

# Unit – 2: Relational Model

## Table of Contents

|                                  |   |
|----------------------------------|---|
| Relational Model Structure ..... | 1 |
| Keys in DBMS .....               | 2 |
| Why Keys are required? .....     | 2 |
| # Types of Keys .....            | 3 |
| # Super Key .....                | 4 |
| Candidate Key .....              | 5 |
| Primary Key .....                | 6 |
| Alternate Key .....              | 6 |
| Foreign Key .....                | 6 |
| Composite Key .....              | 7 |
| Unique Key .....                 | 7 |

## Relational Model Structure

A database consists of a collection of tables (relations), each having a unique name.

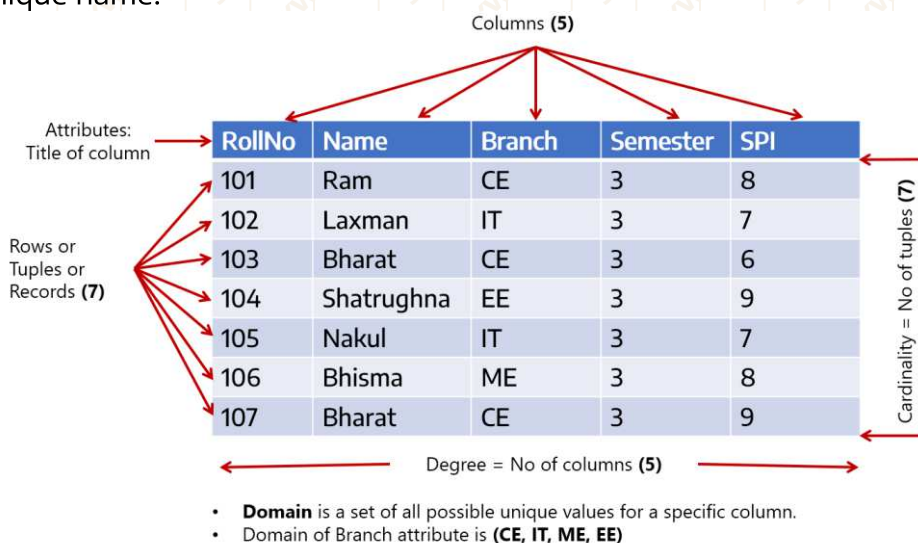


Fig 1 – Relational Data Model

**Table (Relation):** A database object that holds a collection of data for a specific topic. Table consist of rows and columns.

**Column (Attribute):** The vertical component of a table. A column has a name and a particular data type; e.g., varchar, decimal, integer, datetime etc.

**Degree:** Degree refers to the total number of attributes/columns in a table.

**Row (Record / Tuple):** The horizontal component of a table, consisting of a sequence of values, one for each column of the table. It is also known as row / record / tuple.

**Cardinality:** Number of rows in any table is called Cardinality. For example, in above table cardinality is 7

**Domain:** Domain is set of possible unique value in any particular column. For example, in above table, domain of branch attribute is (CE, IT, ME, EE)

## Keys in DBMS

- Key in DBMS is attribute (column) or set of attributes that help to uniquely identify a (tuple) record in a table (relation)
- Keys are also used to establish relationship between different tables of the database.

### Why Keys are required?

- Table is used to store the records.
- In a table there may be thousands of records and there should be a way to identify each record separately and uniquely.
- If we can't identify each record individually then our data is of no use.

Let's take an example of students studying in a college. Below is the table where data of student is stored.

| Enrolment No | Roll No | Name     | Class | Course          | Date of Birth | City      |
|--------------|---------|----------|-------|-----------------|---------------|-----------|
| 1001         | 1       | Ram      | C1    | Computer Eng.   | 02-04-1990    | Jamnagar  |
| 1002         | 2       | Lakshman | C1    | Computer Eng.   | 05-06-1990    | Rajkot    |
| 1003         | 3       | Bharat   | C1    | Computer Eng.   | 05-05-1990    | Ahmedabad |
| 1004         | 1       | Arjun    | M1    | Mechanical Eng. | 07-10-1991    | Rajkot    |
| 1005         | 2       | Bhim     | M1    | Mechanical Eng. | 02-04-1990    | Jamnagar  |
| 1006         | 3       | Ram      | M1    | Mechanical Eng. | 02-04-1990    | Dwarka    |
| 1007         | 4       | Arjun    | M1    | Mechanical Eng. | 01-03-1991    | Dwarka    |
| 1008         | 5       | Nakul    | M1    | Mechanical Eng. | 05-08-1990    | Jamnagar  |
| 1009         | 1       | Ram      | C2    | Computer Eng.   | 01-04-1990    | Rajkot    |
| 1010         | 2       | Balram   | C2    | Computer Eng.   | 12-05-1990    | Navsari   |
| 1011         | 1       | Sahdev   | C1    | Chemical Eng.   | 15-06-1993    | Bharuch   |
| 1012         | 2       | Lakshman | C1    | Chemical Eng.   | 19-12-1990    | Dang      |

*Table 1.1 - Student Detail*

As there are so many records in this table, we need to identify each record (student) uniquely, so whenever we want to get information of particular student, we can get it easily.

What attribute of student will identify them uniquely? We can say we can identify any student using name, roll no, class, course, etc.

But such attributes can have duplicate values, for example two students can have same name. So, we can't uniquely identify student only by his/her name. So, we can use enrolment no to identify each student, as each student has unique enrolment no. So here enrolment no column can be a key in this table.

Or we can also combine multiple attributes to identify the record uniquely. We can say combination of (roll no, name, class, course) will be unique and can be used to identify each student. So, this combination (roll no, name, class, course) also can be a key in this table.

## # Types of Keys

There is main 7 types of keys in DBMS

- Super Key
- Primary Key
- Candidate Key
- Alternate Key

- Foreign Key
- Composite Key
- Unique Key

## # Super Key

- Super key is attribute or group of attributes which can be used to identify each tuple (records) of the table uniquely.
- Table can have more than one super key which can be used to identify each tuple of table uniquely

| Super Key    |         | Super Key |       |                 | Not a Super key |           |
|--------------|---------|-----------|-------|-----------------|-----------------|-----------|
| Enrolment No | Roll No | Name      | Class | Course          | Date of Birth   | City      |
| 1001         | 1       | Ram       | C1    | Computer Eng.   | 02-04-1990      | Jamnagar  |
| 1002         | 2       | Lakshman  | C1    | Computer Eng.   | 05-06-1990      | Rajkot    |
| 1003         | 3       | Bharat    | C1    | Computer Eng.   | 05-05-1990      | Ahmedabad |
| 1004         | 1       | Arjun     | M1    | Mechanical Eng. | 07-10-1991      | Rajkot    |
| 1005         | 2       | Bhim      | M1    | Mechanical Eng. | 02-04-1990      | Jamnagar  |
| 1006         | 3       | Ram       | M1    | Mechanical Eng. | 02-04-1990      | Dwarka    |
| 1007         | 4       | Arjun     | M1    | Mechanical Eng. | 01-03-1991      | Dwarka    |
| 1008         | 5       | Nakul     | M1    | Mechanical Eng. | 05-08-1990      | Jamnagar  |
| 1009         | 1       | Ram       | C2    | Computer Eng.   | 01-04-1990      | Rajkot    |
| 1010         | 2       | Balram    | C2    | Computer Eng.   | 12-05-1990      | Navsari   |
| 1011         | 1       | Sahdev    | C1    | Chemical Eng.   | 15-06-1993      | Bharuch   |
| 1012         | 2       | Lakshman  | C1    | Chemical Eng.   | 19-12-1990      | Dang      |

*Table 1.2 - Student Detail*

- For example, in table 1.2
- Enrolment no attribute can be a super key
- (roll no, name, class, course) can be a super key
- (enrolment no, name) can be a super key
- (enrolment no, name, class) can be super key
- But (name, date of birth, city) cannot be a key because it can't identify the records uniquely.
- As we can see, there may be more than one super key in any table.

### Maximum how many super keys can be possible in a table?

For table with n column, maximum no of super key can be

$$\text{Max number of Super Keys} = 2^n - 1$$

For example, table with 3 columns:

$$\text{Max number of Super Keys} = 2^3 - 1$$

$$\text{Max number of Super Keys} = 7$$

## Candidate Key

- Candidate key is attribute or group of attributes which is used to identify each record uniquely.
- Candidate key is minimal subset of super keys.
- In other words, any super key is candidate key if any of its subset is not a super key.
- In short Candidate key is a minimal (i.e., irreducible) super key
- So, we can say that every candidate key is super key, but not every super key is not candidate key.

| A | B | C |
|---|---|---|
| 1 | 1 | 1 |
| 2 | 1 | 2 |
| 3 | 2 | 1 |
| 4 | 2 | 2 |

*Table 1.3 - Sample Table*

As from data we can say that Table 1.3 has below super keys

{A}  
{A, B}  
{A, C}  
{A, B, C}  
{B, C}

But not all above super key can be considered as candidate key as we can see in below table. Only {A} and {B, C} is candidate key.

| Super Key | Is Candidate Key? | Reason  |
|-----------|-------------------|---|
| {A}       | Yes               | Not have any subset which is super key          |
| {A, B}    | No                | Subset {A} is super key                         |
| {A, C}    | No                | Subset {A} is super key                         |
| {A, D}    | No                | Subset {A} is super key                         |
| {A, B, C} | No                | Subset {A}, {A, B}, {A, C}, {A, D} is super key |
| {B, C}    | Yes               | Not have any subset which is super key          |

*Table 1.4 - Candidate Key*

## Primary Key

- Primary key is attribute or group of attributes which is used to identify each record uniquely.
- There can be only one primary key can be in any table.
- Primary key is chosen from the list candidate keys.
- When primary key constraint is assigned to any column or group of columns in table below rules are applied.
  - Every record must have value in primary key.
  - Primary key column(s) cannot be null.
  - Value in Primary key column(s) cannot be duplicate, means two or more record cannot have same value in primary key column(s)

For example, in table 1.4 there are two candidate key {A} and {B, C}. From these 2 candidate keys, we can choose any one candidate key as primary key. e.g. {A}

When choosing primary key, whenever possible we should choose that candidate key which is made up of only one column instead of group of columns.

## Alternate Key

- An alternate key is a candidate key that is not chosen by database designer to identify tuples uniquely in a relation.
- In other words, all candidate keys except primary key is called alternate key.

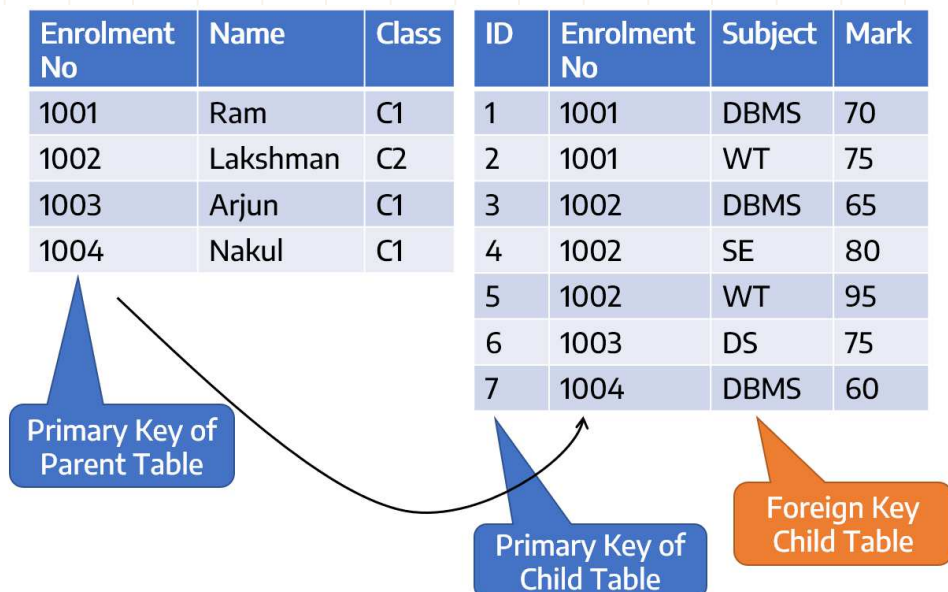
For example, in table 1.4 if we choose {A} as primary key then {B, C} will be the candidate key, and if we choose {B, C} as primary key then {A} is called alternate key.

## Foreign Key

- A foreign key is an attribute or collection of attributes in one table that refers to the primary key in another table.
- A table containing the foreign key is called the child table, and the table containing the primary key is called the parent table.
- It is also known as Referential Integrity Constraint



| Enrolment No | Name     | Class | ID | Enrolment No | Subject | Mark |
|--------------|----------|-------|----|--------------|---------|------|
| 1001         | Ram      | C1    | 1  | 1001         | DBMS    | 70   |
| 1002         | Lakshman | C2    | 2  | 1001         | WT      | 75   |
| 1003         | Arjun    | C1    | 3  | 1002         | DBMS    | 65   |
| 1004         | Nakul    | C1    | 4  | 1002         | SE      | 80   |
|              |          |       | 5  | 1002         | WT      | 95   |
|              |          |       | 6  | 1003         | DS      | 75   |
|              |          |       | 7  | 1004         | DBMS    | 60   |



*Table 1.5 - Foreign Key*

## Composite Key

- When two or more attributes of a table are used to identify each record uniquely.
- If key consists of more than one attribute, that key is known as composite key.

## Unique Key

- Unique key is attribute or set of attributes in a table such that
  - Table does not have duplicate value in that unique column or set of columns.
  - Table does not have null value in that unique column or set of columns.
- Unique key constraint is generally applied to all alternate keys.