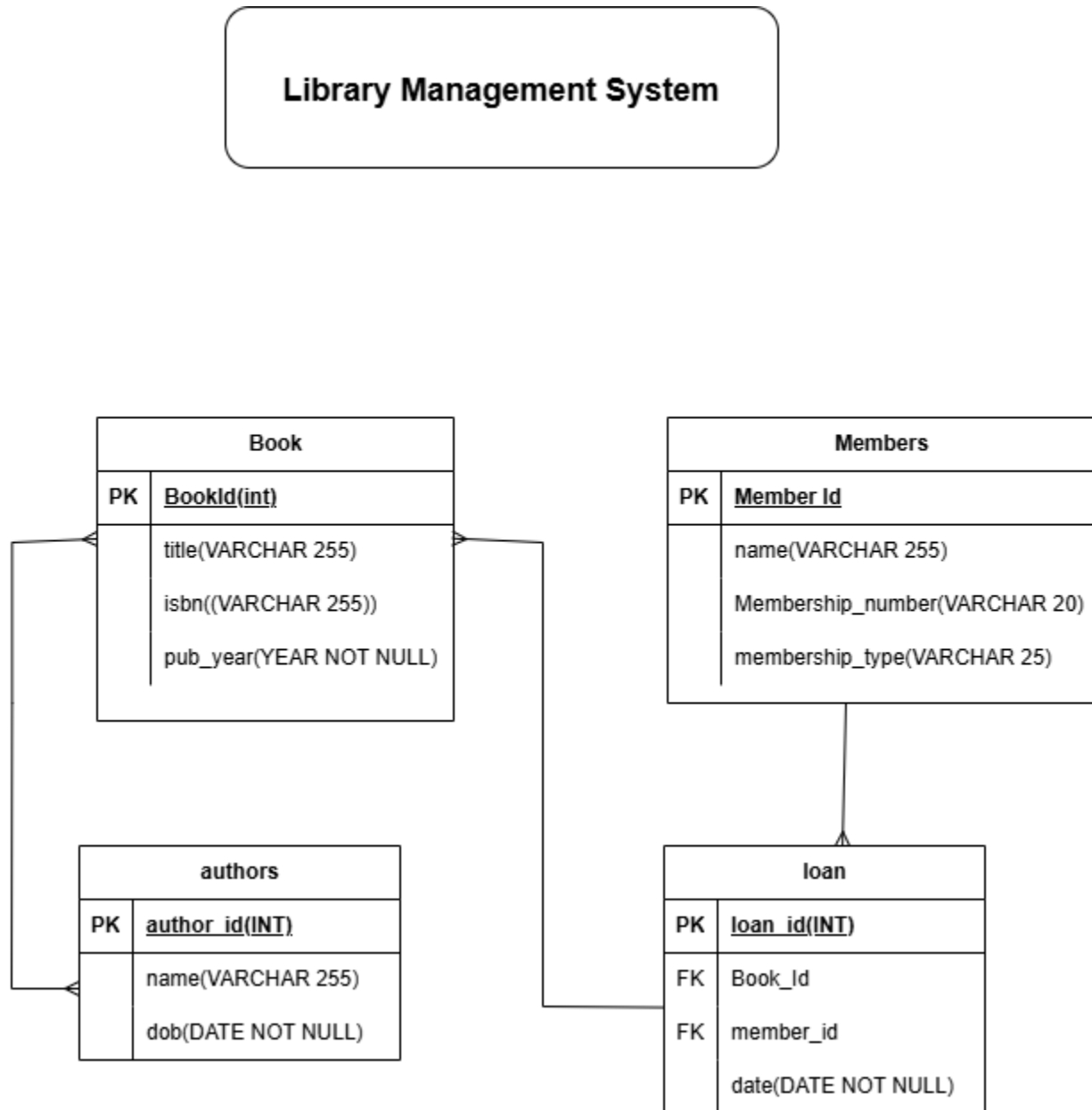


Question : You are tasked with designing a database for a library management system. The system must store information about **books, authors, members, and book loans**. Each **book** has a title, ISBN, and publication year. Each **author** has a name and date of birth. **Members** have a name, membership number, and membership type. Each **loan** involves a book, a member, and a loan date. A book can be written by multiple authors, and an author can write multiple books. A member can borrow many books, but each book can only be borrowed by one member at a time.

Task: Design an ER diagram for this library management system. Be sure to define all the entities, relationships, and any necessary attributes. Identify primary keys, foreign keys, and cardinality constraints .

Solution:



Question:

Given the following unnormalized relation:

StudentID	StudentName	CourseName	InstructorName	InstructorEmail	InstructorPhone
1	Nahim	Django-1	Nahid	a@xyz.com	123-456-789
2	Araf	Django-2	Salman	b@xyz.com	890-456-432
3	Moin	Django-1	Nahid	a@xyz.com	123-456-789

Task:

- Normalize the relation to 3rd Normal Form (3NF).
- Identify any functional dependencies in the given table and explain why normalization is necessary.

Solution :

a)First of all , we all understand about the normalization.

what is normalization?

Normalization is a process used in database design to organize data in a way that reduces redundancy and improves data integrity.

ডেটাকে সুন্দর ভাবে সাজানো ও ডেটার ডিজাইন কে উন্নত করার মাধ্যমে ডাটাবেজকে বোধগম্য করে তোলে ।

Basically there are 3 types of normalization.

1.1nf: প্রতিটি কলামে একটি মান ই থাকবে যেমনঃ courseName এ Django,react বা name এ nahim,nahim থাকবে না।

2:2nf: অবশ্যই 1NF এ থাকতে হবে এবং partial Dependency থাকবে না।
partial dependency কি?

3.3nf : টেবিলটি 2NF-এ থাকতে হবে এবং কোনও ট্রানজিটিভ নির্ভরতা (Transitive Dependency) থাকবে না।

আরো কিছু টাইপ আছে যেমনঃ BCNF ,4nf

এখন আমরা আমাদের টেবিলটাকে 3Nf এ কনভার্ট করবো।

প্রথমেই যদি আমরা খেয়াল করি , তাহলে দেখতে পাবো একটা কলামের আরেকটা কলামের

যোগসূত্র আছে যেমনঃ studentId এর সাথে studentName এর একটা নির্ভরতা রয়েছে সেটিই functional dependency ।
টেবিল থেকে আমরা functional dependency দেখতে পায় । যথাক্রমেঃ

StudentId -> StudentName , courseName

courseName -> courseInstructor, phone,email

আমরা যদি ভাল করে লক্ষ্য করি, সেই টেবিলে একি কলাম বা রো তে মাল্টিপল ডেটা নেই। সুতরাং, আমাদের টেবিলটি 1nf এ রয়েছে। এরপর যদি আমরা দেখি ,courseName এর সাথে courseInstructor এর partial dependency আছে। যেমনঃ আমরা যদি studentName delete করি তাহলে অন্যান্য যে তথ্য আছে তা হারিয়ে যাবে । সুতরাং এটির partial dependency remove করতে হবে । এর জন্য টেবিলটাকে আমরা দুইভাগে ভাগ করবো এবং নতুন একটা টেবিল সংযোজন করবো ।

studentId	StudentName
1	Nahim
2	araf

CourseId	CourseName	CourseInstructor	Email	phone
1	Django-1	Nahid	a@xyz.com	01293231

নতুন আরেকটি টেবিল বানায় যেটার মাধ্যমে এখন আমরা আরো সহজেই বুঝতে পারবো ।

StudentID	CourseID
1	1
2	2

আমরা যদি আমরা student কে ডিলিট করি তাহলে শুধু মাত্র সেই স্টুডেন্টের ডেটায় ডিলিট হবে এবং ডেটাবেজকে সহজ সরল সিম্পল করে তোলেছে ।

3NF: আমাদের টেবিলে যে একটা transitive Dependency ছিলো সেইটাও রিমোভ হয়ে গেছে 2nf এ । অর্থাৎ, StudentID → CourseName এবং

CourseName → InstructorName, InstructorEmail, InstructorPhone

এর মানে, StudentID → InstructorName, InstructorEmail, InstructorPhone একটি **ট্রানজিটিভ নির্ভরশীলতা**, কারণ **InstructorName, InstructorEmail, InstructorPhone** সরাসরি **StudentID**-এর উপর নির্ভরশীল নয়। এটি **CourseName**-এর মাধ্যমে নির্ভরশীল। 2nf এ যেহেতু আমরা remove করে ফেলেছি এবং 2nf এ যেহেতু আমরা আর Transitive Dependency দেখতে পাচ্ছি না সুতরাং এটি এখন 3nf Normalization .

Reference:

[Normalization Practice Exercise | Third Normal Form| Denormalization](#)

[Third Normal Form \(3NF\) | Database Normalization | DBMS](#)

Question

You are designing a database for an e-commerce platform. The system needs to store information about products, customers, orders, and payments. A product has a name, description, price, and category. A customer has a name, email, and shipping address. An order includes a customer and a list of products with their quantities. A payment is linked to an order and contains information about the payment method, amount, and payment status.

Task:

- Identify the relations (tables) for this e-commerce database and design the relational schema.
- Define primary keys, foreign keys, and any relevant constraints.

Solution:

