Chapter 2
Relational Database

A relational database matches data by using common characteristics found within the data set. The resulting groups of data use the relational model (schema). The software used in relational database is called a relational database management system (RDBMS). The term "relational database" often refers to RDBMS software, not the database itself.

Relational databases are currently the predominant choice in storing data like protein data, nucleic acid, medical records, genome data, phylogenetic, microarray data etc.; a database is a collection of relations (called tables). Other items are frequently considered part of the database, as they help to organize and structure the data, in addition to forcing the database to conform to a set of requirements.

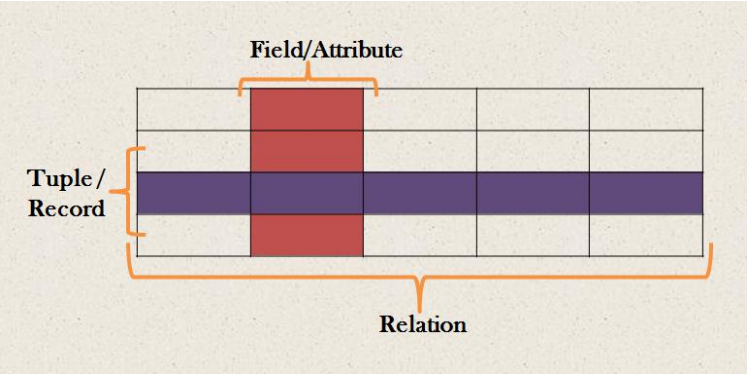

Figure 2.1 Shows the Relational database terminology.

2.1 Attributes and tuples

A relation is defined as a set of tuples that have the same attributes. A tuple usually represents an object and information about that object. Objects are typically physical objects or concepts. A relation is usually described as a table, which is organized into rows and columns. All the data referenced by an attribute are in the same domain and conform to the same constraints.

The relational model specifies that the tuples of a relation have no specific order and that the tuples, in turn, impose no order on the attributes. Applications access data by specifying queries, which use operations such as select to identify tuples, project to identify attributes, and join to combine relations. Relations can be modified using the insert, delete, and update

operators. New tuples can supply explicit values or be derived from a query. Similarly, queries identify tuples for updating or deleting. It is necessary for each tuple of a relation to be uniquely identifiable by some combination (one or more) of its attribute values. This combination is referred to as the primary key.

2.2 Domain

A domain describes the set of possible values for a given attribute, and can be considered a constraint on the value of the attribute. Mathematically, attaching a domain to an attribute means that any value for the attribute must be an element of the specified set.

2.3 Constraints

Constraints make it possible to further restrict the domain of an attribute. SQL implements constraint functionality in the form of check constraints. The restriction is put on the data that is to be stored in relations. These are usually defined using expressions that result in a boolean value, indicating whether or not the data satisfies the constraint. Constraints can apply to single attributes, to a tuple (restricting combinations of attributes) or to an entire relation.

Since every attribute has an associated domain, there are constraints (domain constraints). The two principal rules for the relational model are known as entity integrity and referential integrity.

2.3.1 Primary keys

A primary key uniquely defines a relationship within a database. In order for an attribute to be a good primary key it must not repeat. While natural attributes are sometimes good primary keys, surrogate keys are often used instead.

2.3.2 Surrogate key

A surrogate key is an artificial attribute assigned to an object which uniquely identifies it. The surrogate key has no intrinsic (inherent) meaning, but rather is useful through its ability to uniquely identify a tuple.

2.3.3 Composite key

The key is designed especially in regards to N:M cardinality is the composite key. A composite key is a key made up of two or more attributes within a table that (together) uniquely identify a record.

2.3.4 Foreign key

A foreign key is a reference to a key in another relation, meaning that the referencing table has, as one of its attributes, the values of a key in the referenced table. Foreign keys need not have unique values in the referencing relation. Foreign keys effectively use the values of attributes in the referenced relation to restrict the domain of one or more attributes in the referencing relation.
A foreign key could be described formally as: "For all tuples in the referencing relation projected over the referencing attributes, there must exist a tuple in the referenced relation projected over those same attributes such that the values in each of the referencing attributes match the corresponding values in the referenced attributes."

2.4 Normalization

Normalization was first proposed by Codd as an integral part of the relational model. It encompasses a set of procedures designed to eliminate non simple domains (non-atomic values) and the redundancy (duplication) of data, which in turn prevents data manipulation anomalies and loss of data integrity. The most common forms of normalization applied to databases are called the normal forms. Normalization trades reducing redundancy for increased information entropy. Normalization is criticized because it increases complexity and processing overhead required to join multiple tables representing what are conceptually a single item.

2.5 Relational database management systems

Relational databases, as implemented in relational database management systems, have become a predominant choice for the storage of information in new databases used for financial records, manufacturing and logistical information, personnel data and much more. Relational databases have often replaced legacy hierarchical databases and network databases because they are easier to understand and use, even though they are much less efficient. As computer power has increased, the inefficiencies of relational databases, which made them impractical in earlier times, have been outweighed by their ease of use.