4.7

```
create table employee
( employee_name char(20),
  street char(20),
  city char(20),
primary key( employee_name) )

create table works
( employee_name char(20),
  company_name char(20),
  salary int
primary key( employee_name),
foreign key( company_name) references company,
foreign key( employee_name) references employee )

create table company
( company_name char(20),
  city char(20),
primary key( company_name) )

create table managers
( employee_name char(20),
  manager_name char(20),
primary key( employee_name),
foreign key( employee_name) references employee )
```

4.9

所有级别的经理的所有员工的元组也会被删除!这是通过一系列步骤进行的。最初的删除将触发删除经理直接员工对应的所有元组。这些删除会反过来导致第二级员工元组的删除，以此类推，直到所有直接和间接的员工元组都被删除。

4.16

a.

```
create table expression for address
foreign key(name) references salaried_worker or hourly_worker
```

b.

每当一个元素被插入到 address 中时，必须先在 salaried_worker 中查找他的名字，找不到则在 hourly_worker 中查找；

5.8

```
create trigger check-delete-trigger after on account
referencing old row as orow
for each row
delete from depositor
where depositor.customer_name not in( select customer_name
                                      from depositor
                                      where account_number <> orow.account_number )
end
```

5.21

```
create trigger on-delete-cascade after delete on s
referencing old as orow
for each row
delete from r
where r.b=orow.a
end
```