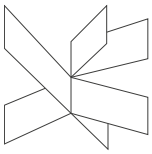
Bring ideas to life
VIA University College



Taha Mohamed Alzain(2669055)

Muhammad Nadeem 266704

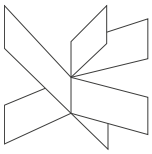
Fadi Atia Dasus(266265)

Oskars Arajs(266534)

Alexandru Vieru(267013)

Supervisor

Jens Cramer Alkjærsig



Preparing some data

update employee set superssn = 'null' where ssn = '888665555';

update employee set superssn = 'null' where ssn = '999887777';

update employee set superssn = '888665555' where ssn = '123456789';

update employee set superssn = '888665555' where ssn = '453453453';

update employee set superssn = '999887777' where ssn = '987654321';

update employee set superssn = '999887777' where ssn = '666884444';

update employee set superssn = '999887777' where ssn = '443445555';

Exercise 1 Views

1.1

create or replace view firstView as select * from works_on;

select * from firstView;2

1.2

create or replace view secondView as select sum(hours) from works_on;

select * from secondView;

1.3

create or replace view thirdView as

select essn as "EMP#", fname as "EMP_Name", pno as "PROJ#", pname as "Project Name" , hours, hours*300 as "Cost"

from works_on inner join employee on essn = ssn inner join project on pno = pnumber

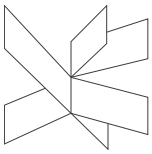
select * from thirdView

1.4

create or replace view FourthView as

select dname as "Department Name", fname as "Manager Name", salary as "Manager Salary"

from department inner join employee on ssn=mgrssn



select* from FourthView

1.5

create or replace view supervisor as

select fname as "name1", ssn as "ssn1" from employee where superssn = 'null'

select * from Supervisor

create or replace view departmentView as

select dname as "departmentname", dnumber from department

create or replace view FifthView as

select fname as "EmployeeName", name1 as "SuperVisorName", salary from employee inner join supervisor on ssn1 = superssn inner join departmentView on dno = dnumber

select * from FifthView

1.6

create or replace view SixtView as

select pname as "Project name", dname as "Dept_Name", count(ssn) as "NmbrOfEmployees", sum(hours) as "Total hours"

from works_on inner join employee on essn = ssn

inner join project on pno = pnumber

inner join department on pnumber = dnumber

group by pname, dname, ssn

having count(ssn)>0

order by ssn desc;

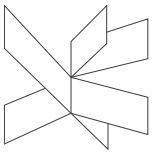
select*from SixtView

1.7

create or replace view TempView as

select pname as "projectname", dname as "deptname", count(ssn) as "nmbrofemployees", sum(hours) as "totalhours"

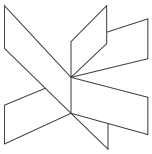
from works_on inner join employee on essn = ssn



```
inner join project on pno = pnumber

inner join department on pnumber = dnumber

group by pname, dname, ssn

having count(ssn)>0

order by ssn desc;

create or replace view SeventhView as

select projectname, deptname, sum(nmbrofemployees) as "totalemployees",
sum(totalhours) as "totalhours" from TempView

group by projectname, deptname

select * from SeventhView
```

1.8

```
create or replace view tempView1 as

select fname, MAX(salary) as "salary"

from employee inner join department on dno = dnumber

group by fname, salary

order by salary desc;


select * from tempView1


create or replace view tempView2 as

select max(salary) from tempView1


select * from  tempView2


create or replace view EightView as

select * from employee where dno = (select  dno  from employee inner join tempView2 on
salary = max )
```



```
select * from  EightView
```

1.9

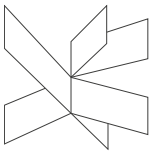
```
create or replace view nine as
select * from employee where superssn = '888665555'
```

1.10

```
create or replace view TempView as
select dname as "deptname", count(ssn) as "nmbrofemployees"
from employee
inner join department on dno = dnumber

group by dname, ssn
having salary>=30000
order by ssn desc;

create or replace view EightView as
select deptname, sum(nmbrofemployees) as "totalemployees" from TempView
group by deptname

select * from EightView
```

1.11

```
create or replace view EleventhView as+
select fname as "Name", lname as "Last Name", salary as "Salary", address as "Address"
```



from employee group by fname, lname, salary, address

select * from EleventhView

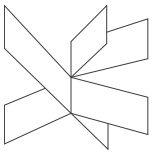
Exercise 2 Triggers

set search_path = 'COMPANY';

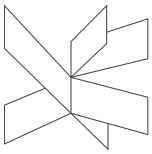
select * from works_on

2.1

```
create table log_works_on
(
   essn varchar(9),
   day_time timestamp
);
```



```
create or replace function log_for_works_on() returns trigger as $BODY$
declare
   count_works_on integer;
begin
   if(tg_op='INSERT') then
   insert into log_works_on (essn, day_time)
   values (new.essn, now());
   return new;
   end if;
   return null;
   end;
   $BODY$ language plpgsql;
create trigger log_insert
before insert on works_on for each row
execute procedure log_for_works_on();
create trigger log_update
before update on works_on for each row
execute procedure log_for_works_on();
create trigger log_delete
after delete on works_on for each row
execute procedure log_for_works_on();
insert into works_on (essn, pno, hours)
values ('234567891', 2 ,34);
update works_on set hours = 35 where essn = '666884444';
delete from works_on where essn = '3334455S5';
select * from works_on;
```



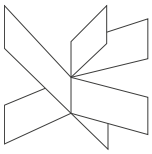
```
select * from log_works_on;
```

2.2

```
create or replace function max_4_projects() returns trigger  as $$
declare
   log_count integer;
Begin
   select pnumber into log_count from project where pnumber = new.pnumber;
   if log_count > 4 then raise exception 'Too many projetcs';
   end if;
   return new;
   end;
   $$ LANGUAGE plpgsql;
   create trigger max_4_projects
   before insert or update on project
   for each row
   execute procedure max_4_projects();
  insert into project (pname, pnumber, plocation,dnum)
values ('PC', 7 ,'horsen',4);
 insert into project (pname, pnumber, plocation,dnum)
values ('window', 2 ,'aarhus',5);
select * from project
```

2.3

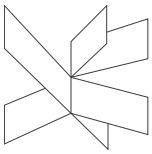
```
create or replace function max_4_projects_works_on() returns trigger  as $$
declare
```



```
        log_count integer;
Begin
        select pno into log_count from works_on where pno = new.pno;
        if log_count >4 then raise exception 'Too many projetcs';
        end if;
        return new;
        end;
        $$ LANGUAGE plpgsql;
    create trigger max_4_projects_works_on
        before insert or update on works_on
        for each row
        execute procedure max_4_projects_works_on();

insert into works_on (essn, pno, hours)
values ('888889999', 5 ,30);
insert into works_on (essn, pno, hours)
values ('222223333', 2 ,35);
```

2.4

```
create table log_department
(
   dname varchar(20),
   dnumber integer,
   mgrssn char(9),
   day_time timestamp
);
```



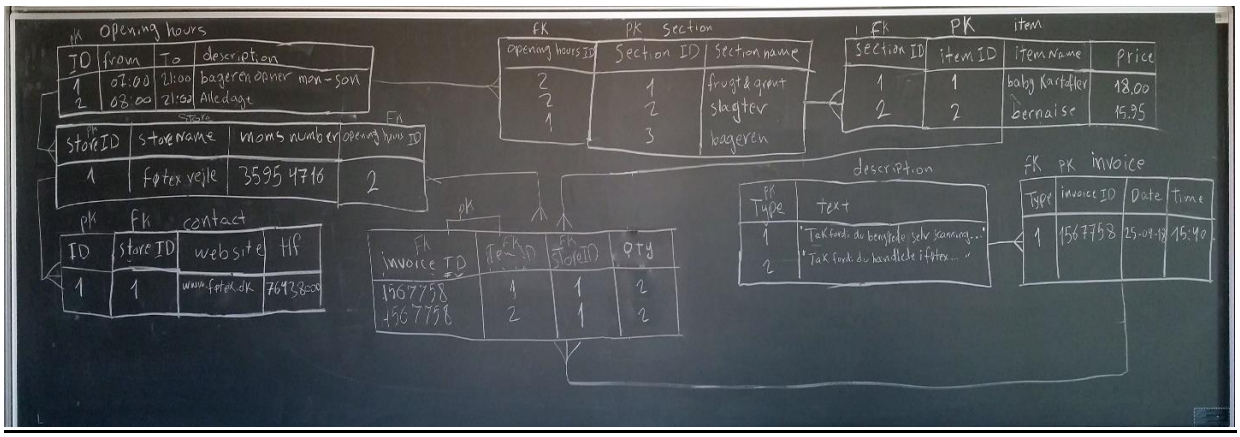
```
create or replace function log_for_department() returns trigger as $BODY$
declare
   count_department integer;
begin
   if(tg_op='INSERT') then
   insert into log_department (dname,dnumber,mgrssn,day_time)
   values (new.dname,new.dnumber,new.mgrssn, now());
   return new;
   end if;
   return null;
   end;
   $BODY$ language plpgsql;
create trigger log_insert
before insert on department for each row
execute procedure log_for_department();
create trigger log_update
before update on department for each row
execute procedure log_for_department();
create trigger log_delete
after delete on department for each row

execute procedure log_for_department();
insert into department (dname,dnumber,mgrssn)
values ('Secretary', 2 ,'222223333');
update department set dnumber=6 where mgrssn = '333445555';
delete from department where mgrssn = '222223333';
```

select * from department;

select * from log_department;

Exercise 4

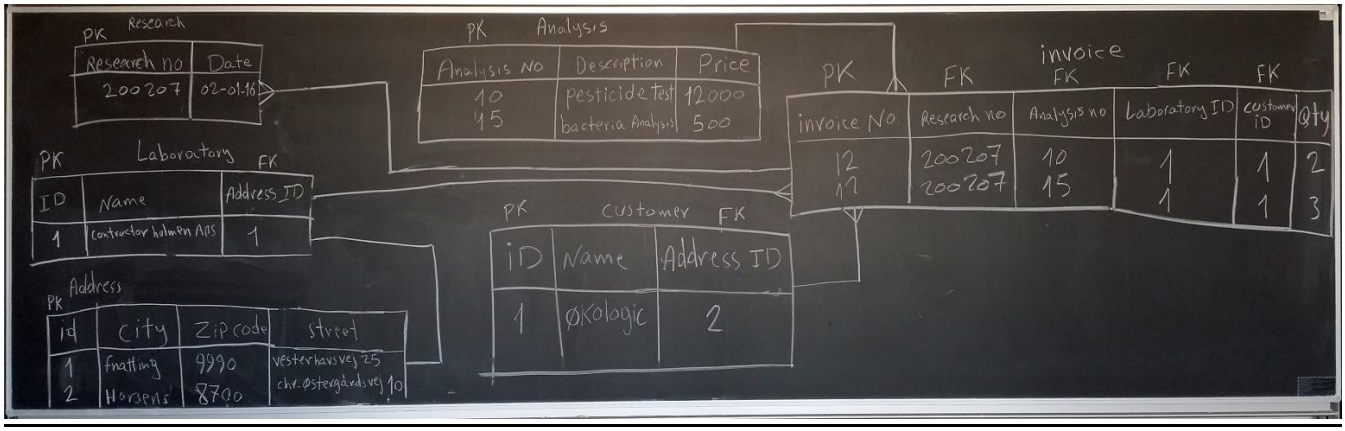
Reverse engineering



Exercise 5

Reverse engineering

0



Exercise 6 Transactions

6.1 Log file

postgresql-2018-05-10_091019 - Notepad

File Edit Format View Help

```
        from works_on inner join employee on essn = ssn inner join project on pno = pnumber
2018-05-10 11:01:58 CEST ERROR:  syntax error at end of input at character 36
2018-05-10 11:01:58 CEST STATEMENT:  create or replace view thirdView as
2018-05-10 11:35:06 CEST LOG:  could not receive data from client: An existing connection was forcibly closed by the remote host.

2018-05-10 11:35:06 CEST LOG:  could not receive data from client: An existing connection was forcibly closed by the remote host.

2018-05-10 11:35:49 CEST ERROR:  relation "employee" does not exist at character 42
2018-05-10 11:35:49 CEST STATEMENT:  select superssn as "superVisor ssn" from employee where superssn != null
2018-05-10 11:36:39 CEST ERROR:  relation "employee" does not exist at character 42
2018-05-10 11:36:39 CEST STATEMENT:  select superssn as "superVisor ssn" from employee
2018-05-10 11:37:15 CEST ERROR:  syntax error at or near "not" at character 67
2018-05-10 11:37:15 CEST STATEMENT:  select superssn as "superVisor ssn" from employee  where superssn not null
2018-05-10 11:37:36 CEST ERROR:  syntax error at or near "Exercise" at character 227
2018-05-10 11:37:36 CEST STATEMENT:  set search_path = 'COMPANY';
        select * from department;
        select * from dependent;
        select * from dept_locations;
        select * from employee;
        select * from project;
        select * from works_on;
        ------------------------------------------
        Exercise 1.1
        create or replace view firstView as select * from works_on;
        select * from firstView;
        ------------------------------------------

        Exercsie 1.2
        create or replace view secondView as select sum(hours) from works_on;
        select * from secondView;
        ------------------------------------------

        Exercsie 1.3
        create or replace view thirdView as
        select essn as "EMP#", fname as "EMP_Name", pno as "PROJ#",  pname as "Project Name" , hours, hours*300 as "Cost"
        from works_on inner join employee on essn = ssn inner join project on pno = pnumber

        select * from thirdView
        ------------------------------------------

        Exercsie 1.4
        create or replace view FourthView as
        select dname as "Department Name", fname as "Manager Name", salary as "Manager Salary"
        from department inner join employee on ssn = mgrssn

        select * from FourthView
        ------------------------------------------

        Exercsie 1.5
        create or replace view FifthView as
        select superssn as "superVisor ssn" from employee  where superssn = null
```

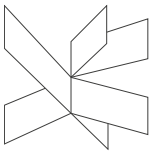
6.2 Dirty read problem

Dirty read anomaly occurs when someone is making changes to a database and does not commit the modifications. Once a user tries to access that data which hasn't been committed he will be able to read the data but not see the changes that have been made.

In other words, he can see certain data that might not be there anymore.

6.3 Non-repeatable read

The non-repeatable anomaly occurs when one operator issues a read transaction command but after an update is issued by a second user on the same table in which the first command was originally executed. If the first command is issued again than it will not work anymore



as the first time. This is due because the table has been updated after the first read command has been executed and changes which do not support it have occurred.

6.4 Phantom read

The phantom read anomaly occurs when there are multiple transactions and in the case of one transaction changing the content of a table, then another transaction will have different results when calling that table due to it being modified from the previous transaction.

Exercise 7

7 Invoices who were paid

7.1.1 Inner joins

```
SELECT P.*,D.*
    FROM PAYMENTS P,DEBTORS D
    WHERE P.INVOICENUMBER=D.INVOICENUMBER;
```

invoicenu... integer	customer character varying (20)	value integer	invoicenu... integer	customer character varying (20)	value integer
123	Peter	200	123	Peter	200
234	Soren	500	234	Soren	500
345	Soren	400	345	Soren	400
456	Peter	60	456	Peter	66
567	Trine	50	567	Trine	50

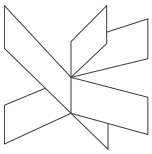
7.1.2 Intersect

```
(SELECT INVOICENUMBER FROM PAYMENTS)
    INTERSECT
      (SELECT INVOICENUMBER FROM DEBTORS)
```

Data Output | Explain

invoicenu... integer
234
123
456
567
345

7.1.3 Left outer join



```
SELECT P.*,D.*
FROM DEBTORS D LEFT JOIN
PAYMENTS P ON D.INVOICENUMBER= P.INVOICENUMBER;
```

Data Output | Explain | Messages | History

invoicenu... integer	customer character varying (20)	value integer	invoicenu... integer	customer character varying (20)	value integer
[null]	[null]	[null]	12	Hans	600
123	Peter	200	123	Peter	200
234	Soren	500	234	Soren	500
345	Soren	400	345	Soren	400
456	Peter	60	456	Peter	66
567	Trine	50	567	Trine	50

7.2 Invoices that have not been paid

Except

```
(SELECT INVOICENUMBER FROM DEBTORS)
    EXCEPT
(SELECT INVOICENUMBER FROM PAYMENTS)
```

Data Output | Explain

invoicenu... integer
12

7.3 Make a list of customers who have an invoice but have not paid

Except

```
(SELECT INVOICENUMBER FROM PAYMENTS)
    EXCEPT
(SELECT INVOICENUMBER FROM DEBTORS)
```

Data Output | Explain | Messages

invoicenu... integer
1313
1212