**Terminology**

Relational Database Management System (RDBMS)

* All data is logically structured within relations (tables).
* Each **relation** has a name and is made up of named **attributes** (columns) of data
* Each tuple (row) contains one value per attribute

Normalized Relations – relations that have no repeating groups

Relation – A table with columns and rows

Attribute – a named column of a relation

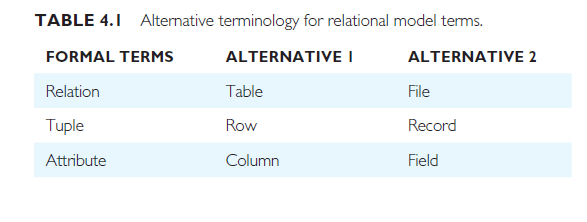
Domain – set of allowable values for one or more attributes

Tuple – row of a relation

Degree – the number of attributes a relation contains

Cardinality – the number of tuples a relation contains

Relational database – a collection of normalized relations with distinct relation names



Relation schema – a named relation defined by a set of attribute and domain name pairs

Relational database schema – a set of relationships, each with a distinct name

Super key – an attribute, or set of attributes, that uniquely identifies a tuple within a relation

Candidate key- a super key such that no proper subset is a super key within the relation

Primary key- the candidate key that is selected to identify tuples uniquely with in the relation

Foreign key- an attribute, or set of attributes, within one relation that matches the candidate key of some (possibly the same) relation

**Integrity Constraints**

Null – represents a value for an attribute that is currently unknown or is not applicable for this tuple

Entity Integrity rules

1. Entity integrity – in a base relation, no attribute of a primary key can be null
2. Referential integrity – If a foreign key exists in a relation, either the foreign key value must match a candidate key value of some tuple in its home relation or the foreign key value must be wholly null

General constraints – Additional rules specified by the users or database administrators of a database that define or constrain some aspect of the enterprise

**Views**

Base relation – a named relation corresponding to an entity in the conceptual schema, whose tuples are physically stored in the database

View - The dynamic result of one or more relational operations operating on the base relations to produce another relation. A view is a virtual relation that does not necessarily exist in the database but can be produced upon request by a particular user, at the time of request.

Purposes of Views –

1. Provides a powerful and flexible security mechanism by hiding parts of the database from certain users
2. Permits users to access data in a way that is customized to their needs
3. Simplify complex operations on the base relation
4. Provides logical data independence

Updating views –

1. Updates are allowed through a view defined using a simple query involving a single base relation and containing either the primary key or a candidate key of the base relation
2. Updates not allowed through views involving multiple base relations
3. Updates are not allowed through views involving aggregation or grouping operations.