**E-R Model**

E-R Model is a logical representation of data in an organization. It views the entire system as a collection of entities related to one another. It is used to describe the elements of a system and their relationships.

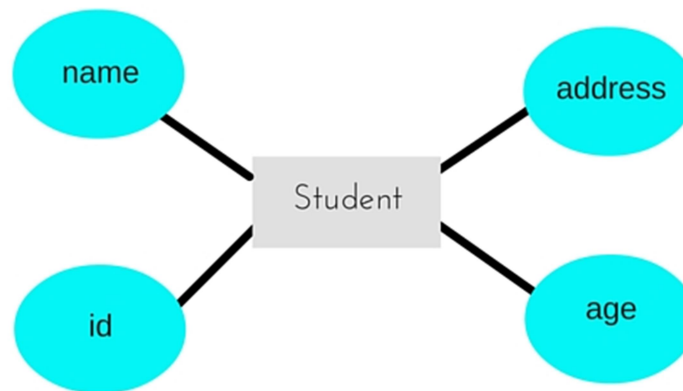**Components of E-R Model**

The E-R Model has three main components.

**1) Entity**

An Entity can be any object, place, person or class. In E-R Diagram, an entity is represented using rectangles.



**2) Attribute**

An Attribute describes a property or characteristic of an entity. An attribute is represented using eclipse.



**3) Relationship**

A Relationship describes relations between entities. Relationship is represented using diamonds.
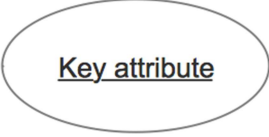
There are three types of relationship that exist between Entities.

- Binary Relationship: Binary Relationship means relation between two Entities.
- Recursive Relationship: When an Entity is related with itself it is known as Recursive Relationship.
- Ternary Relationship: Relationship of degree three is called Ternary relationship.

**ER Diagram Symbols and Notation**
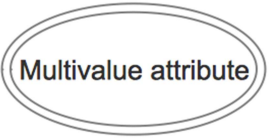
Entity-relationship diagrams are essential to modeling anything from simple to complex databases

| Entities | | |
|---|---|---|
| **Symbol** | **Shape Name** | **Symbol Description** |
| Entity | Entity | These entities are independent from other entities. |
| Weak Entity | Weak Entity | Weak entities depend on some other entity type. |
| Associative Entity | Associative Entity | Associative entities relate the instances of several entity types. |
| | | |

| Attributes | | |
|---|---|---|
| **Symbol** | **Shape Name** | **Symbol Description** |
| Attribute | Attribute | Attributes are characteristics of an entity. |
| Key attribute | Key attribute | Key attribute represents the main characteristic of an Entity. |
| Multivalue attribute | Multivalued attribute | Multivalued attributes are those that can take one or more than one value. |
| Derived attribute | Derived attribute | Derived attributes are attributes whose value can be calculated from related attribute values. |

| Relationships | | |
|---|---|---|
| **Symbol** | **Shape Name** | **Symbol Description** |
| Relationship | Strong relationship | Relationships are associations between or among entities. |
| Relationship | Weak relationship | Weak Relationships are connections between a weak entity and its owner. |

**Mapping Constraints (Cardinalities)**

Mapping cardinalities define the number of association between two entities.
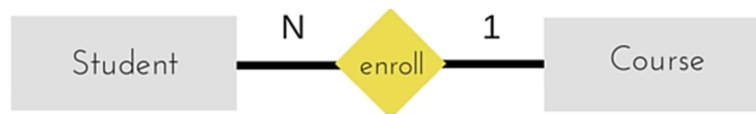
Mapping cardinalities –

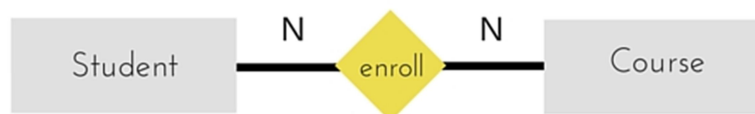**1. One to One:** One entity can be associated with at most one entity and vice versa.
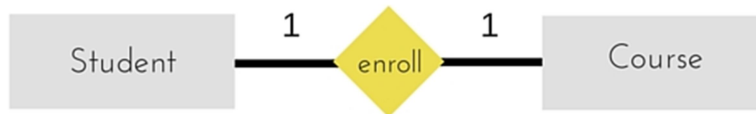


**2. One to Many:** One entity can be associated with more than one entity.
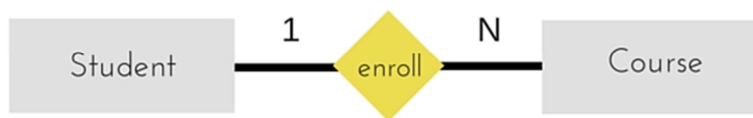


**3. Many to One:** More than one entity can be associated with at most one entity.



**4. Many to Many:** One entity can be associated with more than one entity and vice versa.



**Primary Key**

A primary is a column or set of columns in a table that uniquely identifies tuples (rows) in that table.

**Super Key**

A super key is either a single or a combination of attributes that can be used to uniquely identify a database record.

**Candidate Key**

A candidate key is a column, or set of columns, in a table that can uniquely identify any database record without referring to any other data.

**Secondary or Alternative key**

The candidate keys which are not selected for primary key are known as secondary keys or alternative keys.

**Foreign Key**

A foreign key is a key used to link two tables together. This is sometimes also called as a referencing key.

**Composite Key**

A primary key that is made up of more than one attribute is known as a composite key.

**Generalization**

Generalization is a bottom-up approach in which two lower level entities combine to form a higher level entity. In generalization, the higher level entity can also combine with other lower level entity to make further higher level entity.
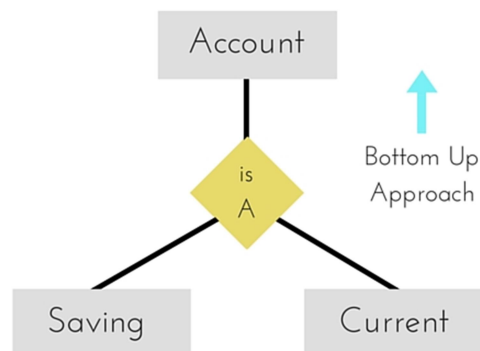


**Specialization**

Specialization is opposite to Generalization. It is a top-down approach in which one higher level entity can be broken down into two lower level entities. In specialization, some higher level entities may not have lower-level entity sets at all.
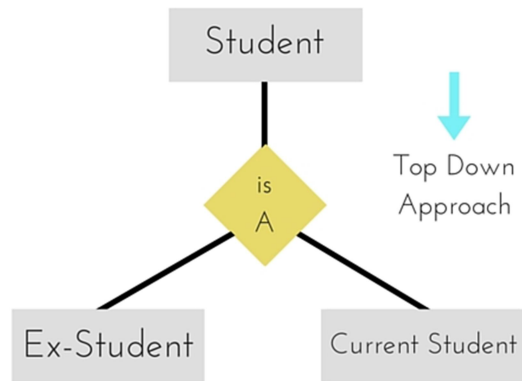
**Aggregation**

Aggregation is a process when relation between two entities is treated as a single entity. Here the relation between Center and Course is acting as an Entity in relation with Visitor.
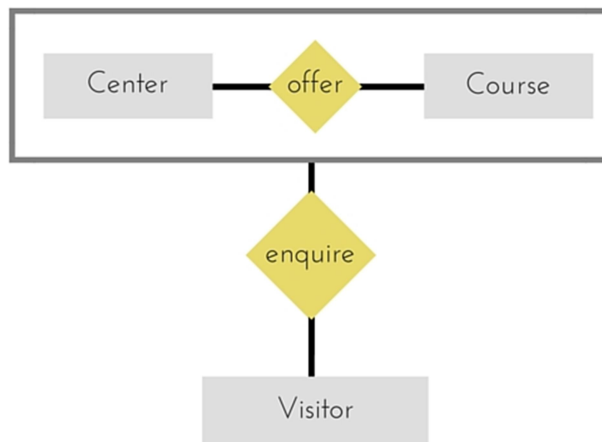


**Relational Algebra**

The relational algebra is a procedural query language. It consists of a set of operations that take one or two relations as input and produces a new relation as output.

**Fundamental Operations**

- SELECT
- PROJECT
- UNION
- SET DIFFERENCE
- CARTESIAN PRODUCT
- RENAME

### Select Operation (σ)

It selects tuples that satisfy the given predicate from a relation.

**Notation:–**   $\sigma_p(r)$

### Project Operation (∏)

It projects column(s) that satisfy a given predicate.

**Notation:–**   $\prod_{A1, A2, An}(r)$

### Union Operation (∪)

It performs binary union between two given relations.

**Notation:–**   r ∪ s

### Set Difference (−)

The result of set difference operation is tuples, which are present in one relation but are not in the second relation.

**Notation:-**   r − s

### Cartesian Product (X)

Combines information of two different relations into one.

**Notation:–**   r X s

### Rename Operation (ρ)

The rename operation allows us to rename the output relation.

**Notation:–**   $\rho_x(E)$