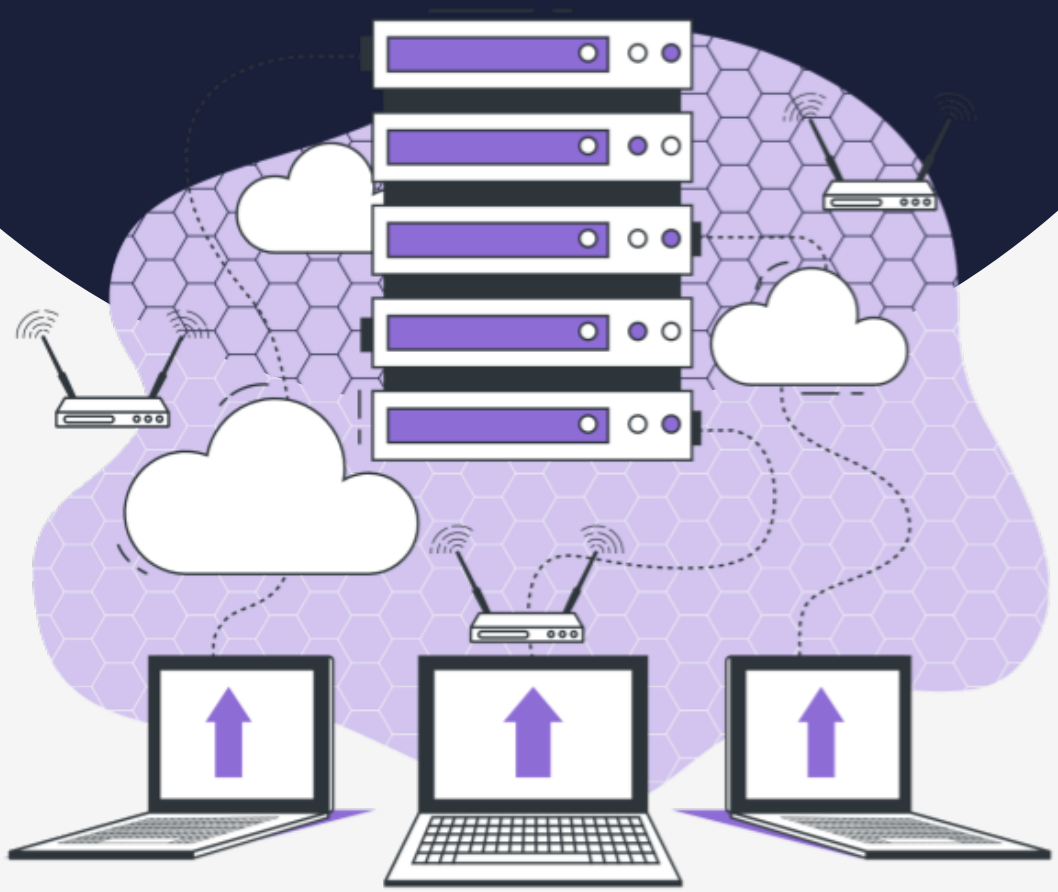
Lesson:

Keys in RDBMS



PW SKILLS

Lecture Checklist

1. Introduction to Keys in RDBMS.
2. Super Key.
3. Candidate Key.
4. Primary Key.
5. Foreign Key.
6. Alternate Key.
7. Composite Key.
8. Artificial Key.

In a relational database management system (RDBMS), keys play a significant role in organizing and managing data. In this lecture let's explore what are keys in RDBMS and look at some of the important keys in RDBMS which will help us in designing our database.

Introduction to Keys in RDBMS

The term Keys in RDBMS does not refer to any type of secret which will open a lock. But, they are very important in helping us organize and manage databases.

A key in RDBMS is a column or a group of columns that uniquely identifies a row in a table. Keys are mainly used to identify records and establish relationships between different tables.

Let's now look at some of the keys in RDBMS along with their practical example.

Super Key.

In relational database management systems (RDBMS), a super key is a set of one or more columns in a table that can uniquely identify each row in the table.

A super key is a concept used in database design to identify all the possible combinations of columns that can be used to uniquely identify rows in a table. It helps in understanding the different ways in which rows in a table can be uniquely identified.

Employee Table:

Column Name
Emp_ID
Emp_Name
Pan_Number
License_Number
Aadhar_number

Considering the above table let's try to identify some super keys.

- Emp_ID.
- Pan_Number.
- License_Number.
- Aadhar_Number.

These are some of the super keys. Along with these, we can also have some combinations.

- Emp_ID and Emp_Name: Here even though Emp_Name is not a unique identifier as multiple employees can have the same name, on combining it with Emp_ID it becomes a unique identifier. Even these combinations are a part of super keys.

A superset is simply a representation of all the possible keys which can be a single column or a group of columns that can identify a row uniquely in a table.

Candidate Key.
Candidate keys are similar to super keys, the only difference is that they can have a single column or group of columns which can necessarily identify each row uniquely.

In simple words, a candidate key is a super key that can individually identify the row uniquely in the table.

Let's understand the candidate key with an example.

Employee Table:

Column Name
Emp_ID
Emp_Name
Pan_Number
License_Number
Aadhar_number

We know that the combination of Emp_ID and Emp_Name is a super key. But, this cannot be a candidate key. This is because Emp_Name cannot be individually used to identify the rows uniquely in the table. It is treated as unnecessary in uniquely identifying the rows in a table and candidate keys cannot include unnecessary columns.

Let's look at some of the possible candidate keys:

- Emp_ID.
- Pan_Number.
- License_Number.
- Aadhar_Number.

These are some of the candidate keys. Along with these, we can also have some combinations.

- Emp_ID and Pan_Number: Since both are necessary columns in identifying each row uniquely in a table we can consider this combination as a candidate key.

Primary Key.
A primary key is a chosen candidate key that is used as a unique identifier for each row in a table.

In the Indian political system, during the elections for the post of Prime Minister, there can be multiple candidates from different political parties trying for the position. However, at the end of the election process, only one person is elected as the Prime Minister of the country. This elected person becomes the unique identifier for the position of Prime Minister, and no other person can hold that position simultaneously.

Similarly, in a relational database table, there can be multiple candidate keys, which are like the candidates from different political parties. These candidate keys are sets of columns that can uniquely identify each row in the table. However, to ensure that each row has a unique identifier, one of the candidate keys is chosen as the primary key. Just like the elected Prime Minister becomes the unique identifier for the highest political position in the country, the primary key becomes the unique identifier for each row in the table.

Employee Table:

Column Name
Emp_ID
Emp_Name
Pan_Number
License_Number
Aadhar_number

In the above table, Emp_ID is chosen as the primary key and will be represented as PK.

Column Name
Emp_ID (PK)
Emp_Name
Pan_Number
License_Number
Aadhar_number

Foreign Key.
A foreign key is a column or a set of columns in a table that refers to the primary key of another table, establishing a relationship between the two tables in a relational database.

Employee Table:

Column Name
Emp_ID
Emp_Name
Pan_Number
License_Number
Aadhar_Number
Department_ID

Department Table:

Column Name
Department_ID
Department_Name
Location
Manager_ID

In the above images we have two tables: The employee table and the Department table. We know that both have a relationship. Every employee must be a part of at least one department and every department must have at least one employee. To represent this relationship we use the foreign key.

In this case, Depart_Id is the foreign key of the Employee table as it establishes a relationship between the Employee table and the primary key of the Department table.

Employee Table:

Column Name	Data Type	Primary Key	Foreign Key
Emp_ID	Integer	Yes	
Emp_Name	Text		
Pan_Number	Text		
License_Number	Text		
Aadhar_Number	Text		
Department_ID	Integer		Yes

In a database, a primary key is a unique identifier for a record in a table, similar to how a ruling party has a prime minister in Indian politics holds the majority, and has the power to govern. However, just like in Indian politics, there can be multiple political parties that oppose the ruling party and offer alternative choices to the voters.

Similarly, in a database table, there can be multiple candidate keys - combinations of columns that can uniquely identify records, just like different political parties as alternates. However, only one of these candidate keys is chosen as the primary key, just like the ruling party that holds the majority.

The remaining candidate keys, which are not chosen as the primary key, can be considered as alternate keys. They provide alternative choices for uniquely identifying records in the absence of the primary key, just like the opposition parties in Indian politics offer alternative choices to voters in the absence of the ruling party. These alternate keys are not used as the primary means of identification, but they are still unique and can be used to uniquely identify records if needed.

Composite Key.
A composite key is a type of primary key in a relational database that consists of multiple columns or fields as the primary key in a table. In simple words, a composite key uses a combination of two or more columns as the primary key to uniquely identify records.

OrderDetails Table:

Column Name
Cust_ID
Order_ID
Prod_Code
Prod_Name

In the above table:

1. "Cust_ID": Represents the customer ID, this cannot serve as the primary key as a single user can place multiple orders.
2. "Order_ID": Represents the order ID, this cannot serve as the primary key as a single order id can contain multiple products.
3. "Prod_Code": Represents the product code, this cannot serve as the primary key as the same product can be ordered by multiple users.
4. "Prod_Name": Represents the product name, this cannot serve as the primary key as the same product can be ordered by multiple users.

In this condition, we cannot have a single-column primary key to uniquely identify the rows. Here we choose a combination of columns like Cust_ID, Order_ID, and Prod_Code as the primary key. This is also called the composite product key. If we just group and don't choose as a primary key it is simply a composite key.

Artificial Key.
If we look at the composite key which we have discussed in the previous section.

OrderDetails Table:

Column Name
Cust_ID
Order_ID
Prod_Code
Prod_Name

For this table, we cannot have a single-column primary key to uniquely identify the rows. Here we choose a combination of columns like Cust_ID, Order_ID, and Prod_Code as the primary key.

This primary key is very lengthy so we introduce a new column named "id" which has no significance to the records but is used to uniquely identify the rows. This is called the artificial key.

An artificial key is a key generated by a database management system to uniquely identify records in a database table. This is usually done if the primary key combination is too lengthy or there exists no primary key.

OrderDetails Table:

Field	Description
ID	Artificial key for uniquely identifying each order detail record
Cust_ID	Customer ID associated with the order
Order_ID	Order ID
Prod_Code	Product code
Prod_Name	Name of the product associated with the order