

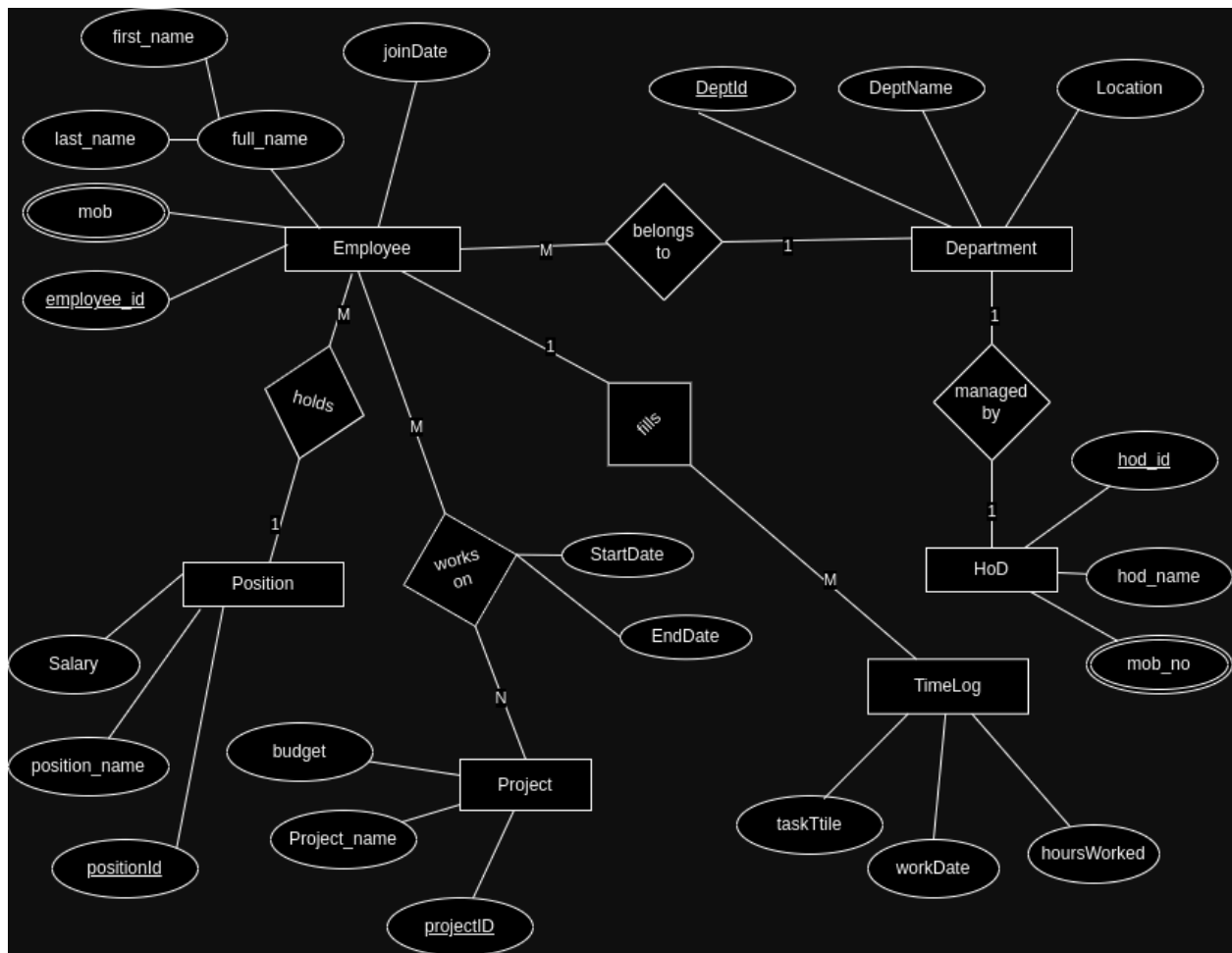
c. Aggregation

- i. In aggregation, the relation between two entities is treated as a single entity.

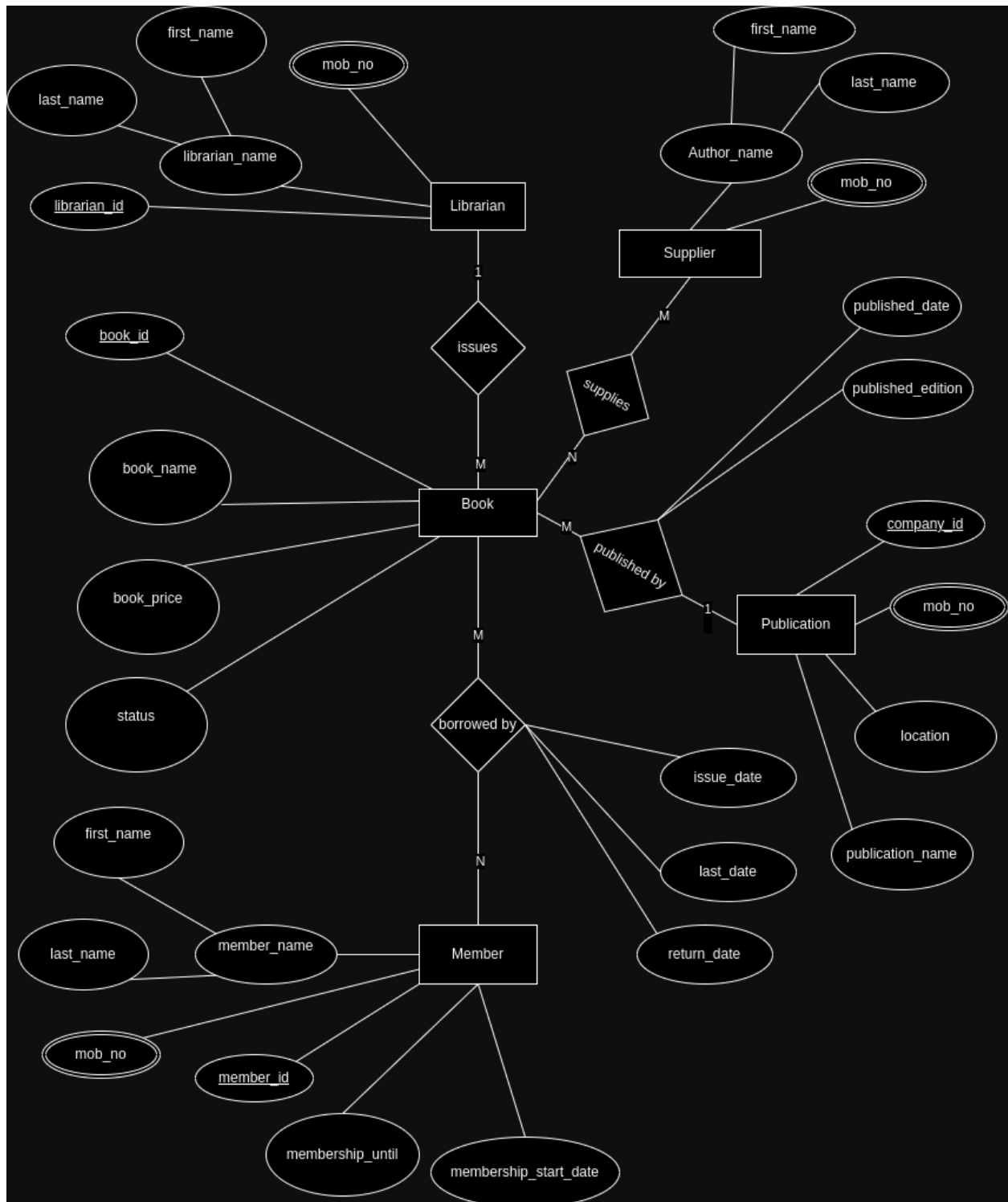
Day16

ER Diagrams

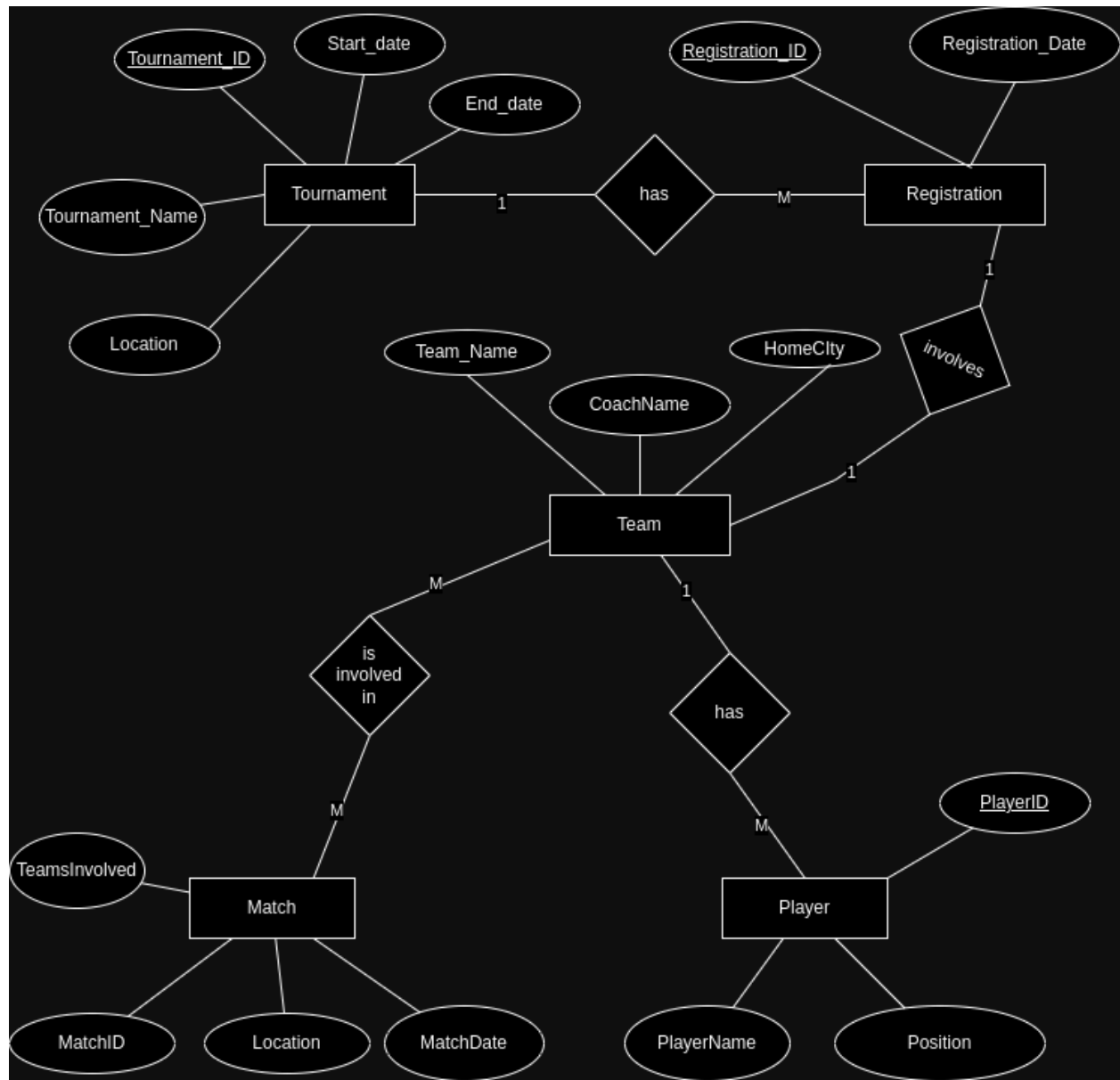
Employee Management System



Library Management System



Tournament Management System



What is RDBMS?

- a. based on the relational model introduced by E.F. Codd.
- b. Data is represented in terms of tuples (rows) in RDBMS.
- c. The total number of tuples at any one time in a relation is known as the table's cardinality.
- d. The Domain refers to the possible values each attribute can contain.
- e. Types of Integrity
 - i. Entity integrity
 - 1. ensures that each table has a unique identifier, which is often the primary key.
 - 2. prevents duplicate rows in a table.
 - ii. Domain integrity
 - 1. ensures that data entered into a column falls within a valid set of values.
 - 2. These constraints can include data type restrictions, format constraints, or value range restrictions.
 - 3. Ensures data validity within each column.
 - iii. Referential Integrity
 - 1. prevents orphaned records by ensuring that a foreign key value always points to an existing, valid record in another table.
 - iv. User-defined Integrity
 - 1. specific to the particular application and are not covered by the standard integrity constraints.

Datatypes

- a. Integer
 - i. store whole numbers (both positive and negative) without any decimal places.
- b. Char
 - i. store fixed-length character strings.
 - ii. If the string is shorter than specified length, then it is padded with spaces.
- c. Varchar
 - i. used to store variable-length character strings.
- d. DECIMAL
 - i. store fixed-point numbers
 - ii. Eg : Salary DECIMAL(10, 2)
- e. DATE
 - i. store calendar dates
- f. Datetime
 - i. stores date and time information.

- g. Bytea
 - i. byte array
 - ii. Postgresql datatype
 - iii. uses psycopg2 : popular python library for working with PostgreSQL
 - iv. used to store binary data, such as images, audio files, or any other large object (BLOB). [raw data]
- h. JSON
 - i. Lightweight text-based data interchange format.
 - ii. Human readable
 - iii. Uses a key-value pair structure with keys as strings, and values can be strings, numbers, objects, arrays, true, false, or null.
- i. BSON (JSONB)
 - i. BSON is a binary representation of JSON-like documents.
 - ii. Designed for efficient machine parsing and storage.
 - iii. Non human readable
 - iv. Supports more data types than JSON, including int, long, date, binary data, decimal, etc

Main differences between JSON and BSON

JSON data contains its data basic in JSON format.	BSON gives extra datatypes over the JSON data.
Database like AnyDB, redis, etc stores information in JSON format.	MongoDB stores data in BSON format.
JSON uses less space in comparison to BSON.	BSON uses more space as compared to JSON.
It is slow as compared to BSON.	It is faster than JSON.
It is used for the transmission of data.	It is used for the storage of the data.
It has no encoding and decoding technique.	It has encoding and decoding technique.

RDBMS vs NOSQL

Relational Database	NoSQL
It is used to handle data coming in low velocity.	It is used to handle data coming in high velocity.
It gives only read scalability.	It gives both read and write scalability.
It manages structured data.	It manages all type of data.
Data arrives from one or few locations.	Data arrives from many locations.
It supports complex transactions.	It supports simple transactions.
It has single point of failure.	No single point of failure.
It handles data in less volume.	It handles data in high volume.
Transactions written in one location.	Transactions written in many locations.
support ACID properties compliance	doesn't support ACID properties
Its difficult to make changes in database once it is defined	Enables easy and frequent changes to database
schema is mandatory to store the data	schema design is not required
Deployed in vertical fashion.	Deployed in Horizontal fashion.

ACID Properties

They are a set of properties which ensure that database transactions are completed efficiently.

a. Atomicity

- i. By this, we mean that either the entire transaction takes place at once or doesn't happen at all. There is no midway i.e. transactions do not occur partially.
- ii. Involves two operations
 1. Abort : If a transaction aborts, changes made to the database are not visible.
 2. Commit : If a transaction commits, changes made are visible.

b. Consistency

- i. integrity constraints must be maintained so that the database is consistent before and after the transaction. It refers to the correctness of a database.
- ii. correctness of a database.

c. Isolation

- i. Transactions occur concurrently and independently without interference.