**What is SQL ?**

SQL stands for structured query language

SQL is a case in-sensitive language

SQL is used to create database | table structured

SQL is used to | delete | update | select data from database

SQL is used to join data from tables

**Types of SQL commands or query**

a) DDL (data definition language)

b) DML (data manipulations language)

c) DQL (data query language)

d) TCL (transactional query language)

**DDL** : ddl stands for data definition language

Ex: create | alter | drop | truncate | change | rename

a) how to create a database

syntax : create database databasename;

ex: create database shopsdb;

b) how to create tables

syntax : create table tablename

(

columnname datatype(size) primary key auto\_increment,

columnname datatype(size)

)

Table datatype chart

Fieldname datatype(size)

Id int(default size(11))

name char, varchar(0-255)

mobile bigint(default size(20))

text,address,message text

date date

datetime datetime

images varchar, blob

salary float

ex: create table contactus

(

contactid int AUTO\_INCREMENT primary key,

name varchar(255),

email varchar(255),

firstname varchar(155),

lastname varchar(125),

mobile bigint,

message text

);

create table country

(

countryid int AUTO\_INCREMENT primary key,

countryname varchar(255)

);

**Alter :**

Alter is used to add , modify or change column name after create table

a) how to add new column

alter table country add addeddate date;

b) how to modify or change columnanme

alter table contactus change mobile phone bigint;

c) alter table employee add address text;

d) alter table employee add photo varchar(255) after emplid;

e) alter table employee add hobby varchar(255) after address;

**change** : change should be change the columnname used in alter

ex: alter table contactus change mobile phone bigint;

**drop** : drop is used to drop database or table

drop delete database structured and tables structured and data both

after drop we cant rollback anything

ex: drop database databasename;

drop database shopsdb;

ex: drop table country;

**truncate :** truncate are used to truncate data from tables

**ex:** truncate table tablename;

truncate table country;

Note : after truncate we never rollback data

**Rename :** rename is used to change table name or update table name

rename table contactus to tbl\_contactus

rename table country to tbl\_country

**DML : data manipulation language**

**a) insert b) delete c) update**

**a) insert :** insert is used to insert data in tables

syntax : insert into tablename(columnname) values(‘values’);

ex: insert into tbl\_country(countryname,addeddate) values ('india','2023/07/15');

or

insert into tbl\_country(countryname,addeddate) values ('afganistan','15/07/2023'),('france','15/07/2023'),('dubai','15/07/2023');

or

insert into tbl\_country values ('null','australia','15/07/2023'),('null','Africa','15/07/2023'),('null','uk','15/07/2023');

**delete :** delete is used to delete all data and particular data and random data from table

a) delete from tablename;

ex: delete from tbl\_country;

b) delete from tablename where id=2;

ex: delete from tbl\_country where countryid=3

c) delete from tbl\_country where countryname='africa';

d) delete from tbl\_country where countryid in(2,5,8);

e) delete from tbl\_country where countryid BETWEEN 5 and 30;

**Note : after delete we will rollback our data**

**Update** : update is used to update particular data or rows

Ex: update tablename set columnname=’value’ where id=2;

Ex: update tbl\_country set countryname='canada' where countryid=4;

Ex: update tbl\_country set countryname='romania',addeddate='2023-07-15' where countryid=4;

**DQL :** DQL stands for data query language

a) select

select is used to select all data from table or fetch data

ex: select \* from tbl\_country

ex: select \* from tbl\_country where countryid=1;

ex: select \* from tbl\_country where countryname='india';

ex: select \* from tbl\_country where countryid in(1,4,5,6,8,9);

ex: select \* from tbl\_country where countryid BETWEEN 1 and 3;

ex: select \* from tbl\_country where countryid BETWEEN 1 and 10;

ex: select countryname from tbl\_country;

ex: select \* from tbl\_country where countryid limit 0,10;

ex: how to give alise name of any columnname

select countryname as cname from tbl\_country

**order by :** filter data **from tables in ascending and descending** order there we used order by

difference between truncate | delete | drop

**truncate** : truncate is used to empty data from tables

after truncate we can not rollback our data

ex: truncate table tbl\_salary;

insert into tbl\_salary values('null','adarsh',99500),('null','nezal',99600),('null','neha',99700),('null','meena',99800),('null','sohil',99900),('null','uttam',99800),('null','femish',99900);

**delete :** delete are used to delete particular rows or data or delete all data or delete alternate data

**Note : after delete we will rollback our data**

**a) delete from tbl\_salary;**

**b) delete from tbl\_salary where name=’nezal’;**

**b) delete from tbl\_salary where salaryid=10;**

**c) delete from tbl\_salary where id in(8,10,12,14);**

**drop :** drop is used to drop database and table structured and data both after drop we can not rollback our data and structured both.

Ex: drop database databasename;

Ex: drop table tablename;

difference between **order by** and **group by** :

**Order by** : order by is used to filter data in asending or desending order

Ex: select \* from tbl\_salary order by name asc;

Ex: select \* from tbl\_salary order by name;

Ex: select \* from tbl\_salary order by name desc;

**Group by :** filter data from tables in group of columns

Ex: select sum(salary),department from tbl\_salary GROUP by department;

**Like operator :** like operator is used to search data from tables using wild card in SQL

Ex: select \* from tbl\_salary where name like 'a%';

Ex: select \* from tbl\_salary where name like '%h';

Ex: select \* from tbl\_salary where name like '%h%';

**SQL functions** : sql function is used to provides some queries where we find max, min, avg, sum, etc

**Types of SQL function**

a) aggregate function

ex: max

min

avg

sum

count

b) scalar function

ex: first

last

ucase

lcase

a) select max(salary) as maximum\_salary from tbl\_salary

b) select min(salary) as min\_salary from tbl\_salary

c) select avg(salary) as average\_salary from tbl\_salary

d) select sum(salary) as sumofsalary from tbl\_salary

e) select count(salaryid) as numberofemployee from tbl\_salary;

**subquery** : query within another query i.e called subquery

find second highest salary from tables :

ex: select max(salary) from tbl\_salary where salary<(SELECT max(salary) from tbl\_salary)

**scalar functions :**

**a) first :** find first data from table

select first(name) from tbl\_salary;

**b) last :** find last data from table

select last(name) from tbl\_salary;

**c) ucase :** convert name in uppercase

select ucase(name) from tbl\_salary;

**a) lcase :** find lower case name data from table

select lcase(name) from tbl\_salary;

**SQL key constraints :** sql key constraints is setting limit on tables

**Types of sql key contsraints**

a) primary key

b) foreign key

c) unique key

**a) Primary key : A** pk is only defined once time in a tables

A pk is never return null values

A primary key is always auto\_increments

Tbl\_users

Uid(pk) name fname lname address pincode

create table tbl\_state

(

sid int PRIMARY key AUTO\_INCREMENT,

sname varchar(255)

)

**b) foreign key :** A fk is provides more than once time in a table

A fk never return null values

A fk is provides relationship between one table to another table

Tbl\_state

Sid(pk) sname

1 gujrat

2 rajsthan

Tbl\_city

Ctid(pk) sid(fk) ctname

1 1 rajkot

2 2 jaipur

3 1 ahemdabad

CREATE table tbl\_city

(

ctid int primary key AUTO\_INCREMENT,

sid int REFERENCES tbl\_state(sid),

ctname varchar(155)

)

create table tbl\_user

(

uid int PRIMARY key AUTO\_INCREMENT,

username varchar(255),

password varchar(155),

name varchar(255),

address text,

sid int REFERENCES tbl\_state(sid),

ctid int REFERENCES tbl\_city(ctid)

)

**c) unique key :**

**unique key : unique key is used to provides more than one columns in a tables i.e used to identify uniquely in a table**

**note : uk is used to return once time a null values in a tables**

**ALTER TABLE `tbl\_employees` ADD UNIQUE(`mobile`);**

**ALTER TABLE `tbl\_employees` ADD UNIQUE(`email`);**

**create table tbl\_department**

**(**

**depid int PRIMARY KEY AUTO\_INCREMENT,**

**depname varchar(255)**

**)**

**create table tbl\_employees**

**(**

**emplid int AUTO\_INCREMENT primary key,**

**empname varchar(155),**

**password varchar(155),**

**firstname varchar(155),**

**lastname varchar(155),**

**salary float,**

**mobile bigint,**

**depid int REFERENCES tbl\_department(depid),**

**compid int REFERENCES tbl\_company(compid)**

**)**

**TCL : stands for trasactional control language**

**a) tcl is used to commit data**

**after delete user can seve data there we used commit**

**ex: START TRANSACTION;**

**delete from tbl\_city where ctid=10;**

**commit;**

**b) rollback : rollback after delete we can rollback our data**

**note : rollback is not support in mysql**

**ex: START TRANSACTION;**

**select \* from tbl\_city where ctid=10;**

**rollback;**

**SQL join :**

**sql join is used to join more than one tables itself and return data if first table of data matched with second table of data.**

**types of join :**

**a) inner join**

**b) join**

**c) outer join**

**a) left join**

**b) right join**

**c) full join**

**d) cross join**

**a) inner join : inner join is used to join first table of matched data with second table of matched data if data matched join tables otherwiser return null values.**

**syntax : select firstable.\*,columnname from firsttable inner join secondtable name on firsttable.commonfield=secondtable.commonfieild;**

**ex:select tbl\_employees.\*,depname from tbl\_employees inner join tbl\_department on tbl\_employees.depid=tbl\_department.depid;**

**ex: select tbl\_employees.\*,depname,compname from tbl\_employees inner join tbl\_department on tbl\_employees.depid=tbl\_department.depid inner join tbl\_company on tbl\_employees.compid=tbl\_company.compid;**

**b) join : join is used to join first table of matched data with second table of matched data if data matched join tables otherwisr return null values.**

**select tbl\_employees.\*,depname,compname from tbl\_employees join tbl\_department on tbl\_employees.depid=tbl\_department.depid join tbl\_company on tbl\_employees.compid=tbl\_company.compid;**

**c) outer join**

**a) left join : left join is used to join first table of left rows matched data with second table of left rows if data matched join tables otherwise return null values.**

**ex: select tbl\_employees.\*,depname,compname from tbl\_employees left join tbl\_department on tbl\_employees.depid=tbl\_department.depid left join tbl\_company on tbl\_employees.compid=tbl\_company.compid;**

**b) right join : right join is used to join second table of right rows matched data with first table of right rows if data matched join tables otherwise return null values.**

**ex: select tbl\_employees.\*,depname,compname from tbl\_employees right join tbl\_department on tbl\_employees.depid=tbl\_department.depid right join tbl\_company on tbl\_employees.compid=tbl\_company.compid;**

**ex:**

**c) full join**

**note : full join is not support mysql**