Exam 2

Your name:

Problem 1:
Consider the relational database, where the primary keys are underlined.
Give an expression in SQL for each of the following queries:

Employee(person_name,street,city)

Works(person_name,company_name,salary)

Company(company_name,city)

Manages(person_name,manager_name)

a. Find the names of all employees who work for the First Bank Corporation.

```
Select person_name from Works
Where company_name=’First Bank Corporation’
```

b. Find the names of all employees who live in the same city and on the same street as do their managers.

```
Select E1.person_name
From Employee as E1, Employee as E2, Manages as M
Where E1.person_name=M.person_name and E2.person_name=M.manager_name
and E1.stree=E2.street and E1.city=E2.city
```

c. Find the names, street address, and cities of residence of all employees who work for First Bank Corporation and earn more than $10,000 per annum.

```
select * from employee
where person_name in
(select person_name from Works
where company_name=’First Bank Corporation’ and salary>10000
```

```
select E.person_name, street, city
from Employee as E, Works as W
where E.person_name=W.person_name and W.company_name=’First Bank Corporation’
and W.salary>10000
```

d. Find the names of all employees who earn more than every employee of Small Bank Corporation.

```
select person_name from Works
where salary > all
( select salary from Works
where company_name=’Small Bank Corporation’)

select person_name from Works
where salary>(select max(salary) from Works
where company_name=’Small Bank Corporation’)
```

e. Find the company that has the most employees.

```
select company_name from Works
group by company_name
having count(distinct person_name)>=all
( select count(distinct person_name) from Works
group by company_name)

with company_person_num as
(select company_name, count(distinct person_name) as employee_num
from Works
group by company_name)
select company_name from Works
group by company_name
having count(distinct person_name)=(select max(employee_num)
from company_person_num)
```

f. Find those companies whose employees earn a higher salary, on average,
than the average salary at First Bank Corporation.

```
select company_name
from Works
group by company_name
having avg(salary)>(select avg(salary)
                    from Works
                    where company_name=’First Bank Corporation’)
```

g. Find the names of all employees in this database who live in the same city as the company
for which they work

```
select E.person_name
from Employee as E, Works as W, Company as C
where E.person_name=W.person_name and E.city=C.city
and W.company_name=C.company_name
```

h. Give all employees of First Bank Corporation a 10 percent salary raise.

```
update Works
set salary=salary*1.1
where company_name=’First Bank Corporation’
```

i. Delete all tuples in the works relation for employees of Small Bank Corporation.

```
delete from Works
where company_name=’Small Bank Corporation’
```

Problem 2:

Using the “banking” example, write SQL to define the following views:

branch (branch-name, branch-city, assets)

customer (customer-name, customer-street, customer-city)

account (account-number, branch-name, balance)

loan (loan-number, branch-name, amount)

depositor (customer-name, account-number)

borrower (customer-name, loan-number)

employee (employee-name, branch-name, salary)

a. A view containing the account numbers and customer names (but not the balances) for all accounts at the Stow branch.

```
create view myview as
(select account-number, customer-name
from depositor, account
where depositor.account-number=account.account-number
and account.branch-name=’Stow’
```

b. A view containing the names and addresses of all customers who have an account with the bank, but do not have a loan.

```
create view myview as
(select C.customer-name, C.customer-street, Ccustomer-city
from customer as C, depositor as D
where C.customer-name=D.customer-name
and C.customer-name not in
(select customer-name
from Borrower)
```

c. A view containing the names and average account balance of every customer of the Kent branch.

```
create view myview as
(select customer-name, avg(balance)
from account as A, depositor as D
where A.account-number=D.account-number
and A.branch-name='kent'
```