Logo

Description automatically generated

rEvision

Relational Database Development

|  |  |
| --- | --- |
| ERD |  |

# Entity Relationship Diagram Revision

An Entity Relational Diagram is a visual representation of a database that shows how the elements and data are related. Within an ERD there are a few components to focus on:

* Entities: Groups of related data stored together in a single component, a table.
* Attributes: Are characteristics of an entity, a column.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity | |  |  | | --- | --- | | Customer | | |  | customerID | |  | firstName | |  | lastName | |  | street | |  | postcode | |  | phone | | Attribute |

Note: It is important to understand that the table above is a representation of the table structure and not the table data. The table data will look like below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Customer | | | | | |
| customerID | firstName | lastName | street | postcode | Phone |
| 10001 | Hiroshi | Tanaka | 123 Sakura St | 100-0001 | +81-3-1234-5678 |
| 10002 | Maria | Garcia | 202 Crescent Rd | 28013 | +34-91-123-4567 |
| 10003 | Amina | Khan | 789 Maple Drive | M5V 2T6 | +1-416-123-4567 |
| 10004 | Wei | Zhang | 101 Bamboo Lane | 200120 | +86-21-1234-5678 |
| 10005 | Fatima | Ali | 202 Crescent Rd | 122001 | +91-124-123-4567 |

* Relationships: Relationships between entities are represented as lines.

An entity is a person, place, or thing that can be stored in a database. For example, when you buy a laptop online, the transaction can be recorded in a database as represented below.

Lines between tables represent how entities interact with each other, showing connections between the two tables.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Customer | | |  | customerID | |  | firstName | |  | lastName | |  | street | |  | postcode | |  | phone | | |  |  | | --- | --- | | Order | | |  | orderID | |  | orderDate | |  | shipTo | |  | shippingDate | | |  |  | | --- | --- | | Product | | |  | productID | |  | name | |  | description | |  | price | |

The notation on each line is called cardinality, defining the context of the relationship.

|  |  |  |
| --- | --- | --- |
| * Cardinalities: Symbols at the ends of   Relationship lines show how  tables interact with each other. | |  | | --- | |  | |

In the example above, cardinality is explained as follows:

Customer to Order: A customer can have zero to many orders.

An order must have one and only one customer.

Order to Product: An order must have one product. A product can be associated with zero to many orders.

## Primary Key (pk)

Imagine you have a table with over a million customers and you need to find a specific one. This is where primary keys come into play. A primary key is a unique identifier for each record in a table.

In this table, the primary key is customerID.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Customer | | | | | |
| customerID | firstName | lastName | street | postcode | Phone |
| 10001 | Hiroshi | Tanaka | 123 Sakura St | 100-0001 | +81-3-1234-5678 |
| 10002 | Maria | Garcia | 202 Crescent Rd | 28013 | +34-91-123-4567 |
| 10003 | Amina | Khan | 789 Maple Drive | M5V 2T6 | +1-416-123-4567 |
| 10004 | Wei | Zhang | 101 Bamboo Lane | 200120 | +86-21-1234-5678 |
| 10005 | Fatima | Ali | 202 Crescent Rd | 122001 | +91-124-123-4567 |
| 10006 | John | Smith | 303 Elm St | 02110 | +1-617-123-4567 |
| 10007 | Ekaterina | Ivanova | 404 Red Square Ave | 109012 | +7-495-123-4567 |
| 10008 | Fatima | Ali | 505 Pyramid Blvd | 11511 | +20-2-123-4567 |
| 10009 | Chipo | Ncube | 606 Victoria Falls Rd | 263 | +263-4-123-456 |
| 10010 | Sofia | Rossi | 707 Via Roma | 00184 | +39-06-123-4567 |

By using the customerID of 10005, you can quickly retrieve all information about Fatima Ali.

Primary Key Rules:

1. Each value in the primary key column must be unique across the table.
2. Once a primary key is assigned to a record, it cannot be modified.
3. A primary key must always have a value; it cannot be left empty (null).

Choosing a Primary Key

When selecting a primary key for the Customer table, you cannot use firstName, lastName, or a combination of both, because there can be duplicate names. For example, Fatima Ali appears twice but represents two different customers. Similarly, you cannot rely on the address (202 Crescent Rd appears twice) or phone number, as these can also change, breaking the primary key rules. This leaves customerID, which is designed to increment by one with each new customer.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Customer | | | pk | customerID | |  | firstName | |  | lastName | |  | street | |  | postcode | |  | phone | | |  |  | | --- | --- | | Order | | | pk | orderID | |  | customerID | |  | productId | |  | orderDate | |  | shipTo | |  | shippingDate | | |  |  | | --- | --- | | Product | | | pk | productID | |  | name | |  | description | |  | price | |

customerID meets all the primary key criteria: it is unique and non-null.

## Foreign Key (fk)

A foreign key is a field in one table that uniquely identifies a row of another table. It establishes a link between the data in the two tables. While a primary key uniquely identifies a record within its own table, a foreign key is used to reference that primary key from a different table. This allows you to understand how tables relate to each other.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Customer | | | pk | customerID | |  | firstName | |  | lastName | |  | street | |  | postcode | |  | phone | | |  |  | | --- | --- | | Order | | | pk | orderID | | fk | customerID | | fk | productId | |  | orderDate | |  | shipTo | |  | shippingDate | | |  |  | | --- | --- | | Product | | | pk | productID | |  | name | |  | description | |  | price | |

In the example above, customerID is the primary key for the Customer table. By including customerID as a foreign key in the Order table, you can link each order to the corresponding customer. This relationship allows you to retrieve all orders associated with a specific customer, thereby connecting the two tables.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Customer | | | | | |
| pkCustomerID | firstName | lastName | street | postcode | Phone |
| 10001 | Hiroshi | Tanaka | 123 Sakura St | 100-0001 | +81-3-1234-5678 |
| 10002 | Maria | Garcia | 202 Crescent Rd | 28013 | +34-91-123-4567 |
| 10003 | Amina | Khan | 789 Maple Drive | M5V 2T6 | +1-416-123-4567 |
| 10004 | Wei | Zhang | 101 Bamboo Lane | 200120 | +86-21-1234-5678 |
| 10005 | Fatima | Ali | 202 Crescent Rd | 122001 | +91-124-123-4567 |
| 10006 | John | Smith | 303 Elm St | 02110 | +1-617-123-4567 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Order | | | | |
| orderID | fkCustomerID | shipTo | orderDate | shippingDate |
| 70001 | 10001 | 123 Sakura St | 2023-01-15 | 2023-01-20 |
| 70002 | 10002 | 202 Crescent Rd | 2023-01-18 | 2023-01-23 |
| 70003 | 10005 | 202 Crescent Rd | 2023-01-15 | 2023-01-20 |
| 70004 | 10006 | 303 Elm St | 2023-01-20 | 2023-01-25 |
| 70005 | 10005 | 202 Crescent Rd | 2023-01-20 | 2023-01-25 |
| 70006 | 10003 | 789 Maple Drive | 2023-01-20 | 2023-01-25 |

Foreign Key Rules:

1. The foreign key indicates that the table containing it (child table) is related to another table where the primary key resides (parent table).
2. The foreign key in the child table must match a primary key value in the parent table.
3. A foreign key does not have to be unique. Multiple records in the child table can reference the same primary key value in the parent table.
4. The foreign key and the referenced primary key must have the same data type and size to ensure proper matching and indexing.

## Bridge tables

A bridge table is used to handle many-to-many relationships between two tables. Leaving a many-to-many relationship directly between two tables without a bridge table leads to data redundancy and integrity issues.

In the current example, there is a many-to-many relationship between the Order and Product tables.

Order to Product: An order can have many products.

A product can have many orders.

To manage this relationship, a bridge table is required that references the primary keys from both the Order and Product tables.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Order | | | | | |
| orderID | fkCustomerID | shipTo | orderDate | shippingDate | quantity |
| 70001 | 10001 | 123 Sakura St | 2023-01-15 | 2023-01-20 | 2 |
| 70002 | 10002 | 202 Crescent Rd | 2023-01-18 | 2023-01-23 | 1 |
| 70003 | 10005 | 202 Crescent Rd | 2023-01-15 | 2023-01-20 | 4 |
| 70004 | 10006 | 303 Elm St | 2023-01-20 | 2023-01-25 | 6 |
| 70005 | 10005 | 202 Crescent Rd | 2023-01-20 | 2023-01-25 | 3 |
| 70006 | 10003 | 789 Maple Drive | 2023-01-20 | 2023-01-25 | 1 |

|  |  |  |  |
| --- | --- | --- | --- |
| Product | | | |
| pkProductID | name | description | price |
| 30001 | Laptop | 15-inch, 16GB RAM | 1500 |
| 30002 | Smartphone | 6-inch, 128GB | 800 |
| 30003 | Headphones | Noise-cancelling | 200 |

Bridge Table: OrderProduct

The OrderProduct table will contain foreign keys referencing the primary keys from both the Order and Product tables.

|  |  |  |  |
| --- | --- | --- | --- |
| OrderProduct | | | |
| pkOrderProductID | fkOrderID | fkProductID | quantity |
| 1 | 70001 | 30001 | 1 |
| 2 | 70001 | 30003 | 2 |
| 3 | 70002 | 30002 | 1 |
| 4 | 70003 | 30001 | 1 |
| 5 | 70003 | 30003 | 3 |

This table will also include an additional quantity attribute to indicate how many units of a product are included in an order.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Customer | | | pk | customerID | |  | firstName | |  | lastName | |  | street | |  | postcode | |  | phone | | |  |  | | --- | --- | | Order | | | pk | orderID | |  | customerID | |  | orderDate | |  | shipTo | |  | shippingDate | | |  |  | | --- | --- | | OrderProduct | | | pk | orderID | | pk | productID | |  | price | | |  |  | | --- | --- | | Product | | | pk | productID | |  | name | |  | description | |  | price | |