Logical Relational Schema Design

Introduction

Project Overview

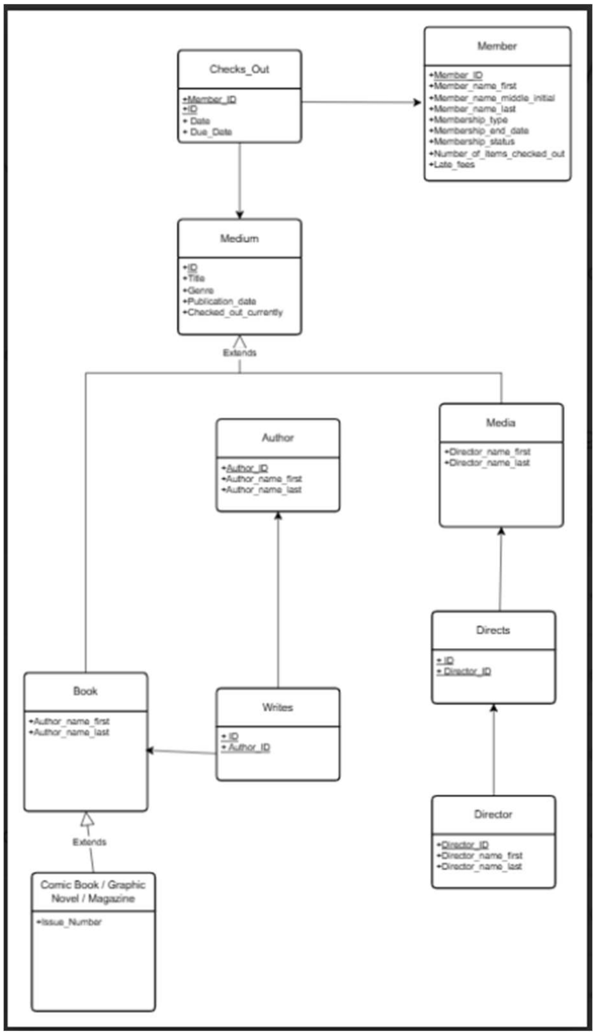
This database is intended to track the memberships and various media that a given library may store. This should include which items are checked out, for how long, what is overdue, any fees charged to members, and other important details about items available.

Scope

The database will not track the content or condition of every item. It will track the availability of certain items, and the membership fees associated with a client account.

Schema Documentation

Relational Diagram



Data Dictionary

Member

Attribute	Datatype	Domain	Constraints
Member_ID	int	A positive integer	Primary key
Member_name_first	varchar	A string consisting of 15 characters	Not null
Member_name_middle_initial	char	A single character	Null allowed
Member_name_last	varchar	A string consisting of 15 characters	Not null
Membership_type	varchar	A string consisting of 7 characters	Not null
Membership_end_date	date	A calendar date	
Membership_status	varchar	A string consisting of 8 characters	Not null
Number_of_items_checked_out	int	A positive integer	Not null
Late_fees	int	A positive integer	Not null

Medium

Attribute	Datatype	Domain	Constraints
ID	int	A positive integer	Primary key
Title	varchar	A string consisting of 100 characters	Not null
Genre	varchar	A string consisting of 50 characters	Not null
Publication_date	date	A calendar date	Not null
Checked_out_currently	varchar	A string of 3 characters “yes” or “no”	Not null

Checks Out (relationship between Member and Medium)

Attribute	Datatype	Domain	Constraints
Member_ID	int	A positive integer	Primary Key, Foreign key (Member)
Medium_ID	int	A positive integer	Primary Key, Foreign key (Medium)
Checkout_date	date	A calendar date	Not null
Due_date	date	A calendar date	Not null

Comic Book/Graphic Novel/Magazine (inherits from Medium)

Attribute	Datatype	Domain	Constraints
ID	int	A positive integer	Primary key, Foreign key (Book)
Issue_Number	int	A positive integer	Not null

Author

Attribute	Datatype	Domain	Constraints
Author_ID	int	A positive integer	Primary key
Author_name_first	varchar	A string consisting of 50 characters	Not null
Author_name_last	varchar	A string consisting of 50 characters	Not null

Writes (relationship between Author and Book)

Attribute	Datatype	Domain	Constraints
Author_ID	int	A positive integer	Primary key, Foreign key (Author)
Book_ID	int	A positive integer	Primary key, Foreign key (Book)

Director

Attribute	Datatype	Domain	Constraints
Director_ID	int	A positive integer	Primary key
Director_name_first	varchar	A string consisting of 50 characters	Not null
Director_name_last	varchar	A string consisting of 50 characters	Not null

Directs (relationshp between Director and Media)

Attribute	Datatype	Domain	Constraints
Director_ID	int	A positive integer	Primary key, Foreign key (Director)
Media_ID	int	A positive integer	Primary key, Foreign key (Media)

Generated DDL

```
CREATE TABLE Member (

        Member_ID INT NOT NULL,

        Member_name_first VARCHAR (15) NOT NULL,

        Member_name_middle_initial CHAR,

        Member_name_last VARCHAR (15) NOT NULL,

        Membership_type VARCHAR(7) NOT NULL,

        Membership_end_date DATE,

        Membership_status VARCHAR(8) NOT NULL,

        Number_of_items_checked_out INT NOT NULL,

        Late_fees INT NOT NULL,

        PRIMARY KEY (Member_ID),

        CHECK (Late_fees >= 0),

        CHECK (Number_of_items_checked_out >= 0),

        CHECK (Membership_status=”active” OR Membership_status=”inactive”),
        CHECK (Membership_type=”senior” OR Membership_type=”student” OR
        Membership_type=”child” OR Membership_type=”adult”)
);
CREATE TABLE Medium (
        ID INT NOT NULL,
        Title VARCHAR(100) NOT NULL,
        Genre VARCHAR(50) NOT NULL,
        Publication_date DATE NOT NULL,
        Checked_out_currently VARCHAR(3) NOT NULL,
        CHECK (Checked_out_currently=”yes” OR Checked_out_currently=”no”)
);
CREATE TABLE Checks Out (
        Member_ID INT NOT NULL,
        Medium_ID INT NOT NULL,
        Checkout_date DATE NOT NULL,
        Due_date DATE NOT NULL,
        PRIMARY KEY (Member_ID, Medium_ID)
        FOREIGN KEY (Member_ID) REFERENCES Member
        FOREIGN KEY (Medium_ID) REFERENCES Medium
);
CREATE TABLE Magazine (
        ID INT NOT NULL,
        Issue_number INT NOT NULL,
        PRIMARY KEY (ID),
        FOREIGN KEY (ID) REFERENCES Medium
);
CREATE TABLE Author (
        Author_ID INT NOT NULL,
        Author_first_name VARCHAR(50) NOT NULL,
        Author_last_name VARCHAR(50) NOT NULL,
        PRIMARY KEY (Author_ID)
);
CREATE TABLE Writes (
        Author_ID INT NOT NULL,
        Book_ID INT NOT NULL,
        PRIMARY KEY (Author_ID, Book_ID),
        FOREIGN KEY (Author_ID) REFERENCES Author,
        FOREIGN KEY (Book_ID) REFERENCES Medium
);
CREATE TABLE Director (
        Director_ID INT NOT NULL,
        Director_name_first VARCHAR(50) NOT NULL,
        Director_name_last VARCHAR(30) NOT NULL,
        PRIMARY KEY (Director_ID)
);
CREATE TABLE Directs (
        Director_ID INT NOT NULL,
        Media_ID INT NOT NULL,
        PRIMARY KEY (Director_ID, Media_ID),
        FOREIGN KEY (Director_ID) REFERENCES Director
        FOREIGN KEY (Media_ID) REFERENCES Medium
```