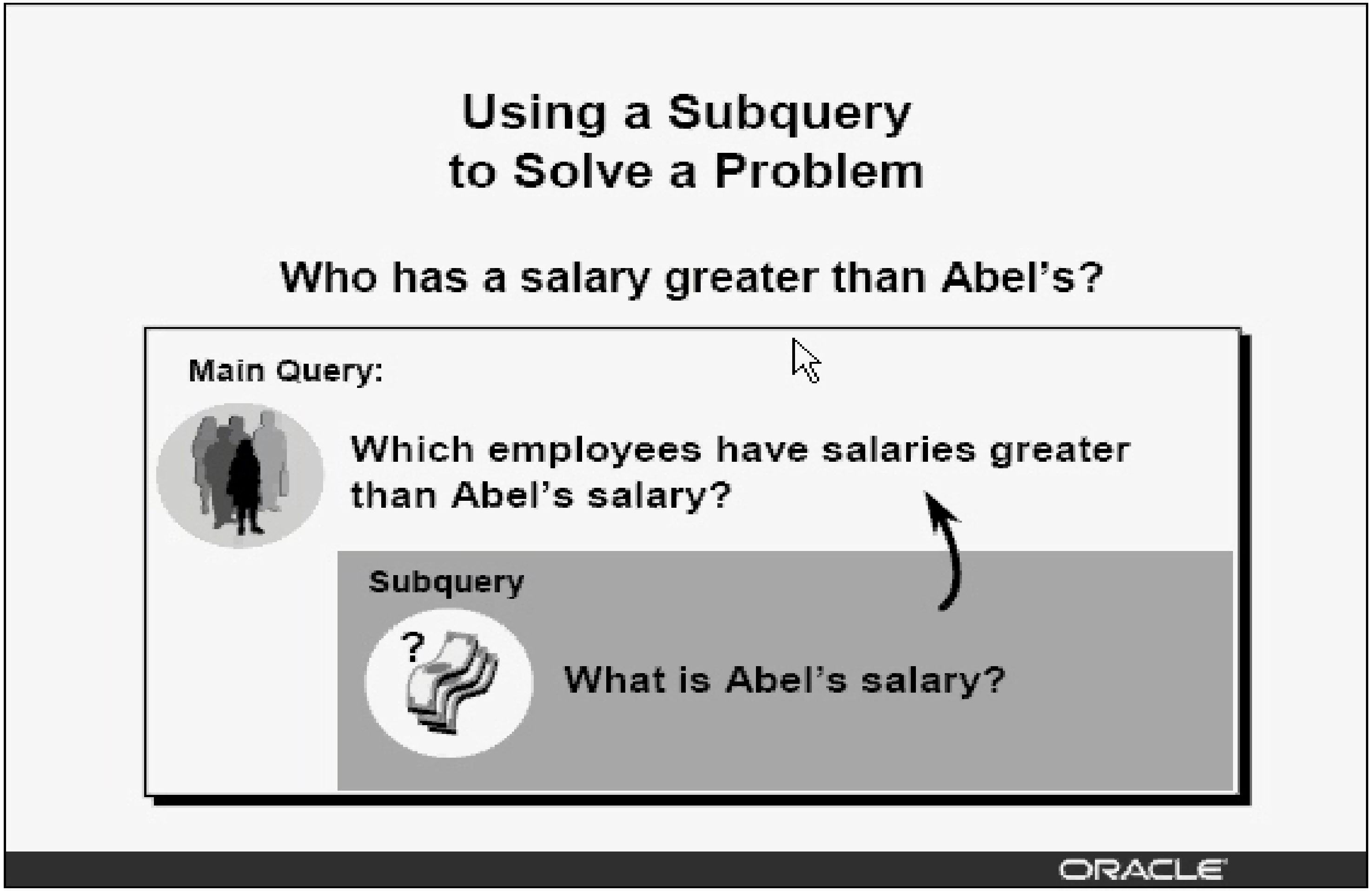
ORACLE®

Subquery



Using a Subquery to Solve a Problem

Who has a salary greater than Abel's?

Main Query:

Which employees have salaries greater than Abel's salary?

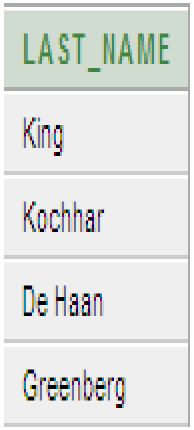
Subquery

What is Abel's salary?

Subquery

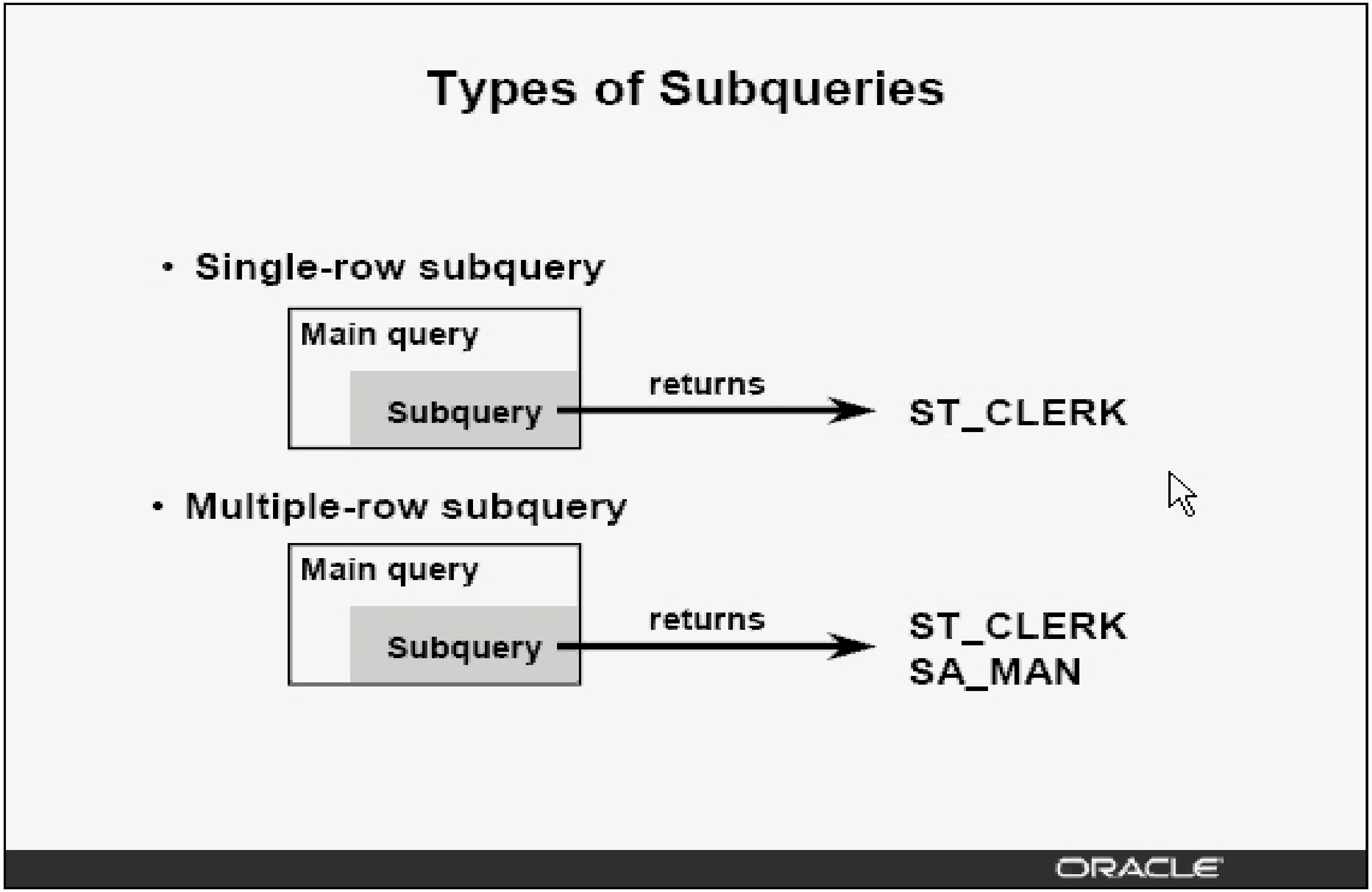
```
SELECT last_name
FROM employees
WHERE salary>(SELECT salary FROM employees where last_name='Abel')
```

>(ئەنجام)



LAST_NAME
King
Kochhar
De Haan
Greenberg

Subquery

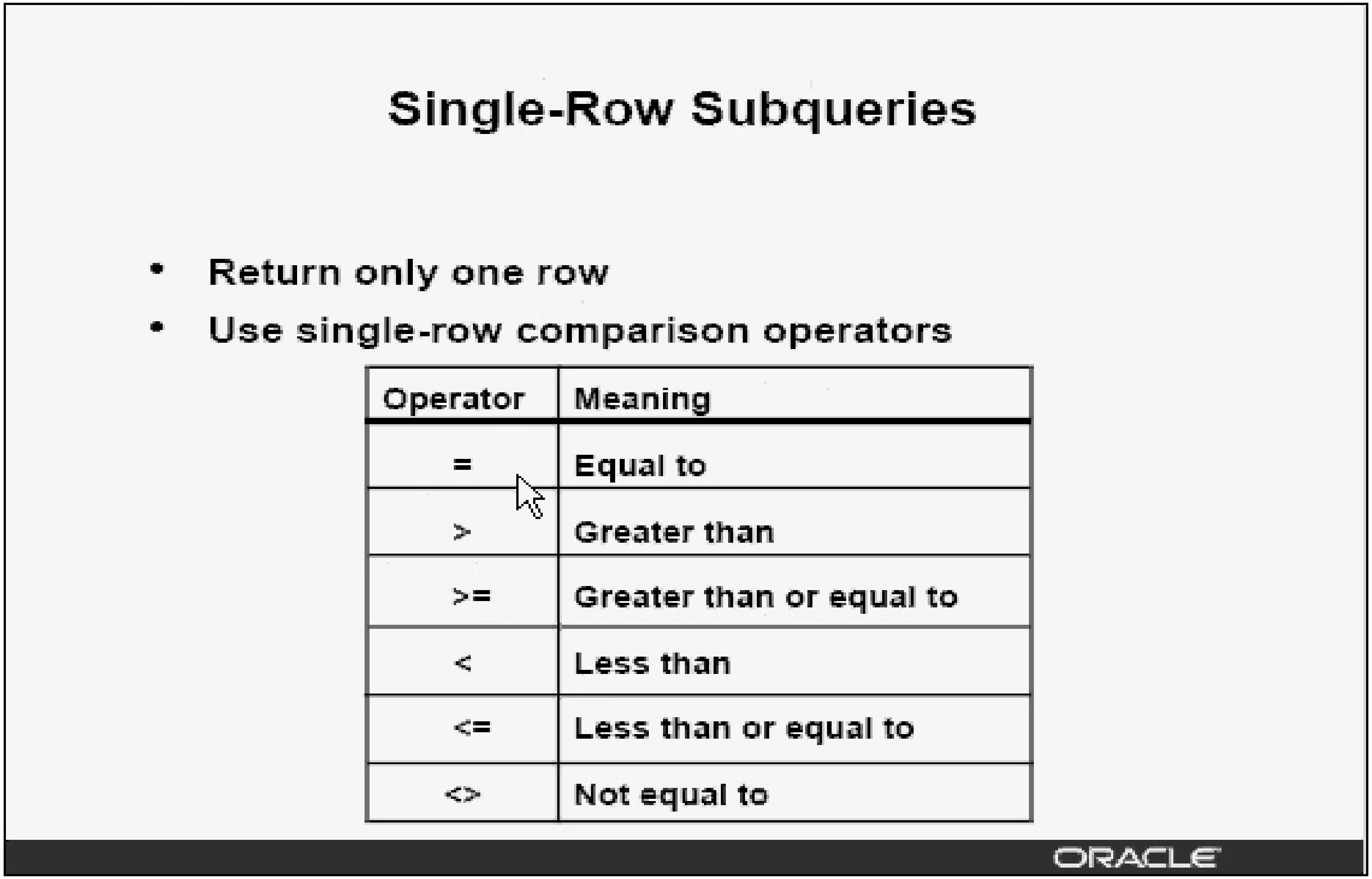


Types of Subqueries

- Single-row subquery

- Multiple-row subquery

Subquery



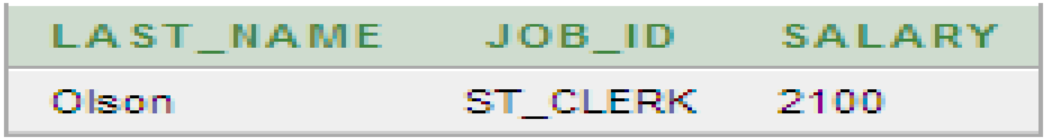
Single-Row Subqueries

- Return only one row
- Use single-row comparison operators

Operator	Meaning
=	Equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>	Not equal to

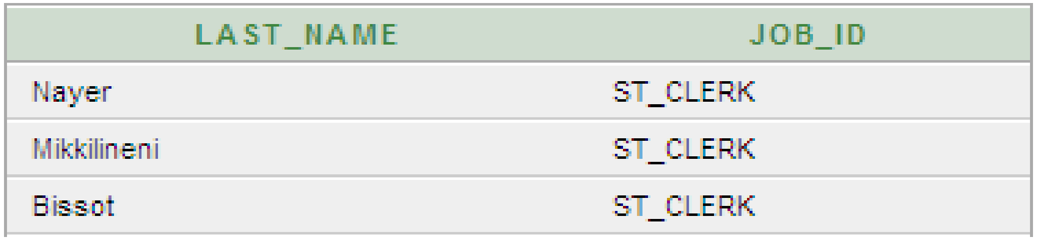
Subquery

```
SELECT last_name,job_id,salary
FROM employees
WHERE salary=(SELECT MIN(salary) FROM employees)
```



LAST_NAME	JOB_ID	SALARY
Olson	ST_CLERK	2100

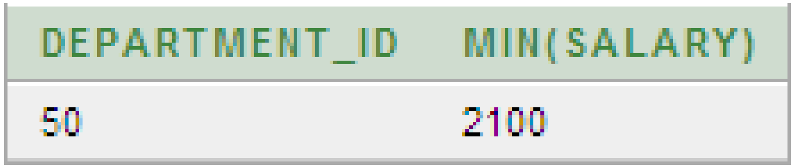
```
SELECT last_name,job_id
FROM employees
WHERE job_id=(SELECT job_id FROM employees where
employee_id=141)
AND salary>(SELECT salary FROM employees where employee_id=143)
```



LAST_NAME	JOB_ID
Nayer	ST_CLERK
Mikkilineni	ST_CLERK
Bissot	ST_CLERK

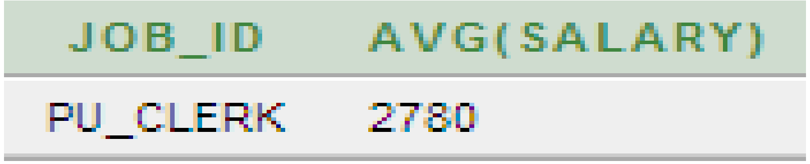
Subquery

```
SELECT department_id,MIN(salary)
FROM employees
GROUP BY department_id
HAVING MIN(salary)=(SELECT MIN(salary) FROM employees WHERE
department_id=50)
```



DEPARTMENT_ID	MIN(SALARY)
50	2100

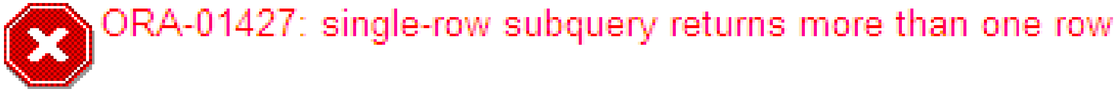
```
SELECT job_id,AVG(salary)
FROM employees
GROUP BY job_id
HAVING AVG(salary)=(SELECT MIN(AVG(salary)) FROM employees
GROUP BY job_id)
```



JOB_ID	AVG(SALARY)
PU_CLERK	2780

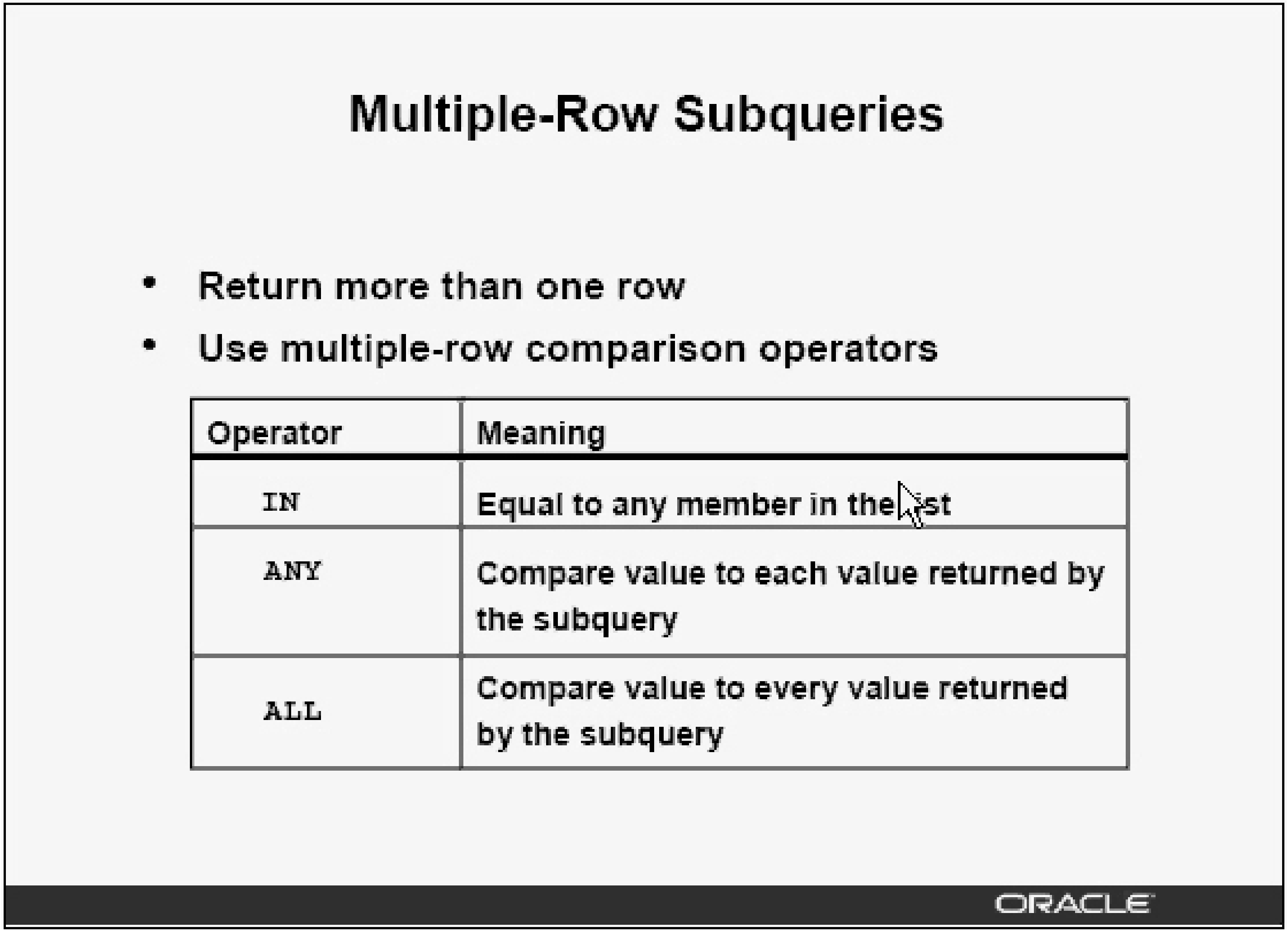
Subquery

```
SELECT employee_id,last_name
FROM employees
WHERE salary= (SELECT MIN(salary) FROM employees GROUP BY department_id)
```



ORA-01427: single-row subquery returns more than one row

Multiple Row Subqueries



Multiple-Row Subqueries

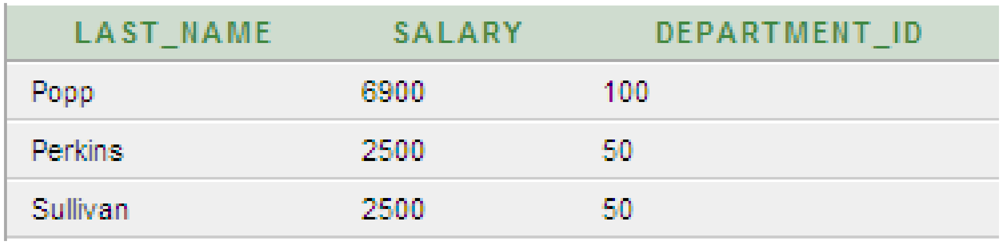
- Return more than one row
- Use multiple-row comparison operators

Operator	Meaning
`IN`	Equal to any member in the list
`ANY`	Compare value to each value returned by the subquery
`ALL`	Compare value to every value returned by the subquery

Example

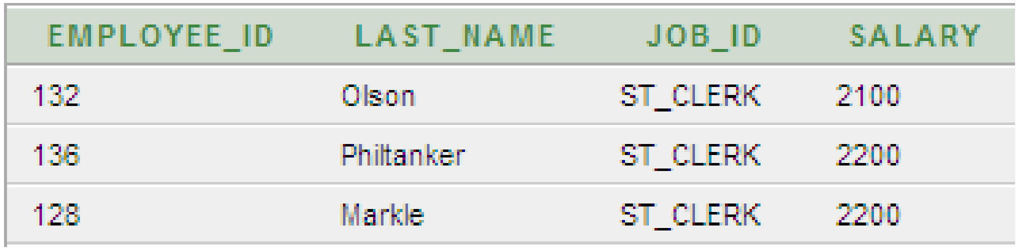
Multiple Row Subqueries

```
SELECT last_name,salary,department_id
FROM employees
WHERE salary IN (SELECT MIN(salary) FROM employees GROUP BY department_id)
```

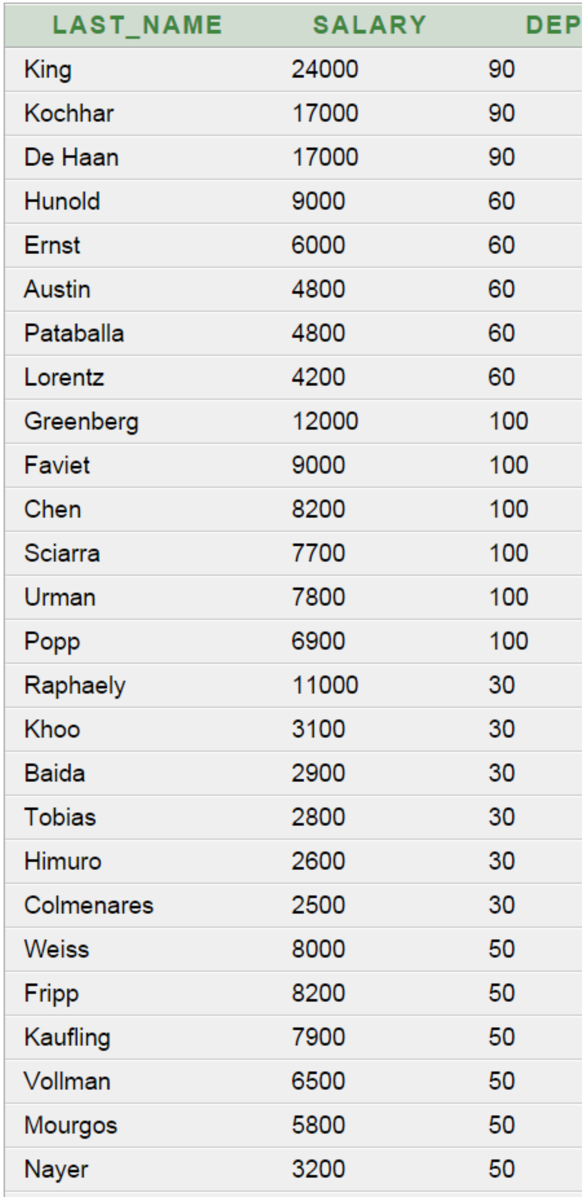


LAST_NAME	SALARY	DEPARTMENT_ID
Popp	6900	100
Perkins	2500	50
Sullivan	2500	50

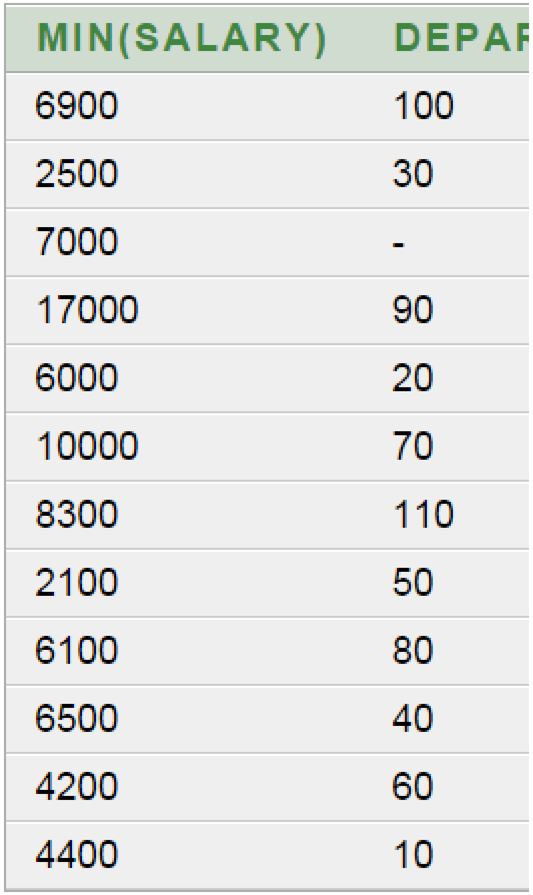
```
SELECT employee_id,last_name,job_id,salary
FROM employees
WHERE salary < ANY (SELECT salary FROM employees WHERE job_id='IT_PROG') AND job_id<> 'IT_PROG'
```



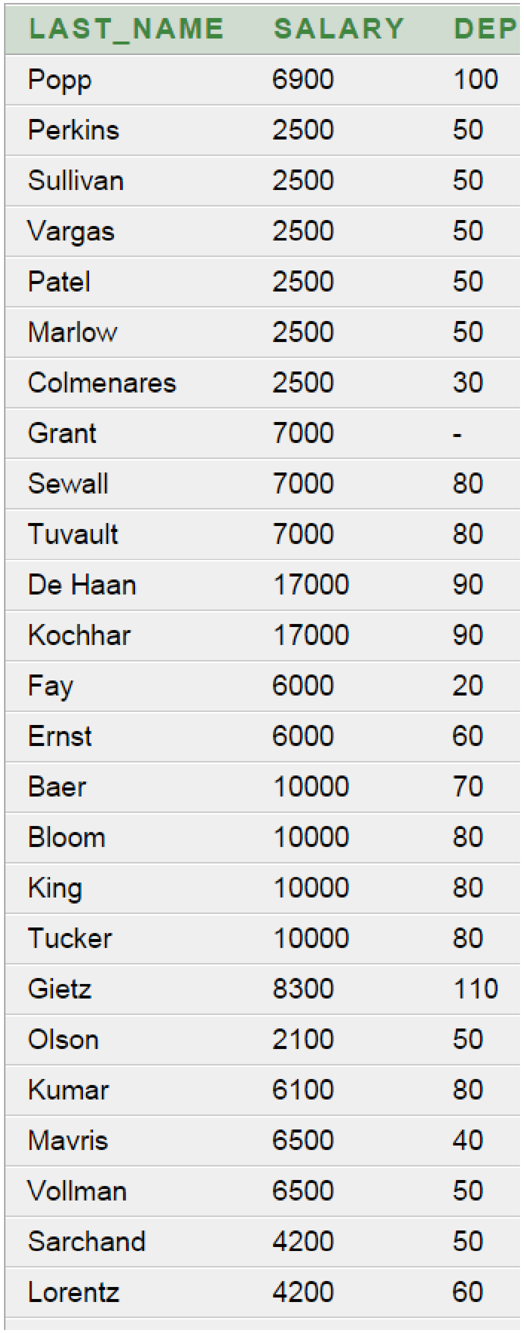
EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
132	Olson	ST_CLERK	2100
136	Philtanker	ST_CLERK	2200
128	Markle	ST_CLERK	2200



LAST_NAME	SALARY	DEP
King	24000	90
Kochhar	17000	90
De Haan	17000	90
Hunold	9000	60
Ernst	6000	60
Austin	4800	60
Pataballa	4800	60
Lorentz	4200	60
Greenberg	12000	100
Faviet	9000	100
Chen	8200	100
Sciarra	7700	100
Urman	7800	100
Popp	6900	100
Raphaely	11000	30
Khoo	3100	30
Baida	2900	30
Tobias	2800	30
Himuro	2600	30
Colmenares	2500	30
Weiss	8000	50
Fripp	8200	50
Kaufling	7900	50
Vollman	6500	50
Mourgos	5800	50
Nayer	3200	50



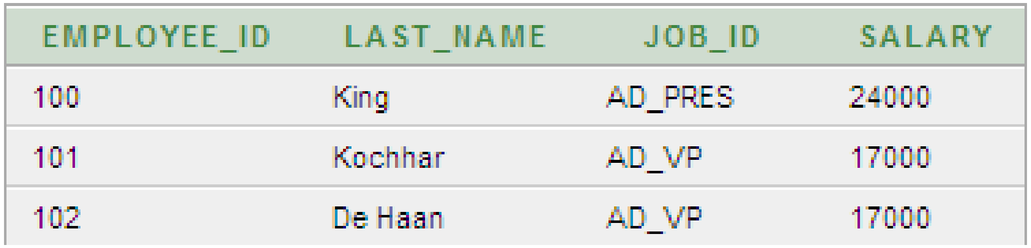
MIN(SALARY)	DEPAR
6900	100
2500	30
7000	-
17000	90
6000	20
10000	70
8300	110
2100	50
6100	80
6500	40
4200	60
4400	10



LAST_NAME	SALARY	DEP
Popp	6900	100
Perkins	2500	50
Sullivan	2500	50
Vargas	2500	50
Patel	2500	50
Marlow	2500	50
Colmenares	2500	30
Grant	7000	-
Sewall	7000	80
Tuvault	7000	80
De Haan	17000	90
Kochhar	17000	90
Fay	6000	20
Ernst	6000	60
Baer	10000	70
Bloom	10000	80
King	10000	80
Tucker	10000	80
Gietz	8300	110
Olson	2100	50
Kumar	6100	80
Mavris	6500	40
Vollman	6500	50
Sarchand	4200	50
Lorentz	4200	60

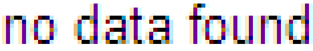
Multiple Row Subqueries

```
SELECT employee_id,last_name,job_id,salary
FROM employees
WHERE salary <> ALL (SELECT salary FROM employees WHERE
job_id='IT_PROG') AND job_id<> 'IT_PROG'
```



EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
100	King	AD_PRES	24000
101	Kochhar	AD_VP	17000
102	De Haan	AD_VP	17000

```
SELECT emp.last_name
FROM employees emp
WHERE emp.employee_id NOT IN (SELECT mgr.manager_id FROM
employees mgr)
```



```
no data found
```

Excercise

```
SELECT last_name,hire_date
FROM employees emp
WHERE department_id = (SELECT department_id FROM employees
WHERE last_name='Zlotkey')
AND last_name <> 'Zlotkey'
```

```
SELECT employee_id,last_name
FROM employees emp
WHERE salary > (SELECT AVG(salary) FROM employees )
ORDER BY salary
```

```
SELECT employee_id,last_name
FROM employees
WHERE department_id IN (SELECT department_id FROM employees
WHERE last_name like '%u%' )
```

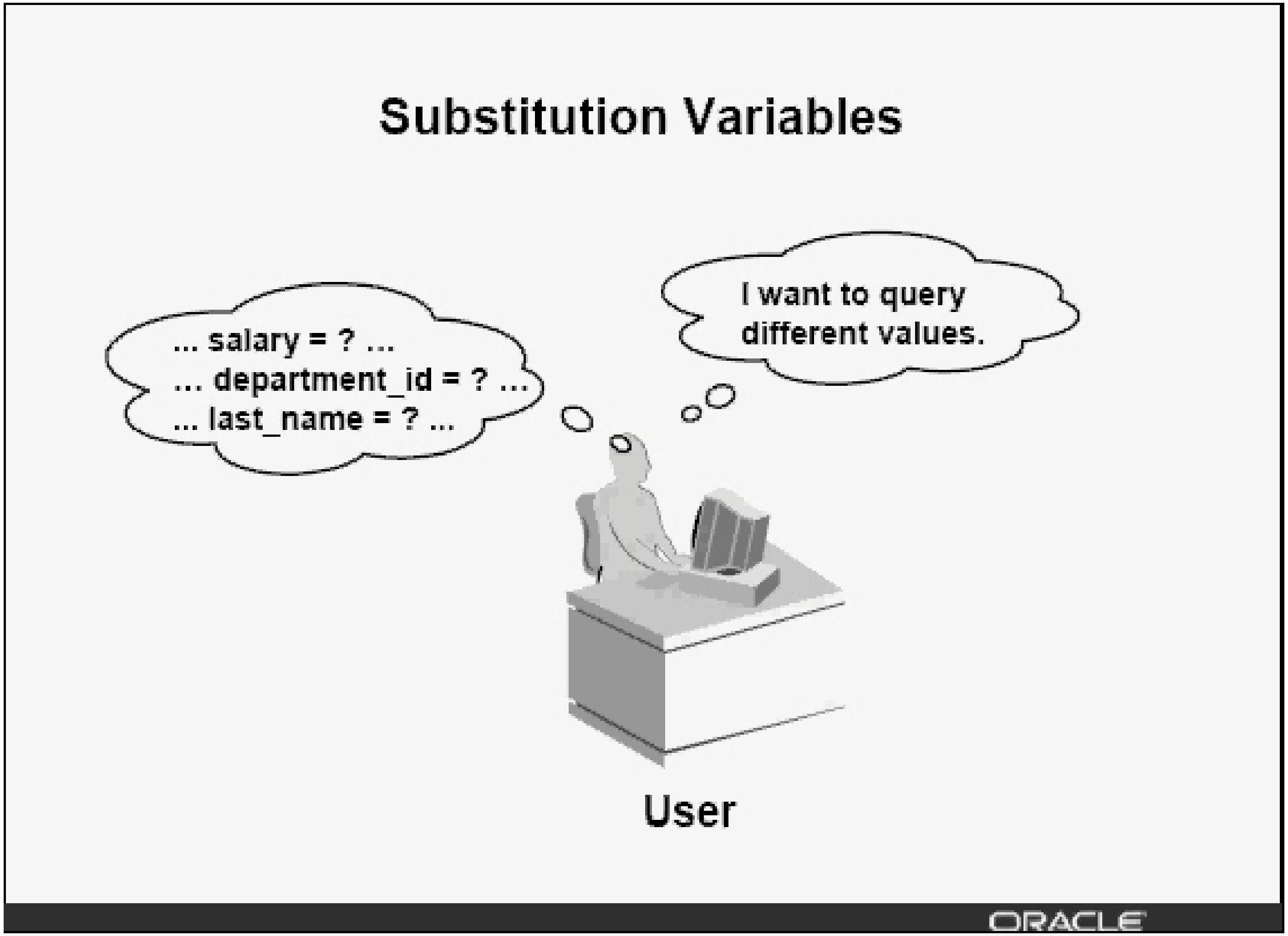
Excercise

```
SELECT last_name,department_id,job_id
FROM employees
WHERE department_id IN (SELECT department_id FROM departments
WHERE location_id=1700 )

SELECT last_name,salary
FROM employees
WHERE manager_id IN (SELECT employee_id FROM employees WHERE
last_name='King' )

SELECT department_id,last_name,job_id
FROM employees
WHERE department_id IN (SELECT department_id FROM departments
WHERE department_name='Executive')
```

Excercise

```
SELECT employee_id,last_name,salary
FROM employees
WHERE salary >(SELECT AVG(salary) FROM employees) AND
department_id IN (SELECT department_id FROM
employees WHERE last_name LIKE '%u%')
```

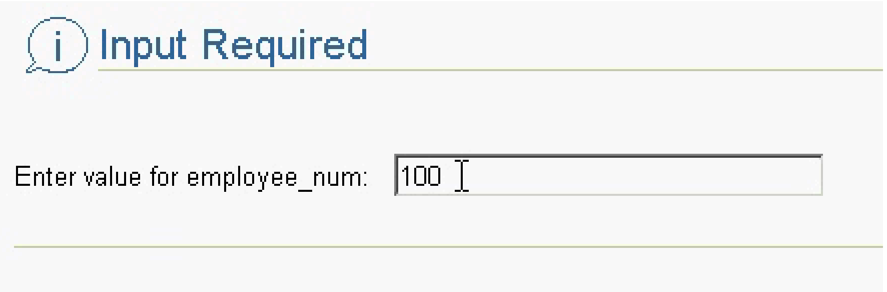
Substitution Variables



Substitution Variables

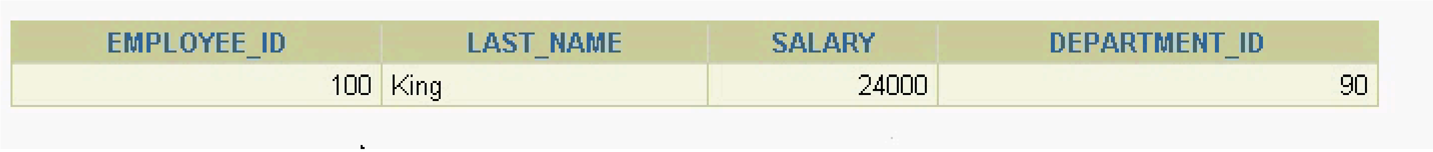
Substitution Variables

```
SELECT employee_id,last_name,salary,department_id
FROM employees
WHERE employee_id=:employee_num
```



Input Required

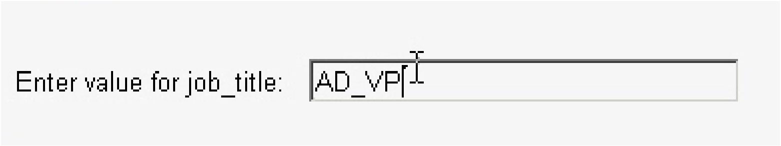
Enter value for employee_num: 100



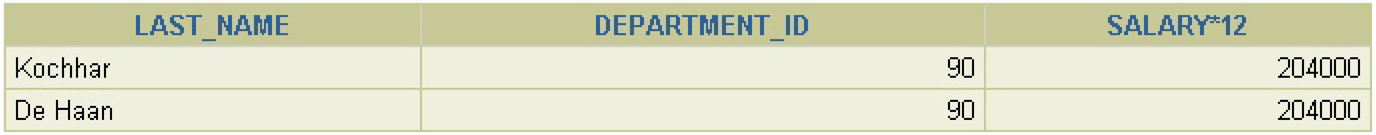
EMPLOYEE_ID	LAST_NAME	SALARY	DEPARTMENT_ID
100	King	24000	90

Substitution Variables

```
SELECT last_name,department_id, salary*12
FROM employees
WHERE job_id=:job_title
```

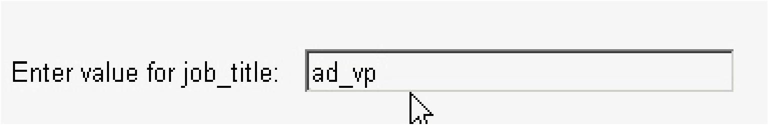


Enter value for job_title: AD_VP



LAST_NAME	DEPARTMENT_ID	SALARY*12
Kochhar	90	204000
De Haan	90	204000

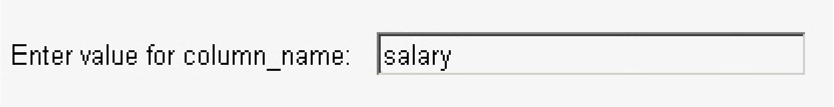
```
SELECT last_name,department_id, salary*12
FROM employees
WHERE job_id=UPPER(:job_title)
```



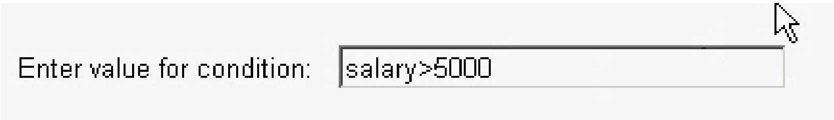
Enter value for job_title: ad_vp

Substitution Variables

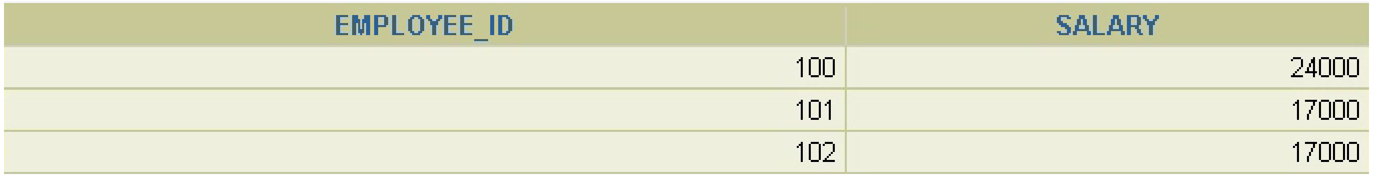
```
SELECT employee_id, &column_name
FROM employees
WHERE &condition
```



Enter value for column_name: salary



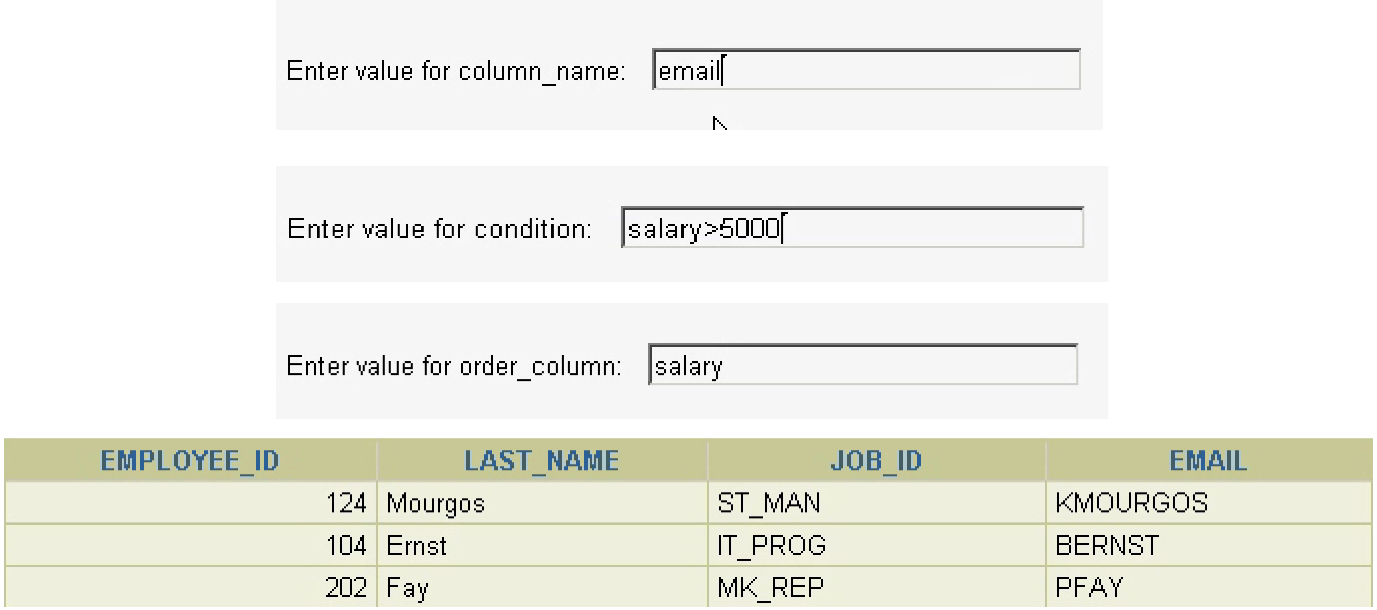
Enter value for condition: salary>5000



EMPLOYEE_ID	SALARY
100	24000
101	17000
102	17000

Substitution Variables

```
SELECT employee_id,last_name,job_id, &column_name
FROM employees
WHERE &condition
ORDER BY &order_column
```



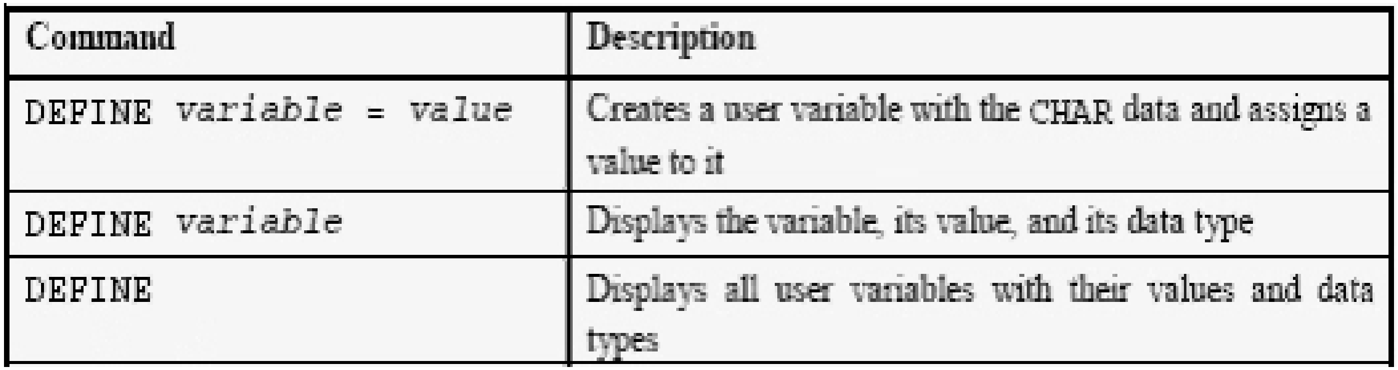
Enter value for column_name: email

Enter value for condition: salary>5000

Enter value for order_column: salary

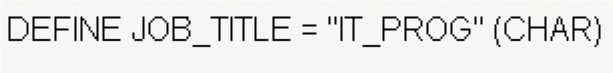
EMPLOYEE_ID	LAST_NAME	JOB_ID	EMAIL
124	Mourgos	ST_MAN	KMOURGOS
104	Ernst	IT_PROG	BERNST
202	Fay	MK_REP	PFAY

DEFINE Variable



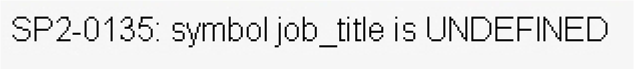
Command	Description
DEFINE *variable* = *value*	Creates a user variable with the CHAR data and assigns a value to it
DEFINE *variable*	Displays the variable, its value, and its data type
DEFINE	Displays all user variables with their values and data types

DEFINE job_title=‘IT_PROG’
DEFINE job_title



```
DEFINE JOB_TITLE = "IT_PROG" (CHAR)
```

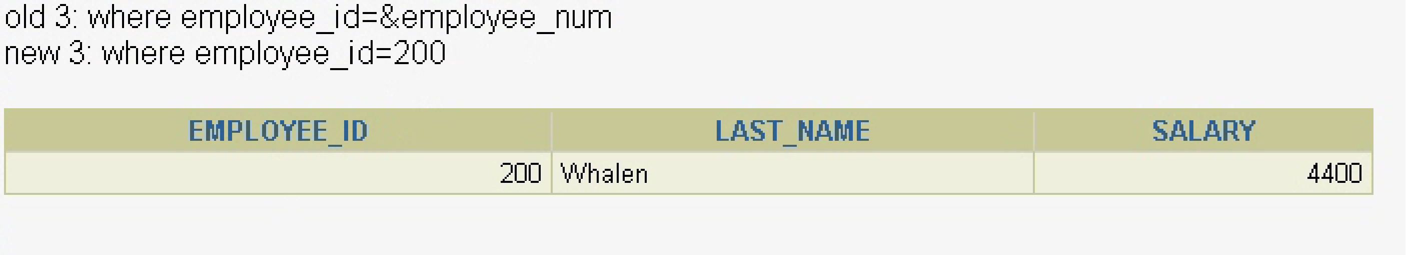
UNDEFINE job_title
DEFINE job_title



```
SP2-0135: symbol job_title is UNDEFINED
```

DEFINE Variable

```
DEFINE employee_num=200
SELECT employee_id,last_name,salary
FROM employees
WHERE employee_id=&employee_num
```

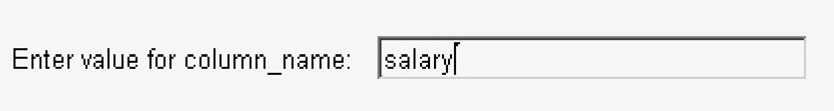


```
old 3: where employee_id=&employee_num
new 3: where employee_id=200
```

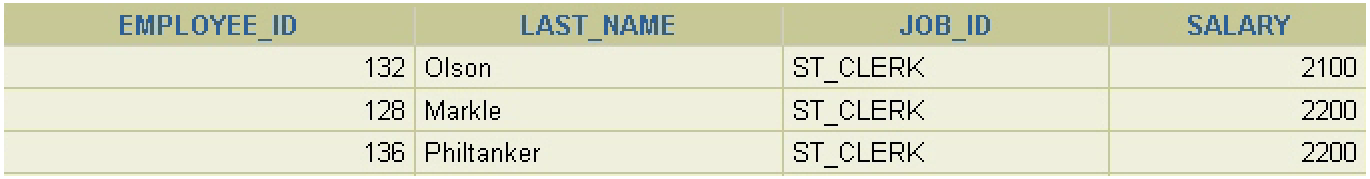
EMPLOYEE_ID	LAST_NAME	SALARY
200	Whalen	4400

DEFINE Variable

```
SELECT employee_id,last_name,job_id, &&column_name
FROM employees
ORDER BY &order_column
```



Enter value for column_name: salary



EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
132	Olson	ST_CLERK	2100
128	Markle	ST_CLERK	2200
136	Philtanker	ST_CLERK	2200

After that if you execute the same query more than one times It does not ask for column_name

```
UNDEFINE column_name
```

You must undefine column_name for new value