# Chapter 02: KEYS

## Definition

Database keys are attributes or sets of attributes in a relational database that are used to uniquely identify records within a table and establish relationships between tables. They are crucial for maintaining the integrity, consistency, and efficiency of the data in the database.

## Different Types of keys in DBMS with example

### 1. Super Key

**Definition**: A super key is any set of one or more columns that can uniquely identify a record in a table.
A super key can have extra columns that are not necessary for uniqueness.

**Purpose**: Represents all possible sets of columns that can uniquely identify a record.

### Example:

| Emp_id | Emp_Name | Email | Joining Date |
|---|---|---|---|
| 1 | Jhon | jhonuk@gmail.com | 11/11/2019 |
| 2 | Michle | Michleuk@gmail.com | 16/08/2019 |
| 4 | Jhon | Jhon22uk@gmail.com | 16/11/2019 |
| 5 | Sara | Sarauk@gmail.com | 11/11/2019 |

For the above table the possible super keys are: {Emp_id}, {Email}, {Emp_id, Emp_Name, Email}, {Emp_Name, Email, JoiningDate} etc.

But EmpName & JoiningDate can not work as superkey because if you use these columnstoidentifyUnique values then you can not do it because there remains duplicate values.

### SQL Queries for Super Key :

Using Emp_id and Email as a Super Key

```
SELECT * FROM Employee WHERE Emp_id = 1 AND Email = 'jhonuk@gmail.com';
```

Using Emp_id, Emp_Name, and Email as a Super Key

```
SELECT * FROM Employee WHERE Emp_id = 1 AND Emp_Name = 'Jhon' AND Email = 'jhonuk@gmail.com';
```

### 2. Candidate Key

**Definition**: A candidate key is a column or a group of columns that can potentially be used as a primary key. A table can have multiple candidate keys, but only one can be chosen as the primary key. A candidate key can never be NULL or empty. And its value should be unique.

**Purpose**: Represents a set of columns that uniquely identify a record.

### Example:

Suppose, the super key is: {Emp_id, Emp_Name,Email}. We can use {Emp_id, Emp_Name} or {Emp_id,Email} as a candidate key
But candidate key is the minimal set of Super Key. {Emp_id, EmpName} from here we get {Emp_id} , {EmpName} but {EmpName} is not a candidate key. Again, {Emp_id,Email} from here we get {Emp_id} , {Email}.
Both can be called candidate key.

### SQL Queries for Candidate Key :

Using Emp_id as a Candidate Key

```
SELECT * FROM Employee WHERE Emp_id = 1;
                    OR
SELECT DISTINCT Emp_id FROM Employee;
```

## 3. Alternate Key

**Definition**: 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 canbesetastheprimary key.
All the keys which are not primary key are called an Alternate Key.

**Purpose**: It serves as a unique identifier for records, similar to a primary key, but is not the primary key.

Example:

| Emp_id | Emp_Name | Email | Joining Date |
|---|---|---|---|
| 1 | Jhon | jhonuk@gmail.com | 11/11/2019 |
| 2 | Michle | michleuk@gmail.com | 16/08/2019 |
| 4 | Jhon | jhon22uk@gmail.com | 16/11/2019 |
| 5 | Sara | sarauk@gmail.com | 11/11/2019 |

In this table, Emp_id & Email are qualified to become a primary key. But since Emp_id is the primary key, so Email becomes the alternative key.

### SQL Queries for Alternate Key :

Using Email as a Alternate Key

```
SELECT * FROM Employee WHERE Email = 'jhonuk@gmail.com';
                    OR
SELECT DISTINCT Email FROM Employee;
```

## 4. Primary Key

**Definition**: A primary key is a unique identifier for each record in a table. It cannot contain NULL values, and each value must be unique across the table.

**Purpose**: Ensures that each record can be uniquely identified.

Example:

| Emp_id | Emp_Name | Email | Joining Date |
|---|---|---|---|
| 1 | Jhon | jhonuk@gmail.com | 11/11/2019 |
| 2 | Michle | michleuk@gmail.com | 16/08/2019 |
| 4 | Jhon | jhon22uk@gmail.com | 16/11/2019 |
| 5 | Sara | sarauk@gmail.com | 11/11/2019 |

In the above table, Emp_id can be used as a primary key through which data/records can be uniquely identified.

### SQL Queries for Primary Key :

1

```
CREATE TABLE Employee (
    Emp_id INT PRIMARY KEY,
    Emp_Name VARCHAR(50),
    Email VARCHAR(100),
    Joining_Date DATE
);
```

2

```
INSERT INTO Employee (Emp_id, Emp_Name, Email, Joining_Date)
VALUES
(1, 'Jhon', 'jhonuk@gmail.com', '2019-11-11'),
(2, 'Michle', 'michle@gmail.com', '2019-10-05'),
(3, 'Jhon', 'Jhon22@gmail.com', '2018-05-05'),
(4, 'Sara', 'sara@gmail.com', '2019-11-11');
```

3

```
SELECT * FROM Employee;
```

4

```
-- This will fail because Emp_id 1 already exists
INSERT INTO Employee (Emp_id, Emp_Name, Email, Joining_Date)
VALUES
(1, 'New Jhon', 'newjhon@gmail.com', '2020-01-01');
```

**Adding a Primary Key to an Existing Table**

```
ALTER TABLE Table_name
ADD PRIMARY KEY (column_name);
```

## 5. Foreign Key

**Definition**: A foreign key is a column or a group of columns in one table that creates a link between two tables. It refers to the primary key in another table.

**Purpose**: Enforces referential integrity by ensuring that a value in the foreign key column corresponds to a valid value in the related table's primary key.

The table containing the foreign key is called the child table, andthetablecontaining the candidate key is called the referenced or parent table.

### Example:

| S_id | S_Name | S_CGPA |
|---|---|---|
| 13231003 | x | 3.70 |
| 13231004 | y | 3.50 |
| 13231005 | z | 3.99 |

TableName: Student_Info

| S_id | Session | Dept |
|---|---|---|
| 13231003 | Summer-16 | CSE |
| 13231004 | Summer-16 | CSE |
| 13231005 | Summer-16 | CSE |

TableName: Dept_Info

```
Create table Student_info(
S_id int not null,
S_Name varchar(20),
S_CGPA float,
primary key(S_Id));
```

```
Create table Dept_info(
S_id int,
Session varchar(20),
Dept varchar(20),
Foreign key(S_id) references Student_info(S_id));
```

Set as primary key

| S_id | S_Name | S_CGPA |
|---|---|---|
| 13231003 | x | 3.70 |
| 13231004 | y | 3.50 |
| 13231005 | z | 3.99 |

Set as foreign key

| S_id | Session | Dept |
|---|---|---|
| 13231003 | Summer-16 | CSE |
| 13231004 | Summer-16 | CSE |
| 13231005 | Summer-16 | CSE |