------create a table category to store (cid,cname,description)

product(pid,pname, exp\_dt,price,qty,cid,sid)

salesman(sid,sname,address)

constraints

1. pname cannot be null -------- not null
2. cname has to be unique -------unique
3. sname cannot be null
4. qty > 0 ---------------------check
5. price default 100.00 ---------- default

create table category(

cid int primary key,

cname varchar(20) unique,

description varchar(20));

unique allows to add any number of null value but not null values cannot be duplicate

create table salesman(

sid int primary key,

sname varchar(20) not null,

address varchar(30))

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

product(pid,pname,exp\_dt,price,qty,cid,sid)

create table product(

pid int primary key,

pname varchar(20) not null,

exp\_dt date,

price decimal(9,2),

qty int check(qty>0),

cid int references category(cid)

on delete set null

on update cascade,

sid int references salesman(sid)

on delete no action

on update cascade);

--------table level constraint ---🡪 we may write after last filed

primary key ,foreign key,unique,check can be defined as table level constraint.

create table product(

pid int,

pname varchar(20) not null,

exp\_dt date,

price decimal(9,2) default 100.00,

qty int check(qty>0),

cid int,

sid int ,

constraint pk\_pid primary key(pid),

constraint fk\_cid foreign key(cid) references category(cid)

on delete set null

on update cascade,

constraint fk\_sid foreign key(sid)

references salesman(sid)

on delete no action

on update cascade);

example 2

faculty(fid,fname,spsub)

102 trupti database

100 bakul Java

103 tejal HTML

104 ashu Java

room(rid,rname,rloc)

rname unique

10 Mogra 1st floor

20 Jasmin 2nd floor

30 Lotus 1st floor

course(cid,cname,rid,fid)

1 DBDA 10 100

2 DAC 20 102

3 DTISS

cname- unique not null

faculty(fid,fname,spsub)

create table faculty(

fid int primary key,

fname varchar(20) not null,

spsub varchar(30)

)

room(rid,rname,rloc)

create table room(

rid int auto\_increment,

rname varchar(20) unique not null,

rloc varchar(20),

constraint rp\_rid primary key(rid)

);

insert into room values(default,'xxx','vvv');

insert into room(rname) values('xxx');

course(cid,cname,rid,fid)

create table course(

cid int primary key,

cname varchar(20) unique not null,

roomid int,

fid int,

constraint fk\_rid foreign key(roomid) references room(rid)

on delete set null

on update cascade,

constraint fk\_fid foreign key(fid) references faculty(fid)

on delete set null

on update cascade);

subject(fid,spsub)

create table subject(

fid int,

spsub varchar(20),

constraint pk\_fid\_sub primary key(fid,supsub),

constrinst fk\_fid foreign key(fid) references faculty(fid)

)

select \* from emp e

where sal < (select sal

from emp s

where s.empno=e.mgr)

category

|  |  |  |
| --- | --- | --- |
| cid | Cname | description |
| 10 | Chips | ksdk |
|  |  |  |

product

|  |  |  |  |
| --- | --- | --- | --- |
| Pid | Pname | Price | cid |
| 12 | Lays | 2345 | 10 |
| 13 | Marie | 23 | null |
| 14 | Pringles | 120 | 10 |

student-marks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sid | Sname | Marks | sub |  |
| 1 | raj | 95 | dbms |  |
| 2 | Ashu | 98 | dbms |  |
| 3 | rajesh | 98 | dbms |  |
| 1 | raj | 98 | java |  |
| 2 | Ashu | 98 | java |  |

create table student\_marks

(sid int,

sub varchar(20) primary key ,

sname varchar(20) not null,

marks int check (marks>0),

constraint pk\_key primary key(sid,sub));

room

|  |  |  |
| --- | --- | --- |
| rid | Rname | rloc |
| 1 | Rose | First floor |
| 2 | Lotus | Second floor |

create table room(

rid int auto\_increment,

rname varchar(20),

rloc varchar(20),

constraint pk\_key primary key(rid));

insert into room values(default,’Rose’,’ksdjfksdj’)------- rid-🡪1

insert into room(rname,rloc) values(’Rose’,’ksdjfksdj’)------ rid -🡪 2

insert into room values(default,’Rose’,’ksdjfksdj’)-----rid🡪3 ------- if error occurs

insert into room values(default,’Rose’,’ksdjfksdj’)--------rid--🡪4

----------modify structure of table

alter table room auto\_increment=100

alter table

1.add column

2.drop column

3. modify column

4. rename column

3.add constraint

4. delete constraint

5

add new column in the table

alter table room

add details varchar(20) after rname;

--------------To find all constraints on a table

select COLUMN\_NAME, CONSTRAINT\_NAME, REFERENCED\_COLUMN\_NAME, REFERENCED\_TABLE\_NAME

from information\_schema.KEY\_COLUMN\_USAGE

where TABLE\_NAME = 'product';

product

1. add new column reorder in product table after qty column

alter table product

add reorder varchar(20) after qty

1. add new column description in product table before qty column

alter table product

add description varchar(20) before qty

1. to change the size of description column to varchar(50)

alter table product

modify description varchar(50)

1. to change name of description column to pinfo

alter table product

change column description pinfo

1. to delete column reorder column

alter table product

drop column reorder

1. to drop primary key constraint

alter table product

drop primary key;

1. to add primary constraint

alter table product

add primary key(pid)

1. To delete the foreign key constraints

alter table product

drop foreign key fk\_cid;

---------delete the table data and structure

drop table product;

-------delete the table data and you want to keep empty table

truncate table product;

or

delete

from product;

------- to delete all products with price > 20;

delete

from product

where price>20;

-------delete all product with price > price of Maggie;

delete

from product

where price > (select price

from (select \* from product) p

where pname=’maggie’)

----------- increase the price of all products with category 1 by 10%

update product

set price=price+(0.10\*price)

where cid=1;

----------- increase the price of all products by 10% if the category of product is same as marie

update product

set price=price+(0.1\*price)

where cid=(select cid

from (select \* from product) p

where pname=’marie’);

-------- set price of lays same as price of pringles

update product

set price=(select price from (select \* from product) p where pname='50-50')

where pname='lays';

----------change the price to 123 of all products where pname=’good day’

update product

-> set price=123

-> where price>(select price

-> from (select \* from product) p

-> where pname='Good day');

-------- joins in mysql

cross join ---- if you combine data from 2 tables

without any condition then it is cross join

table 1 contains 10 rows

table 2 contains 4 rows

the total rows =10\*4=40

inner join

table 1 contains 10 rows

table 2 contains 4 rows

the total rows =10

------ equi join

-------display all employees with their dept name

select empno,ename,e.deptno,d.deptno,dname

-> from emp e,dept d

-> where e.deptno=d.deptno;

------- non equi

-------- self join

if in join if you combine a table with itself then it is called as self join

outer join

----- left

------ right

------ full outer ------ does not mysql but works in oracle.

------------------ to display empno,ename along with manager no and managers name

select e.empno,e.ename,m.empno "mgrno",m.ename "mgrname"( example self join)

-> from emp e,emp m

-> where e.mgr=m.empno;

----- to display all products along with salaes man name

select pid,pname,p.sid,s.sid,sname

-> from product p,salesman s

-> where p.sid=s.sid;

------ to display pname,cname and salesman name

select pname,cname,sname

-> from product p,category c,salesman s

-> where p.cid=c.cid and p.sid=s.sid;

----to display ename along with grade (non equijoin)

select empno,ename,sal,grade,losal,hisal

-> from salgrade s,emp e

-> where sal between losal and hisal;

example 2

faculty(fid,fname,spsub)

102 trupti database

100 bakul Java

103 tejal HTML

104 ashu Java

room(rid,rname,rloc)

rname unique

10 Mogra 1st floor

20 Jasmin 2nd floor

30 Lotus 1st floor

course(cid,cname,rid,fid)

1 DBDA 10 100

2 DAC 20 102

3 DTISS

---display all courses along with faculty name

select cname,f.fid,f.fname

from faculty f,course c

where f.fid=c.fid;

other syntax

select cname,f.fid,f.fname

from faculty f inner join course c on f.fid=c.fid

---- which room is allocated to each course

select cname,rname

from room r,course c

where r.rid=c.rid;

select cname,rname

from room r inner join course c on r.rid=c.rid;

----to display DAC course along with faculty name,room name

select cname,fname,rname

from faculty f,course c,room r

where f.fid=c.fid and r.rid=c.rid and cname=’DAC;

------ to find all products along with product name, category name with matching and non matchin rows from both tables

select pid,pname,p.cid,c.cid,cname

from product p left join category c on p.cid=c.cid

union

select pid,pname,p.cid,c.cid,cname

from product p right join category c on p.cid=c.cid;