IT TECHS

RELATIONAL MODEL

PRESENTED BY

Abueg, Josiah
Auguis, Ronecia Mae
Escalona Rodel
Gallarte, Mark
Kabigting, Caizen Dee
Rivas, Raymond

TABLE OF CONTENTS

RELATIONAL MODEL

- represents how data is stored and managed in Relational Databases where data is organized into tables, each known as a relation.
- Introduced by **Edward F. Codd** of IBM in 1970
- Based on **mathematical set theory**
- Each table consists of **rows (tuples) and columns (attributes)**
- Relationships between tables are well-defined
- Serves as the foundation of modern **Relational Database Management Systems (RDBMS)**

RELATIONAL MODEL CONCEPTS

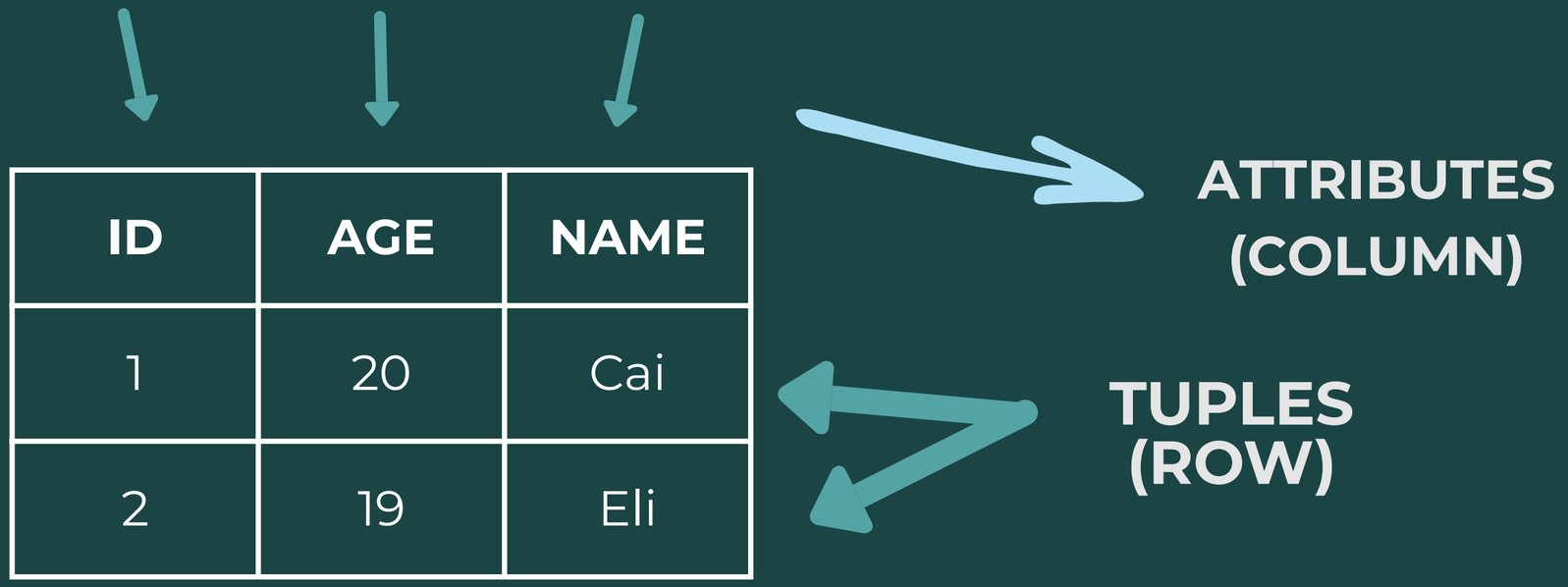
Relation (Table): is a two-dimensional table used to store a collection of data.

Customer_ID	Name	Contact_Number	Address
101	Maria Cruz	09171234567	Manila
102	Juan Dela	09281234567	Quezon City
103	Ana Reyes	09193456789	Cebu

RELATIONAL MODEL CONCEPTS

Tuple (Row): A single row in a table that represents a complete record.

Attribute (Column): A column in a table that contains values for a specific data point.



RELATIONAL MODEL CONCEPTS

Relation Schema: is the metadata or blueprint that defines the structure and organization of a table (or "relation") in a relational database.

Customer (Customer_ID, Name, Contact_Number, Address)

RELATIONAL MODEL CONCEPTS

Domain: is the set of all possible, valid, and atomic (indivisible) values for an attribute (column).

Degree: The number of attributes (columns) in a relation

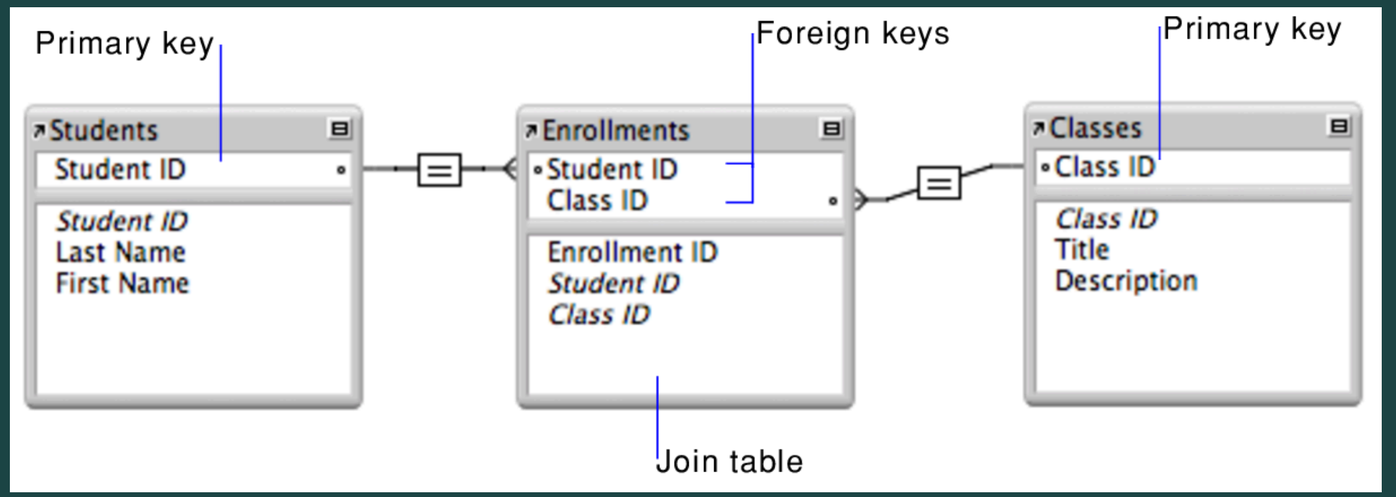
Cardinality: The number of tuples (rows) in a relation.

ID	AGE	NAME
1	20	Cai
2	19	Eli

RELATIONAL MODEL CONCEPTS

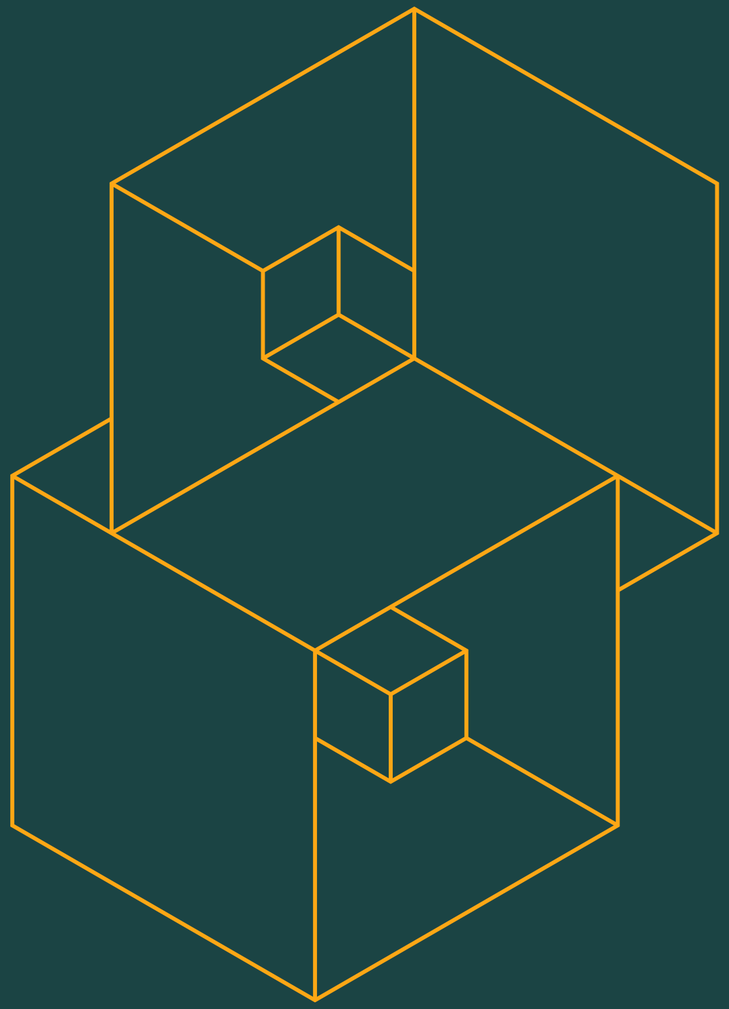
Primary Key: is a column or a set of columns in a relational database table that uniquely identifies each record (row) in that table.

Foreign Key: is a column or a set of columns in one table that links to the primary key of another table, establishing a relationship between them.



RELATIONAL INTEGRITY CONSTRAINTS

Integrity constraints are rules that maintain the accuracy, consistency, and reliability of data in a relational database.



TYPES OF RELATIONAL INTEGRITY CONSTRAINTS

- **Domain Constraints: Restrict the values an attribute can take to its defined domain, such as enforcing data types like INTEGER or DATE.**

- **Key Constraints: Ensure that every tuple in a relation is unique. The primary key is the main mechanism for enforcing this.**

- **Entity Integrity Constraints: Dictate that the primary key of a relation cannot have a NULL value.**

- **Referential Integrity Constraints: Ensure that a foreign key value in one table must have a matching primary key value in the table it references, preventing "orphaned" records.**

EXAMPLES

Domain Constraints

Column Name	Data Type	Constraint/Rule	Example Data
employee_age	INTEGER	Must be between 18 and 65	25 (Valid), 16 (Invalid)
order_date	DATE	Must be a valid date format	'2024-09-24' (Valid), 'September 24th' (Invalid)
is_active	BOOLEAN	Can only be TRUE or FALSE	TRUE (Valid), 'yes' (Invalid)

EXAMPLES

Key Constraints

Table	Column Name	Constraint	Example Data
Products	product_id	PRIMARY KEY	P-101, P-102 (Valid). Cannot insert another P-101.
Employees	email	UNIQUE	johnfrost@email.com, janesmith@email.com (Valid). Cannot insert another johnfrost@email.com.

EXAMPLES

Entity Integrity Constraints

Table	Primary Key	Constraint	Example Data
Customers	customer_id	**NOT NULL**	C-5001, C-5002 (Valid). Cannot insert a new customer with a NULL customer_id.

EXAMPLES

Referential Integrity Constraints

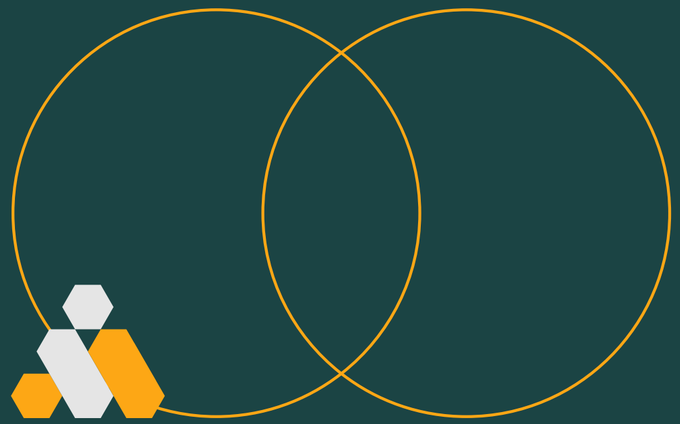
Table	Column Name	Constraint	Example Data
Orders	customer_id	**FOREIGN KEY (references customer_id in Customers table)**	C-5001 (Valid, as C-5001 exists in the Customers table).
			Cannot insert an order for a customer_id like C-9999 if that customer does not exist in the Customers table.

OPERATIONS IN RELATIONAL MODEL

THE RELATIONAL MODEL SUPPORTS TWO MAIN TYPES OF OPERATIONS:

- DATA MANIPULATION
- RELATIONAL ALGEBRA

OPERATIONS IN RELATIONAL MODEL



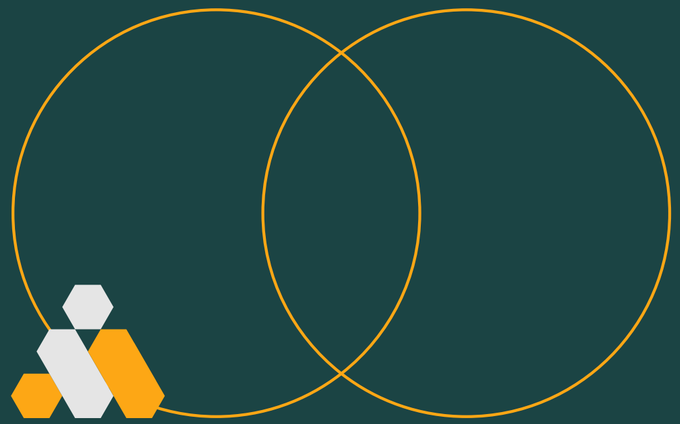
THE DATA MANIPULATION LANGUAGE (DML)

- is a part of a database language, most commonly SQL, used for managing existing data in a database.

THESE OPERATIONS MODIFY THE DATA WITHIN TABLES:

· **INSERT**:
Adds a new tuple (row) to a relation.
· **UPDATE**:
Modifies attribute values within one or more existing tuples.
· **DELETE**:
Removes one or more tuples from a relation.
· **SELECT**:
Retrieves a specific range of data from a relation based on certain criteria.

OPERATIONS IN RELATIONAL MODEL



RELATIONAL ALGEBRA

- is a procedural query language based on a set of fundamental mathematical operations.

Selection (σ): Filters rows from a table based on a given condition, returning a subset of tuples.

Projection (πpi): Selects specific columns from a table, returning a vertical subset of the relation.

Union (∪): Combines the tuples of two relations that have the same set of attributes, removing duplicates.

Set Difference (-): Returns tuples from the first relation that do not exist in the second.

Cartesian Product (×): Combines every tuple of the first relation with every tuple of the second relation.

Join (⋈): Combines related tuples from two or more relations based on a common attribute.

ADVANTAGES OF USING RELATIONAL MODEL

Simple model: Relational Model is simple and easy to use in comparison to other languages.

Flexible: Relational Model is more flexible than any other relational model present.

Secure: Relational Model is more secure than any other relational model.

Data Accuracy: Data is more accurate in the relational data model.

Data Integrity: The integrity of the data is maintained in the relational model.

Operations can be Applied Easily: It is better to perform operations in the relational model.

Disadvantages of the Relational Model

- Performance: The relational model can experience performance issues with very large databases.
- Complexity for Complex Data: The model struggles with hierarchical or complex data relationships, which might be better handled with other models like the Graph or Document model.
- Normalization Overhead: Extensive use of normalization can result in complex queries and slower performance.

DISADVANTAGES OF USING RELATIONAL MODEL

BEST PRACTICES FOR CREATING A RELATIONAL MODEL

1. Understand Business Requirements: Before designing, clarify what data is needed and how it will be used by all stakeholders.
2. Follow Normalization Principles: Minimize data redundancy and improve consistency by normalizing tables to at least the Third Normal Form (3NF).
3. Define Clear Naming Conventions: Use consistent and descriptive names for tables, columns, and constraints to enhance readability and maintainability.

BEST PRACTICES FOR CREATING A RELATIONAL MODEL

4. Use Appropriate Data Types: Select the correct data type for each column (e.g., integer, date, text) to ensure accuracy and optimize storage.

5. **Utilize Constraints Effectively**: Enforce data integrity by properly defining primary keys, foreign keys, and other constraints.

6. **Optimize with Indexing**: Add indexes to frequently queried columns to improve retrieval speed, but use them wisely to avoid slowing down write operations.

BEST PRACTICES FOR CREATING A RELATIONAL MODEL

7. Document the Design: Clearly document the schema, relationships, and business rules. This is essential for future maintenance and new team members.

8**. Plan for Scalability:** Consider future growth by using strategies like partitioning large tables or sharding data across multiple servers.

THANK YOU FOR LISTENING!!!