

Module I

Chapter 1.2

(Keys in detail)

What are Keys in DBMS?

KEYS in DBMS is an attribute or set of attributes which helps you to identify a row(tuple) in a relation(table). They allow you to find the relation between two tables. Keys help you uniquely identify a row in a table by a combination of one or more columns in that table. Key is also helpful for finding unique record or row from the table. Database key is also helpful for finding unique record or row from the table.

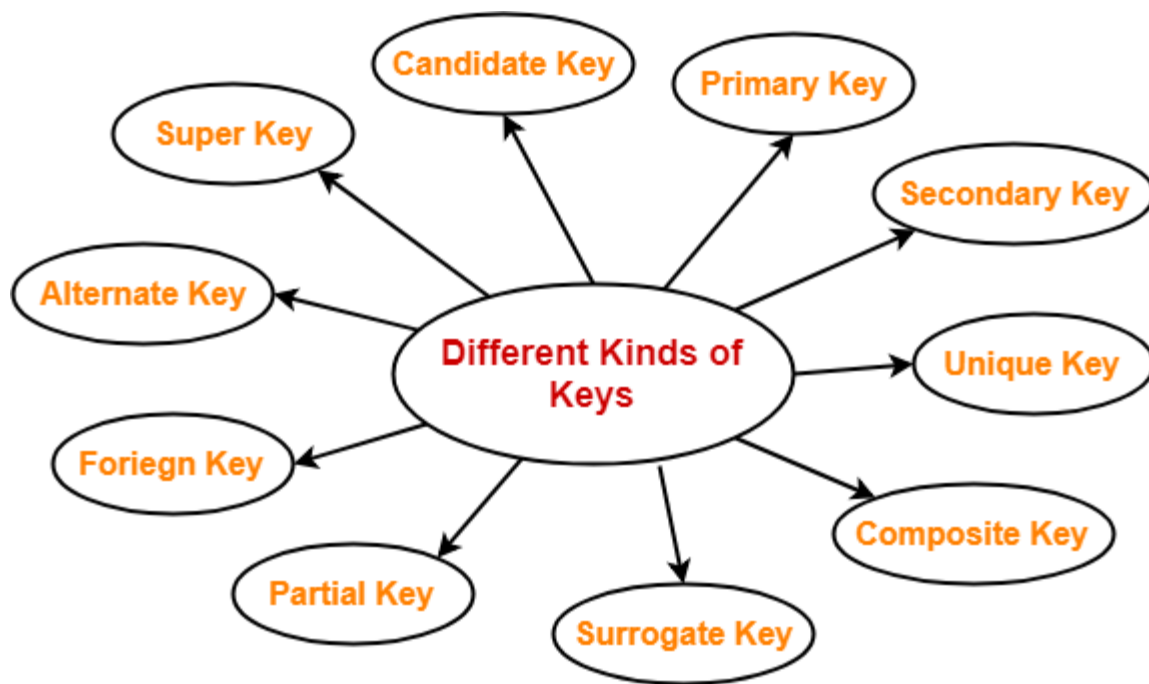
Why we need a Key?

Here are some reasons for using sql key in the DBMS system.

- Keys help you to identify any row of data in a table. In a real-world application, a table could contain thousands of records. Moreover, the records could be duplicated. Keys ensure that you can uniquely identify a table record despite these challenges.
- Allows you to establish a relationship between and identify the relation between tables
- Help you to enforce identity and integrity in the relationship.

Different Types Of Keys in DBMS-

There are following 10 important keys in DBMS-



1. Super key
2. Candidate key
3. Primary key
4. Alternate key
5. Foreign key
6. Partial key
7. Composite key
8. Unique key
9. Surrogate key
10. Secondary key

Types of Keys in Database Management System intro

There are mainly seven different types of Keys in DBMS and each key has its different functionality:

- **Super Key** - A super key is a group of single or multiple keys which identifies rows in a table.
- **Primary Key** - is a column or group of columns in a table that uniquely identify every row in that table.
- **Candidate Key** - is a set of attributes that uniquely identify tuples in a table. Candidate Key is a super key with no repeated attributes.
- **Alternate Key** - is a column or group of columns in a table that uniquely identify every row in that table.

- **Foreign Key** - is a column that creates a relationship between two tables. The purpose of Foreign keys is to maintain data integrity and allow navigation between two different instances of an entity.
- **Compound Key** - has two or more attributes that allow you to uniquely recognize a specific record. It is possible that each column may not be unique by itself within the database.
- **Composite Key** - An artificial key which aims to uniquely identify each record is called a surrogate key. These kind of key are unique because they are created when you don't have any natural primary key.
- **Surrogate Key** - An artificial key which aims to uniquely identify each record is called a surrogate key. These kind of key are unique because they are created when you don't have any natural primary key.

NOTE-

Before proceeding further, Kindly note-

- The terms 'relation' and 'table' are used interchangeably.
- The terms 'tuple' and 'record' are used interchangeably.

So, don't get confused!

1. Super Key-

- A super key is a set of attributes that can identify each tuple uniquely in the given relation.
- A super key is not restricted to have any specific number of attributes.
- Thus, a super key may consist of any number of attributes.
- A superkey is a group of single or multiple keys which identifies rows in a table. A Super key may have additional attributes that are not needed for unique identification.

Example 1-

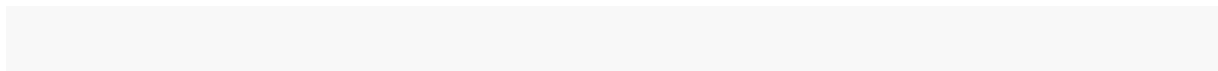
Consider the following Student schema-

Student (roll , name , sex , age , address , class , section)

Given below are the examples of super keys since each set can uniquely identify each student in the Student table-

- (roll , name , sex , age , address , class , section)
- (class , section , roll)
- (class , section , roll , sex)
- (name , address)

Example 2-



In the above-given example, EmpSSN and EmpNum name are superkeys.

NOTE-

All the attributes in a super key are definitely sufficient to identify each tuple uniquely in the given relation but all of them may not be necessary.

2. Candidate Key-

A minimal super key is called as a candidate key.

OR

A set of minimal attribute(s) that can identify each tuple uniquely in the given relation is called as a candidate key.

CANDIDATE KEY is a set of attributes that uniquely identify tuples in a table. Candidate Key is a super key with no repeated attributes. The Primary key should be selected from the candidate keys. Every table must have at least a single candidate key. A table can have multiple candidate keys but only a single primary key.

Properties of Candidate key:

- It must contain unique values
- Candidate key may have multiple attributes
- Must not contain null values
- It should contain minimum fields to ensure uniqueness
- Uniquely identify each record in a table

Example 1-

Consider the following Student schema-

Student (roll , name , sex , age , address , class , section)

Given below are the examples of candidate keys since each set consists of minimal attributes required to identify each student uniquely in the Student table-

- (class , section , roll)
- (name , address)

Example 2-

In the given table Stud ID, Roll No, and email are candidate keys which help us to uniquely identify the student record in the table.

--

NOTES-

- All the attributes in a candidate key are sufficient as well as necessary to identify each tuple uniquely.
- Removing any attribute from the candidate key fails in identifying each tuple uniquely.

- The value of candidate key must always be unique.
- The value of candidate key can never be NULL.
- It is possible to have multiple candidate keys in a relation.
- Those attributes which appears in some candidate key are called as **prime attributes**.

3. Primary Key-

A primary key is a candidate key that the database designer selects while designing the database.

OR

Candidate key that the database designer implements is called as a primary key.

PRIMARY KEY is a column or group of columns in a table that uniquely identify every row in that table. The Primary Key can't be a duplicate meaning the same value can't appear more than once in the table. A table cannot have more than one primary key.

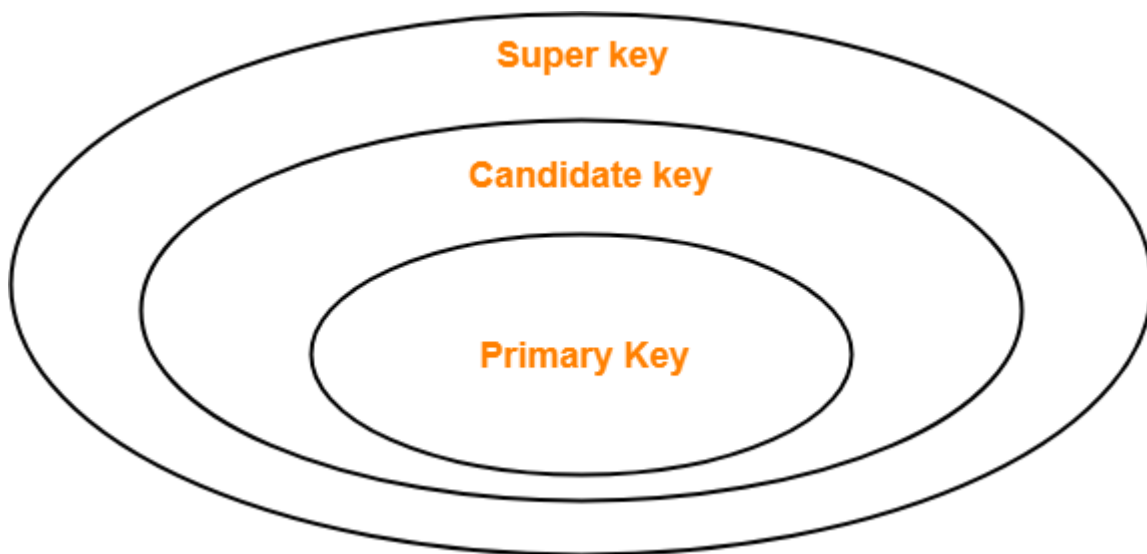
Rules for defining Primary key:

- Two rows can't have the same primary key value
- It must for every row to have a primary key value.
- The primary key field cannot be null.
- The value in a primary key column can never be modified or updated if any foreign key refers to that primary key.

NOTES-

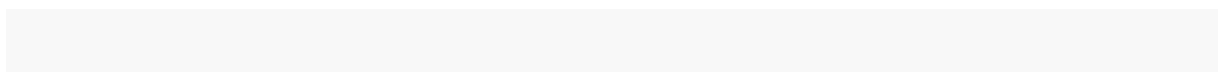
- The value of primary key can never be NULL.
- The value of primary key must always be unique.
- The values of primary key can never be changed i.e. no updation is possible.
- The value of primary key must be assigned when inserting a record.
- A relation is allowed to have only one primary key.

Remember-



Example:

In the following example, `<code>StudID</code>` is a Primary Key.



4. Alternate Key-

Candidate keys that are left unimplemented or unused after implementing the primary key are called as alternate keys.

OR

Unimplemented candidate keys are called as alternate keys.

ALTERNATE KEYS is a column or group of columns in a table that uniquely identify every row in that table. A table can have multiple choices for a primary key but only one can be set as the primary key. All the keys which are not primary key are called an Alternate Key.

Example:

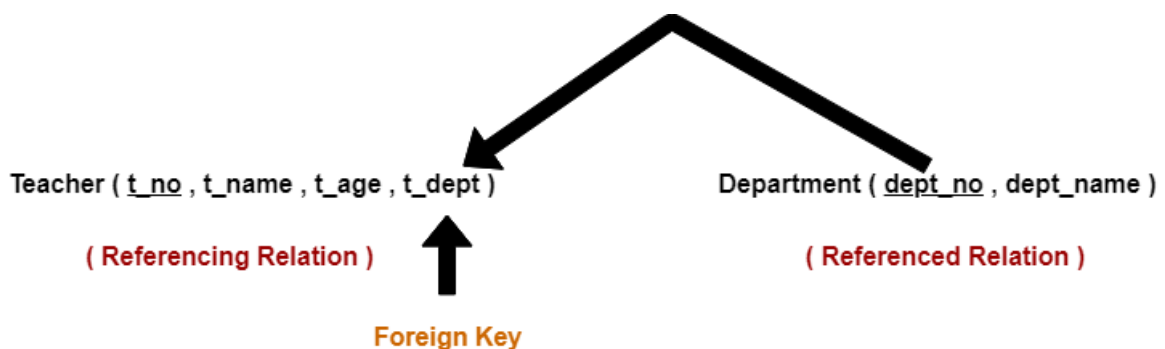
In this table, StudID, Roll No, Email are qualified to become a primary key. But since StudID is the primary key, Roll No, Email becomes the alternative key.

5. Foreign Key-

- An attribute 'X' is called as a foreign key to some other attribute 'Y' when its values are dependent on the values of attribute 'Y'.
- The attribute 'X' can assume only those values which are assumed by the attribute 'Y'.
- Here, the relation in which attribute 'Y' is present is called as the **referenced relation**.
- The relation in which attribute 'X' is present is called as the **referencing relation**.
- The attribute 'Y' might be present in the same table or in some other table.

Example-

Consider the following two schemas-



Here, t_dept can take only those values which are present in dept_no in Department table since only those departments actually exist.

NOTES-

- Foreign key references the primary key of the table.
- Foreign key can take only those values which are present in the primary key of the referenced relation.
- Foreign key may have a name other than that of a primary key.
- Foreign key can take the NULL value.
- There is no restriction on a foreign key to be unique.
- In fact, foreign key is not unique most of the time.
- Referenced relation may also be called as the master table or primary table.
- Referencing relation may also be called as the foreign table.

FOREIGN KEY is a column that creates a relationship between two tables. The purpose of Foreign keys is to maintain data integrity and allow navigation between two different instances of an entity. It acts as a cross-reference between two tables as it references the primary key of another table.

This concept is also known as Referential Integrity.

6. Partial Key-

- Partial key is a key using which all the records of the table can not be identified uniquely.
- However, a bunch of related tuples can be selected from the table using the partial key.

Example-

Consider the following schema-

Department (Emp_no , Dependent_name , Relation)

Emp_no Dependent_name Relation

E1 Suman Mother

E1 Ajay Father

E2 Vijay Father

E2 Ankush Son

Here, using partial key Emp_no, we can not identify a tuple uniquely but we can select a bunch of tuples from the table.

7. Composite Key-

A primary key comprising of multiple attributes and not just a single attribute is called as a composite key.

COMPOSITE KEY is a combination of two or more columns that uniquely identify rows in a table. The combination of columns guarantees uniqueness, though individually uniqueness is not guaranteed. Hence, they are combined to uniquely identify records in a table.

The difference between compound and the composite key is that any part of the compound key can be a foreign key, but the composite key may or maybe not a part of the foreign key.

8. Unique Key-

Unique key is a key with the following properties-

- It is unique for all the records of the table.
- Once assigned, its value can not be changed i.e. it is non-updatable.
- It may have a NULL value.

Example-

The best example of unique key is Adhaar Card Numbers.

- The Adhaar Card Number is unique for all the citizens (tuples) of India (table).
- If it gets lost and another duplicate copy is issued, then the duplicate copy always has the same number as before.
- Thus, it is non-updatable.
- Few citizens may not have got their Adhaar cards, so for them its value is NULL.

9. Surrogate Key-

Surrogate key is a key with the following properties-

- It is unique for all the records of the table.
- It is updatable.
- It can not be NULL i.e. it must have some value.

Example-

Mobile Number of students in a class where every student owns a mobile phone.

SURROGATE KEYS is An artificial key which aims to uniquely identify each record is called a surrogate key. This kind of partial key in dbms is unique because it is created when you don't have any natural primary key. They do not lend any meaning to the data in the table. Surrogate key is usually an integer. A surrogate key is a value generated right before the record is inserted into a table.

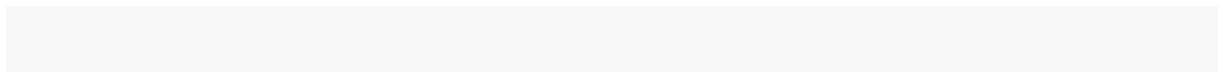
Above, given example, shown shift timings of the different employee. In this example, a surrogate key is needed to uniquely identify each employee.

Surrogate keys in sql are allowed when

- No property has the parameter of the primary key.
- In the table when the primary key is too big or complicated.

10. Secondary Key-

Secondary key is required for the indexing purpose for better and faster searching.



Other References

- [DBMS Keys: Candidate, Super, Primary, Foreign \(Example\) \(guru99.com\)](#)
- [DBMS Keys: Primary, Foreign, Candidate and Super Key - javatpoint](#)
- [Types of Keys in DBMS | Definitions | Examples | Gate Vidyalay](#)

Suggested Book References

1. J. Date, "An Introduction to Database Systems", Addison Wesley.
2. Navathe, "Fundamentals of Database System", The Benjamin / Cummings Publishing Co.