5)

a)

SELECT \*

FROM EMPLOYEE

WHERE EMPLOYEE\_NAME IN ( SELECT EMPLOYEE\_NAME

FROM WORKS

WHERE COMPANY\_NAME=”First Bank Corporation”

AND SALARY > 10000)

b)

SELECT E.EMPLOYEE\_NAME

FROM EMPLOYEE AS E, WORKS AS W, COMPANY AS C

WHERE E.EMPLOYEE\_NAME=W.EMPLOYEE\_NAME

AND W.COMPANY\_NAME=C.COMPANY\_NAME

AND E.CITY=C.CITY

c)

SELECT E1.EMPLOYEE \_NAME

FROM EMPLOYEE AS E1, EMPLOYEE AS E2, MANAGES AS M

WHERE E1.EMPLOYEE \_NAME=M.EMPLOYEE \_NAME

AND E2.EMPLOYEE \_NAME=M.MANAGER\_NAME

AND E1.STREET=E2.STREET AND E1.CITY=E2.CITY

d)

SELECT EMPLOYEE\_NAME

FROM WORKS

WHERE COMPANY\_NAME !=”FIRST BANK CORPORATION”

e)

SELECT EMPLOYEE\_NAME

FROM WORKS

WHERE SALARY > ALL

(SELECT SALARY

FROM WORKS

WHERE COMPANY\_NAME=”SMALL BANK CORPORATION”)

f)



select Company\_name

from company

where city in( select city

from company

where company\_name = "Small Bank Corporation")

and Company name!= "Small Bank Corportaion";

g)

SELECT EMPLOYEE\_NAME

FROM WORKS

WHERE SALARY> ALL (SELECT AVG(SALARY)

FROM WORKS

GROUP BY COMPANY\_NAME) (this is wrong)

Correct answer

with dt as(select employee\_name,company\_name, avg(salary) as avg\_sal

from Works

group by company\_name)

select employee\_name

from works w, dt d

where w.company\_name = d.company\_name

and w.salary > d.avg\_salary;

h)

SELECT COMPANY\_NAME

FROM WORKS W

GROUP BY COMPANY\_NAME

ORDER BY SUM(SALARY)

LIMIT 1

6)

a)

MATCH (r1:Reader)--(r2:Reader),

(r2)--(b:Book)--(g:Genre)

WHERE r1.name='Seppe vanden Broucke' AND g.name='fantasy'

RETURN r2.name

b)

MATCH (r1:Reader)--(r2:Reader),

(r2)--(book:Book)--(g:Genre)

WHERE NOT (r1)--(book) AND r1.name='Bart Baesens' AND g.name='education'

RETURN r2.name,book.title

c)

MATCH (b1:Book)--(g:Genre)--(b2:Book)

WITH b1,b2, count(\*) AS common\_genres

WHERE common\_genres > 1

RETURN b1.title,b2.title,common\_genres

d)

MATCH (b1:Book), (b2:Book)

WITH b1,b2

OPTIONAL MATCH (b1)--(g:Genre)--(b2)

WHERE g IS NULL

RETURN b1.title,b2.title

7)

a)

db.people.mapReduce(

function()

{

emit(this.restaurant\_id,{rating:this.rating, count:1});

},

function(restaurant\_id,val)

{

var x=0;

var temp=0;

val.forEach(

function(check){

x+=check.average\*check.count;

temp+=check.count;

});

return { rating:(x/temp), count:temp};

},

{

out:{inline:1}

}

).find();

b)

db.people.aggregate([

{$group:{ \_id:"$restaurant\_id", avgrating:{$avg:"$rating"}}},

{$sort:{avgrating:1}}

]);

c)

db.people.mapReduce(

function()

{

emit(this.restaurant\_id,{maximum:this.rating});

},

function(restaurant\_id,val)

{

var temp=0;

val.forEach(

function(check){

temp = Math.max(temp,check.max)

});

return {maximum:temp};

},

{

out:{inline:1}

}

).find();

d)

db.people.aggregate([

{$group:{\_id:"$restaurant\_id",maxrating:{$max:"$rating"}}},

{$sort:{maxrating:1}}

]);