Multiple Choice: 1C, 3C, 4A, 5A, 6A, 7B, 8C, 9C, 10B

Problems:

```
1. create table employees(ssn varchar(11),
                          name varchar(30),
                          address varchar(50),
                          isFaculty boolean,
                          primary key (ssn));

   create table contact(phone varchar(20),
                        ssn varchar(11),
                        primary key (phone),
                        foreign key (ssn) references employees(ssn));

   create table university(uname varchar(50),
                          campus varchar(30),
                          uphone varchar(20),
                          uaddress varchar(50),
                          primary key (uname,campus));

   create table studyat(ssn varchar(30),
                        uname varchar(50),
                        campus varchar(30),
                        major varchar(30),
                        minor varchar(30),
                        primary key (ssn,uname,campus),
                        foreign key (ssn) references employees(ssn),
                        foreign key (uname,campus)
                          references university(uname,campus));

   create table documents(docid int,
                          issueDate date,
                          type enum('diploma','certificate','transcript'),
                          ssn varchar(11),
                          uname varchar(50),
                          campus varchar(30),
                          primary key (docid),
                          foreign key (ssn) references employees(ssn),
                          foreign key (uname,campus)
                            references university(uname,campus));
```

2.
```
select storeId,
       sum(if(productType='computer',quantity,0)) computers,
       sum(if(productType='appliance',quantity,0)) appliances,
       sum(if(productType='accessory',quantity,0)) accessories
from products
group by storeId;
```
Anther choice is to include a subquery like

```
(select storeid,sum(quantity) qty from products group by storeid)
```

for each product and do a join on storeid.

It is also possible to include a similar subqueries in the select.

3.

	min	max
(a)	max{n,m}	m+n
(b)	0	n*m
(c)	0	n*m
(d)	0	n

4.

(a)

```
sid:2     name:Bob     cid:198:336    exam number:2     cname:Databases
1) What is data?
   (A) facts
 x(B) Knowledge

2) What does DBMS stand for?
   (A) Data binary module
 x(B) Database Mgmt system
   (C) Db Manipulation Storage
```

(b)

```
select e.sid
       s.name
       e.grade
from Exam e
join Students s using (sid)
where e.cid='119:380'
  and e.number=1;
```

(c)

```
select eq.sid,
       s.name,
       sum(o.correct * eq.studentAnswer) nCorrect
from ExamQuestions eq
join Options o using (qid,optid)
join Students s using (sid)
where eq.cid='198:336'
  and eq.number=2
group by eq.sid,
         s.name;
```

(d)

```
select eq.sid,
       s.name,
       count(distinct qid) totalQuestions
from ExamQuestions eq
join Options o using (qid,optid)
join Students s using (sid)
where eq.cid='198:336'
  and eq.number=2
group by eq.sid,
         s.name;
```

(e)

```
update Exam e,
       (select eq.sid,
          s.name,
          100*sum(o.correct * eq.studentAnswer)/count(distinct qid) grade
        from ExamQuestions eq
        join Options o using (qid,optid)
        join Students s using (sid)
        where eq.cid='198:336'
          and eq.number=2
        group by eq.sid,
                 s.name) t1
set e.grade=t1.grade
where e.sid=t1.sid
  and e.cid='198:336'
  and e.number=2;
```

(f)

```
select e.sid,
       s.name
from Students s
join Exam e using (sid)
where e.cid='198:336'
  and e.number=2
  and e.grade= (
                select max(e.grade)
                from Exam e
                where e.cid='198:336'
                  and e.number=2);
```

(g)

```
select s.sid
from Students s
where s.sid not in (
                select distinct sid
                from Exam);
```

(h)

$$\Pi_{sid}(Students) - \Pi_{sid}(Exam)$$