Experiment No 7
Aim: Perform nested and complex queries
Class: SE Comp
Year: 2020-21
Performed by: Danyl Fernandes, 72
Performance Date: 31-03-2021
Submission Date: 04-04-2021

Theory:

SQL 'IN'

The IN operator allows you to specify multiple values in a WHERE clause. The IN operator is a shorthand for multiple OR conditions.

Syntax:
expression **IN** (value1, value2, ... value_n);
OR
expression **IN** (subquery);

Parameters or Arguments:

expression

This is a value to test.

value1, value2 ..., value_n
These are the values to test against expression. If any of these values matches expression, then the IN condition will evaluate to true.

subquery
This is a SELECT statement whose result set will be tested against expression. If any of these values matches expression, then the IN condition will evaluate to true.

Example:

```
SELECT * FROM suppliers
WHERE supplier_name = 'Microsoft'
OR supplier_name = 'Oracle'
OR supplier_name = 'Flowers Foods';
```

SQL 'EXISTS'

- The EXISTS operator in MySQL is a type of Boolean operator which returns the true or false result.
- It is used in combination with a subquery and checks the existence of data in a subquery.
- It means if a subquery returns any record, this operator returns true. Otherwise, it will return false.
- The true value is always represented numeric value 1, and the false value represents 0. We can use it with SELECT, UPDATE, DELETE, INSERT statement

Syntax:

```
SELECT column_name(s)
FROM table_name
WHERE column_name [NOT] EXISTS (
        SELECT column_name(s)
        FROM table_name
        WHERE condition
);
```

- The NOT operator is used to negate the EXISTS operator. It returns true when the subquery does not return any row.
- Otherwise, it returns false. Generally, the EXISTS query begins with SELECT *, but it can start with the SELECT column, SELECT a_constant, or anything in the subquery.
- It will give the same output because MySQL ignores the select list in the SUBQUERY.

Example:

```
SELECT name, occupation FROM customer WHERE EXISTS (
        SELECT * FROM Orders WHERE customer.cust_id = Orders.cust_id
);
```

Conclusion:

We were successfully able to implement the concept of IN and EXISTS queries and the experiment demonstrated the concept clearly.

Employee (<u>eid</u>, ename, address, city)

Query:

```
create table employee (
     eid int PRIMARY KEY,
     ename varchar(30),
     address varchar(30),
     city varchar(30)
);
```

Works (eid(FK), cid(FK), salary)

Query:

```
create table works (
     eid int,
     cid int,
     salary int,
     FOREIGN key (eid) REFERENCES employee(eid),
     FOREIGN key (cid) REFERENCES company(cid)
);
```

Company (<u>cid</u>, cname, city)

Query:

```
create table company (
     cid int PRIMARY KEY,
     cname varchar(30),
     city varchar(30)
);
```

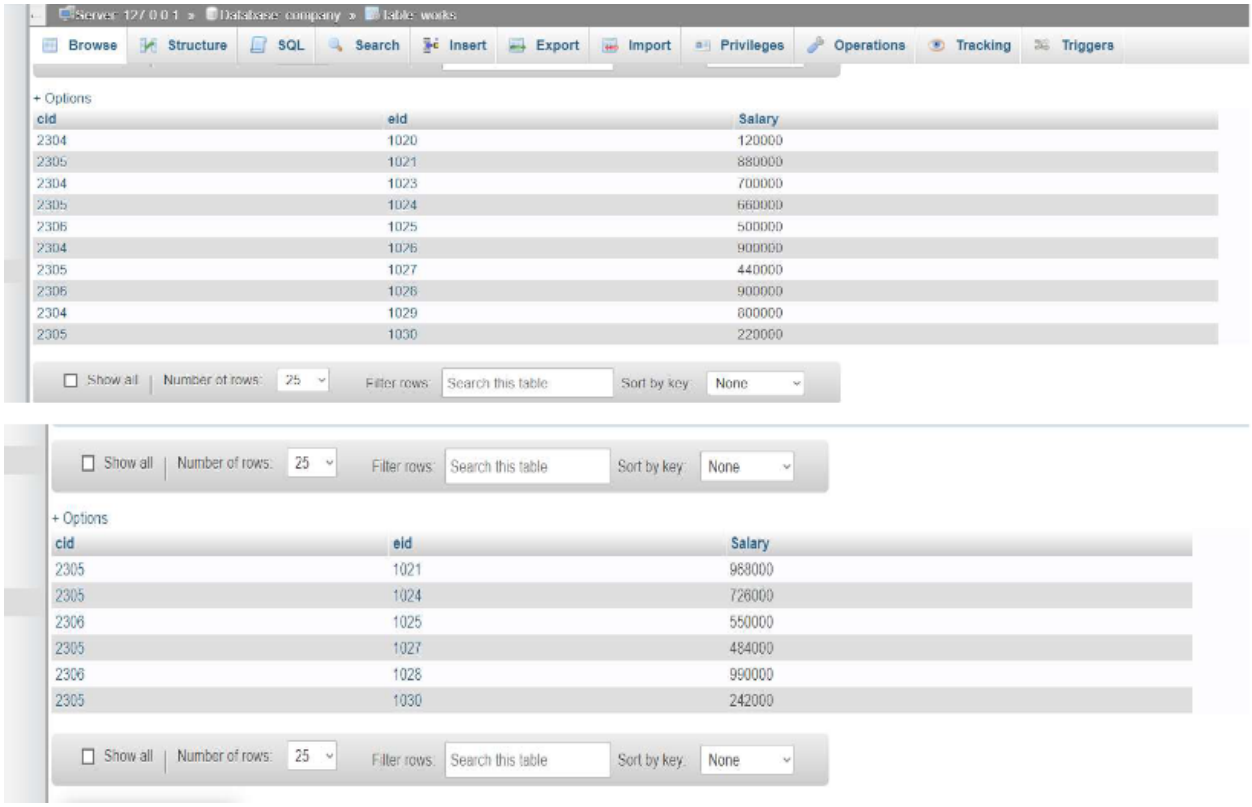
Give all employees of “ANZ corporation” a 10% increase in salary

QUERY 1:

Update works SET Salary = Salary *1.1where (cid) IN (select cid from Company where cname = 'ANZ Corporation')

QUERY 2:

UPDATE works SET Salary = Salary *1.1WHERE EXISTS (SELECT*FROM company WHERE company.cid=works.cidAND cname="ANZ Corporation")



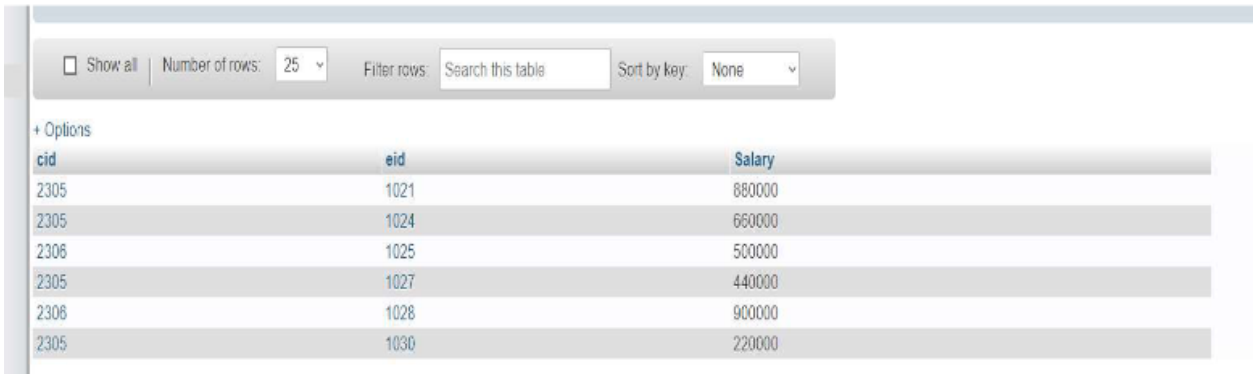
Delete the rows from works relation for employees of “Small Bank Corporation”

QUERY 1:

DELETE FROM works where (cid) IN (select cid from Company where cname = 'small bank corporation')

QUERY 2:

DELETE FROM works WHERE EXISTS (SELECT*FROM company WHERE works.cid= company.cidAND company.cname= 'Small Bank Corporation')



cid	eid	Salary
2305	1021	880000
2305	1024	660000
2306	1025	500000
2305	1027	440000
2306	1028	900000
2305	1030	220000

Find name of employees who are working in “ANZ corporation”

QUERY 1:

SELECT ename as ANZ_List FROM employee WHERE (eid) IN (select eid from works w where (cid) IN(SELECT cid FROM company c WHERE cname='ANZ Corporation'))

QUERY 2:

SELECT ename as ANZ_List FROM employee e WHERE EXISTS (select eid from works w where w.eid=e.eidAND EXISTS(SELECT cid FROM company c WHERE w.cid=c.cidAND cname='ANZ Corporation’))

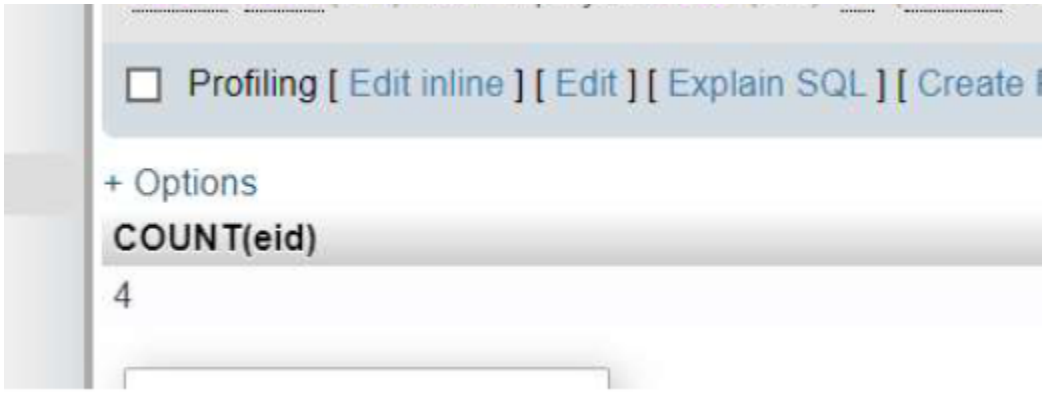
Find total number of employees in “ABC” company

QUERY 1:

SELECTCOUNT(eid) FROM employee WHERE (eid) IN (select eid from works where (cid) IN(SELECT cid FROM company c WHERE cname='ANZ Corporation'))

QUERY 2:

SELECT COUNT(eid) FROM employee e WHERE EXISTS (select eid from works w where e.eid =w.eid AND EXISTS (SELECT cid FROM company c WHERE c.cid=w.cid AND cname='ANZ Corporation'))



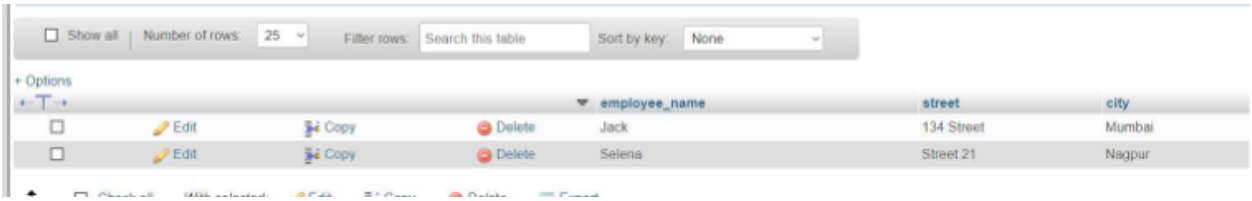
Find the names, street address, and cities of residence for all employees who work for “First Bank Corporation” and earn more than $10,000.

QUERY 1:

SELECT employee_name, street,city FROM employee WHERE (employee_name) IN (select employee_name from works where salary>10000 AND (company_name) IN(SELECT company_name from company WHERE company_name='First Bank Corporation'))

QUERY 2:

SELECT employee_name, street,city FROM employee e WHERE EXISTS (select employee_name from works w where w.employee_name=e.employee_name AND salary>10000 AND EXISTS (SELECT company_name from company c WHERE c.company_name=w.company_name AND company_name='First Bank Corporation'))



Find the names of all employees in the database who live in the same cities as the companies for which they work.

QUERY 1:

SELECT employee_name FROM employee e WHERE (employee_name) IN (select employee_name from works where (company_name) IN(SELECT company_name from company c WHERE e.city=c.city))

QUERY 2:

SELECT employee_name FROM employee e WHERE EXISTS (select employee_name from works w where e.employee_name=w.employee_name AND EXISTS (SELECT company_name from company c WHERE c.company_name=w.company_name AND e.city=c.city))

Find the names of all employees in the database who live in the same cities and on the same streets as do their managers.

QUERY 1:

Select e1.employee_name,e1.street,e1.city from employee as e1, employee as e2, manages as m where exists(select manager_name from manages where e1.employee_name = m.manager_name and e2.employee_name = m.manager_name and e1.street = e2.street and e1.city = e2.city)

QUERY 2:

Select e1.employee_name,e1.street,e1.city from employee as e1, employee as e2, manages as m where (manager_name) IN(select manager_name from manages where e1.employee_name = m.manager_name and e2.employee_name = m.manager_name and e1.street = e2.street and e1.city = e2.city)

				employee_name	street	city
☐	Edit	Copy	Delete	Jonas	0	Pune
☐	Edit	Copy	Delete	Lincy	185 Street	Lucknow
☐	Edit	Copy	Delete	Martha	134 Street	Mumbai
☐	Edit	Copy	Delete	Selena	185 Street	Nagpur
☐	Edit	Copy	Delete	Taylor	Street 21	Nashik

Find the names of all employees in the database who earn more than every "Small Bank Corporation"

QUERY 1:

SELECT employee_name from works w where w.salary>(SELECT MAX(salary) FROM works w WHERE (employee_name) IN (select employee_name from works where (company_name) IN(SELECT c.company_name FROM company c WHERE company_name='Small Bank Corporation')))

QUERY 2:

SELECT employee_name from works w where w.salary>(SELECT MAX(salary) FROM works w WHERE EXISTS (select employee_name from works where w.employee_name=w.employee_name AND EXISTS (SELECT c.company_name FROM company c WHERE c.company_name=w.company_name AND company_name='Small Bank Corporation')))

Find the names of all employees who earn more than the average salary of all employees of their company

QUERY 1:

SELECT employee_name from works w where w.salary>(SELECT AVG(salary) FROM works w WHERE (employee_name) IN (select employee_name from works where (company_name) IN(SELECT c.company_name FROM company c WHERE w.salary=w.salary)))

QUERY 2:

SELECT employee_name from works w where w.salary>(SELECT AVG(salary) FROM works w WHERE EXISTS (select employee_name from works where EXISTS(SELECT c.company_name FROM company c WHERE w.salary=w.salary)))

Find the employee whose salary is highest

QUERY 1:

SELECT employee_name from works w where (salary) IN(SELECT MAX(salary) FROM works w)

QUERY 2:

SELECT employee_name, MAX(Salary) from works w where EXISTS(SELECT salary FROM works)

Find the second largest salary

QUERY 1:

SELECT employee_name, MAX(salary) FROM works WHERE salary IN (SELECT salary FROM works EXCEPT SELECT MAX(salary) FROM works)

QUERY 2:

SELECT employee_name,salary FROM works WHERE EXISTS (SELECT salary FROM works EXCEPT SELECT MAX(salary) FROM works)