

# RDBMS

RDBMS stands for Relational Database Management Systems. It is a program that allows us to create, delete, and update a relational database. A Relational Database is a database system that stores and retrieves data in a tabular format organized in the form of rows and columns. It is a smaller subset of DBMS which was designed by E.F Codd in the 1970s. The major DBMSs like [SQL](#), [My-SQL](#), and [ORACLE](#) are all based on the principles of relational DBMS.

Relational Database Management Systems maintains data integrity by simulating the following features:

- **Entity Integrity:** No two records of the database table can be completely duplicate.
- **Referential Integrity:** Only the rows of those tables can be deleted which are not used by other tables. Otherwise, it may lead to [data inconsistency](#).
- **User-defined Integrity:** Rules defined by the users based on confidentiality and access.
- **Domain integrity:** The columns of the database tables are enclosed within some structured limits, based on default values, type of data or ranges.

## What is a Database Table?

A table is a **collection of related data** in an organized manner in the form of rows and columns. It is an organized arrangement of data and information in **tabular form** containing rows and columns, making it easier to understand and compare data. Here is the pictorial representation of the table and its different components containing the data about different students that is ID, name, Age, and course.

## Features of RDBMS

- Data must be stored in tabular form in DB file, that is, it should be organized in the form of rows and columns.
- Each row of table is called record/tuple . Collection of such records is known as the cardinality of the table
- Each column of the table is called an attribute/field. Collection of such columns is called the arity of the table.
- No two records of the DB table can be same. Data duplicity is therefore avoided by using a candidate key. Candidate Key is a minimum set of attributes required to identify each record uniquely.
- Tables are related to each other with the help for foreign keys.
- Database tables also allow NULL values, that is if the values of any of the element of the table are not filled or are missing, it becomes a NULL value, which is not equivalent to zero. (NOTE: Primary key cannot have a NULL value).

# SQL Query in RDBMS

## Creating a Table

### Syntax:

```
CREATE TABLE table_name (  
  
column1_name datatype constraint,  
  
column2_name datatype constraint,  
  
);
```

### Example:

```
CREATE TABLE Employees (  
  
EmployeeID INT PRIMARY KEY,  
  
FirstName VARCHAR(50),  
  
LastName VARCHAR(50),  
  
BirthDate DATE,  
  
Salary DECIMAL(10, 2)  
  
);
```

## 2. Inserting Data into a Table

**Syntax:**

```
INSERT INTO table_name (column1_name, column2_name, ...)
```

```
VALUES (value1, value2, ...);
```

**Example:**

```
INSERT INTO Employees (EmployeeID, FirstName, LastName,  
BirthDate, Salary)
```

```
VALUES (1, 'John', 'Doe', '1985-06-15', 55000.00);
```

**3. Querying Data (SELECT)****Syntax:**

```
SELECT column1_name, column2_name, ...
```

```
FROM table_name
```

```
WHERE condition;
```

**Example:**

```
SELECT FirstName, LastName, Salary
```

```
FROM Employees
```

```
WHERE Salary > 50000;
```

**4. Deleting Data from a Table**

**Syntax:**

```
DELETE FROM table_name
```

```
WHERE condition;
```

**Example:**

```
DELETE FROM Employees
```

```
WHERE EmployeeID = 1;
```

## 5. . Dropping a Table

**Syntax:**

```
DROP TABLE table_name;
```

**Example:**

```
DROP TABLE Employees;
```

## Advantages of RDBMS

- **Easy to Manage:** Each table can be independently manipulated without affecting others.
- **Security:** It is more secure consisting of multiple levels of security. Access of data shared can be limited.
- **Flexible:** Updating of data can be done at a single point without making amendments at multiple files. Databases can easily be extended to incorporate more records, thus providing

greater scalability. Also, facilitates easy application of SQL queries.

- **Users:** RDBMS supports [client-side architecture](#) storing multiple users together.
- Facilitates storage and retrieval of large amount of data.
- **Easy Data Handling:**
  - Data fetching is faster because of relational architecture.
  - Data redundancy or duplicity is avoided due to keys, indexes, and normalization principles.
  - Data consistency is ensured because RDBMS is based on [ACID properties](#) for data transactions (Atomicity Consistency Isolation Durability).
- **Fault Tolerance:** Replication of databases provides simultaneous access and helps the system recover in case of disasters, such as power failures or sudden shutdowns.

## Disadvantages of RDBMS

- **High Cost and Extensive Hardware and Software Support:**  
Huge costs and setups are required to make these systems functional.

- **Scalability:** In case of addition of more data, servers along with additional power, and memory are required.
- **Complexity:** Voluminous data creates complexity in understanding of relations and may lower down the performance.
- **Structured Limits:** The fields or columns of a relational database system is enclosed within various limits, which may lead to loss of data.
-