

SQL

Celestin Niyomugabo

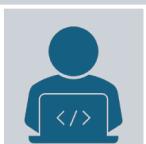
Introduction to Databases



Databases are organized collections of structured information, enabling efficient data storage, management, and sharing.



Types: Relational databases (organized in tables) are most common (e.g., MySQL, PostgreSQL), while NoSQL databases are designed for flexible data structures and high scalability.



SQL stands for **Structured Query Language**, a standard language for interacting with relational databases.

Key functions

1

Defining database
structure (create,
alter, drop tables)

2

Manipulating data
(insert, update,
delete)

3

Retrieving data
(SELECT)

Database Structure and Concepts



Tables: Core components with rows (records) and columns (fields).



Columns: Represent attributes with data types (e.g., integer, text, date).



Rows: Represent individual entries.



Primary Key: Unique identifier for each row.

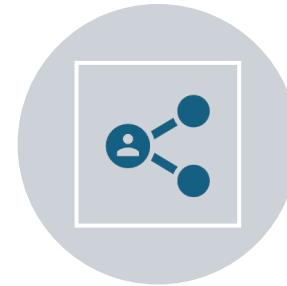


Foreign Key: Links tables by referencing the primary key of another table.

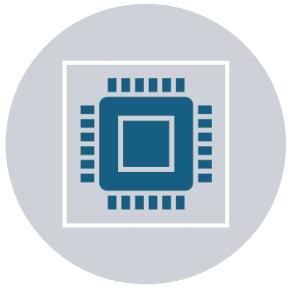
Business Rules of Data Modeling



Data Integrity: Ensuring data accuracy and consistency.



Entity-Relationship Diagram (ERD): Visual representation of data relationships.



Normalization: Organizing data to minimize redundancy and improve integrity.



Examples: "Each customer must have a unique ID." "An order must be associated with only one customer."

Do's

Plan	Plan thoroughly: Define clear business requirements, identify entities and relationships, and create an ERD.
Normalize	Normalize data: Minimize redundancy and improve data integrity.
Use	Use meaningful and consistent naming conventions: Make your database easy to understand and maintain.
Document	Document your database: Create clear documentation of the database schema, tables, and relationships.
Test	Test thoroughly: Test your database with sample data to ensure it functions as expected.
Review and maintain	Regularly review and maintain: Update your database schema as needed to accommodate changes in business requirements.

Don'ts



Design without a plan: Avoid ad-hoc design decisions; plan your database structure carefully.



Over-normalize: Excessive normalization can lead to complex queries and decreased performance.



Use ambiguous or generic names: Avoid using names like "Table1" or "Data"; choose descriptive names.



Ignore data integrity: Ensure data accuracy and consistency through constraints and validation rules.



Neglect security: Implement appropriate security measures to protect your data from unauthorized access.

ERD



An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system.



ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research.

ERD



Entity: A definable thing—such as a person, object, concept or event—that can have data stored about it.



Entities are categorized as strong or weak. A strong entity can be defined solely by its own attributes, while a weak entity cannot.



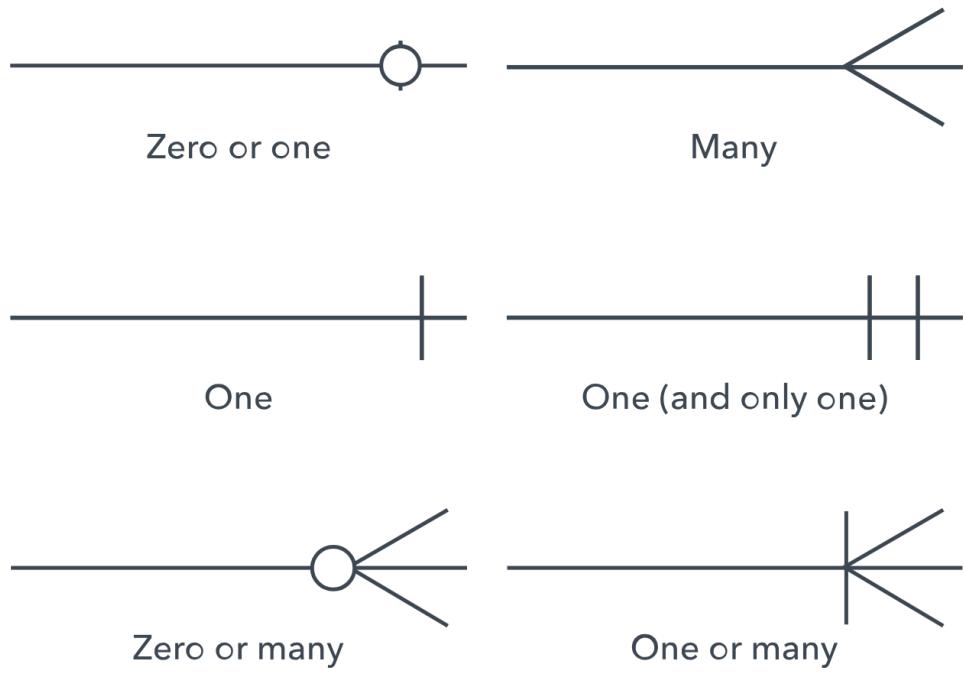
Entity keys: Refers to an attribute that uniquely defines an entity in an entity set. Entity keys can be super, candidate or primary.



Relationship: How entities act upon each other or are associated with each other. Think of relationships as verbs.



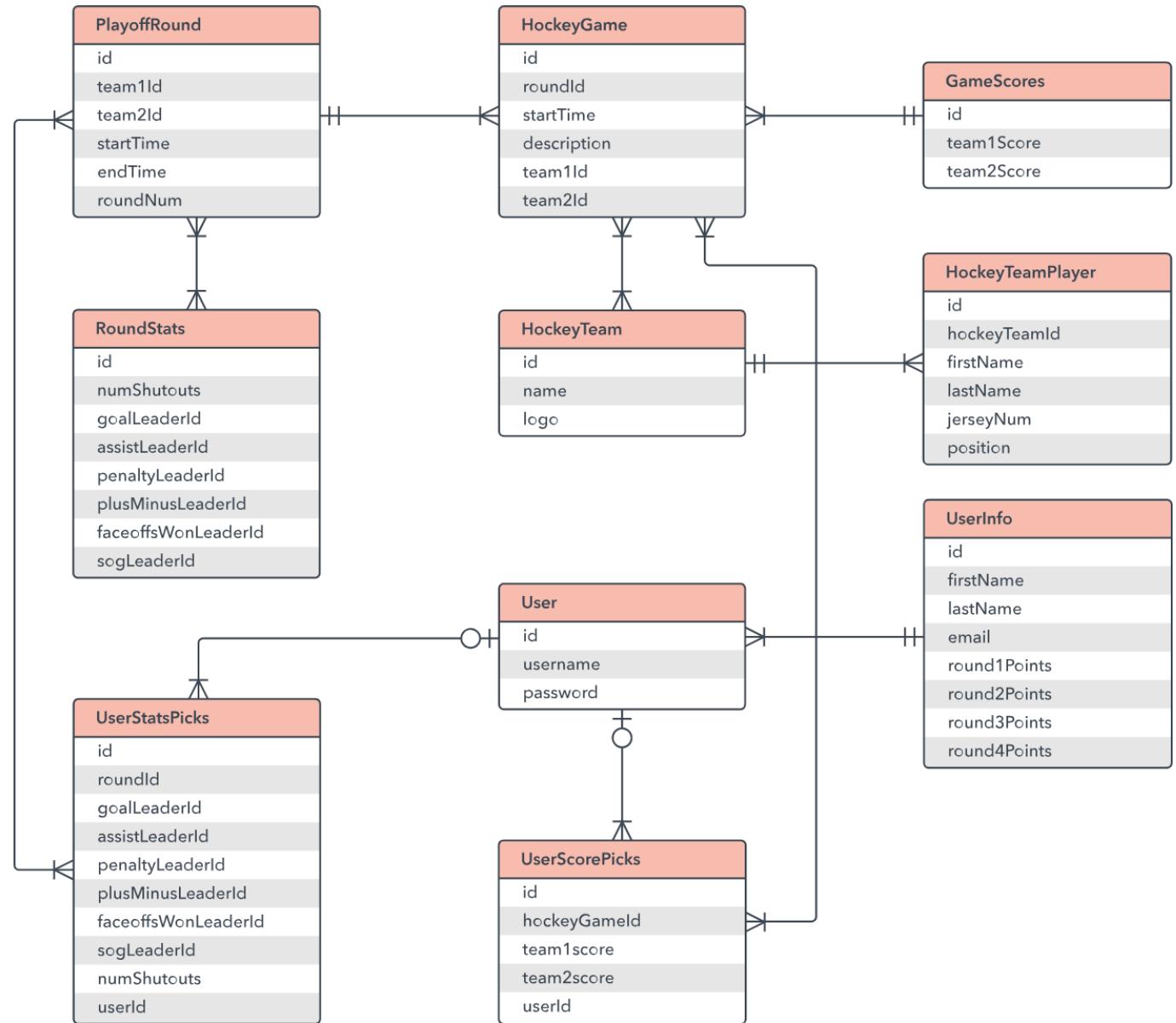
Recursive relationship is when an entity has a relationship with itself.



Cardinality

- Cardinality defines the numerical attributes of the relationship between two entities or entity sets.
- The three main cardinal relationships are **one-to-one**, **one-to-many**, and **many-many**.

ERD



ERD

