

PES University
Department of Computer Applications
Programme: BCA, Sem: IV

Instructions for the Exercises

- i) Draw conceptual Schema.
- ii) Create the relations using primary key, foreign key and on delete cascade
- iii) Display the structure of the relations.
- iv) Enter minimum 5 rows in each relation.
- v) Display the contents of the relations.
- vi) Perform the queries and the results of the queries may be displayed directly

Exercise-1

doctor (doctor_id, dname, dob, specialization, city)

check-up (docid, patid, cdate, diagnosis, fee)

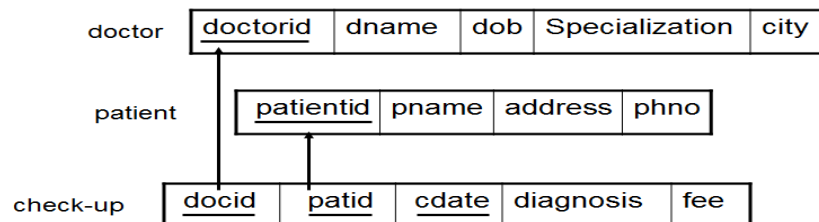
patient (patient_id, pname, address, dob)

Queries to be performed

- a. Find the name, address and birth date of the patients whose name starts with 'r'.
- b. Find the name of the patient, name of doctor, date of check-up and diagnosis.
- c. Display each specialization and number of doctors available for that specialization.
- d. Print the numbers of doctors who have checked Hari, also print average fees.

Solution

Conceptual Schema



Creation of Database

```
mysql> create database ex1;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> use ex1;
Database changed
```

Creation of Tables

```
mysql> create table doctor
  (doctor_id tinyint primary key,
   dname varchar(30) not null,
   dob date not null,
   spec varchar(5) not null,
   city varchar(30) not null);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> create table patient
  (patient_id tinyint primary key,
   pname varchar(30) not null,
   address varchar(50),
   dob date not null);
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> create table check_up
      (doctor_id tinyint,
       patient_id tinyint,
       foreign key (doctor_id) references doctor(doctor_id) on delete cascade,
       foreign key(patient_id) references patient(patient_id) on delete cascade,
       diagnosis varchar(50),
       cdate date not null,
       fee real not null,
       primary key(doctor_id, patient_id, cdate));
Query OK, 0 rows affected (0.00 sec)
```

Structure of Tables

```
mysql> desc doctor;
```

Field	Type	Null	Key	Default	Extra
doctor_id	tinyint(4)	NO	PRI	NULL	
dname	varchar(30)	NO		NULL	
dob	date	NO		NULL	
spec	varchar(5)	NO		NULL	
city	varchar(30)	NO		NULL	

5 rows in set (0.00 sec)

```
mysql> desc patient;
```

Field	Type	Null	Key	Default	Extra
patient_id	tinyint(4)	NO	PRI	NULL	
pname	varchar(30)	NO		NULL	
address	varchar(50)	YES		NULL	
dob	date	NO		NULL	

4 rows in set (0.01 sec)

```
mysql> desc check_up;
```

Field	Type	Null	Key	Default	Extra
doctor_id	tinyint(4)	NO	PRI	0	
patient_id	tinyint(4)	NO	PRI	0	
diagnosis	varchar(50)	YES		NULL	
cdate	date	NO	PRI	NULL	
fee	double	NO		NULL	

5 rows in set (0.00 sec)

Insertion of Tuples

```
mysql> insert into doctor values (11, 'krishna', '1975=01-17', 'physician',
'bangalore');
Query OK, 1 row affected, 1 warning (0.00 sec)
```

```
mysql> insert into patient values(1, 'hari', '507, rrnagar', '1969-08-14');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into check_up values(11, 5, 'viral', '2016-01-11', 200);
Query OK, 1 row affected (0.00 sec)
```

View of Tables

```
mysql> select * from doctor;
```

doctor_id	dname	dob	spec	city
11	krishna	1975-01-17	physi	bangalore
22	rama	1975-02-22	physi	bangalore
33	sita	1980-03-25	eye	bangalore
44	rita	1978-05-04	eye	bangalore
55	john	1969-08-14	ortho	mysore

5 rows in set (0.00 sec)

```
mysql> select * from patient;
```

patient_id	pname	address	dob
1	hari	507, rrnagar	1969-08-14
2	suma	234, jainagar	1992-05-24
3	manav	133, vijaynagar	2001-03-04
4	ishwar	302, kingeri	2021-09-11
5	seema	302, kingeri	2011-09-11
6	hari	500, kingeri	2012-02-10

6 rows in set (0.00 sec)

```
mysql> select * from check_up;
```

doctor_id	patient_id	diagnosis	cdate	fee
11	5	viral	2016-01-11	200
22	4	viral	2016-01-11	200
33	1	infection	2016-01-31	300
44	1	infection	2016-01-31	300
55	2	arthritis	2016-01-12	500
55	6	infection	2016-01-31	700

6 rows in set (0.00 sec)

Queries

Query-i: Find the name, address and birth date of the patients whose name starts with 's'.

```
mysql> select pname, address, dob from patient where pname like 's%';
```

pname	address	dob
suma	234, jainagar	1992-05-24
seema	302, kingeri	2011-09-11

2 rows in set (0.00 sec)

Query-ii: Find the name of the patient, name of doctor, date of check-up and diagnosis.

```
mysql> select pname, dname, cdate, diagnosis
        from doctor d, patient p, check_up c
        where d.doctor_id = c.doctor_id
        and p.patient_id = c.patient_id;
```

pname	dname	cdate	diagnosis
seema	krishna	2016-01-11	viral
ishwar	rama	2016-01-11	viral
hari	sita	2016-01-31	infection
hari	rita	2016-01-31	infection
suma	john	2016-01-12	arthritis
hari	john	2016-01-31	infection

6 rows in set (0.00 sec)

Query-iii: Display each specialization and number of doctors available for that specialization.

```
mysql> select spec, count(doctor_id) from doctor group by spec;
```

spec	count(doctor_id)
eye	2
ortho	1
physi	2

3 rows in set (0.00 sec)

Query-iv: Print the numbers of doctors who have checked Hari, also print average fees.

```
mysql> select count(c.doctor_id),
        avg(fee) from doctor d, patient p, check_up c
        where d.doctor_id = c.doctor_id
        and p.patient_id = c.patient_id
        and pname = 'hari'
        group by c.patient_id;
```

count(c.doctor_id)	avg(fee)
2	300
1	700

2 rows in set (0.00 sec)

Exercise-2

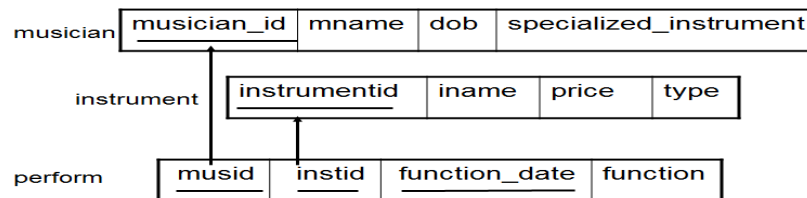
musician (musician_id, mname, dob, specialized_instrument)
perform (musid, instid, function_date, function,)
instrument (instrument_id, iname, price, type)

Queries to be performed

- Find the name and price of the string type instruments.
- Display the names of instruments along with their price which were used in New Year function.
- Display names of musicians, their specialized instrument and function held after 2005.
- Print the name of instrument for which number of musicians specialized is more than 1.

Solution

Conceptual Schema



Creation of Database

```
mysql> create database ex2;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> use ex2;
Database changed
```

Creation of Tables

```
mysql> create table musician
(musician_id tinyint primary key,
mname varchar(30) not null,
dob date not null,
spec_inst varchar(30));
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> create table instrument
(instrument_id tinyint primary key,
iname varchar(30) not null,
price real not null,
type varchar(30) not null);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> create table perform
(musician_id tinyint,
instrument_id tinyint,
function varchar(50),
fdate date not null,
foreign key (musician_id) references musician on delete cascade,
foreign key (instrument_id) references instrument on delete cascade,
primary key(musician_id, instrument_id, fdate));
Query OK, 0 rows affected (0.00 sec)
```

Structure of Tables

```
mysql> desc musician;
```

Field	Type	Null	Key	Default	Extra
musician_id	tinyint(4)	NO	PRI	NULL	
mname	varchar(30)	NO		NULL	
dob	date	NO		NULL	
spec_inst	varchar(30)	YES		NULL	

```
4 rows in set (0.00 sec)
```

```
mysql> desc instrument;
```

Field	Type	Null	Key	Default	Extra
instrument_id	tinyint(4)	NO	PRI	NULL	
iname	varchar(30)	NO		NULL	
price	double	NO		NULL	
type	varchar(30)	NO		NULL	

```
4 rows in set (0.01 sec)
```

```
mysql> desc perform;
```

Field	Type	Null	Key	Default	Extra
musician_id	tinyint(4)	NO	PRI	0	
instrument_id	tinyint(4)	NO	PRI	0	
function	varchar(50)	YES		NULL	
fdate	date	NO	PRI	NULL	

```
4 rows in set (0.00 sec)
```

Insertion of Tuples

```
mysql> insert into musician values(1, 'pt.hariprasad', '1980-02-15', 'bansuri');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into instrument values(11, 'sitar', 35000, 'string');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into perform values(3, 11, 'new-year', '2015-01-21');
Query OK, 1 row affected (0.00 sec)
```

View of Tables

```
mysql> select * from musician;
```

musician_id	mname	dob	spec_inst
1	pt.hariprasad	1980-02-15	bansuri
2	pt.zakir	1981-04-25	tabla
3	pt.ravishankar	1965-03-21	sitar
4	pt.shivkumar	1975-06-01	santoor
5	pt.bismillah	1978-08-08	shahnai
6	pt.rama	1970-08-08	sitar

```
6 rows in set (0.00 sec)
```

```
mysql> select * from instrument;
```

instrument_id	iname	price	type
11	sitar	35000	string
22	tabla	30000	percussion
33	bansuri	20000	wind
44	flute	25000	wind
55	drums	25000	percussion

5 rows in set (0.00 sec)

```
mysql> select * from perform;
```

musician_id	instrument_id	function	fdate
1	33	new-year	2015-01-01
1	33	convocation	2015-01-18
2	22	new-year	2015-02-01
3	11	new-year	2014-01-01
3	11	new-year	2015-01-21

5 rows in set (0.00 sec)

Queries

Query-i: Find the name and price of the string type instruments.

```
mysql> select iname, price from instrument where type = 'string';
```

iname	price
sitar	35000

1 row in set (0.00 sec)

Query-ii: Display the names of instruments along with their price which were used in New Year function.

```
mysql> select distinct iname, price from instrument i, perform p
       where p.instrument_id = i.instrument_id
       and function = 'new-year';
```

iname	price
sitar	35000
sitar	35000
tabla	30000
bansuri	20000

4 rows in set (0.00 sec)

Query-iii: Display names of musicians, their specialized instrument and function held after 2014.

```
mysql> select mname, spec_inst, function
       from musician m, perform p
       where p.musician_id = m.musician_id
       and fdate > '2014-12-31';
```

mname	spec_inst	function
pt.hariprasad	bansuri	new-year
pt.hariprasad	bansuri	convocation
pt.zakir	tabla	new-year
pt.ravishankar	sitar	new-year

4 rows in set (0.27 sec)

Query-iv: Print the name of instrument for which number of musicians specialized is more than 1.

```
mysql> select spec_inst from musician m
        group by spec_inst
        having count(distinct(m.musician_id))>1;
```

spec_inst
sitar

1 row in set (0.01 sec)

Exercise-3

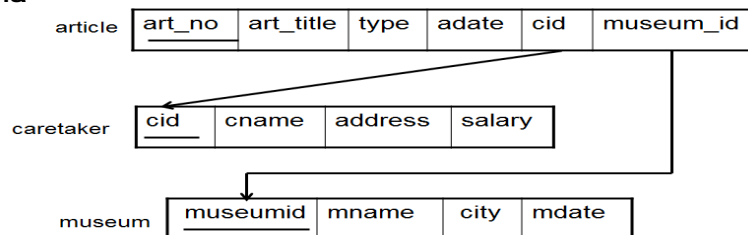
article (art_no, art_title, type, adate, cid, museum_id)
caretaker (cid, cname, address, salary)
museum (museum_id, mname, city, mdate)

Queries to be performed

- Print the details of articles which are cared by person living in Delhi.
- Find the details of care takers taking care of more than 2 articles.
- Print the details of museum which has paintings and located in Hyderabad.
- List the museum name, article title and name of the caretaker taking care of those articles.

Solution

Conceptual Schema



Creation of Database

```
mysql> create database ex3;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> use ex3;
Database changed
```

Creation of Tables

```
mysql> create table caretaker
  (cid tinyint primary key,
   cname varchar(30) not null,
   address varchar(50),
   salary numeric(8,2));
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> create table museum
  (museum_id tinyint primary key,
   mname varchar(30) not null,
   city varchar(50) not null,
   mdate date not null);
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> create table article
  (art_no tinyint primary key,
   art_title varchar(30) not null,
   type varchar(20) not null,
   adate date not null,
   cid tinyint,
   museum_id tinyint,
   foreign key (cid) references caretaker (cid) on delete cascade,
   foreign key (museum_id) references museum (museum_id) on delete cascade);
Query OK, 0 rows affected (0.00 sec)
```

Structure of Tables

```
mysql> desc caretaker;
```

Field	Type	Null	Key	Default	Extra
cid	tinyint(4)	NO	PRI	NULL	
cname	varchar(30)	NO		NULL	
address	varchar(50)	YES		NULL	
salary	decimal(8,2)	YES		NULL	

4 rows in set (0.00 sec)

```
mysql> desc museum;
```

Field	Type	Null	Key	Default	Extra
museum_id	tinyint(4)	NO	PRI	NULL	
mname	varchar(30)	NO		NULL	
city	varchar(50)	NO		NULL	
mdate	date	NO		NULL	

4 rows in set (0.01 sec)

```
mysql> desc article;
```

Field	Type	Null	Key	Default	Extra
art_no	tinyint(4)	NO	PRI	NULL	
art_title	varchar(30)	NO		NULL	
type	varchar(20)	NO		NULL	
adate	date	NO		NULL	
cid	tinyint(4)	YES		NULL	
museum_id	tinyint(4)	YES		NULL	

6 rows in set (0.00 sec)

Insertion of Tuples

```
mysql> insert into caretaker values (1, 'ram', 'rr nagar, bangalore', 15000);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into museum values (11, 'salarjung', 'hyderabad', '1970-01-01');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into article values (101, 'sitar', 'instrument', '1990-05-02', 1,
22);
Query OK, 1 row affected (0.00 sec)
```

View of Tables

```
mysql> select * from caretaker;
```

cid	cname	address	salary
1	ram	rr nagar, bangalore	15000.00
2	shyam	vijay nagar, hyderabad	14000.00
3	john	jai nagar, mysore	20000.00
4	meena	mg road, hyderabad	10000.00
5	seema	bansankari, delhi	12000.00

5 rows in set (0.00 sec)

mysql> select * from museum;

museum_id	mname	city	mdate
11	salarjung	hyderabad	1970-01-01
22	gandhi smriti	delhi	1972-11-11
33	national science centre	delhi	1972-11-11
44	visvesvaraya	bangalore	1980-09-21
55	mural art	thrissur	1985-05-26

5 rows in set (0.00 sec)

mysql> select * from article;

art_no	art_title	type	adate	cid	museum_id
101	sitar	instrument	1990-05-02	1	22
102	portrait	painting	1991-06-02	4	11
103	chair	furniture	1991-04-12	3	44
104	table	furniture	1991-04-29	3	44
105	jacket	cloths	1988-04-29	2	11
106	necklace	jewellery	1988-04-29	3	44

6 rows in set (0.00 sec)

Queries

Query-i: Print the details of articles which are cared by person living in Bangalore.

mysql> select * from article where cid in
(select cid from caretaker where address like '%bangalore');

art_no	art_title	type	adate	cid	museum_id
101	sitar	instrument	1990-05-02	1	22

1 row in set (0.00 sec)

Query-ii: Find the details of care takers taking care of more than 2 articles.

mysql> select * from caretaker where cid in
(select cid from article
group by cid having count(art_no)>2);

cid	cname	address	salary
3	john	jai nagar, mysore	20000.00

1 row in set (0.01 sec)

Query-iii: Print the details of museum which has paintings and located in Hyderabad.

```
mysql> select * from museum where city = 'hyderabad'
      and museum_id in
      (select museum_id from article where type = 'painting');
```

museum_id	mname	city	mdate
11	salarjung	hyderabad	1970-01-01

1 row in set (0.00 sec)

Query-iv: List the museum name, article title and name of the caretaker taking care of those articles.

```
mysql> select mname, art_title, cname
      from museum m, caretaker c, article a
      where m.museum_id = a.museum_id
      and c.cid = a.cid;
```

mname	art_title	cname
gandhi smriti	sitar	ram
salarjung	jacket	shyam
visvesvaraya	chair	john
visvesvaraya	table	john
visvesvaraya	necklace	john
salarjung	portrait	meena

6 rows in set (0.00 sec)

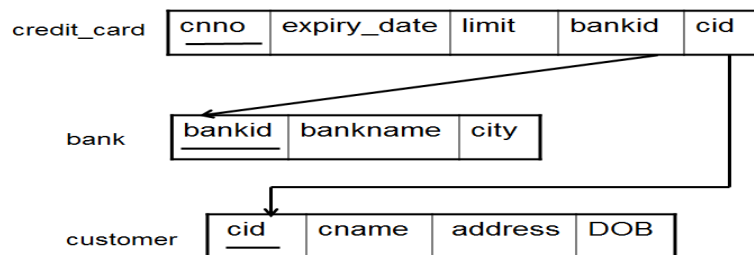
Exercise-4

credit_card (ccno, expiry_date, limit, bankid, cid)
bank (bankid, bankname, city)
customer (cid, cname, address, DOB)

Queries to be performed

- Display the details of bank having India in its name.
- Find the customer names and address who have cards from the bank present in Delhi.
- Print the total number of cards as '_Total-Cards', minimum limit as '_Min-Limit' and maximum limit as '_Max-Limit' of those cards
- Find the name of bank which has issued more than 3 cards.

Conceptual Schema



Creation of Database

```
mysql> create database ex4;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> use ex4;
Database changed
```

Creation of Tables

```
mysql> create table bank
    (bankid tinyint primary key,
    bankname varchar(30) not null,
    city varchar(30) not null);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> create table customer
    (cid tinyint primary key,
    cname varchar(30) not null,
    address varchar(50),
    dob date not null);
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> create table credit_card
    (ccno int primary key,
    expiry_date date not null,
    climit numeric(6) not null,
    bankid tinyint,
    cid tinyint,
    foreign key (bankid) references bank on delete cascade,
    foreign key (cid) references customer on delete cascade);
Query OK, 0 rows affected (0.00 sec)
```

Structure of Tables

```
mysql> desc bank;
```

Field	Type	Null	Key	Default	Extra
bankid	tinyint(4)	NO	PRI	NULL	
bankname	varchar(30)	NO		NULL	
city	varchar(30)	NO		NULL	

3 rows in set (0.00 sec)

```
mysql> desc customer;
```

Field	Type	Null	Key	Default	Extra
cid	tinyint(4)	NO	PRI	NULL	
cname	varchar(30)	NO		NULL	
address	varchar(50)	YES		NULL	
dob	date	NO		NULL	

4 rows in set (0.00 sec)

```
mysql> desc credit_card;
```

Field	Type	Null	Key	Default	Extra
ccno	int(11)	NO	PRI	NULL	
exppiry_date	date	NO		NULL	
climit	decimal(6,0)	NO		NULL	
bankid	tinyint(4)	YES		NULL	
cid	tinyint(4)	YES		NULL	

5 rows in set (0.00 sec)

Insertion of Tuples

```
mysql> insert into bank values(1, 'State Bank of India', 'bangalore');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into customer values(11, 'ram', 'mg road, delhi', '1992-01-01');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into credit_card values(106, '2018-07-06', 160000, 3, 55);
Query OK, 1 row affected (0.00 sec)
```

View of Tables

```
mysql> select * from bank;
```

bankid	bankname	city
1	State Bank of India	bangalore
2	Union Bank India	bangalore
3	Vijya	mysore
4	syndicate	mysore
5	punjab	delhi

5 rows in set (0.00 sec)

```
mysql> select * from customer;
```

cid	cname	address	dob
11	ram	mg road, delhi	1992-01-01
22	shyam	rr road, bangalore	1990-11-21
33	sita	btm, mysore	1980-09-16
44	nita	abc, mysore	1985-08-26
55	john	phase-1, hyderabad	1988-02-23

```
5 rows in set (0.00 sec)
```

```
mysql> select * from credit_card;
```

ccno	exppiry_date	climit	bankid	cid
101	2020-01-01	200000	1	33
102	2025-01-01	100000	1	22
103	2022-11-11	150000	1	11
104	2022-11-11	150000	1	44
105	2021-08-21	150000	5	44
106	2018-07-06	160000	3	55

```
6 rows in set (0.00 sec)
```

Queries

Query-i: Display the details of bank having India in its name.

```
mysql> select * from bank where bankname like '%India%';
```

bankid	bankname	city
1	State Bank of India	bangalore
2	Union Bank India	bangalore

```
2 rows in set (0.00 sec)
```

Query: Find the customer names and address who have cards from the bank present in Delhi.

```
mysql> select cname, address from customer
       where cid in (select cid from credit_card where bankid in
                    (select bankid from bank where city = 'delhi'));
```

cname	address
nita	abc, mysore

```
1 row in set (0.00 sec)
```

Query-iii: Print the total number of cards as 'Total-Cards', minimum limit as 'Min-Limit' and maximum limit as 'Max-Limit' of those cards.

```
mysql> select count(ccno) "Total-Cards",
               min(climit) " Min-Limit",
               max(climit) "Max-Limit" from credit_card;
```

```
+-----+-----+-----+
| Total-Cards | Min-Limit | Max-Limit |
+-----+-----+-----+
|          6 |    100000 |    200000 |
+-----+-----+-----+
1 row in set, 1 warning (0.00 sec)
```

Query-iv: Find the name of bank which has issued more than 3 cards.

```
mysql> select bankname from bank where bankid in
       (select bankid from credit_card
        group by bankid having count(ccno)>3);
```

```
+-----+
| bankname          |
+-----+
| State Bank of India |
+-----+
1 row in set (0.00 sec)
```


Exercise-5

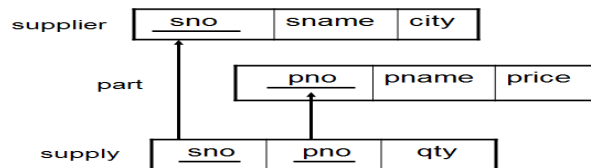
supplier (Sno, Sname, city)
part (Pno, Pname, Price)
supply (Sno, Pno, qty)

Queries to be performed

- Display average and sum of price of all the parts.
- Display the price of those parts for which name ends with 't';
- Display the part details of part which are supplied by supplier lives in city 'Bangalore'.
- Select the supplier number for the suppliers who supply exactly two parts.

Solution

Conceptual Schema



Creation of Database

```
mysql> create database ex5;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> use ex5;
Database changed
```

Creation of Tables

```
mysql> create table supplier
(sno tinyint primary key,
sname varchar(20),
city varchar(20));
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> create table part
(pno tinyint primary key,
pname varchar(20),
price numeric(8,2));
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> create table supply
(sno tinyint, pno tinyint,
foreign key(sno) references supplier(sno) on delete cascade,
foreign key(pno) references part (pno) on delete cascade,
qty tinyint, primary key(sno, pno));
Query OK, 0 rows affected (0.01 sec)
```

Structure of Tables

```
mysql> desc supplier;
```

Field	Type	Null	Key	Default	Extra
sno	tinyint(4)	NO	PRI	NULL	
sname	varchar(20)	YES		NULL	
city	varchar(20)	YES		NULL	

3 rows in set (0.00 sec)

mysql> desc part;

Field	Type	Null	Key	Default	Extra
pno	tinyint(4)	NO	PRI	NULL	
pname	varchar(20)	YES		NULL	
price	decimal(8,2)	YES		NULL	

3 rows in set (0.00 sec)

mysql> desc supply;

Field	Type	Null	Key	Default	Extra
sno	tinyint(4)	NO	PRI	0	
pno	tinyint(4)	NO	PRI	0	
qty	tinyint(4)	YES		NULL	

3 rows in set (0.00 sec)

Insertion of Tuples

mysql> insert into supplier values(11, 'rama', 'bangalore');

Query OK, 1 row affected (0.00 sec)

mysql> insert into part values(101, 'nut', 30);

Query OK, 1 row affected (0.00 sec)

mysql> insert into supply values(15, 101, 4);

Query OK, 1 row affected (0.00 sec)

View of Tables

mysql> select * from supplier;

sno	sname	city
11	rama	bangalore
12	shyam	delhi
13	ramesh	chennai
14	john	chennai
15	yusuf	hyderabad

5 rows in set (0.00 sec)

mysql> select * from part;

pno	pname	price
101	nut	30.00
102	bolt	40.00
103	rope	50.00
104	cutter	50.00
105	tape	20.00

5 rows in set (0.00 sec)

```
mysql> select * from supply;
```

sno	pno	qty
11	104	5
12	101	14
14	102	10
14	103	5
15	101	4

```
5 rows in set (0.00 sec)
```

Queries

Query-i: Display average and sum of price of all the parts.

```
mysql> select avg(price), sum(price) from part;
```

avg(price)	sum(price)
38.000000	190.00

```
1 row in set (0.00 sec)
```

Query-ii: Display the price of those parts for which name ends with 't'.

```
mysql> select price from part where pname like '%t';
```

price
30.00
40.00

```
2 rows in set (0.00 sec)
```

Query-iii: Display the part details of part which are supplied by supplier lives in city 'Bangalore'.

```
mysql> select * from part where pno in (select pno from supply where sno in (select sno from supplier where city = 'bangalore'));
```

pno	pname	price
104	cutter	50.00

```
1 row in set (0.00 sec)
```

Query-iv: Select the supplier number for the suppliers who supply exactly two parts.

```
mysql> select sno, count(*) from supply group by sno having count(*)=2;
```

sno	count(*)
14	2

```
1 row in set (0.00 sec)
```