```
show databases;
show tables;
delimiter $$

delimiter ;



create database vita;
use vita;

use testdb;
desc dept;

select * from dept;



use vita;

create table dept
(deptid int,
deptname varchar(100));



select * from dept;

insert into dept(deptid,deptname)
values(1,'HR');

insert into dept
values(2,'IT');



insert into dept
values(3,'Finance'),
(4,'OPR');



select current_date();
```

```
select curdate();

select current_time();
select curtime();

select now();

select * from dept;

select deptname,deptid from dept;

select * from dept where deptid=1;
select * from dept where deptid!=1;
select * from dept where deptid<>1;


select * from dept where deptid>1;
select * from dept where deptid>=1;

select * from dept where deptid<2;
select * from dept where deptid<=2;

select * from dept where deptname<'HR';

select * from dept where deptname<='HR';

select * from dept where deptname>='HR';

select * from dept where deptid between 1 and 3;

select * from dept where deptid not between 1 and 3;

insert into dept values(5,null);

insert into dept(deptid) values(6);

-- null means value not yet defined

select * from dept where deptname is null;

select * from dept where deptname is not null;

select * from dept where deptname like 'H%';
```

```
select * from dept where deptname like 'H_';

select * from dept where deptname like '%R';

select * from dept where deptname like '_R';

select * from dept where deptname like '__R';

select * from dept where deptname not like '__R';


update dept
set deptname='Supply Chain'
where deptid=5;

update dept
set deptname='Supply Chain' ,deptid=5
where deptid=5;


update dept
set deptname='Supply Chain' ,deptid=5
where deptid=5 and deptname='Supply Chain';


delete from dept where deptname='Supply Chain';



alter table dept add c1 int;

alter table dept drop column c1;

alter table dept add c1 int;


update dept set c1=deptid+1;
alter table dept modify column c1 varchar(100);
alter table dept modify column c1 varchar(100) not null;
```

```
insert into dept values(7,'New',null); -- fails


insert into dept(deptid,deptname) values(8,'D');  -- fails
alter table dept modify column c1 varchar(100) not null default 'Something';
insert into dept(deptid,deptname) values(8,'D');  -- works
alter table dept rename to department;
alter table department rename to dep;
alter table dep rename to dept;

alter table dept rename column c1 to col1;


create table t_dt(id int, dt date);


insert into t_dt values(1,'2022-12-30');


-- unique key


create table t_uk(id int unique , name varchar(100));
insert into t_uk values(1,'a'); -- works

insert into t_uk values(1,'b'); -- fails

insert into t_uk values(null,'b'); -- works
insert into t_uk values(null,'b'); -- works




-- composite unique key
create table t_uk_comp
(c1 int, c2 int, c3 varchar(100), unique(c1,c2));

insert into t_uk_comp values(1,1,'a'); -- works

insert into t_uk_comp values(1,2,'b'); -- works

insert into t_uk_comp values(2,2,'c'); -- works
```

```
insert into t_uk_comp values(2,2,'d'); -- fails


insert into t_uk_comp values(2,null,'d'); -- works

insert into t_uk_comp values(2,null,'e'); -- works



-- multi unique keys

create table t_uk_multi(c1 int unique, c2 int unique , c3 varchar(100));

insert into t_uk_multi values(1,1,'a'); -- works

insert into t_uk_multi values(1,1,'b'); -- fails
insert into t_uk_multi values(1,2,'b'); -- fails




select table_name,column_name,COLUMN_KEY from information_schema.columns
where table_name='t_uk_comp' and table_schema='vita';


select * from information_schema.KEY_COLUMN_USAGE
where table_name='t_uk_comp' and table_schema='vita';

show create table t_uk_comp;


create table t_no_unq(c1 int, c2 int, c3 varchar(100));

insert into t_no_unq values(1,1,'a');
insert into t_no_unq values(1,1,'a');


alter table t_no_unq add unique(c1); -- fails

delete from t_no_unq limit 1;


alter table t_no_unq add unique(c1,c2);
```

```
-- primary key


create table t_pk (id int primary key, name varchar(100));

insert into t_pk values(1,'a'); -- works
insert into t_pk values(1,'b'); -- fails
insert into t_pk values(null,'b'); -- fails
insert into t_pk values(2,null); -- works

--Comp primary key

create table t_comp_pk (c1 int, c2 int, name varchar(100), primary key(c1,c2));

--Multiple primary key

create table t_pk_multi (c1 int primary key, c2 int primary key, name varchar(100));
-- ERROR 1068 (42000): Multiple primary key defined

-- null constraint

create table t_nn(id int, name varchar(100) not null);


insert into t_nn values(1,null); -- fails
insert into t_nn(id) values(1); -- fails


-- check constraint

create table t_check (eid int, ename varchar(100), salary int check(salary>0));


insert into t_check values(1,'a',0); -- fails

insert into t_check values(1,'a',1000); -- works

insert into t_check values(1,'a',null); -- works
```

```
create table t_check2 (eid int, ename varchar(100), gender varchar(10) check(gender in
('Male','Female')));


insert into t_check2 values(1,'a',null); -- works
insert into t_check2 values(1,'a','male'); -- works
insert into t_check2 values(1,'a','male2'); -- fails




create table t_check_comp
(eid int, ename varchar(100), salary int, gender varchar(100),
check((salary<=100000 and gender='Male') or (salary<=200000 and gender='Female')));


insert into t_check_comp values(1,'a',200000,'Male'); -- fails

insert into t_check_comp values(1,'a',200000,'Female'); -- works



-- foreign key


create table t_parent(deptid int primary key , deptname varchar(100));

create table t_child(eid int, ename varchar(100),
deptid int,
foreign key fk1(deptid) references t_parent(deptid));


insert into t_child values(1,'a',1); -- fails

insert into t_child values(1,'a',null); -- works
insert into t_parent values(1,'HR');

insert into t_child values(1,'a',1); -- works

delete from t_parent where deptid=1; -- fails

delete from t_parent ; -- fails
```

```
-- comp FK

create table t_parent_comp
(c1 int , c2 int, c3 varchar(100), unique (c1,c2));

create table t_child_comp
(c1 int,c2 int, c4 varchar(100),
foreign key fk2(c1,c2) references t_parent_comp(c1,c2));


insert into t_child_comp valueS(1,1,'a'); -- fails

insert into t_child_comp valueS(1,null,'a'); -- works

insert into t_child_comp valueS(null,1,'a'); -- works

insert into t_child_comp valueS(null,null,'a'); -- works


-- default


create table t_def (id int , salary int default 10);

insert into t_def(id) values(1);

insert into t_def(id,salary) values(1,null);


create table t_name_def(id int, name varchar(100) default 'NA');

insert into t_name_def(id) values(1);

select * from t_name_def;

 grant select on dept to u1@localhost;

revoke select on dept from u1@localhost;


show variables like '%commit%';
```

```
start transaction;
update dept set col1=300 where deptid=1;
rollback;

-- Joins

create table t1 (c1 int, c2 varchar(10));
create table t2 (c1 int, c3 varchar(10));

insert into t1 values(1,'a'),(2,'b'),(3,'c');
insert into t2 values(3,'x'),(4,'y'),(5,'z');

-- cross join
-- old syntax

select * from t1, t2;

-- new/Ansi Syntax


select * from t1 cross join t2;

select c1,c2,c3 from t1 cross join t2; -- fails

select t1.c1,c2,c3 from t1 cross join t2; -- works

-- inner join

-- old syntax

select * from t1, t2 where t1.c1=t2.c1;

-- new/Ansi Syntax


select * from t1 inner join t2 on t1.c1=t2.c1;

select * from t1 join t2 on t1.c1=t2.c1;

select c1,c2,c3 from t1  join t2 on t1.c1=t2.c1; -- fails
```

```
select t1.c1,c2,c3 from t1 join t2 on t1.c1=t2.c1; -- works
```

left join

-- new/Ansi Syntax

```
select * from t1 left outer join t2 on t1.c1=t2.c1;

select * from t1  left join t2 on t1.c1=t2.c1;

select c1,c2,c3 from t1  left join t2 on t1.c1=t2.c1; -- fails

select t1.c1,c2,c3 from t1 left join t2 on t1.c1=t2.c1; -- works
```

Right join

-- new/Ansi Syntax

```
select * from t1 Right outer join t2 on t1.c1=t2.c1;

select * from t1  Right join t2 on t1.c1=t2.c1;

select c1,c2,c3 from t1  Right join t2 on t1.c1=t2.c1; -- fails

select t1.c1,c2,c3 from t1 Right join t2 on t1.c1=t2.c1; -- works



select ename,deptname
from t_parent, t_child
where t_parent.deptid=t_child.deptid;



select ename,deptname
from t_parent right join t_child
on t_parent.deptid=t_child.deptid;
```

```
create table a (c1 int);
create table b (c1 int);

insert into a values(1),(2),(3);
insert into b values(3),(4),(5);

select * from a
union
select * from b;

select * from b
union
select * from a;


select * from a
union all
select * from b;

select * from b
union all
select * from a;

-- full join workround in mysql
select * from t1 left outer join t2 on t1.c1=t2.c1
union
select * from t1 Right outer join t2 on t1.c1=t2.c1;


select count(*) from t_parent;
select count(*) from t_child;
select count(1) from t_child;

select count(deptid) from t_child;

select sum(eid) from t_child;

select max(eid) from t_child;
select min(ename) from t_child;

select avg(eid) from t_child;
```

```
drop table dept;



create table dept (deptid int , deptname varchar(100));
insert into dept values ( 1, 'HR'),
( 2, 'IT'),
( 3, 'Finance'),
( 4, 'OPR');

create table emp(eid int, ename varchar(100), salary int, deptid int);

select version();

insert into emp
with recursive tab
as
(select 1 eid , cast(concat('a',1) as char(100)) ename,rand()*100000 salary,mod(0,3)+1 deptid
union all
select  eid+1 , concat('a',eid+1) ename,rand()*100000 salary,mod(eid,3)+1 deptid
from tab where eid<100)
select * from tab;


select  deptname, sum(salary) dep_total_salary from emp join dept on emp.deptid=dept.deptid
group by deptname;



select  deptname, sum(salary) as dep_total_salary from emp join dept on
emp.deptid=dept.deptid
group by deptname;


select  deptname, sum(salary) as dep_total_salary from emp right join dept on
emp.deptid=dept.deptid
group by deptname;


select  deptname, sum(salary) as dep_total_salary from emp e right join dept d on
e.deptid=d.deptid
group by deptname;
```

```
-- intersect workaround
select * from a where c1 in (select * from b);

-- Minus workaround
select * from a where c1  not in (select * from b);

select * from b where c1  not in (select * from a);
```

```
select  deptname, sum(salary) as dep_total_salary from emp e right join dept d on
e.deptid=d.deptid
group by deptname
having sum(salary)>1800000;
```