<https://www.geeksforgeeks.org/introduction-of-er-model/>

Types of Keys:

1. Candidate key
2. Primary key
3. Composite key
4. Alternate key
5. Unique key
6. Foreign key

**Composite Key:**

A key can be specified on just one column of a database table or on a combination of multiple columns. When a key is specified on a combination of multiple columns then such key is referred to as composite key.

**Candidate Key:**

It is any set of columns that have a unique combination of values in each row, with the additional constraint that removing any column could produce duplicate combinations of values. Also, the value of a candidate key column cannot be NULL.

**Primary Key:**

One of the candidate keys which has been chosen as the primary unique identifier of each row of a table is referred to as *primary key*. The primary key values are unique and not null.

**Alternate Key:**

Any candidate key which has not been chosen as the primary key of a table is referred to as *alternate key*.

**Unique Key:**

It is any set of columns that have a unique combination of values in each row, but NULL values are allowed.

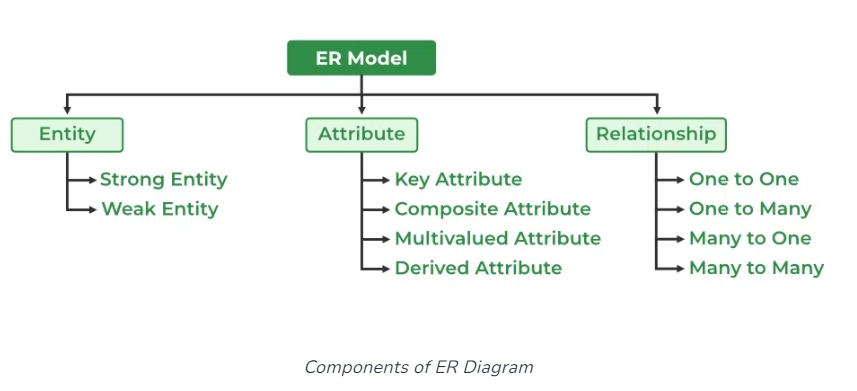
(NOTE: No two NULL values are not the same. So having two or more rows with NULL values in a column is not considered as duplication.)

**Foreign Key:**

Foreign Key imposes a constraint that the values in a set column are subset or equal to (⊆) the values of a primary key column of the same table or another table.

Customer data of a bank or e-commerce business (for .e.g HDFC or Amazon)





**Diagrammatic representation of Customer entity with its attributes:**

CUSTOMER

**NOTE:** A rectangle is used to represent an entity, ellipses for attributes, and lines to indicate that the attributes are related to a particular entity. The key attribute (primary key) will be underlined.

**Representation of Customer table in a Entity Model diagram:**

**CUST**

**CUSTNO**

CUST\_NAME

EMAIL

PAN

MOBILE\_NUM

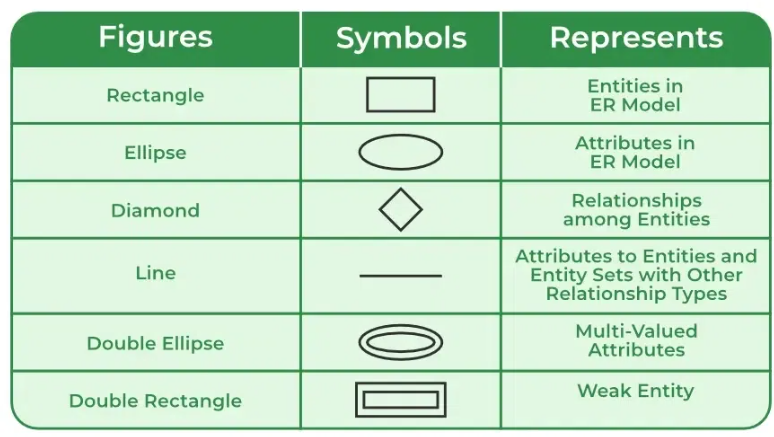
GENDER

DOB

**Symbols Used in ER Model**

ER Model is used to model the logical view of the system from a data perspective which consists of these symbols:

1. Rectangles: Rectangles represent Entities in ER Model.
2. Ellipses: Ellipses represent Attributes in ER Model.
3. Diamond: Diamonds represent Relationships among Entities.
4. Lines: Lines represent attributes to entities and entity-sets with other relationship types.
5. Double Ellipse: Double Ellipses represent Multi-Valued Attributes.
6. Double Rectangle: Double Rectangle represents a Weak Entity.
7. Composite Attribute: A composite attribute will be split further into its component attributes



**Above CUSTOMER entity has been modified to composite attributes:**

1. Cust Name has been split into First Name, Middle Name and Last Name.
2. A new attribute Address has been added to Customer entity, which in turn can be split into Address Line 1, Address Line 2, City, State, Postal Code and Country

* *Notice the double ellipse used for CUST NAME and ADDRESS attributes of Customer as they are composite attributes.*

CUSTOMER

**Simple Attribute vs. Composite Attribute:**

**Simple Attribute:** An attribute that cannot be split into further components is a simple attribute/

**Composite Attribute:** An attribute that can be further split into component parts is a composite attribute.

In the above example of Customer entity, the two attributes CUST NAME and ADDRESS are composite attributes, while the others are simple attributes.

**Single-Valued Attribute vs. Multi-Valued Attribute:**

**Single-Valued Attribute:** An attribute for which for a given entity instance there can exist at most one value. *[If the attribute is nullable it is zero or one value; if it is not-nullable, then it is one-and-only-one.}*

Eg: One customer can have only one value for Gender, Date of Birth, PAN etc.

**Multi-Valued Attribute:** An attribute for which for a given entity instance there can exist multiple values.

Eg: One customer can have multiple email ids, multiple phone numbers and multiple addresses (in a B2B scenario a customer company can have different addresses like Corporate address, one or more plant address, billing address etc. Even in case of an individual there could be permanent address and current address)

**Derived Attribute:** An attribute whose value is derived (computed) from another attribute is known as a derived attribute. Generally, derived attributes are not stored in the database.

A derived attribute is depicted with a dashed ellipse.

CUSTOMER

**NOTE:** Age must be derived (computed) based on the date of birth.

* Similarly, while storing sales or purchasing data based on quantity and price of a material or product the amount of each item in the order and the total value of the order can be derived.

**Assignments:**

Prepare entity model representations for below requirements and identify the different types of keys, simple and composite attributes, single-valued & multi-valued attributes.

1. Entity: Employee

Attributes: Employee Number, name, date of birth, date of joining, address, designation, salary (salary is split into basic pay, HRA, PF, medical allowance, and variable pay)

1. Entity: Material

Attributes: Material Number, Material Description, Material type, Industry sector

Material type values could include raw material, semi-finished material, finished material, spares etc.

Industry sector values could include Chemical, Beverages, Auto parts, Automobile, Electrical, Electronics, Pharmaceuticals etc.

**Entity, Entity Type, Entity Instance and Entity Set:**

**Entity:** A thing with distinct existence. (in plain English)

In RDBMS “entity” is a thing about which data is stored. The data related to an entity is stored in a database table.

The thing about which data is stored could be a physical or a non-physical entity.

Physical Entities: Employee, Product, Manufacturing Plant, Bank, Customer, Vendor, Student, Faculty etc.

Non-Physical Entities: Sale, Sales Return, Purchase, Purchase Return, Course etc.

**Entity Type, Entity Instance and Entity Set:**

Entity Type is a generic representation of a thing, and Entity Instance is one specific representation.

Eg: If Customer is an entity type, a customer identified by the attributes and their values as

Customer number = 1001, Customer Name = Vishal, Email: [vishal@gmail.com](mailto:vishal@gmail.com) is an entity instance. In other words, each row in the database table represents one entity instance.

A unique collection of entity instances is an Entity Eet.

|  |  |  |
| --- | --- | --- |
| **Customer Number** | **Customer Name** | **Email** |
| 1001 | VISHAL | [vishal@gmail.com](mailto:vishal@gmail.com) |
| 1002 | KIRAN | [kiran@yahoo.co.in](mailto:kiran@yahoo.co.in) |
| 1003 | BEENA | beena@gmail.com |

**Strong Entity vs Weak Entity:**

EMPLOYEE

DEPENDENT

HAS

Both Employee and Dependents have attributes of their own but the existence of Dependents is possible only when an Employee exists. In this example Employee is a strong entity and Dependent is a weak entity.

While a strong entity is depicted as a single rectangle, a weak entity is depicted as a double rectangle.