Introduction to SQL

Database: It is a set of related information.



Fig:1. Telephone book: Classic example of database

Disadvantages of Telephone book database:

1. Time consuming
2. Not updated
3. Specific checking like finding people who live in the same street is difficult as it is indexed based on the first/last name.

These led to the development of computational databases. They are fast, updated on the fly and efficient for complex findings.

# **Introduction to Databases:**

## A database is an organized collection of data that can be easily accessed, managed, and updated.

## Databases are used to store and manage data for a wide range of applications, from small personal projects to large enterprise systems.

## There are two main types of databases: relational databases and non-relational databases

# **Types of Databases**

## Hierarchical Database: These are the first computerized database systems. In this database, data is stored in the form of one or more trees. Tree traversal algorithms are employed to access the data.

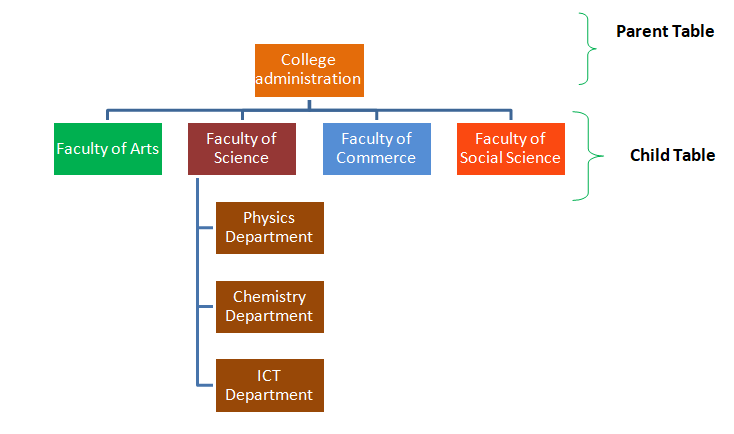


Fig: 2. A hierarchical model of an educational institute.

Tree traversal algorithms are computationally cost and the overall structure of the database will become complex as the data grows.

* 1. Relational Database: This is coined by Dr. E. F. Codd of IBM’s research laboratory in 1970. It proposes data to be represented as sets of tables and connecting these tables effectively.

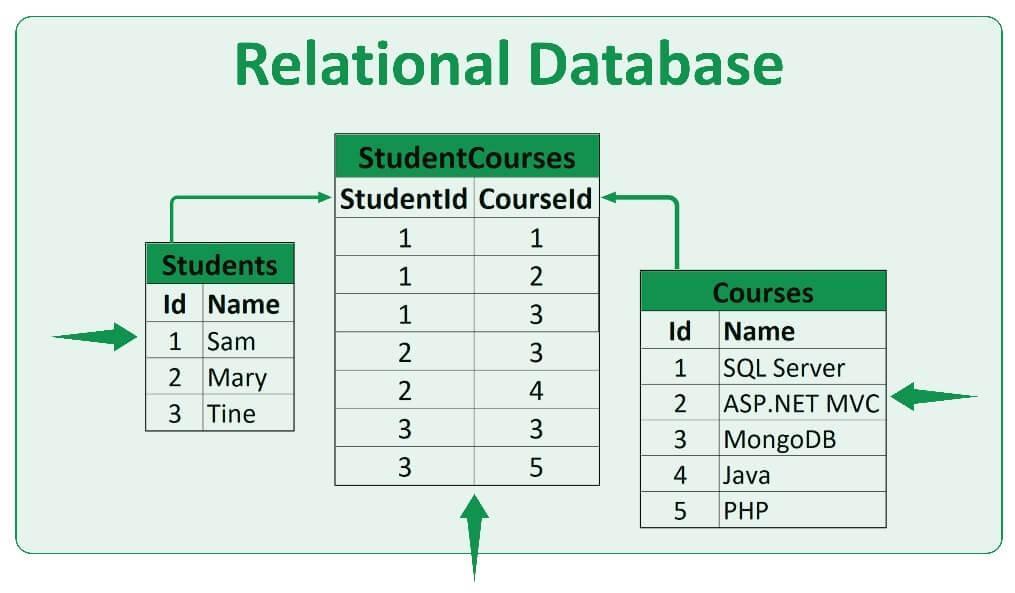


Fig: 3. A relational database of students and the courses.

Relational database models are fast and can represent any complex data effectively. One can easily travel from one table to another table to get the related information. Dr. Codd proposed a programming language DSL/Alpha for manipulating the data in relational models. A simplified version of DSL/Alpha was developed by IBM and called in SEQUEL which eventually shortened to SQL.

Note:

1. SQL is NOT an acronym for anything (even Structured Query language).
2. SQL is a non-procedural language. It means that the programmer has limited control over the flow of execution of the code when compared to the procedural languages like C, C++, Python etc. The flow of execution in SQL will be taken care by the *optimizer.*
3. SQL is a case-insensitive language.

# Relational Database Management System (RDBMS)

RDBMS is a software which supports SQL language to work on relational databases. SQL language is available on many RDBMS systems. Some of the popular RDBMS which supports SQL are:

1. Oracle Database from Oracle corporation
2. SQL Server from Microsoft
3. DB2 Universal Database from IBM
4. MySQL

Out of the above four RDBMS, MySQL is the only open-source RDBMS. In our entire course, we employ MySQL. There won’t be much differences in the code between MySQL and the other RDBMS as all them comply with the latest ANSI SQL standard.

Using SQL language, we can

1. Create Tables
2. Populate the tables with data
3. Retrieve the data from single/multiple tables
4. Update the data
5. Delete the data/table

In short, the above operations can be said as CRUD (Create, Retrieve, Update and Delete).

# Relational Databases

A relational database stores information in the form of tables. For example, consider the following table:

Table1: An example Database of Indian films.

| **Film** | **Lead\_Actor** | **Release\_Year** |
| --- | --- | --- |
| Baahubali-2 | Prabhas | 2017 |
| RRR | Jr. NTR & Ram Charan | 2022 |
| KGF-2 | Yash | 2022 |
| Pushpa | Allu Arjun | 2021 |
| Major | Adivi Sesh | 2022 |

Table1 has three columns. They are Film, Lead\_Actor and Release\_Year. Columns are also called as attributes. The table has 5 rows. Rows are also called as records.

3.1Traversing a table:

Suppose we want to retrieve the data from the table. We can traverse along the table and retrieve the data we are interested in. For retrieving the data, SQL provides three commands (also called as clauses). They are: SELECT, FROM, WHERE.

FROM – which table/tables

SELECT – which columns (selects all columns if \* is used)

WHERE – which rows (select all rows if Where is omitted)

3.2Query:

Query is the code used to retrieve the data from the database. The code will be written in SQL language. Although, SQL is a case-insensitive language, most of the practitioners will write all the SQL commands in upper case and the remaining code in lower case. So, we too follow the same standards.

Query example 1:

SELECT \*

FROM Table1

Result:

| Film | Lead\_Actor | Release\_Year |
| --- | --- | --- |
| Baahubali-2 | Prabhas | 2017 |
| RRR | Jr. NTR & Ram Charan | 2022 |
| KGF-2 | Yash | 2022 |
| Pushpa | Allu Arjun | 2021 |
| Major | Adivi Sesh | 2022 |

Query example 2:

SELECT Release\_Year

FROM Table1

Result:

| Release\_Year |
| --- |
| 2017 |
| 2022 |
| 2022 |
| 2021 |
| 2022 |

Query example 3:

SELECT Lead\_Actor

FROM Table1

WHERE Film = KGF-2

Result:

| Lead\_Actor |
| --- |
| Yash |

* WHERE is an optional command use to filter the rows from the table.

# DataTypes

The information stored in any table can be broadly classified into three types. Numeric, Text and Date/Time. Numeric form of data consists of numbers whereas Text form consists the textual data. Date/Time type is used to capture any eventual information. Every programming language has different classes to store these data. These classes are called as datatypes. In a RDBMS, we have data in columns and each column has to be defined with a datatype. To put it in simple, a datatype tells the SQL server, what type of data is being stored in a particular column.

Consider the table below. In the table, both Film and Lead\_actor columns store the text data whereas Release\_Date column stores the Date/Time data.

| Film | Lead\_Actor | Release\_Date |
| --- | --- | --- |
| Baahubali-2 | Prabhas | 2017-04-28 |
| RRR | Jr. NTR & Ram Charan | 2022-03-24 |
| KGF-2 | Yash | 2022-04-14 |
| Pushpa | Allu Arjun | 2021-12-17 |
| Major | Adivi Sesh | 2022-06-03 |

The datatypes allowed in MySQL can be broadly classified into three types:

1. Character Data: CHAR, VARCHAR
2. Numeric Data: INT, FLOAT
3. Date (Temporal) Data: DATE, DATETIME

Character Data:

CHAR: It is a datatype in SQL which is used to store character string of fixed length specified. If the length of the string is less than set or fixed-length then it is padded with extra blank spaces. We should use this datatype when we expect the data values in a column are of the same length.

Varchar: It is a datatype in SQL which is used to store character string of variable length but a maximum of the set length specified. If the length of the string is less than set or fixed-length then it will store as it is without padded with extra blank spaces. We should use this datatype when we expect the data values in a column are of variable length.

*Example:*

CREATE TABLE student(name VARCHAR(20), gender CHAR(1));

INSERT INTO student VALUES('Ravi', 'M');

INSERT INTO student VALUES('Rizwan', 'M');

INSERT INTO student VALUES(‘Rani’, ’F’);

Numeric Data:

INT

In MySQL, INT stands for the integer that is a whole number. An integer can be written without a fractional component e.g., 1, 100, 4, -10, and it cannot be 1.2, 5/3, etc. An integer can be zero, positive, and negative.

Column\_name INT;

Decimal:

*Example:* 3.14, 4.25, -3.254

Column\_name DECIMAL(P,S);

* P is the precision that represents the number of significant digits. The range of P is 1 to 65.
* D is the scale that that represents the number of digits after the decimal point. The range of D is 0 and 30.

Temporal data:

Used to store time events.

| Type | Usage | Data type format | Range |
| --- | --- | --- | --- |
| DATE | Stores only date information in the table column | YYYY-MM-DD format | from ‘1000-01-01’ to ‘9999-12-31’ |
| TIME | Displays only time | HH:MM:SS format | from ‘-838:59:59’ to ‘838:59:59’ |
| DATETIME | Stores both date and time in the column | YYYY-MM-DD HH:MM:SS | from ‘1000-01-01 00:00:00’ to ‘9999-12-31 23:59:59’ |

*Example:*

CREATE TABLE Person(

first\_name VARCHAR(20),

last\_name VARCHAR(20),

date\_of\_birth DATE

);

INSERT INTO employees (first\_name, last\_name, date\_of\_birth) VALUES (1, 'John', 'Sanders', '2000-01-19');

# Keys in MySQL

### **Primary key**

MySQL primary key is a single or combination of the field, which is used to identify each record in a table uniquely. A table may have duplicate columns, but it can contain only one primary key. It always contains a unique value into a column.

#### Rules for Primary key

1. The primary key column value must be unique.
2. Each table can contain only one primary key.
3. The primary key column cannot be null or empty.
4. MySQL does not allow us to insert a new row with the existing primary key.
5. It is recommended to use INT or BIGINT data type for the primary key column.

### **Natural and surrogate keys**

Sometimes the primary key is made up of real data and these are normally referred to as natural keys, while other times the key is generated when a new record is inserted into a table. When a primary key is generated at runtime, it is called a surrogate key.

### **Unique key**

A unique key in MySQL is a single field or combination of fields that ensure all values going to store into the column will be unique. It means a column cannot store duplicate values. For example, the email addresses and roll numbers of students in the "student\_info" table or contact number of employees in the "Employee" table should be unique.

### **Need of Unique Key**

* It is useful in preventing the two records from storing identical values into the column.
* It stores only distinct values that maintain the integrity and reliability of the database for accessing the information in an organized way.
* It can contain null value into the table.

### **Foreign key**

The foreign key is used to link one or more than one table together. It is also known as the referencing key. A foreign key matches the primary key field of another table. It means a foreign key field in one table refers to the primary key field of the other table. It identifies each row of another table uniquely that maintains the referential integrity in MySQL.

# **Basic queries on data (SELECT, FROM, WHERE):**

In SQL, a query is used to retrieve data from a database table. The basic query syntax includes the SELECT, FROM, and WHERE clauses.

**SELECT:** The SELECT statement is used to retrieve data from a database table. It specifies the columns that you want to retrieve. For example: SELECT column1, column2, column3. You can also use an asterisk (\*) to select all columns in the table. For example: SELECT \*.

**FROM:** The FROM statement specifies the table or tables from which you want to retrieve data. For example: FROM table1. You can also join multiple tables using the JOIN clause.

**WHERE:** The WHERE statement is used to filter data based on certain conditions. For example: WHERE column1 = 'value'. You can use logical operators such as AND, OR, and NOT to combine multiple conditions. For example: WHERE column1 = 'value' AND column2 > 10.