A **subquery** is a SQL query nested inside a larger query.

- A subquery may occur in :
  - A SELECT clause
  - A FROM clause
  - A WHERE clause
- The subquery can be nested inside a SELECT, INSERT, UPDATE, or DELETE statement or inside another subquery.
- A subquery is usually added within the WHERE Clause of another SQL SELECT statement.
- You can use the comparison operators, such as >, <, or =. The comparison operator can also be a multiple-row operator, such as IN, ANY, or ALL.

```
--SQL query to find those employees whose salary is more than the salary of JONAS. Return complete information about the employees.
select * from  employees
where salary >(select salary from  employees  where emp_name = 'JONAS');
```

Output:

68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001
67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001

```
--find those employees who work as 'MANAGERS' and 'ANALYST' and working in ‘SYDNEY’ or ‘PERTH’ with an experience more than 5 years without receiving the commission. Sort the result-set in ascending order by department location. Return employee ID, employee name, salary, and department name.
select e.emp_id,e.emp_name,e.salary ,d.dep_name from  Employees e , Department d
where e.dep_id in (select dep_id from  Department where d.dep_location in ('SYDNEY' ,'PERTH')) and
d.dep_id=e.dep_id
and e.job_name in ('MANAGER' , 'ANALYST') and e.commission is null
and DATEDIFF(year,hire_date,getdate())>5 order by dep_location;
--or
select emp_id,emp_name,salary ,dep_name from   Employees e , Department d
where emp_id in (select emp_id from  Employees where e.job_name in ('MANAGER','ANALYST'))
and e.dep_id=d.dep_id and dep_location in ('SYDNEY','PERTH') and commission is null
and DATEDIFF(year,hire_date,getdate())>5 order by dep_location;
```

Output:

66928	BLAZE	2750	Marketing
67832	CLARE	2550	Finance

```
--find those employees work at SYDNEY or working in the FINANCE department with an annual salary above 28000, but the monthly salary should not be 3000 or 2800 and who do not work as a MANAGER and whose ID contain a digit of '3'or'7' in 3rd position.Sort the result-set in ascending order by department ID and descending order by jobname.Return empID, empname, salary, dep name, dep location, depID, and job name.

select * from  Employees e ,Department d
```

```
where e.emp_id in(select emp_id from  Employees e where (e.salary*12)>28000
and e.salary not in(3000,2800)
and (emp_id like '__3%' or e.emp_id like '__7%')  )
and e.dep_id =d.dep_id
and (dep_location in ('Sydney') or e.job_name not in ('manager'))
order by e.dep_id asc , e.job_name desc;
```
Output:

68319	KAYLING PRESIDENT	NULL	1991-11-18	6000	NULL	1001	1001 Finance	Sydney

--find those employees who work as same designation of FRANK. Return complete information about the employees.
```
select * from  employees where job_name = (select job_name  from  employees  where emp_name = 'FRANK');
```
Output:

67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001

--find those employees who are senior to ADELYN. Return complete inform ation about the employees.
```
select * from  employees where hire_date <  (select hire_date  from  employees  where emp_name = 'ADELYN');
--or
select * from  employees where emp_id < (select emp_id  from  employees  where emp_name = 'ADELYN');
```
Output:

63679	SANDRINE	CLERK	69062	1990-12-18	900	NULL	2001

find those employees of department ID 2001 and whose designation is same as of the designation of department ID 1001.
```
select e.emp_name ,e.job_name,d.dep_id,d.dep_location from  employees e, department d
where d.dep_id = 2001   and e.dep_id = d.dep_id
and e.job_name IN (select e.job_name from  employees e, department d  where e.dep_id = d.dep_id and d.dep_id
=1001);
```
Output:

JONAS	MANAGER	2001	Melbourne
SANDRINE	CLERK	2001	Melbourne
ADNRES	CLERK	2001	Melbourne

--find those employees whose salary is the same as the salary of FRANK or SANDRINE.Sort the result-set in descending order by salary.
```
select * from  employees
where salary IN (select salary from  employees e where (emp_name = 'FRANK' OR emp_name = 'BLAZE')
and employees.emp_id <> e.emp_id)
order by salary desc;
```
Output:

67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001

--find those employees whose designation are the same as the designation of MARKER or salary is more than the salary of ADELYN.

```
select * from employees where salary > (select salary from Employees where emp_name='ADELYN')
or job_name =(select job_name from Employees where emp_name ='MARKER');
```
Output:

68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001
66928	BLAZE	MANAGER	68319	1991-05-01	2750	NULL	3001
67832	CLARE	MANAGER	68319	1991-06-09	2550	NULL	1001
65646	JONAS	MANAGER	68319	1991-04-02	2957	NULL	2001
67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001
63679	SANDRINE	CLERK	69062	1990-12-18	900	NULL	2001
68736	ADNRES	CLERK	67858	1997-05-23	1200	NULL	2001
69000	JULIUS	CLERK	66928	1991-12-03	1050	NULL	3001
69324	MARKER	CLERK	67832	1992-01-23	1400	NULL	1001

```
--Find those employees whose salary is more than the total remuneration (salary + commission) of the designation SALESMAN.
select * from employees where salary > (select max(salary+commission) from employees WHERE job_name = 'SALESMAN');
```
Output:

68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001
65646	JONAS	MANAGER	68319	1991-04-02	2957	NULL	2001
67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001

```
-- find those employees who are senior to BLAZE and working at PERTH or BRISBANE.
select * from employees e, department d
where d.dep_location IN ('PERTH', 'BRISBANE')
and e.dep_id = d.dep_id
and e.hire_date < (select e.hire_date from employees e where e.emp_name = 'BLAZE') ;
```
Output:

64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001	3001	Marketing	Perth
65271	WADE	SALESMAN	66928	1991-02-22	1350	600	3001	3001	Marketing	Perth

```
--Find those employees of grade 3 and 4 and work in the department of FINANCE or AUDIT and whose salary is more than the salary of ADELYN and experience is more than FRANK
select * from employees e where e.dep_id IN(select d.dep_id from department d where d.dep_name IN ('FINANCE', 'AUDIT') )
and e.salary > (select salary from employees where emp_name = 'ADELYN')
and e.hire_date < (select hire_date from employees where emp_name = 'FRANK')
and e.emp_id IN (select e.emp_id from employees e, salary_grade s where e.salary BETWEEN s.min_sal AND s.max_sal AND s.grade IN (3, 4))
order by e.hire_date ASC;
```
Output:

65646	JONAS	MANAGER	68319	1991-04-02	2957	NULL	2001
67832	CLARE	MANAGER	68319	1991-06-09	2550	NULL	1001

--find the highest paid employees in the department MARKETING.Return complete information about the employees

```
select * from Employees where salary in (select max(salary) from Employees where
dep_id in (select dep_id from Department where dep_name ='MARKETING'));
```

Output:

66928	BLAZE	MANAGER	68319	1991-05-01	2750	NULL	3001

--find the employees of grade 3 who joined recently and location at PERTH.

```
select emp_name,emp_id,job_name,hire_date from Employees  e  , Department d where d.dep_location ='Perth'
and hire_date in (select max(hire_date)  from Employees e , salary_grade s where e.salary between s.min_sal and
s.max_sal
and s.grade =3);
```

Output:

TUCKER	68454	SALESMAN	1991-09-08

--find those employees who are senior to recently hired employee and work under KAYLING.

```
select * from Employees where hire_date < (select max(hire_date) from Employees where
manager_id in (select emp_id from Employees where emp_name='KAYLING'));
```

Output:

66928	BLAZE	MANAGER	68319	1991-05-01	2750	NULL	3001
65646	JONAS	MANAGER	68319	1991-04-02	2957	NULL	2001
63679	SANDRINE	CLERK	69062	1990-12-18	900	NULL	2001
64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001
65271	WADE	SALESMAN	66928	1991-02-22	1350	600	3001

--find those employees of grade 3 to 5 and location at SYDNEY. The employees are not in PRESIDENT designated and salary is more than the highest paid employee of PERTH where no MANAGER and SALESMAN are working under KAYLING.

```
select * from Employees e
where e. emp_id in (select emp_id from Employees e , salary_grade s where e.salary between s.min_sal and
s.max_sal and s.grade in (3,4,5))
and dep_id in (select dep_id from Department d where d.dep_location='SYDNEY')
and e.job_name<>'PRESIDENT' and job_name in ('MANAGER' ,'SALESMAN')
and salary > ( select max(salary) from Employees where dep_id in (select dep_id from Department d where
d.dep_location='PERTH')
                        and manager_id not in ( select emp_id from Employees where emp_name
                        ='KAYLING'));
```

Output:

67832	CLARE	MANAGER	68319	1991-06-09	2550	NULL	1001

-- find those employees who are senior employees as of year 1991. Return complete information about the employees.

```
select * from Employees
where hire_date=(select min(hire_date) from Employees where FORMAT(hire_date,'yyyy')='1991');
```

Output:

64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001

--SQL query to compute the total salary of employees of grade 3. Return total salary.

```
select sum(salary)'totalsal' from Employees where emp_id in (select emp_id from Employees e , salary_grade s
where e.salary between s.min_sal and s.max_sal and grade =3);
```

Output:

totalsal
3300

--Find those departments where maximum number of employees work.Return department ID, department name, location and number of employees

```
select * from Department  where dep_id in (select dep_id from Employees
group by dep_id
having count(*) in
                    (select max(mycount)  from (select count(*)  mycount from Employees group by dep_id ) a) );
 --or
select d.dep_id,d.dep_name ,d.dep_location,count(*) as 'no.of Emp' FROM employees e, department d
WHERE e.dep_id = d.dep_id
GROUP BY d.dep_id,d.dep_name,d.dep_location
HAVING count(*) =
  (select MAX (mycount)FROM (select COUNT(*) mycount FROM employees GROUP BY dep_id) a);
```

Output:

3001	Marketing	Perth	
3001	Marketing	Perth	6

--find those employees who are working as a manager. Return employee name, job name, department name, and location

```
select e.emp_name,e.job_name,d.dep_name,d.dep_location from Employees e , Department d
where e.emp_id in( select manager_id from Employees )
And e.dep_id=d.dep_id;
```

Output:

KAYLING	PRESIDENT	Finance	Sydney
CLARE	MANAGER	Finance	Sydney
JONAS	MANAGER	Audit	Melbourne
SCARLET	ANALYST	Audit	Melbourne
FRANK	ANALYST	Audit	Melbourne
BLAZE	MANAGER	Marketing	Perth

--find those employees who receive the highest salary of each department. Return employee name and department ID.

```
select emp_name,e.dep_id from Employees e where salary in (select max(salary) from Employees group by dep_id)
```

Output:

KAYLING	1001
BLAZE	3001
SCARLET	2001
FRANK	2001

--find those employees whose salary is equal or more to the average of maximum and minimum salary.

```
select * from Employees where salary >=(select (max(salary)+min(salary))/2 from Employees);
```

Output:

68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001

--find those managers whose salary is more than the average salary of his employees
```
select* from Employees m where m.emp_id in (select manager_id from Employees )
and m.salary >(select avg(e.Salary) from Employees e where e.manager_id=m.emp_id);
```
Output:

66928	BLAZE	MANAGER	68319	1991-05-01	2750	NULL	3001
67832	CLARE	MANAGER	68319	1991-06-09	2550	NULL	1001
67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001

--find those employees whose salary is less than the salary of his manager but more than the salary of any other manager.
```
select w.emp_name from employees w, employees m
where w.manager_id = m.emp_id
AND w.salary < m.salary
and w.salary > any  (select salary from employees  where emp_id in (select manager_id from employees));
```
Output:

BLAZE
JONAS

--SQL query to find five lowest paid workers
```
select emp_id,emp_name,salary from  employees e where 5> (select count(*) from employees where e.salary
>salary);
```
Output:

63679	SANDRINE	900
65271	WADE	1350
66564	MADDEN	1350
68736	ADNRES	1200
69000	JULIUS	1050

--Find those managers who are not working under the PRESIDENT
```
select * from Employees where emp_id in (select manager_id from Employees)
and manager_id not in (select emp_id from Employees where job_name='President');
```
Output:

67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001

--find those employees who joined in the company on the same date.
```
select * from employees e where hire_date IN(select hire_date from employees where e.emp_id <> emp_id);
```
Output:

69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001
69000	JULIUS	CLERK	66928	1991-12-03	1050	NULL	3001

```
--find those managers who handle maximum number of employees. Return managers name, number of employees.
select m.emp_name,count(*)from employees w, employees m
where w.manager_id = m.emp_id group by m.emp_name
having count(*) =
                (select MAX (mycount)from (select count(*) mycount from employees group by manager_id) a);
```

Output:

BLAZE	5

```
--find those managers who receive less salary then the employees work under them.
select distinct m.emp_name, m.salary from employees w,employees m where w.manager_id = m.emp_id AND
w.salary>m.salary;
--or
select * from employees w where emp_id IN (select manager_id from employees where w.salary<salary);
```

Output:

JONAS				2957			
65646	JONAS	MANAGER	68319	1991-04-02	2957	NULL	2001

```
--find those employees who work as managers. Return complete information about the employees. Use co-related
subquery.
select * from Employees where emp_id in (select manager_id from Employees);
```

Output:

68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001
66928	BLAZE	MANAGER	68319	1991-05-01	2750	NULL	3001
67832	CLARE	MANAGER	68319	1991-06-09	2550	NULL	1001
65646	JONAS	MANAGER	68319	1991-04-02	2957	NULL	2001
67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001

```
--SQL query to find those employees who are sub-ordinates of BLAZE.
select * from Employees where manager_id in (select emp_id from Employees where emp_name ='BLAZE');
```

Output:

64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001
65271	WADE	SALESMAN	66928	1991-02-22	1350	600	3001
66564	MADDEN	SALESMAN	66928	1991-09-28	1350	1500	3001
68454	TUCKER	SALESMAN	66928	1991-09-08	1600	0	3001
69000	JULIUS	CLERK	66928	1991-12-03	1050	NULL	3001

```
--SQL query to find those employees who receive minimum salary for a designation
select * from Employees where salary in(select min(salary) from Employees group by job_name);
```

Output:

68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001
67832	CLARE	MANAGER	68319	1991-06-09	2550	NULL	1001

67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001
63679	SANDRINE	CLERK	69062	1990-12-18	900	NULL	2001
65271	WADE	SALESMAN	66928	1991-02-22	1350	600	3001
66564	MADDEN	SALESMAN	66928	1991-09-28	1350	1500	3001

```
--find those employees who receive maximum salary for a designation.
select * from Employees where salary in(select max(salary) from Employees group by job_name);
```

68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001
65646	JONAS	MANAGER	68319	1991-04-02	2957	NULL	2001
67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001
64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001
69324	MARKER	CLERK	67832	1992-01-23	1400	NULL	1001

```
--find recently hired employees of every department. Sort the result-set in descending order by hire date.
select * from Employees e
where hire_date in (Select max(hire_date) from Employees where e.dep_id=dep_id)
order by hire_date desc;
```

Output:

68736	ADNRES	CLERK	67858	1997-05-23	1200	NULL	2001
69324	MARKER	CLERK	67832	1992-01-23	1400	NULL	1001
69000	JULIUS	CLERK	66928	1991-12-03	1050	NULL	3001

```
--find those employees who receive a salary higher than the average salary of their department.
select * from Employees e where salary >(select AVG(salary) from Employees  where e.dep_id=dep_id) order by
dep_id;
```

Output:

68319	KAYLING	PRESIDENT	NULL	1991-11-18	6000	NULL	1001
65646	JONAS	MANAGER	68319	1991-04-02	2957	NULL	2001
67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001
64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001
66928	BLAZE	MANAGER	68319	1991-05-01	2750	NULL	3001

```
--find those employees who earn a commission and receive maximum salary.
select * from Employees where salary in ( select max(Salary) from Employees where commission is not null);
```

Output:

64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001

```
--find those employees who get a commission percent and works as a SALESMAN and earn maximum net salary
select *,(salary+commission) as net_salary from Employees
where job_name = 'SALESMAN'
and (salary+commission ) in (select max(Salary+commission) from Employees where commission is not null ) ;
```

Output:

66564	MADDEN	SALESMAN	66928	1991-09-28	1350	1500	3001	2850

```
--Find those employees who gets a commission and earn the second highest net salary (salary + commission).
select top 1 *,salary+commission as ne_sal from Employees
where  salary+commission not in (Select max(salary+commission) as netsal from Employees where commission is not null) ;
--or
select * from (select *,(salary+commission) as net_salary , DENSE_RANK() over (order by (salary+commission) desc) as netsal_rank
from Employees e where commission is not null)as emp where netsal_rank=2;
```

Output:

64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001	2100
64989	ADELYN	SALESMAN	66928	1991-02-20	1700	400	3001	2100	2

```
--Find those departments where the average salary is less than the averages for all departments.
select dep_id,avg(salary) 'avg' from Employees
group by dep_id
having avg(salary)< (select avg(salary) from Employees );
```

3001	1633

```
-- find the unique department of the employees
 select dep_id from Department where dep_id in (select distinct dep_id from Employees);
```

1001
2001
3001

```
--find those employees who work in the department where KAYLING works.
select  e.emp_name,d.dep_id,d.dep_location,e.salary
from employees e,department d where e.dep_id=d.dep_id
and e.dep_id in
        (select dep_id from employees w  where emp_name = 'KAYLING' AND w.emp_id <> e.emp_id);
```

Output:

CLARE	1001	Sydney	2550
MARKER	1001	Sydney	1400

```
find those employees whose salary grade is greater than the grade of MARKER.
select e.emp_id,e.emp_name,s.grade from Employees e, salary_grade s
where e.salary between s.min_sal and s.max_sal and
s.grade >(select s.grade from Employees e , salary_grade s  where e.emp_name='MARKER' and e.salary between s.min_sal and s.max_sal) ;
```

Output:

64989	ADELYN	3
68454	TUCKER	3
66928	BLAZE	4
67832	CLARE	4
65646	JONAS	4

67858	SCARLET	4
69062	FRANK	4
68319	KAYLING	5

```
--find those employees whose grade same as the grade of TUCKER or experience is more than SANDRINE and who are belonging to SYDNEY or PERTH.
select e.emp_name,d.dep_location,d.dep_id,e.hire_date from Employees e , salary_grade s,Department d
where e.dep_id =d.dep_id
and d.dep_location in ('SYDNEY','PERTH')
and e.salary between s.min_sal and s.max_sal
and (s.grade in (select s.grade from Employees e, salary_grade s where e.emp_name ='TUCKER' and e.salary between s.min_sal and s.max_sal)
        or DATEDIFF(year,hire_date,getdate()) >(Select DATEDIFF(year,hire_date,getdate()) from Employees where emp_name= 'SANDRINE'));
```

Output:

ADELYN	Perth	3001	1991-02-20
TUCKER	Perth	3001	1991-09-08

```
--find those employees whose salary is same as any one of the employee
select  * from employees where salary in(select salary from employees e where employees.emp_id <> e.emp_id);
```

Output:

67858	SCARLET	ANALYST	65646	1997-04-19	3100	NULL	2001
69062	FRANK	ANALYST	65646	1991-12-03	3100	NULL	2001
65271	WADE	SALESMAN	66928	1991-02-22	1350	600	3001
66564	MADDEN	SALESMAN	66928	1991-09-28	1350	1500	3001

```
--find the recently hired employees of department 3001
select *from employees
where hire_date in  (select max(hire_date) from employees where dep_id = 3001)
AND dep_id=3001;
```

Output:

69000	JULIUS	CLERK	66928	1991-12-03	1050	NULL	3001

```
--SQL query to count the number of employees who work as a manager.
select count(*) as manager_count from Employees e
where emp_id in (select manager_id from Employees);
```

Output : 6

```
--SQL query to find those departments where no employee works.
select d.dep_id,COUNT(e.dep_id) as empCount
from Department d  left join employees e
on e.dep_id=d.dep_id
group by  d.dep_id
having count(e.dep_id) = 0;
```

Output:

4001	0