovic', '105 American Ash Avenue', '3905409744'),
('Thiel, Hintz and Kulas', '42107 Weeping Birch Way', '2027579172'),
('Gleichner, Cronin and Heaney', '36 Meadow Valley Plaza', '8246181254'),
('Ratke and Sons', '45713 Buhler Hill', '3328772359'),
('Little, Doyle and Roberts', '193 Lakewood Gardens Crossing', '6997922933'),
```

```
('Howe-Ortiz', '90 Cherokee Drive', '7081463715'),  
('Ernser Group', '36 Maple Wood Circle', '4893183966'),  
('Bashirian, Carroll and Cormier', '7014 Starling Terrace', '7569295354'),  
('Rodriguez Inc', '40413 Anthes Avenue', '1196334374'),  
('Wintheiser LLC', '5263 Bluestem Trail', '6818485152'),  
('Walker-Powlowski', '9088 American Street', '1029363403'),  
('Aufderhar Group', '3 Hayes Court', '2582924077');
```

GO

```
/* ## BookInfo ### */  
insert into BookInfo (BookISBN, BookTitle, CategoryID, PublisherID, NumberOfCopies,  
RackNum)  
values  
(2783161484, 'Introduction to Algorithms', 8, 21, 1, 81),  
(2921571537, 'The C Programming Language', 8, 16, 1, 5),  
(3059981591, 'Big Data Big Challenges', 8, 16, 1, 52),  
(3198391644, 'Designing Data', 8, 22, 1, 75),  
(3336801698, 'Here Comes Peter Cottontail', 10, 8, 1, 29),  
(3475211752, 'Thomas Crown Affair, The', 4, 19, 1, 47),  
(3613621805, 'At Sword''s Point', 2, 9, 1, 4),  
(3752031859, 'Weary River', 6, 16, 1, 5),  
(3890441912, 'Fatty Drives the Bus', 9, 4, 1, 7),  
(4028851966, 'Young & Beautiful', 10, 9, 1, 26),  
(4167262020, 'Bag of Hammers, A', 8, 11, 1, 8),  
(4305672073, 'Animals United', 2, 14, 1, 32),  
(4444082127, 'There Will Be Blood', 10, 10, 1, 42),  
(4582492180, 'Ski Patrol', 7, 4, 1, 29),  
(4720902234, 'Thousand Words, A', 5, 17, 1, 49),  
(4859312288, 'Ragamuffin', 7, 20, 1, 37),  
(4997722341, 'Hyde Park on Hudson', 3, 7, 1, 15),  
(5136132395, 'Good Job: Stories of the FDNY, A', 6, 22, 1, 2),  
(5274542448, 'Ishaqzaade', 1, 18, 1, 15),  
(5412952502, 'Blind (Beul-la-in-deu)', 4, 10, 1, 6),
```

```
(5551362556, 'Taxi Blues', 4, 17, 1, 29),  
(5689772609, 'Dead Silence', 10, 2, 1, 27),  
(5828182663, 'Bat, The', 6, 11, 1, 48),  
(5966592716, 'Choices', 7, 16, 1, 14),  
(6105002770, 'Babysitter, The', 2, 6, 1, 9),  
(6243412824, 'Alfie', 6, 12, 1, 24),  
(6381822877, 'Ordinary Decent Criminal', 2, 4, 1, 17),  
(6520232931, 'Honeymoon', 3, 24, 1, 6),  
(6658642984, '388 Arletta Avenue', 8, 4, 1, 19),  
(6797053038, 'Taira Clan Saga, The (Shin heike monogatari)', 5, 6, 1, 44),  
(6935463092, 'Restraint', 1, 22, 1, 25),  
(7073873145, 'Pajama Party', 9, 3, 1, 6),  
(7212283199, 'Sport, Sport, Sport', 7, 24, 1, 2),  
(7350693252, 'Beethoven''s 5th', 1, 8, 1, 4),  
(7489103306, 'Trial by Jury', 2, 5, 1, 33),  
(7627513360, 'Stolen Seas', 9, 19, 1, 50),  
(7765923413, 'Marathon Family, The (Maratonci Trce Pocasni Krug)', 4, 9, 1, 30),  
(7904333467, 'Burn, Witch, Burn (Night of the Eagle)', 3, 11, 1, 17),  
(8042743520, 'The Hire: Star, The', 7, 6, 1, 8),  
(8181153574, 'Judith of Bethulia', 1, 18, 1, 9),  
(8319563628, 'Bolero', 10, 10, 1, 15),  
(8457973681, 'Dead Man''s Walk', 4, 25, 1, 20),  
(8596383735, 'Secret Agent, The', 3, 12, 1, 29),  
(8734793788, 'Iron Island (Jazireh Ahani)', 7, 12, 1, 11),  
(8873203842, 'Teen Spirit', 9, 18, 1, 40),  
(9011613896, 'Most Wanted Man, A', 2, 22, 1, 9),  
(9150023949, 'Snowriders', 9, 25, 1, 32),  
(9288434003, 'War on Democracy, The', 5, 10, 1, 15),  
(9426844056, 'Road to Zanzibar', 1, 5, 1, 8),  
(9565254110, 'Inconvenient Tax, An', 2, 13, 1, 33);
```

GO

```
/* ## Book ### */
```

```
insert into Book (BookISBN, BranchID, BookStatus)
values
(2783161484, 10, 'Available'),
(2921571537, 5, 'Borrowed'),
(3059981591, 10, 'Available'),
(3198391644, 6, 'Borrowed'),
(3336801698, 10, 'Borrowed'),
(3475211752, 2, 'Borrowed'),
(3613621805, 5, 'Available'),
(3752031859, 2, 'Available'),
(3890441912, 6, 'Available'),
(4028851966, 1, 'Available'),
(4167262020, 5, 'Available'),
(4305672073, 7, 'Borrowed'),
(4444082127, 10, 'Borrowed'),
(4582492180, 6, 'Available'),
(4720902234, 1, 'Borrowed'),
(4859312288, 1, 'Available'),
(4997722341, 5, 'Available'),
(5136132395, 2, 'Available'),
(5274542448, 5, 'Borrowed'),
(5412952502, 9, 'Borrowed'),
(5551362556, 6, 'Available'),
(5689772609, 3, 'Borrowed'),
(5828182663, 7, 'Available'),
(5966592716, 3, 'Borrowed'),
(6105002770, 4, 'Borrowed'),
(6243412824, 4, 'Borrowed'),
(6381822877, 6, 'Available'),
(6520232931, 4, 'Available'),
(6658642984, 2, 'Borrowed'),
(6797053038, 1, 'Available'),
(6935463092, 7, 'Available'),
(7073873145, 2, 'Borrowed'),
```

```
(7212283199, 8, 'Available'),  
(7350693252, 7, 'Available'),  
(7489103306, 6, 'Borrowed'),  
(7627513360, 6, 'Available'),  
(7765923413, 8, 'Available'),  
(7904333467, 9, 'Available'),  
(8042743520, 7, 'Borrowed'),  
(8181153574, 2, 'Available'),  
(8319563628, 10, 'Available'),  
(8457973681, 1, 'Available'),  
(8596383735, 10, 'Available'),  
(8734793788, 9, 'Borrowed'),  
(8873203842, 8, 'Available'),  
(9011613896, 7, 'Available'),  
(9150023949, 1, 'Borrowed'),  
(9288434003, 7, 'Borrowed'),  
(9426844056, 10, 'Available'),  
(9565254110, 2, 'Available');
```

GO

```
/* ## Borrow Table ### */  
insert into Borrow (CardID, BookID, BorrowDate, ReturnDueDate, RenewedNum)  
values  
(1000000018, 1, '2022-05-12', '2022-06-10', 0),  
(1000000016, 13, '2022-03-22', '2022-04-20', 0),  
(1000000023, 15, '2022-05-28', '2022-06-26', 0),  
(1000000012, 9, '2021-11-01', '2021-11-30', 0),  
(1000000022, 10, '2022-08-20', '2022-09-18', 0),  
(1000000008, 7, '2021-10-22', '2021-11-20', 0),  
(1000000005, 12, '2021-12-18', '2022-01-16', 0),  
(1000000008, 3, '2022-09-09', '2022-10-07', 0),  
(1000000021, 8, '2022-07-26', '2022-08-24', 0),  
(1000000010, 7, '2022-07-14', '2022-08-12', 0),
```

```
(1000000001, 18, '2022-06-05', '2022-07-03', 0),
(1000000014, 4, '2022-08-19', '2022-09-17', 0),
(1000000020, 9, '2022-09-10', '2022-10-08', 0),
(1000000001, 3, '2022-08-16', '2022-09-14', 0),
(1000000009, 8, '2021-09-29', '2021-10-27', 0),
(1000000021, 14, '2022-01-29', '2022-02-27', 0),
(1000000014, 10, '2022-05-12', '2022-06-10', 0),
(1000000024, 12, '2021-10-28', '2021-10-26', 0),
(1000000008, 13, '2022-04-08', '2022-05-08', 0),
(1000000002, 4, '2022-07-26', '2022-08-24', 0),
(1000000018, 5, '2021-12-09', '2022-01-07', 0),
(1000000006, 9, '2022-07-30', '2022-08-28', 0),
(1000000004, 12, '2022-07-03', '2022-08-01', 0),
(1000000014, 18, '2021-11-01', '2021-11-29', 0),
(1000000019, 8, '2022-08-16', '2022-09-14', 0);
```

GO

```
/* ## Authors ### */
insert into Authors (AuthorName)
values
('Roma Astlatt'),
('Doris Crossby'),
('Joscelin Heape'),
('Timi Laroux'),
('Allard Kadd'),
('Angelina Gillingham'),
('Mora Paulon'),
('Koral Pittway'),
('Jacobo Dauber'),
('Penelopa Dewire'),
('Aeriel Fentem'),
('Andrej Geeve'),
('Ambrosi Pirouet'),
('Elmo Forrest'),
```

```
('Willy Oleszczuk'),  
('Marga Defond'),  
('Victoir Spellacey'),  
('Mallory Rutledge'),  
('Freemon Kightly'),  
('Tracy Routledge'),  
('Brantley Kinny'),  
('Lyndsay Brimilcombe'),  
('Ivie Straw'),  
('Bobbie Bestar'),  
('Sarette Andrioletti');  
  
GO  
/* ## BookAuthors ### */  
insert into BookAuthors (AuthorID, BookISBN)  
values  
(24, 2783161484),  
(21, 2921571537),  
(21, 3059981591),  
(14, 3198391644),  
(7, 3336801698),  
(8, 3475211752),  
(17, 3613621805),  
(5, 3752031859),  
(17, 3890441912),  
(6, 4028851966),  
(12, 4167262020),  
(16, 4305672073),  
(19, 4444082127),  
(12, 4582492180),  
(22, 4720902234),  
(7, 4859312288),  
(22, 4997722341),  
(13, 5136132395),
```

(14, 5274542448),
(20, 5412952502),
(10, 5551362556),
(24, 5689772609),
(12, 5828182663),
(23, 5966592716),
(21, 6105002770),
(18, 6243412824),
(16, 6381822877),
(12, 6520232931),
(24, 6658642984),
(6, 6797053038),
(2, 6935463092),
(11, 7073873145),
(15, 7212283199),
(7, 7350693252),
(14, 7489103306),
(22, 7627513360),
(15, 7765923413),
(19, 7904333467),
(14, 8042743520),
(14, 8181153574),
(9, 8319563628),
(3, 8457973681),
(17, 8596383735),
(21, 8734793788),
(6, 8873203842),
(12, 9011613896),
(7, 9150023949),
(22, 9288434003),
(15, 9426844056),
(24, 9565254110);

```
/* ====== Rules ===== */
-- Case 01: 30 Days is the Maximum number of days allowed to borrow a book
ALTER TABLE Borrow
ADD CHECK((DATEDIFF(year,ReturnDueDate,BorrowDate))<31);
GO

-- Case 02 : 15 books is The maximum number of borrowed books.
SELECT CardID, count(BookID) from Borrow group by CardID having count(BookID)<=15;
GO

-- Case 03: Customers can renew the book return due date twice only
ALTER TABLE Borrow
ADD CHECK((RenewedNum)<3);
GO

-- Case 04: Every Book Should have only one Publisher
-- ##### Added in FOREIGN KEY #####
-- Case 05 : Each Branch should have at least 500 Books
-- ### Note: We used 5 for the test because we don't have 500 record
SELECT BranchName, count(BookID) FROM Book JOIN LibraryBranch ON Book.BranchID =
LibraryBranch.BranchID
group by BranchName having count(BookID)>=5;
GO

-- Case 06: A book can be borrowed if available only.
SELECT BookID, BookStatus FROM Book where BookStatus='Available';
GO

-- Case 07: At Book Authors Table, AuthorsID and BookISBN Should not be Null
-- ##### Added in Table #####
```

```

-- Case 08: Each customer should be born before 2020
ALTER TABLE Customer
ADD CHECK(DateofBirth<'2020-01-01');
GO

-- ##### Procedure #####
CREATE PROCEDURE ProBorrow(@CardID As Numeric(10,0), @BookID AS INT, @BorrowDate AS DATE,
@RenewedNum AS INT, @RenewDate AS DATE, @ReturnDueDate AS DATE)
AS BEGIN
INSERT INTO Borrow VALUES (@CardID, @BookID, @BorrowDate, @RenewedNum, @RenewDate,
@ReturnDueDate);
END;
GO

-- ##### Trigger Update ReturnDueDate #####
CREATE TRIGGER TrigReturnDate
ON Borrow
AFTER INSERT
AS
Update Borrow SET ReturnDueDate = DATEADD(DAY, 30, BorrowDate) WHERE BorrowID=(SELECT TOP
1 BorrowID FROM Borrow ORDER by BorrowID DESC);
GO

-- ##### Test Trigger and Procedure #####
-- DataBase Content 25 Recored in Borrow, Execute ProBorrow will create recored #26
-- TRIGGER TrigReturnDate will Update Return Date Add 30 instead of Null

--Execute
EXECUTE ProBorrow 100000004,1,'2022-09-20',0,NULL,NULL;
GO
SELECT TOP 1 * FROM Borrow ORDER by BorrowID DESC

```