1(a)

```
drop database if exists cs336exam2;

create database cs336exam2;

use cs336exam2;

create table student(

  snum int primary key,

  sname varchar(30),

  major varchar(50),

  level varchar(30),

  age int

);

create table faculty(

  fid int primary key,

  fname varchar(30),

  dept varchar(30)

);

create table class(

  cname varchar(30) primary key,

  meets_at time,

  room varchar(30),

  fid int,

  foreign key (fid) references faculty(fid)

);

create table enrolled(

  snum int,

  cname varchar(30),

  primary key (snum,cname),

  foreign key (snum) references student(snum),

  foreign key (cname) references class(cname)
```

);

1(b)

If populate the tables with outside data, it should filled either table student or faculty first, and then fill table  class, and the last one is enrolled.


2.

```
select fname
from faculty
where fname like 'W%'
    and dept = 'cs';
```


3.

```
select distinct s.major
from student s,
     enrolled e
where s.snum = e.snum
    and e.cname = 'Algorithms';
```


4.

```
select s.sname
from student s
where s.age =
        (select s.age
        from student s
        where s.sname = 'Horatio');
```


5.

```
select s.sname
from student s
```

where s.age > all

      (select s.age

      from student s

      where s.sname like 'A%');


6.

select c.cname

from class c

where not exists

      (select *

      From enrolled e

      where e.cname = c.cname);


7(a)

select f.fname

from faculty f,

      class c,

    enrolled e,

    student s

where f.fid = c.fid

  and  c.cname = e.cname

  and  e.snum = s.snum

  and  s.major ='cs'

union

select f.fname

from faculty f,

      class c,

    enrolled e,

    student s

```
where f.fid = c.fid
  and  c.cname = e.cname
  and  e.snum = s.snum
  and  s.major ='math';
```

7(b)

```
select distinct f.fname
from faculty f,
          class c,
    enrolled e
where f.fid = c.fid
  and  c.cname = e.cname
  and  e.cname in
  (select e.cname
         from enrolled e,
      student s
         where e.snum = s.snum
         and  s.major ='math'
   intersect
         select e.cname
         from enrolled e,
      student s
         where e.snum = s.snum
         and  s.major ='cs')
  ;
```

7(c)

```
select f.fname
from faculty f,
      class c
```

enrolled e,

student s

where f.fid = c.fid

   and c.cname = e.cname

   and e.snum = s.snum

   and s.major = 'cs';

except

select f.fname

from faculty f,

   class c

   enrolled e,

   student s

where f.fid = c.fid

   and c.cname = e.cname

   and e.snum = s.snum

   and s.major = 'math';


8(a)

select avg(age)

from student;


8(b)

select s.sname

from student s

where s.age > any

      (select avg(age)

      from student);


9

```sql
select f.fname,
       avg(s.age)
from faculty f,
     student s,
     class c,
     enrolled e
where f.fid = c.fid
   and    c.cname = e.cname
   and    e.snum = s.snum
group by f.fname;
```

10.

```sql
select t.snum,
       t.sname,
       count(cname)
from
 (select s.snum,
         s.sname,
         e.cname
         from student s
         left  outer join enrolled e
         on s.snum = e.snum
          )t
group by t.snum
;
```