any order and the relation will still be the same relation, and therefore convey the same meaning.

The structure of a relation, together with a specification of the domains and any other restrictions on possible values, is sometimes called its **intension**, which is usually fixed, unless the meaning of a relation is changed to include additional attributes. The tuples are called the **extension** (or **state)** of a relation, which changes over time.

Degree

The degree of a relation is the number of attributes it contains.

The Branch relation in Figure 4.1 has four attributes or degree four. This means that each row of the table is a four-tuple, containing four values. A relation with only one attribute would have degree one and be called a **unary** relation or one-tuple. A relation with two attributes is called **binary**, one with three attributes is called **ternary**, and after that the term ***n*-ary** is usually used. The degree of a relation is a property of the *intension* of the relation.

Cardinality

The cardinality of a relation is the number of tuples it contains.

By contrast, the number of tuples is called the **cardinality** of the relation and this changes as tuples are added or deleted. The cardinality is a property of the *extension* of the relation and is determined from the particular instance of the relation at any given moment. Finally, we define a relational database.

Relational database

A collection of normalized relations with distinct relation names.

A relational database consists of relations that are appropriately structured. We refer to this appropriateness as *normalization*. We defer the discussion of normalization until Chapters 14 and 15.

Alternative terminology

The terminology for the relational model can be quite confusing. We have introduced two sets of terms. In fact, a third set of terms is sometimes used: a relation may be referred to as a **file**, the tuples as **records**, and the attributes as **fields**. This terminology stems from the fact that, physically, the RDBMS may store each relation in a file. Table 4.1 summarizes the different terms for the relational model.

TABLe 4.1 Alternative terminology for relational model terms.

FORMAL TeRMS	ALTeRNATIVe 1	ALTeRNATIVe 2
Relation	Table	File
Tuple	Row	Record
Attribute	Column	Field

4.2.2 Mathematical Relations

To understand the true meaning of the term *relation*, we have to review some concepts from mathematics. Suppose that we have two sets, D_1 and D_2, where $D_1 = \{2, 4\}$ and $D_2 = \{1, 3, 5\}$. The **Cartesian product** of these two sets, written $D_1 \times D_2$, is the set of all ordered pairs such that the first element is a member of D_1 and the second element is a member of D_2. An alternative way of expressing this is to find all combinations of elements with the first from D_1 and the second from D_2. In our case, we have:

$$D_1 \times D_2 = \{(2, 1), (2, 3), (2, 5), (4, 1), (4, 3), (4, 5)\}$$

Any subset of this Cartesian product is a relation. For example, we could produce a relation R such that:

$$R = \{(2, 1), (4, 1)\}$$

We may specify which ordered pairs will be in the relation by giving some condition for their selection. For example, if we observe that R includes all those ordered pairs in which the second element is 1, then we could write R as:

$$R = \{(x, y) \mid x \in D_1, y \in D_2, \text{ and } y = 1\}$$

Using these same sets, we could form another relation S in which the first element is always twice the second. Thus, we could write S as:

$$S = \{(x, y) \mid x \in D_1, y \in D_2, \text{ and } x = 2y\}$$

or, in this instance,

$$S = \{(2, 1)\}$$

as there is only one ordered pair in the Cartesian product that satisfies this condition. We can easily extend the notion of a relation to three sets. Let D_1, D_2, and D_3s be three sets. The Cartesian product $D_1 \times D_2 \times D_3$ of these three sets is the set of all ordered triples such that the first element is from D_1, the second element is from D_2, and the third element is from D_3. Any subset of this Cartesian product is a relation. For example, suppose we have:

$D_1 = \{1, 3\}$ $D_2 = \{2, 4\}$ $D_3 = \{5, 6\}$
$D_1 \times D_2 \times D_3 = \{(1, 2, 5), (1, 2, 6), (1, 4, 5), (1, 4, 6), (3, 2, 5), (3, 2, 6), (3, 4, 5), (3, 4, 6)\}$

Any subset of these ordered triples is a relation. We can extend the three sets and define a general relation on n domains. Let $D_1, D_2, \ldots, D_n$ be n sets. Their Cartesian product is defined as:

$$D_1 \times D_2 \times \ldots \times D_n = \{(d_1, d_2, \ldots, d_n) | d_1 \in D_1, d_2 \in D_2, \ldots, d_n \in D_n\}$$

and is usually written as:

$$\prod_{i=1}^{n} D_1$$

Any set of n-tuples from this Cartesian product is a relation on the n sets. Note that in defining these relations we have to specify the sets, or **domains**, from which we choose values.