

Database Keys

In **database management systems**, a key is a field that you use to sort data. It can also be called a key field, sort key, index, or key word. For example, if you sort records by age, then the age field is a key. Most database management systems allow you to have more than one key so that you can sort records in different ways.

Keys are, as their name suggests, a key part of a relational database and a vital part of the structure of a table. They ensure each record within a table can be uniquely identified by one or a combination of fields within the table. They help enforce integrity and help identify the relationship between tables.

Always make certain that you define the appropriate keys for each table. Doing so will help you guarantee that the table structures are sound, that redundant data within each table is minimal, and that the relationships between tables are solid.

Why Keys Are Important:

Keys are crucial to a table structure for the following reasons:

- They ensure that each record in a table is precisely identified.
- They help establish and enforce various types of integrity. Keys are a major component of table-level integrity and relationship-level integrity.
- They serve to establish table relationships. You'll use keys to establish a relationship between a pair of tables.

Database Keys:

Keys are very important part of Relational database. They are used to establish and identify relation between tables. They also ensure that each record within a table can be uniquely identified by combination of one or more fields within a table.

Super Key: Super Key is defined as a set of attributes with in a table that uniquely identifies each record within a table. Super Key is a superset of Candidate key.

Candidate Key: Candidate keys are defined as the set of fields from which primary key can be selected. It is an attribute or set of attribute that can act as a primary key for a table to uniquely identify each record in that table.

Primary Key: Primary Key is a candidate key that is most appropriate to become main key of the table. It is a key that uniquely identify each record in a table.

Composite Key: Key that consists of two or more attributes that uniquely identify an entity occurrence is called Composite key.

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

Non-key Attribute: Non-key attributes are attributes other than candidate key attributes in a table.

Non-prime Attribute: Non-prime Attributes are attributes other than Primary attribute.

Definition – What does Primary Key means?

A primary key is a special relational database table column (or combination of columns) designated to uniquely identify all table records.

A primary key's main features are:

1. It must contain a unique value for each row of data.
2. It cannot contain null values.

Primary key must hold a unique value for each record.

A key field that identifies records in a different table is called a **foreign key**.

Two types of key are used to access tables in a relational database:

A **Primary Key** uniquely identifies a record in a table

A **Foreign Key** accesses data in some other related table via its Primary Key.

Common Key Terminology

Some common terminology pertaining to keys

Key: A key is one or more data attributes that uniquely identify an entity. In a physical database a key would be formed of one or more table columns whose value(s) uniquely identifies a row within a relational table.

Composite key: A key that is composed of two or more attributes.

Natural key: A key that is formed of attributes that already exist in the real world.

Surrogate key: A key with no business meaning.

Candidate key: An entity type in a logical data model will have zero or more candidate keys, also referred to simply as unique identifiers.

Primary key: The preferred key for an entity type.

Alternate key: Also known as a secondary key is another unique identifier of a row within a table.

Foreign key: One or more attributes in an entity type that represents a key, either primary or secondary, in another entity type.

A **primary key** is a unique identifier for a database record. When a table is created, one of the fields is typically assigned as the primary key. While the primary key is often a number, it may also be a text field or other data type.

What are Foreign Keys?

A foreign key establishes a relationship, or constraint, between two tables.

Here is more description about Database Keys:

I) Super Key – An attribute or a combination of attribute that is used to identify the records uniquely is known as Super Key. A table can have many Super Keys.

E.g. of Super Key

1. ID
2. ID, Name
3. ID, Address
4. ID, Department_ID
5. ID, Salary
6. Name, Address
7. Name, Address, Department_ID

(II) Candidate Key – It can be defined as minimal Super Key or irreducible Super Key. In other words an attribute or a combination of attribute that identifies the record uniquely but none of its proper subsets can identify the records uniquely.

E.g. of Candidate Key

1. Code
2. Name, Address

(III) Primary Key – A Candidate Key that is used by the database designer for unique identification of each row in a table is known as Primary Key. A Primary Key can consist of one or more attributes of a table.

(IV) Foreign Key – A foreign key is an attribute or combination of attribute in one base table that points to the candidate key (generally it is the primary key) of another table. The purpose of the foreign key is to ensure referential integrity of the data i.e. only values that are supposed to appear in the database are permitted.

(V) Composite Key – If we use multiple attributes to create a Primary Key then that Primary Key is called Composite Key (also called a Compound Key or Concatenated Key).

E.g. of Composite Key, if we have used “Name, Address” as a Primary Key then it will be our Composite Key.

(VI) Alternate Key – Alternate Key can be any of the Candidate Keys except for the Primary Key.

E.g. of Alternate Key is “Name, Address” as it is the only other Candidate Key which is not a Primary Key.

(VII) Secondary Key – The attributes that are not even the Super Key but can be still used for identification of records (not unique) are known as Secondary Key.

E.g. of Secondary Key can be Name, Address, Salary, Department_ID etc. as they can identify the records but they might not be unique.

Types of Keys in Database:

Super Key: A combination of one or more columns in a table which can be used to identify a record in a table uniquely, a table can have any number of super keys.

Candidate Key: A Column (or) Combination of columns which can help uniquely identify a record in a table without the need of any external data is called a Candidate Key. Depending on the need and situation a Table may have one or more candidate keys and one of them can be used as a Primary Key of the table.

A candidate key is a sub set of a Super Keys.

Compound Key: A Combination of more than one column identifying records of a table uniquely, all the columns that take part in the combination process is Simple Key's.

Primary Key: A Column in a table (which is a Simple Key) which is a Candidate Key (Uniquely identify a records in a table) and has the Constraint NOT NULL attached to it is known as a Primary Key.

Composite Primary Key (Composite Key):

When we have a Primary Key of a table defined using more than one columns then it is known as a Composite Key, each columns data can be duplicated, but combined values cannot be. The columns which are participating in a composite primary key are not simple keys.

Alternate Key: We cannot define the Alternate Key Separately from a Candidate Key, for a table, if there are two Candidate Key's and one is chosen as a Primary Key the other Candidate Key is known as the Alternate Key of that table.

Unique Key: A column (or) combination of columns which can be used to uniquely identify a record in a table, it can have one NULL Value.

Primary Key can be considered a special case of unique key with a Not Null Constraint.

Foreign Key: A column of one table points to the Primary Key column of another table to implement referential data integrity.

Super Key

A Super key is any combination of fields within a table that uniquely identifies each record within that table.

Candidate Key

A candidate is a subset of a super key. A candidate key is a single field or the least combination of fields that uniquely identifies each record in the table. Every table must have at least one candidate key but at the same time can have several.

Primary Key

A primary key is a candidate key that is most appropriate to be the main reference key for the table. As its name suggests, it is the primary key of reference for the table and is used throughout the database to help establish relationships with other tables. As with any candidate key the primary key must contain unique values, must never be null and uniquely identify each record in the table.

Foreign Key

A foreign key is generally a primary key from one table that appears as a field in another where the first table has a relationship to the second.

Secondary Key or Alternative Key

A table may have one or more choices for the primary key. Collectively these are known as candidate keys. One is selected as the primary key. Those not selected are known as secondary keys or alternative keys.

Simple Key

A simple key consists of a single field to uniquely identify a record. In addition the field in itself cannot be broken down into other fields,

Compound Key

A compound key consists of more than one field to uniquely identify a record. A compound key is distinguished from a composite key because each field, which makes up the primary key, is also a simple key in its own right.

Composite Key

A composite key consists of more than one field to uniquely identify a record. This differs from a compound key in that one or more of the attributes, which make up the key, are not simple keys in their own right.

Keys in DBMS

The key is defined as the column or attribute of the database table. For example if a table has id, name and address as the column names then each one is known as the key for that table. We can also say that the table has 3 keys as id, name and address. The keys are also used to identify each record in the database table. The following are the various types of keys available in the DBMS system.

A **simple** key contains a single attribute.

A **composite key** is a key that contains more than one attribute.

A **candidate key** is an attribute (or set of attributes) that uniquely identifies a row. A candidate key must possess the following properties:

Unique identification - For every row the value of the key must uniquely identify that row.

Non redundancy - No attribute in the key can be discarded without destroying the property of unique identification.

A **primary key** is the candidate key which is selected as the principal unique identifier. Every relation must contain a primary key. The primary key is usually the key selected to identify a row when the database is physically implemented.

A **super key** is any set of attributes that uniquely identifies a row. A super key differs from a candidate key in that it does not require the non redundancy property.

A **foreign key** is an attribute (or set of attributes) that appears (usually) as a non key attribute in one relation and as a primary key attribute in another relation. It is possible for a foreign key to also be the whole or part of a primary key.

A **semantic** or **natural** key is a key for which the possible values have an obvious meaning to the user or the data.

A **technical** or **surrogate** or **artificial** key is a key for which the possible values have no obvious meaning to the user or the data.

Defining a primary key

A primary key consists of a column or columns whose value uniquely identifies a row. There can only be one primary key per table. Defining a primary key for a table, you create a constraint on data in the table, which ensures that any row in the table can be uniquely identified by its key columns.

When you create a table, you can define the primary key after you define the columns.

Defining a foreign key

A foreign key consists of one or more columns in a table whose value in one row uniquely identifies another row in the same or another table.

When you create a new table, you can define the foreign keys after you have defined the columns for the new table as well as the target. The target of the foreign key is another table and its primary key.

Different Types of SQL Keys

A key is a single or combination of multiple fields in a table. Its is used to fetch or retrieve records/data-rows from data table according to the condition/requirement. Keys are also used to create relationship among different database tables or views.

Types of SQL Keys

We have following types of keys in SQL which are used to fetch records from tables and to make relationship among tables or views.

1. Super Key

Super key is a set of one or more than one keys that can be used to identify a record uniquely in a table. Example: Primary key, Unique key, Alternate key are subset of Super Keys.

2. Candidate Key

A Candidate Key is a set of one or more fields/columns that can identify a record uniquely in a table. There can be multiple Candidate Keys in one table. Each Candidate Key can work as Primary Key.

3. Primary Key

Primary key is a set of one or more fields/columns of a table that uniquely identify a record in database table. It can not accept null, duplicate values. Only one Candidate Key can be Primary Key.

4. Alternate key

An Alternate key is a key that can be work as a primary key. Basically it is a candidate key that currently is not primary key.

5. Composite/Compound Key

Composite Key is a combination of more than one fields/columns of a table. It can be a Candidate key, Primary key.

6. Unique Key

Unique key is a set of one or more fields/columns of a table that uniquely identify a record in database table. It is like Primary key but it can accept only one null value and it can not have duplicate values.

7. Foreign Key

Foreign Key is a field in database table that is Primary key in another table. It can accept multiple null, duplicate values.

Different Types of Database Keys

Key: One or more columns in a database table that is used to sort and/or identify rows in a table. E.g. if you were sorting people by the field salary then the salary field is the key.

Here are different types of Database Keys:

Primary key: A primary key is a one or more fields that uniquely identifies a row in a table. The primary key cannot be null (blank). The primary key is indexed.

Foreign key: A foreign key is a relationship between columns in two database tables (one of which is indexed) designed to insure consistency of data

Composite key: A primary key composed of one or more columns e.g. a staff table STAFF contains the fields FNAME and LNAME for first and last names respectively. The primary key can be formed using the fields (though not very advisable as I'll describe later).

Natural key: A composite primary key which is composed of attributes (fields) which already exist in the real world e.g. First Name, Last Name, Social Security Number. .

Surrogate key: A primary key which is internally generated (typically auto-incremental integer value) that does not exist in the real world.

Candidate key: A candidate key is a column or group of columns that can uniquely identify a row in the table without referring to any other source. In a table which has multiple candidate keys one is selected to be the primary key.

Compound key: A composite key consisting of two or more fields that uniquely describe a row in a table. The difference between compound and candidate is that all of the fields in the compound key are foreign keys; in the candidate key one or more of the fields may be foreign keys (but it is not mandatory).