DATABASE MANAGEMENT SYSTEM

DBMS :

WEB APPLICATIONS : Websites + Web Application (Dynamic webpage)

Projects : Data creation (Form)

Form (Designing)

Data implement – Programming

DBMS – Database management system

----------------------------------------------------------------------------------

DBMS : DBMS is a software that is used to manage data at backend side. DBMS provides facility to store , manipulate and access data from the database.

DBMS also provides many facilities to make sure that right data is being stored in the database.

DBMS provides table structure to developer to store and manage data.

There are many applications of DBMS is present , you can use any one of these to store & manage data :

Like : MYSQL, MSSQL , MSACCESS, ORACLE, MONGODB , SQLITE……….

----------------------------

To communicate with the all application of DBMS , there is a systemic way , that is SQL.

SQL : SQL stands for Structured Query Language . It provides some pre-defined syntax of queries. These queries are used to create , manipulate and access data in the DBMS application.

SQL mainly provides a way to work with DBMS application.

-----------------------------------

Standard way to store data in DBMS is :

Database🡪Tables -> Rows/Tuples & Columns/Fields -> Data

--------------------------------------------

TYPES OF SQL COMMANDS :

1. DDL: Data Definition Language
2. DML : Data Manipulation Language
3. TCL : Transaction Control Language
4. DCL : Data Control Language

SCHOOL MGMT -> DB -> MANY TABLES

DDL : DDL stands for Data Definition Language . It has some queries list, which specifies the schema of table.

That means all DDL commands are applied on the schema of table.

Create

Alter

Truncate

Drop

Rename

DML : DML stands for Data Manipulation Language . It has some queries that Is applied on data/records of table.

DML commands are used to create , manipulate and access the records of table.

INSERT

UPDATE

DELETE

SELECT

TCL : TCL stands for Transaction Control Language . TCL has some commands that is used to control the transactions of database.

Commit

Rollback

Save transaction

DCL : DCL stands for Data Control Language . DCL commands are used to define users of database that who can access the data and who can read and write both on database.

Grant

Revoke

---------------------------

Data Type :

Numeric :

INTEGER

Int

Shortint

longint

float

money

-------------------------------

Char :

Char : name char(100) : fixed length string

Varchar : name varchar(100) : variable length string

Nchar : multi-languages fixed length string

Nvarchar : multi language variable length string

Text : long size string

---------------------------------

Date : Used to store date in pre-defined format like yyyy-mm-dd

Date : only date : yyyy-mm-dd

Time : only time : hh :mm : ss

Datetime : collection of date and time : yyyy-mm-dd hh:mm:ss

----------------------------------------------------------------------------------------------

Create : Create command is used to make a new database or table.

Syntax to create database :

Create database database\_name;

Syntax to create table :

Create table table\_name

(

Field\_name datatype(size) integritity\_constrainits,

Field\_name datatype(size)

-

-

-

);

---------------------------

Integrity Constraints : Integrity constraints are used with create command that is used to make sure that integrity of database will not be disturbed.

So integrity constraints are some keywords that is used to make sure that your table will hold correct data.

Not null : Not null defines that this field can not be null for any tuple.

Default : It sets a by -default value for fields that will be automatic inserted when this field is NULL.

Unique : duplicate value is not accepted , but NULL value is accepted.

Primary key : UNIQUE + NOT NULL

Foreign key : It stores a value that is already a Primary key column value in another table.

Check : check validates the value that is inserted into table before insertion.

---------------------------------------------------------------------------------------------

Create table employee (

Empid int primary key,

Emp\_name char(50) not null,

Salary int check (salary>10000 and salary<50000) ,

Mobile\_no char(13) default ‘7007237006’,

Age int check (age>15)

);

Insert : Insert command is used to add new tuples in tables.

Syntax :

Insert into table\_name values(value\_1, value\_2 , value\_3 , ………….);

Ex :

Insert into employee value(1, ‘Riya Singh’, 5000 , ‘908776655’ , 20);

Syntax \_2 :Insert values in specified columns , remains left NULL,

Insert into table\_name (column\_name\_1, column\_name\_2 , column\_name\_3……….) values(value\_1, value\_2 , value\_3 , ……..);

Ex :

Insert into employee (emp\_id, salary) values(2, 40000);

BULK INSERTION : You can insert multiple rows with single insert command by seprating each rows by comma.

Insert into table\_name values(3,’’,30000,’’,20),(4,’’, 40000, ‘’ , 25);

Insert into table\_name(emp\_id, empname) values (), (), (), () , ();

---------------------------------------------------------------------------------------

Identity : Identity property can be used only with integer type column.

It is used to increment columns value automatic . In some DBMS applications it is also known as Auto\_increment.

Create a table with id should be primary key and name should have default value User.

SYNTAX :

COLUMN\_NAME data\_type(size) identity (seed, increment);

Seed : starting value

Increment : how much value would plus to max value of column

Indentity(100, 5);

Create table trainees

(

Id int primary key identity(1,1),

Name char(20) default ‘user’

);

----------------------------------------------------------------------------------------------

Database -> Technical

DBMS ->

MSSQL , MYSQL, MSACCESS, MONGODB

CRUD Operations

Create - c

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| sn | name | fathername | Mobileno | age | email | Profilephoto |
| 1 | ram | NULL | 8090988776 | 20 | ram@gmail.com | NULL |
| 2 | ram | NULL | 8090988776 | 20 | r@gmail.com |  |
| 3 | ram | NULL | 8090988776 | 20 | [rama@gmail.com](mailto:rama@gmail.com) |  |

Create : DDL

Update , delete , select

Create , alter , truncate, drop , insert

Ddl : create, alter , drop, truncate , rename

Create table table\_name

Alter table table\_name

Drop table table\_name

Truncate table table\_name

Rename table table\_name t

Dml : insert, update, delete, select ,

Insert into table\_name : rows

Update table\_name set : cols

Delete from table\_name : rows

Select \*/column\_name from table\_name

Drop database

Drop table

-----------------------------------------------------------------------------------------------

User input

|  |
| --- |
| [r@gmail.com](mailto:r@gmail.com) |

Select \* from tbl where email=’’

INSERT INTO TBL (email) values(‘ram@gmail.com’)

Insert into tbl (sn,name,mobileno,age,email)values(1,’ram’,’8090’,20,’ram@gmail.com’)

INSERT INTO TBL ()

Read – select / fetch – dml

Registration -------- (insert) ->

Login ----🡪 select with condition

Change password ->update

Change profile photo -> update

Delete account -> delete

Change -> update

View profile -> select

Update -

Delete

DDL – CREATE

DML – INSERT

INTEGRITY CONTSRAINTS :

IDENTITY

Select : select \*/column\_names from table\_name

[where clause]

[group by clause]

[having]

[order\_by ]

Login :

|  |
| --- |
| [D@gmail.com](mailto:D@gmail.com) |

|  |
| --- |
| 123 |

Where clause : delete , update , select

= , <> , > , >= , < , <=, and , or

Login : [saloni@gmail.com](mailto:saloni@gmail.com), password

Your scored : 25 Marks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ram | 24 |  |  |  |
| Hari | 23 |  |  |  |
| priya | 20 |  |  |  |
| siya | 25 |  |  |  |

In , not in , between ,is null , is not null , like

DBMS :

SQL :

USER -> request -> SERVER -> Response

User -> request

Developer - >Database MGMT -> Table -> file/folder

Ready the Schema of table

Columns -> column\_name , data type , size , integrity constraints

Not null , , default check , unique key , primary key , foreign key

Records

DDL : applied on schema : create , alter , drop , truncate , rename

DML : insert , update , delete ,select

TCL : commit, rollback, save transaction

DCL : grant, revoke

------------------------------------------

Select :

Select \*/column\_names from table\_name where <condition>

=

>

>=

<=

<

<>

And

Or

In

Not in

Between

Is null

Is not null

Like : like operator is used to compare data with a part of data or pattern. When you have concept of part of data or you have a pattern behalf of that you want to select data from table then like operator is used with condition.

%

\_

Syntax :

Select \*/column\_names from table\_name where column\_name like ‘pattern’

To create pattern : you can use % or \_

% - ignores characters

\_ - ignores only a single character

Ex :

Where Name like ‘%singh’ : select all rows where name ends with singh

Where college like ‘Feroze%’ :select all rows where college name starts with feroze .

Where college like ‘%gandhi%’ : select all rows where Gandhi presents somewhere

Where dob like ‘2021-\_\_\_\_\_’ :

2021-04-04

2021-

2021-05-09

2021-12-31

2020-04-04

Order by : In table , data is saved in random manner , is you want to arrange data behalf of a column then order by is used.

Numeric – assending(smaller-onwards) / descending (greater - lower)

String – assending(a-z) / descending (z-a)

Syntax :

Select \*/column\_name from table\_name where <condition> order by column\_name <desc>

Ex :

Select \* from tbl where batch=’All’ order by id desc

----------------------------------------------------------------------------------------------

Aggregate Function: Aggregate function are some function that operates on multiple values and provides a single output.

Sum() , count() , avg() , min() , max()

It can be only used with select command . it is only used after select keyword to select a value as a column.

It can be applied only on numeric values/columns.

Select sum(fee)

Select count(college)

select max(fee) from stpayment

select count(\*) from table\_name

select count(column\_name) from table\_name

Note : Count function ignores null values. That means the column which contain NULL value is not counted by count() function.

---------------------------------------------------------------------------------------

String function :

Len() :

select len(name) as Length,lower(name) as studentname from ststudent

Upper() : select upper(name) from table\_name

Lower() : converts string to the lowercase

Ltrim() : select upper(ltrim(rtrim(' techpile technology ')))

Rtrim() : removes extra spaces from the right side of value

Substring() :

select substring(uname,0,13) from ststudent

--------------------------------------------------------------------------------------------

Date & Datetime function :

* Getdate() : IT IS A FUNCTION which returns current date and time.
* Current\_TIMESTAMP : it is a property which returns current date and time

select getdate(),Current\_TIMESTAMP,

datename()

dateadd()

datediff()

convert()

2021-11-08 :

**How to create views**

In SQL, a view is a virtual table based on the result-set of an SQL statement.

A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.

You can add SQL statements and functions to a view and present the data as if the data were coming from one single table.

A view is created with the CREATE VIEW statement.

**CREATE VIEW Syntax**

CREATE VIEW *view\_name* AS  
SELECT *column1*, *column2*, ...  
FROM *table\_name*  
WHERE *condition*;

**Example:**

*create view[studentids] as*

*select studentname,emailid from student where classid=900001*

***How to see view***

*Select \* from view*

## ***SQL Updating a View***

A view can be **updated with the** CREATE OR REPLACE VIEW statement.

alter VIEW[studentids] as

select studentname,emailid,fee from student where classid=900001

**Dropping a View**

**DROP VIEW View\_Name;**