-- create

CREATE TABLE EMPLOYEE (

empId INTEGER PRIMARY KEY,

name TEXT NOT NULL,

dept TEXT,

salary decimal(10,2),

active boolean,

join\_date date

);

-- insert

INSERT INTO EMPLOYEE VALUES (0001, 'Clark', 'Sales', 100000, True, '2024-12-12');

INSERT INTO EMPLOYEE VALUES (0002, 'Dave', 'Accounting', 120000, False,'2025,03-10');

INSERT INTO EMPLOYEE VALUES (0003, 'Ava', 'Sales',110000, True, '2015-05-24');

insert into employee (empId, name, dept, salary, active, join\_date)

values(0004, "Ravindra", 'HR', 150000, True, '2023-10-10');

insert into employee (empId, name, dept, salary, active, join\_date)

values(0005, "Sudarshan", 'HR', 115000, False, '2025-07-01');

insert into employee (empId, name, dept, salary, active, join\_date)

values(0006, "Aruna", 'Accounting', 125000, True, '2010-11-11');

create table department(

deptID integer primary key,

deptName text not null,

block integer

);

insert into department (deptID, deptName, block)

values(2001, "Sales", 1);

insert into department (deptID, deptName, block)

values(2002, "Accounting", 2);

insert into department (deptID, deptName, block)

values(2003, "HR", 3);

select name,

timestampdiff(year, join\_date, current\_date) as service

from employee;

Output:

+-----------+---------+

| name | service |

+-----------+---------+

| Clark | 0 |

| Dave | 0 |

| Ava | 10 |

| Ravindra | 1 |

| Sudarshan | 0 |

| Aruna | 14 |

+-----------+---------+

select name, join\_date

from employee

order by join\_date desc;

Input for the program ( Optional )

STDIN

Output:

+-----------+------------+

| name | join\_date |

+-----------+------------+

| Sudarshan | 2025-07-01 |

| Dave | 2025-03-10 |

| Clark | 2024-12-12 |

| Ravindra | 2023-10-10 |

| Ava | 2015-05-24 |

| Aruna | 2010-11-11 |

+-----------+------------+

select name, join\_date

from employee

order by join\_date;

+-----------+------------+

| name | join\_date |

+-----------+------------+

| Aruna | 2010-11-11 |

| Ava | 2015-05-24 |

| Ravindra | 2023-10-10 |

| Clark | 2024-12-12 |

| Dave | 2025-03-10 |

| Sudarshan | 2025-07-01 |

+-----------+------------+

create table Test(

testID char(5),

testDate date,

marks int

);

insert into Test Values ("t1", '2024-08-15', 10);

insert into Test Values ("t2", '2023-08-15', 50);

insert into Test Values ("31", '2024-09-15', 70);

select \* from Test;

select max(marks) as maxMark,

min(marks) as minMark,

avg(marks) as aveMark

from test

where testDate >= DATE\_ADD(current\_date, INTERVAL -1 year);

output

+--------+------------+-------+

| testID | testDate | marks |

+--------+------------+-------+

| t1 | 2024-08-15 | 10 |

| t2 | 2023-08-15 | 50 |

| 31 | 2024-09-15 | 70 |

+--------+------------+-------+

+---------+---------+---------+

| maxMark | minMark | aveMark |

+---------+---------+---------+

| 70 | 10 | 40.0000 |

+---------+---------+---------+

create table hyderabad(

empID Integer primary key,

empName varchar(20) not null,

project char(20)

);

create table bangalore(

empID Integer primary key,

empName varchar(20) not null,

project char(20)

);

insert into hyderabad (empId, empName, project) values

(1001, "aruna", "java");

insert into hyderabad (empId, empName, project) values

(1002, "arun", "cloud");

insert into hyderabad (empId, empName, project) values

(1003, "arjun", "java");

insert into bangalore (empId, empName, project) values

(1001, "aruna", "java");

insert into bangalore (empId, empName, project) values

(2001, "archana", "BI");

insert into bangalore (empId, empName, project) values

(2002, "arjun", "java");

insert into bangalore (empId, empName, project) values

(2003, "aman", "cloud");

select h.empName, b.empName, b.project from

hyderabad as h inner join bangalore as b on

h.project = b.project

where h.project = 'java';

+---------+---------+---------+

| empName | empName | project |

+---------+---------+---------+

| arjun | aruna | java |

| aruna | aruna | java |

| arjun | arjun | java |

| aruna | arjun | java |

+---------+---------+---------+

select h.empName, b.empName, b.project from

hyderabad as h inner join bangalore as b on

h.project = b.project;

+---------+---------+---------+

| empName | empName | project |

+---------+---------+---------+

| arjun | aruna | java |

| aruna | aruna | java |

| arjun | arjun | java |

| aruna | arjun | java |

| arun | aman | cloud |

+---------+---------+---------+

CREATE TABLE EMPLOYEE (

empId INTEGER PRIMARY KEY,

name varchar(20) DEFAULT 'no name',

dept varchar(10) DEFAULT 'general',

salary integer DEFAULT 0

);

insert into employee (empId, name) values (1, "aruna");

insert into employee values (2, "archana", "mba", 100000);

insert into employee values

(3, "amala", "mca", 120000),

(4, "amar", "DS", 110000),

(5, "akbar", "IT", 115000);

insert into employee (empId, name) values (6, "John");

update employee

set dept = "HR", salary = 75000

where empId = 1 or empId = 6;

select \* from employee;

+-------+---------+------+--------+

| empId | name | dept | salary |

+-------+---------+------+--------+

| 1 | aruna | HR | 75000 |

| 2 | archana | mba | 100000 |

| 3 | amala | mca | 120000 |

| 4 | amar | DS | 110000 |

| 5 | akbar | IT | 115000 |

| 6 | John | HR | 75000 |

+-------+---------+------+--------+

delete -- deleting specific rows using the condition

truncate -- deleting all the rows and the schema will remains after truncating

drop --- deleting entire table(schema will also be deleted)

delete from employee where dept='hr';

select \* from employee;

truncate table employee;

desc employee;

drop table employee;

select \* from employee;

Output:

+-------+---------+------+--------+

| empId | name | dept | salary |

+-------+---------+------+--------+

| 1 | aruna | HR | 75000 |

| 2 | archana | mba | 100000 |

| 3 | amala | mca | 120000 |

| 4 | amar | DS | 110000 |

| 5 | akbar | IT | 115000 |

| 6 | John | HR | 75000 |

+-------+---------+------+--------+

after deleting the rows whose dept=hr

+-------+---------+------+--------+

| empId | name | dept | salary |

+-------+---------+------+--------+

| 2 | archana | mba | 100000 |

| 3 | amala | mca | 120000 |

| 4 | amar | DS | 110000 |

| 5 | akbar | IT | 115000 |

+-------+---------+------+--------+

truncating only schema remains

+--------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+--------+-------------+------+-----+---------+-------+

| empId | int | NO | PRI | NULL | |

| name | varchar(20) | YES | | no name | |

| dept | varchar(10) | YES | | general | |

| salary | int | YES | | 0 | |

+--------+-------------+------+-----+---------+-------+

dropped the table no data remains everything will be deleted

ERROR 1146 (42S02) at line 29: Table 'sandbox\_db.employee' doesn't exist

-- atomocity

start transaction;

update employee set salary=salary-20000 where empid=3;

update employee set salary=salary+20000 where empid=1;

commit;

select \* from employee;

+-------+---------+------+--------+

| empId | name | dept | salary |

+-------+---------+------+--------+

| 1 | aruna | HR | 95000 |

| 2 | archana | mba | 100000 |

| 3 | amala | mca | 100000 |

| 4 | amar | DS | 110000 |

| 5 | akbar | IT | 115000 |

| 6 | John | HR | 75000 |

+-------+---------+------+--------+

primary key -- unique no dupliactes allowed and not null,a table contrains single primary key

unique key -- no duplicates null values are allowed,a table contains multiple unique keys

CREATE TABLE EMPLOYEE (

empId INTEGER AUTO\_INCREMENT primary key,

name varchar(20) DEFAULT 'no name',

email varchar(50) unique not null,

dept varchar(10) DEFAULT 'general',

salary integer

);

INSERT INTO employee (name, email, salary) VALUES

("aruna", "aruna@gmail.com", 10000),

("archana", "archana@yahoo.com", 12000),

("amalla", "amala@mits.edu", 199999),

("amalla", "r@mits.com", 199999);

select \* from employee;

Output:

+-------+---------+-------------------+---------+--------+

| empId | name | email | dept | salary |

+-------+---------+-------------------+---------+--------+

| 1 | aruna | aruna@gmail.com | general | 10000 |

| 2 | archana | archana@yahoo.com | general | 12000 |

| 3 | amalla | amala@mits.edu | general | 199999 |

| 4 | amalla | r@mits.com | general | 199999 |

+-------+---------+-------------------+---------+--------+

ALTER TABLE employee RENAME to emp;

select \* from emp;

ALTER TABLE emp RENAME column salary to remuneration;

select \* from emp;

alter table emp modify remuneration decimal(10,2);

select \* from emp;

alter table emp add mob integer default 99999 ;

select \* from emp;

alter table emp drop column email;

select \* from emp;

Output:

+-------+---------+-------------------+---------+--------+

| empId | name | email | dept | salary |

+-------+---------+-------------------+---------+--------+

| 1 | aruna | aruna@gmail.com | general | 10000 |

| 2 | archana | archana@yahoo.com | general | 12000 |

| 3 | amalla | amala@mits.edu | general | 199999 |

+-------+---------+-------------------+---------+--------+

+-------+---------+-------------------+---------+--------------+

| empId | name | email | dept | remuneration |

+-------+---------+-------------------+---------+--------------+

| 1 | aruna | aruna@gmail.com | general | 10000 |

| 2 | archana | archana@yahoo.com | general | 12000 |

| 3 | amalla | amala@mits.edu | general | 199999 |

+-------+---------+-------------------+---------+--------------+

+-------+---------+-------------------+---------+--------------+

| empId | name | email | dept | remuneration |

+-------+---------+-------------------+---------+--------------+

| 1 | aruna | aruna@gmail.com | general | 10000.00 |

| 2 | archana | archana@yahoo.com | general | 12000.00 |

| 3 | amalla | amala@mits.edu | general | 199999.00 |

+-------+---------+-------------------+---------+--------------+

+-------+---------+-------------------+---------+--------------+-------+

| empId | name | email | dept | remuneration | mob |

+-------+---------+-------------------+---------+--------------+-------+

| 1 | aruna | aruna@gmail.com | general | 10000.00 | 99999 |

| 2 | archana | archana@yahoo.com | general | 12000.00 | 99999 |

| 3 | amalla | amala@mits.edu | general | 199999.00 | 99999 |

+-------+---------+-------------------+---------+--------------+-------+

+-------+---------+---------+--------------+-------+

| empId | name | dept | remuneration | mob |

+-------+---------+---------+--------------+-------+

| 1 | aruna | general | 10000.00 | 99999 |

| 2 | archana | general | 12000.00 | 99999 |

| 3 | amalla | general | 199999.00 | 99999 |

+-------+---------+---------+--------------+-------+

CREATE TABLE EMPLOYEE (

empId INTEGER AUTO\_INCREMENT primary key,

name varchar(20) DEFAULT 'no name',

email varchar(50),

dept varchar(10) DEFAULT 'general',

salary integer

);

insert into employee (name, email, salary, dept) values

("aruna", "aruna@gmail.com", 100000, 'HR'),

("archana", "archana@yahoo.com", 120000, "HR"),

("arun", "arun@yahoo.com", 80000, "HR"),

("amala", "amala@mits.edu", 199999, "IT"),

("anthony", "anthony@mits.edu", 100000, "IT"),

("akbar", "akbar@mits.edu", 120000, "IT");

select \* from employee;

select name, salary from employee

where salary > (select avg(salary) from employee);

select name, salary from employee

where salary < (select avg(salary) from employee);

Output:

+-------+---------+-------------------+------+--------+

| empId | name | email | dept | salary |

+-------+---------+-------------------+------+--------+

| 1 | aruna | aruna@gmail.com | HR | 100000 |

| 2 | archana | archana@yahoo.com | HR | 120000 |

| 3 | arun | arun@yahoo.com | HR | 80000 |

| 4 | amala | amala@mits.edu | IT | 199999 |

| 5 | anthony | anthony@mits.edu | IT | 100000 |

| 6 | akbar | akbar@mits.edu | IT | 120000 |

+-------+---------+-------------------+------+--------+

salary>avg(sal)

+---------+--------+

| name | salary |

+---------+--------+

| archana | 120000 |

| amala | 199999 |

| akbar | 120000 |

+---------+--------+

salary<avg(sal)

+---------+--------+

| name | salary |

+---------+--------+

| aruna | 100000 |

| arun | 80000 |

| anthony | 100000 |

+---------+--------+

where -- individual rows

having -- aggregated rows(groups)