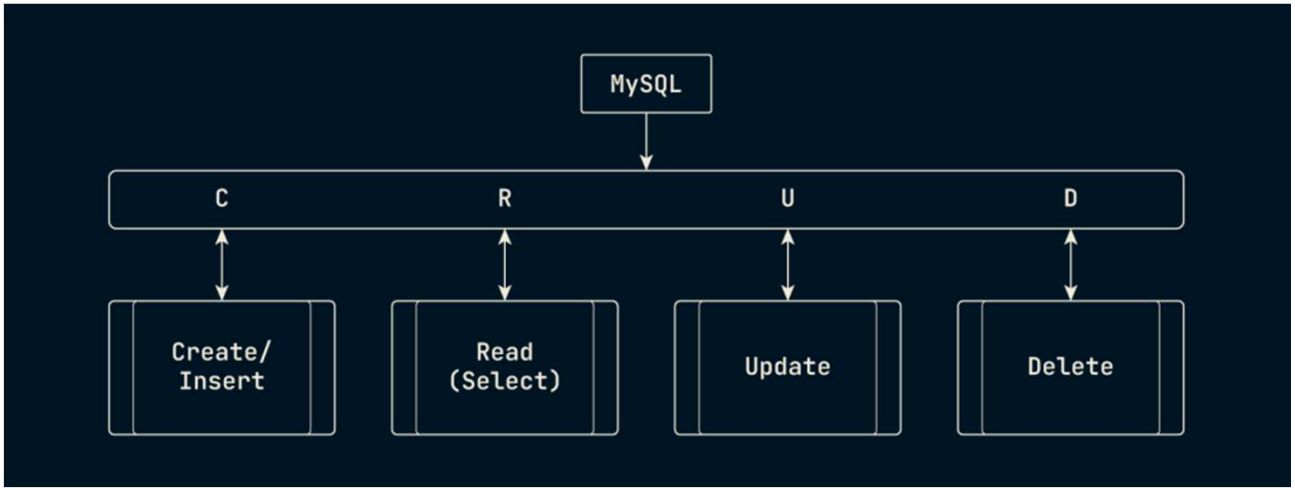
ABESIT
COLLEGE OF ENGINEERING
CRC-Training

MySQL : CRUD Operations SQL



ABESIT
COLLEGE OF ENGINEERING
CRC-Training

MySQL : CRUD Operations SQL

- **CRUD** describes the four basic operations that a user can perform in a database. Users can
  1. ***Create*** or ***insert*** new records into a table,
  2. ***Read*** or ***select*** existing records from a table,
  3. ***Update*** or ***modify*** existing records stored inside a table,
  4. ***Delete*** or ***remove*** existing records from a table.

ABESIT COLLEGE OF ENGINEERING

CRC-Training

MySQL : CRUD Operations SQL - Insert

- **Insert Operation Syntax :**
  - INSERT INTO *table_name* (*column1*, *column2*, *column3*, ...)
    VALUES (*value1*, *value2*, *value3*, ...);
  - INSERT INTO table_name
    VALUES (value1, value2, value3, ...);
- **Example :**
  - INSERT INTO Student_tb (StudentID, StudentName, Branch, City)
    VALUES (1001, 'Shyam Sharma', 'EC', 'Noida');
  - INSERT INTO Student_tb
    VALUES (1002, 'Ram Sharma', 'EC', 'Noida');
  - INSERT INTO Student_tb (StudentName, City)
    VALUES ('Manu Sharma', 'Noida');

ABESIT
COLLEGE OF ENGINEERING
CRC-Training

MySQL : CRUD Operations SQL - Select

- **Select Operation Syntax :**
  - SELECT *column1*, *column2, ...*
    FROM *table_name*;
  - SELECT *
    FROM *table_name*;
- **Example :**
  - SELECT StudentName, Branch, City
    FROM Student_tb;
  - SELECT *
    FROM Student_tb;

ABESIT COLLEGE OF ENGINEERING

CRC-Training

MySQL : CRUD Operations SQL - Select

- **Select - Where Operation Syntax :**
  - SELECT *column1*, *column2, ...*
    FROM *table_name*
    WHERE *condition*;
  - SELECT *
    FROM *table_name*
    WHERE *condition*;
- **Example :**
  - SELECT *
    FROM Student_tb
    WHERE City = 'Delhi';
  - SELECT *
    FROM Student_tb
    WHERE StudentID = 1001;

ABESIT COLLEGE OF ENGINEERING

CRC-Training

MySQL : CRUD Operations SQL - Update

- **Update Operation Syntax :**
  - UPDATE *table_name*
    SET *column1* = *value1*, *column2* = *value2*, ...
    WHERE *condition*;

- **Example :**
  - UPDATE Student_tb
    SET StudentName = 'Ramesh Kumar', City = 'Delhi'
    WHERE StudentID = 1001;

ABESIT COLLEGE OF ENGINEERING

CRC-Training

MySQL : CRUD Operations SQL - Delete

- **Delete Operation Syntax :**
  - DELETE FROM *table_name*
    WHERE *condition*;

- **Example :**
  - DELETE FROM Student_tb
    WHERE StudentName = ‘Ramesh Kumar';

ABESIT COLLEGE OF ENGINEERING

CRC-Training

MySQL : Performing Basic DB Operations (DML)

- **DML(Data Manipulation Language):** DML commands deal with operations on data present in the database and DML commands make up a majority of the SQL statements.

1. **INSERT** – is used to insert data into a table.
2. **UPDATE** – is used to update existing data within a table.
3. **DELETE** – is used to delete records from a database table.

ABESIT
COLLEGE OF ENGINEERING
CRC-Training

MySQL : Join

- Join is used to combines the row of two or more tables based on related columns between them.
- The main purpose of Join is to retrieve the data from multiple tables in other words Join is used to perform multi-table query.
- Types of Join
  - Inner Join
  - Outer join

ABESIT COLLEGE OF ENGINEERING

CRC-Training

MySQL : Join

- **Inner join:** Inner Join is a join operation in DBMS that combines two or more table based on related columns and return only rows that have matching values among tables. Inner join of two types.
  - Equi Join
  - Natural Join

- **Equi Join :** Equi Join is a type of Inner join in which we use equivalence('=') condition in join condition

Example:

Table A

Column A	Column B
a	a
a	b

Table B

Column A	Column B
a	a
a	c

Result:

Column A	Column B
a	a

ABESIT COLLEGE OF ENGINEERING

CRC-Training

MySQL : Join

Natural Join: Natural join is a type of inner join in which we not need of any comparison operators. In natural join columns should have the same name and domain. There should be at least one common attribute between two tables.

Example:

Table A

Number	Square
2	4
3	9

Table B

Number	Cube
2	8
3	27

Result:

Number	Square	Cube
2	4	8
3	9	27