Subquery?

Subquery merupakan suatu query berada di dalam query yang lain, atau bisa juga disebut dengan **nested query**.

Secara teknis, subquery merupakan perintah SELECT yang berada di dalam perintah lain:
1. SELECT
2. INSERT
3. UPDATE
4. DELETE

Implementasi dari subquery
1. WHERE
2. IN and NOT IN
3. FROM
4. EXIST and NOT EXIST

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.com		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Comparison menggunakan Subquery

Comparison biasanya terjadi pada klausa WHERE untuk membandingkan dua nilai.
Subquery dapat digunakan untuk menghasilkan suatu nilai menggunakan perintah SELECT untuk dibandingkan dengan nilai tertentu.

Sintaks

```
non_subquery_operand comparison_operator (subquery)
```

Comparison operator

```
=   >   <   >=   <=   <>   !=   <=>
```

Comparison menggunakan Subquery

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1    -- COMPARISON
2    SELECT lastName, firstName, salary
3    FROM employees
4    WHERE salary >
5        (SELECT MAX(basicSalary)
6        FROM jobs);
```

Hasil

lastName	firstName	salary
Evans	Chris	25000000
Feige	Kevin	27000000

Comparison menggunakan Subquery

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1    -- COMPARISON
2  • SELECT *
3    FROM employees as em
4    WHERE 3 =
5        (SELECT count(*)
6        FROM employees
7        WHERE em.jobId = employees.jobID)
```

Hasil

employeeID	lastName	firstName	email	reportsTo	jobID	Salary
2	Travolta	John	travj@mail.co	NULL	2	19000000
4	Connor	John	connj@mail.co	NULL	2	16000000
6	Rogers	Steve	cap@mail.co	3	2	17500000

Subquery menggunakan ANY, IN, SOME, ALL

Subquery dapat juga digunakan menghasilkan suatu *list,* dimana operand akan mengecek kemunculan suatu nilai berdasarkan data nilai pada *list* tersebut. Operator ANY, IN, SOME, ALL membandingkan nilai dengan setiap nilai yang dikembalikan dari suatu subquery tersebut.

Sintaks

```
operand comparison_operator ANY (subquery)
operand IN (subquery)
operand comparison_operator SOME (subquery)
operand comparison_operator ALL (subquery)
```

Subquery menggunakan ANY

Operator ANY menghasilkan nilai TRUE apabila paling tidak salah satu (ANY) pembandingan nilai dengan hasil subquery menghasilkan nilai TRUE

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1       -- ANY, IN, SOME
2 •     SELECT jobTitle, basicSalary FROM jobs
3       WHERE jobID = ANY
4  ⊖       (SELECT jobID FROM employees
5           WHERE salary>10000000);
```

Hasil

jobTitle	basicSalary
Producer	20000000
Director	15000000
Actor	9000000
Writer	10000000
Executive Producer	22500000

Subquery menggunakan IN

Subquery dapat digunakan dengan operator IN.
Subquery harus mengembalikan satu kolom yang berisi daftar nilai yang dapat digunakan oleh operator IN.

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
SELECT jobTitle, basicSalary
FROM jobs
WHERE jobID IN
        (SELECT jobID
        FROM employees
        WHERE salary>10000000)
```

Hasil

	jobTitle	basicSalary
▶	Producer	20000000
	Director	15000000
	Actor	9000000
	Writer	10000000
	Executive Producer	22500000

IN
sama dengan
= ANY

Subquery menggunakan NOT IN

Operator NOT dapat digunakan untuk membuat negasi dari operator IN.
NOT IN menghasilkan data apabila nilainya tidak ada didalam *list.*

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1    -- ANY, IN, SOME
2 •  SELECT jobTitle, basicSalary FROM jobs
3    WHERE jobID NOT IN
4        (SELECT jobID FROM employees
5        WHERE salary>10000000);
```

Hasil

jobTitle	basicSalary
Fans	0
Stunt Organizer	7500000
VFX Artist	7500000
Stuntman	6000000

Subquery menggunakan SOME

Keyword SOME merupakan alias untuk keyword ANY

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
SELECT jobTitle, basicSalary
FROM jobs
WHERE jobID <> SOME
    (SELECT jobID
    FROM employees
    WHERE salary>10000000)
```

```
SELECT jobTitle, basicSalary
FROM jobs
WHERE jobID <> ANY
    (SELECT jobID
    FROM employees
    WHERE salary>10000000)
```

Hasil

jobTitle	basicSalary
Producer	20000000
Director	15000000
Writer	10000000
Actor	9000000
Fans	0
Executive Producer	22500000
Stunt Organizer	7500000
VFX Artist	7500000
Stuntman	6000000

Subquery menggunakan ALL

Operator ALL menghasilkan nilai TRUE apabila pembandingan untuk setiap nilai dengan hasil subquery menghasilkan nilai TRUE.

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1       -- ANY, IN, SOME
2  •  SELECT * FROM jobs
3       WHERE jobID > ALL
4           (SELECT jobID FROM employees
5               WHERE salary>10000000);
```

Hasil

	jobID	jobTitle	basicSalary
▶	7	Stunt Organizer	7500000
	8	VFX Artist	7500000
	9	Stuntman	6000000

Subqueries using EXISTS

Self join akan mengembalikan semua *record* dari suatu tabel yang digabungkan dengan dirinya sendiri.
Operator EXISTS digunakan untuk memeriksa keberadaan baris/*record* yang dihasilkan query terhadap yang dihasilkan subquery.
Operator EXISTS merupakan operator Boolean. Operator EXIST menghasilkan nilai TRUE apabila subquery menghasilkan paling tidak satu baris/*record*.
Sintaks

```
1    -- EXISTS
2    SELECT column1,column2
3    FROM table1
4    WHERE EXISTS
5        (SELECT column1,column2
6        FROM table2
7        WHERE conditions);
```

Subquery menggunakan EXISTS

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```sql
1    -- EXISTS
2    SELECT firstName,lastName,jobID,salary
3    FROM employees
4    WHERE EXISTS
5        (SELECT jobID
6        FROM jobs
7        WHERE employees.jobID=jobs.jobID AND employees.salary < 1.1*jobs.basicSalary);
```

```sql
SELECT firstName, lastName, jobID, salary
FROM employees
WHERE jobID IN
        (SELECT jobID
        FROM jobs
        WHERE employees.jobID = jobs.jobID AND employees.salary < 1.1 * jobs.basicSalary);
```

Hasil

firstName	lastName	jobID	salary
John	Carter	1	21500000
Peggy	Carter	3	10000000
John	Connor	2	16000000
Liam	Hemsworth	4	9500000

Subqueries menggunakan NOT EXISTS

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1   -- EXISTS
2   SELECT firstName,lastName,jobID
3   FROM employees
4   WHERE NOT EXISTS
5       (SELECT jobID
6       FROM jobs
7       WHERE employees.jobID=jobs.jobID);
```

```
SELECT firstName, lastName, jobID
FROM employees
WHERE jobID NOT IN
        (SELECT jobID
        FROM jobs
        WHERE employees.jobID = jobs.jobID);
```

Hasil

firstName	lastName	jobID
George	Miller	NULL

Subquery dalam klausa FROM

Klausa FROM biasanya mengacu ke nama tabel.
Seperti yang kita ketahui bahwa suatu query menggunakan SELECT untuk menghasilkan suatu tabel, kita dapat menggunakan hasil subquery SELECT sebagai tabel pada klausa FROM.

Sintaks

```
1    -- Subquery in FROM
2    SELECT column1,column2
3    FROM (subquery)
4    WHERE conditions;
```

Subquery dalam klausa FROM

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1       -- Subquery in FROM
2   •   SELECT basic,recent,title
3       FROM
4         (SELECT jo.basicSalary as basic,em.salary as recent, jo.jobTitle as title
5          FROM jobs as jo
6          INNER JOIN employees as em
7          WHERE jo.jobID = em.jobID
8         ) as salaryJob
9       WHERE recent>10000000;
```

Hasil

basic	recent	title
20000000	21500000	Producer
15000000	19000000	Director
15000000	16000000	Director
9000000	15000000	Actor
15000000	17500000	Director
20000000	25000000	Producer
10000000	12000000	Writer
22500000	27000000	Executive Producer

Subquery dalam INSERT dan UPDATE

Hasil dari subquery dapat digunakan sebagai parameter pada klausa INSERT dan UPDATE.

Sintaks

```
1    -- Subquery in INSERT
2    INSERT INTO table_name(column1,column2)
3    VALUES
4        (subquery);
```

Subquery dalam INSERT

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1      -- Subquery in INSERT
2  ●   INSERT INTO jobs(jobTitle,basicSalary)
3      VALUES(
4          'Vice President',
5          (SELECT 1.25*max(salary) FROM employees)
6          );
```

Hasil

jobID	jobTitle	basicSalary
1	Producer	20000000
2	Director	15000000
3	Writer	10000000
4	Actor	9000000
5	Fans	0
6	Executive Producer	22500000
7	Stunt Organizer	7500000
8	VFX Artist	7500000
9	Stuntman	6000000
10	Vice President	33750000

Subquery dalam UPDATE

Tabel Jobs

jobID	jobTitle	basicSalary
1	Producer	20.000.000
2	Director	15.000.000
3	Writer	10.000.000
4	Actor	9.000.000
5	Fans	-
6	Executive Producer	22.500.000
7	Stunt Organizer	7.500.000
8	VFX Artist	7.500.000
9	Stuntman	6.000.000

Tabel Employees

employeeID	lastName	firstName	email	reportsTo	jobID	salary
1	Carter	John	carterj@mail.co		1	21.500.000
2	Travolta	John	travj@mail.co		2	19.000.000
3	Carter	Peggy	carterp@mail.co		3	10.000.000
4	Connor	John	connj@mail.co		2	16.000.000
5	Johnson	Rian	jrian@mail.co		4	15.000.000
6	Rogers	Steve	cap@mail.co	3	2	17.500.000
7	Evans	Chris	evanc@mail.co	3	1	25.000.000
8	Pratt	Chris	prc@mail.co	1	5	100.000
9	Hemsworth	Chris	lebowski@mail.co	2	3	12.000.000
10	Hemsworth	Liam	hliam@mail.co		4	9.500.000
11	Feige	Kevin	kfeige@mail.co		6	27.000.000
12	Miller	George	travj@mail.co			-

Query

```
1       -- Subquery in UPDATE
2   •   UPDATE jobs
3       SET basicSalary = (
4           SELECT 0.1*MIN(salary)
5           FROM employees
6           )
7       WHERE basicSalary = 0 AND
8           jobID IN (SELECT DISTINCT jobID FROM employees);
9   •   SELECT * FROM jobs;
```

Hasil

jobID	jobTitle	basicSalary
1	Producer	20000000
2	Director	15000000
3	Writer	10000000
4	Actor	9000000
5	Fans	10000
6	Executive Producer	22500000
7	Stunt Organizer	7500000
8	VFX Artist	7500000
9	Stuntman	6000000
10	Vice President	33750000

Thank you