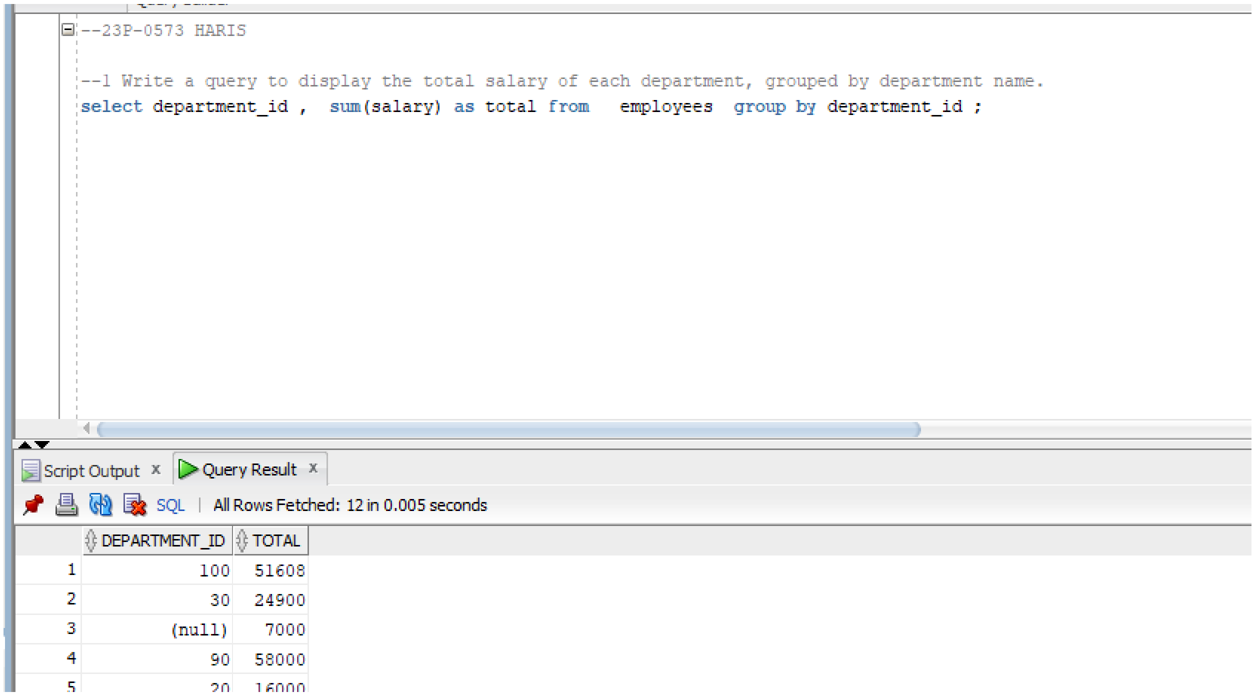
Name : Haris **Roll # 23P-0573**

--23P-0573 HARIS

--1 Write a query to display the total salary of each department, grouped by department name. select department_id , sum(salary) as total from employees group by department_id ;

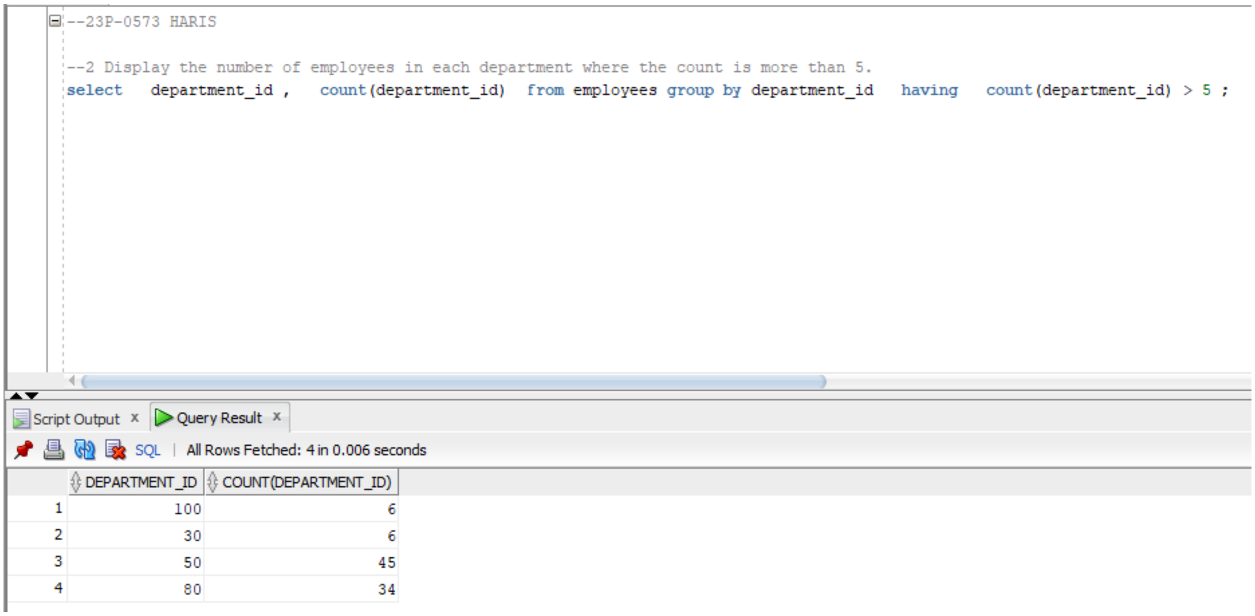


```
--23P-0573 HARIS

--1 Write a query to display the total salary of each department, grouped by department name.
select department_id ,  sum(salary) as total from   employees  group by department_id ;
```

All Rows Fetched: 12 in 0.005 seconds

	DEPARTMENT_ID	TOTAL
1	100	51608
2	30	24900
3	(null)	7000
4	90	58000
5	20	16000

--2 Display the number of employees in each department where the count is more than 5. select department_id , count(department_id) from employees group by department_id having count(department_id) > 5 ;

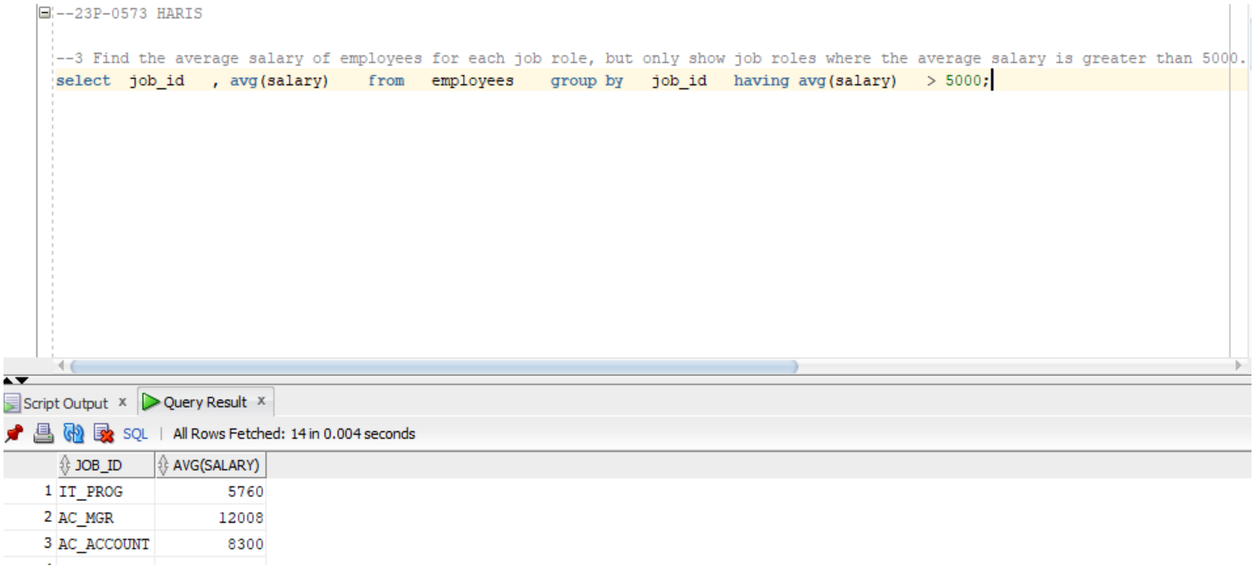


```
--23P-0573 HARIS

--2 Display the number of employees in each department where the count is more than 5.
select    department_id ,    count(department_id)  from employees group by department_id    having    count(department_id) > 5 ;
```

All Rows Fetched: 4 in 0.006 seconds

	DEPARTMENT_ID	COUNT(DEPARTMENT_ID)
1	100	6
2	30	6
3	50	45
4	80	34

--3 Find the average salary of employees for each job role, but only show job roles where the average salary is greater than 5000. select job_id , avg(salary) from employees group by job_id having avg(salary) > 5000;

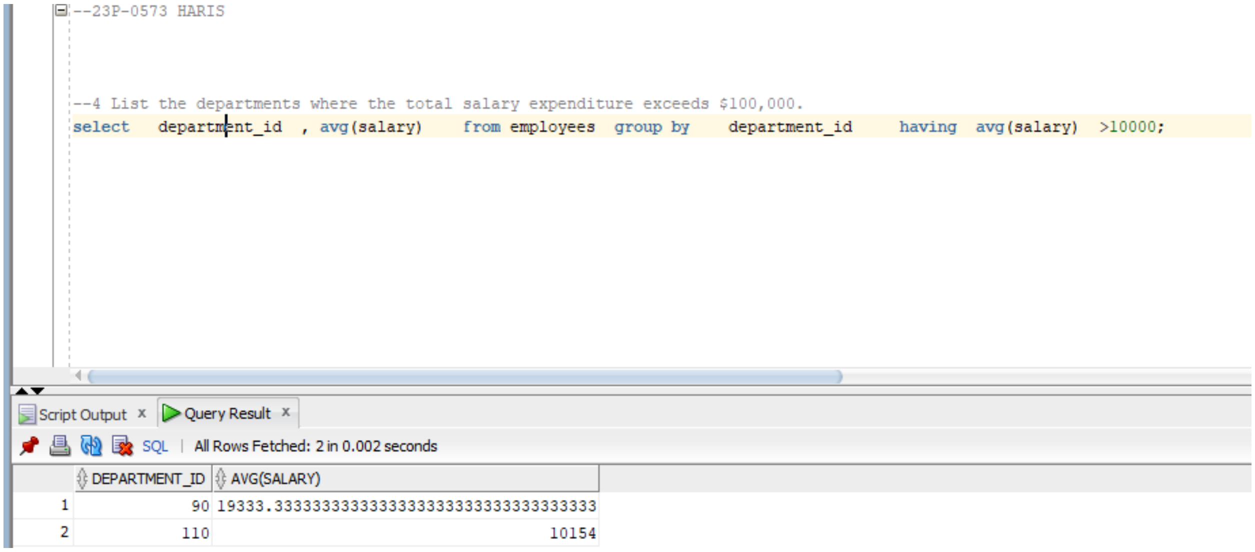


```
--23P-0573 HARIS

--3 Find the average salary of employees for each job role, but only show job roles where the average salary is greater than 5000.
select  job_id   , avg(salary)    from    employees    group by    job_id    having avg(salary)    > 5000;
```

All Rows Fetched: 14 in 0.004 seconds

	JOB_ID	AVG(SALARY)
1	IT_PROG	5760
2	AC_MGR	12008
3	AC_ACCOUNT	8300

--4 List the departments where the total salary expenditure exceeds $100,000. select department_id , avg(salary) from employees group by department_id having avg(salary) >10000;

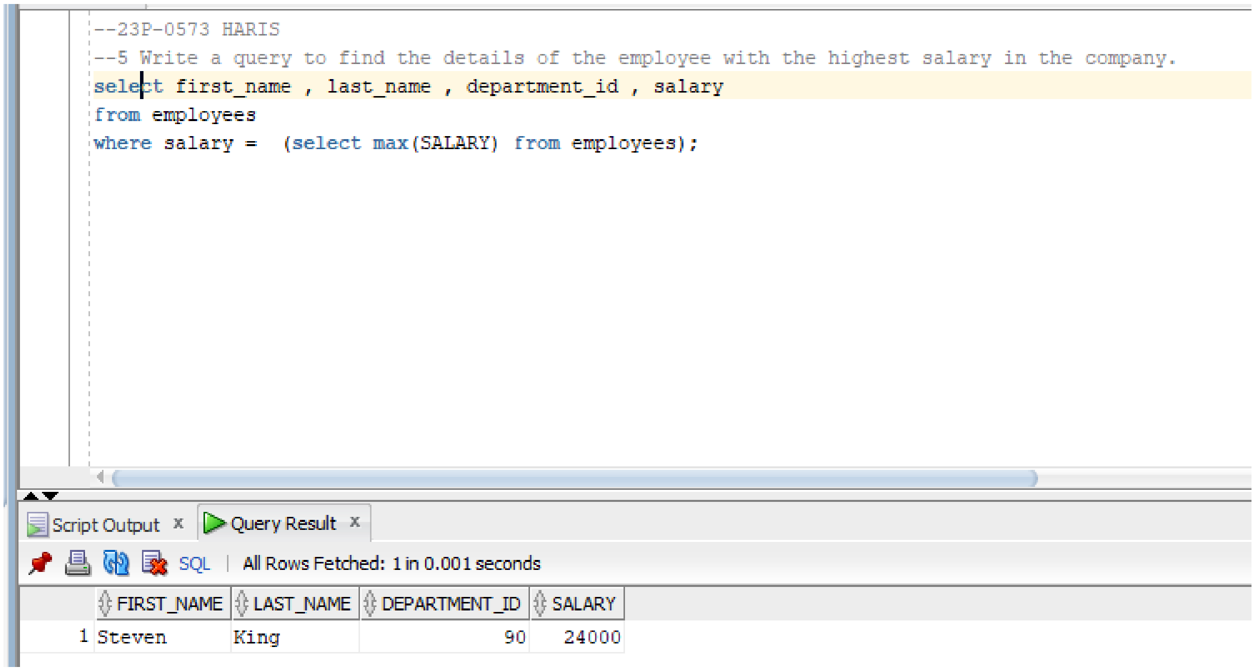


```
--23P-0573 HARIS


--4 List the departments where the total salary expenditure exceeds $100,000.
select   department_id  , avg(salary)    from employees  group by    department_id     having  avg(salary)  >10000;
```

All Rows Fetched: 2 in 0.002 seconds

	DEPARTMENT_ID	AVG(SALARY)
1	90	19333.33333333333333333333333333333333333
2	110	10154

--5 Write a query to find the details of the employee with the highest salary in the company. select first_name , last_name , department_id , salary from employees where salary = (select max(SALARY) from employees);

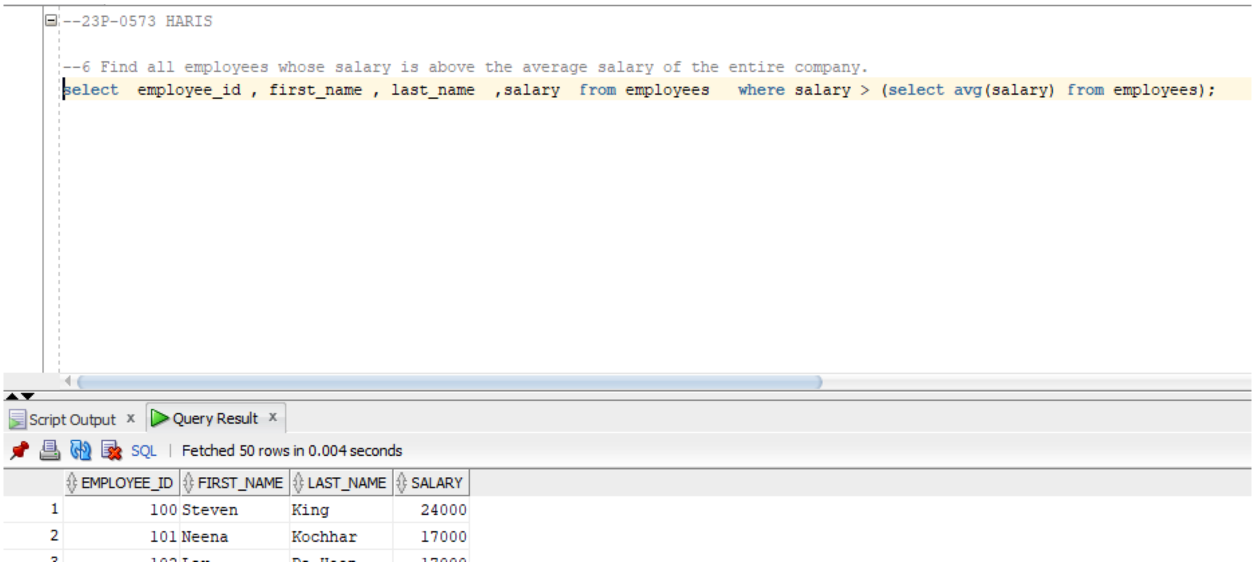


```
--23P-0573 HARIS
--5 Write a query to find the details of the employee with the highest salary in the company.
select first_name , last_name , department_id , salary
from employees
where salary =  (select max(SALARY) from employees);
```

All Rows Fetched: 1 in 0.001 seconds

	FIRST_NAME	LAST_NAME	DEPARTMENT_ID	SALARY
1	Steven	King	90	24000

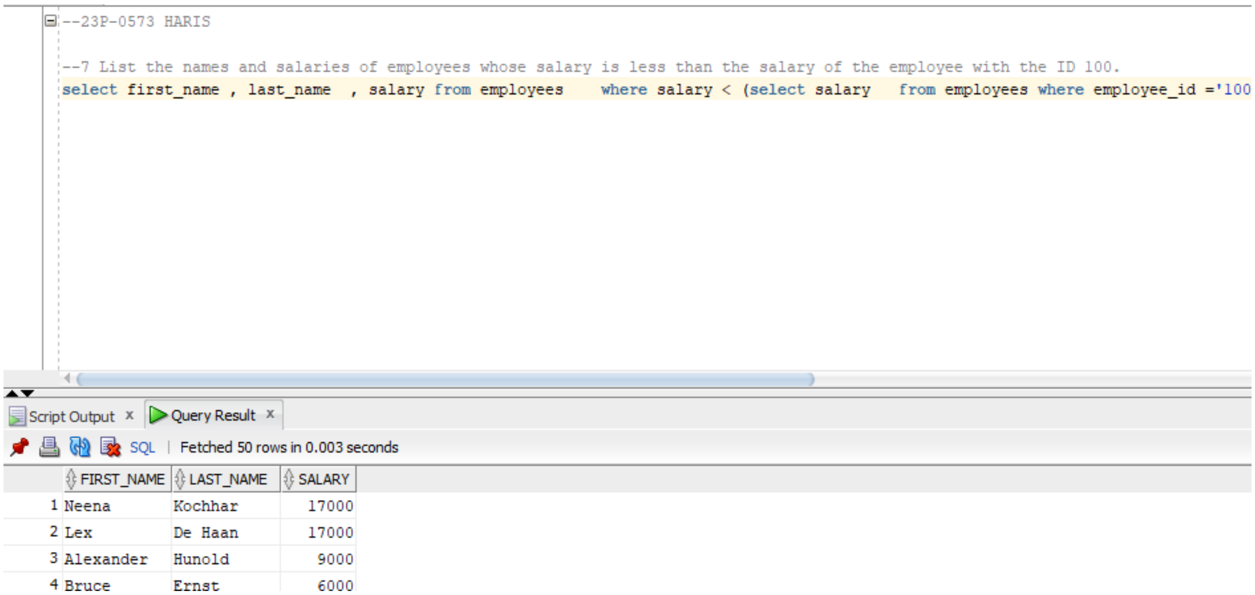
--6 Find all employees whose salary is above the average salary of the entire company. select employee_id , first_name , last_name ,salary from employees where salary > (select avg(salary) from employees);



```
--23P-0573 HARIS

--6 Find all employees whose salary is above the average salary of the entire company.
select  employee_id , first_name , last_name  ,salary  from employees   where salary > (select avg(salary) from employees);
```

```
Fetched 50 rows in 0.004 seconds
   EMPLOYEE_ID FIRST_NAME LAST_NAME SALARY
1          100 Steven     King       24000
2          101 Neena      Kochhar    17000
```

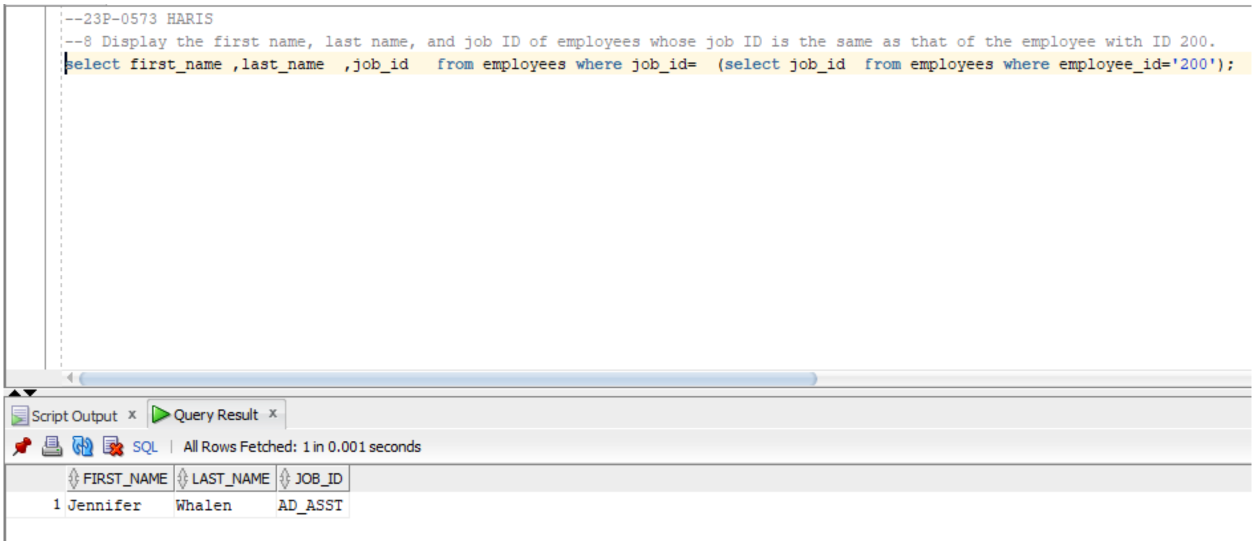
--7 List the names and salaries of employees whose salary is less than the salary of the employee with the ID 100. select first_name , last_name , salary from employees where salary < (select salary from employees where employee_id ='100');



```
--23P-0573 HARIS

--7 List the names and salaries of employees whose salary is less than the salary of the employee with the ID 100.
select first_name , last_name  , salary from employees    where salary < (select salary   from employees where employee_id ='100
```

```
Fetched 50 rows in 0.003 seconds
  FIRST_NAME LAST_NAME SALARY
1 Neena      Kochhar    17000
2 Lex        De Haan    17000
3 Alexander  Hunold      9000
4 Bruce      Ernst       6000
```

--8 Display the first name, last name, and job ID of employees whose job ID is the same as that of the employee with ID 200. select first_name ,last_name ,job_id from employees where job_id= (select job_id from employees where employee_id='200');

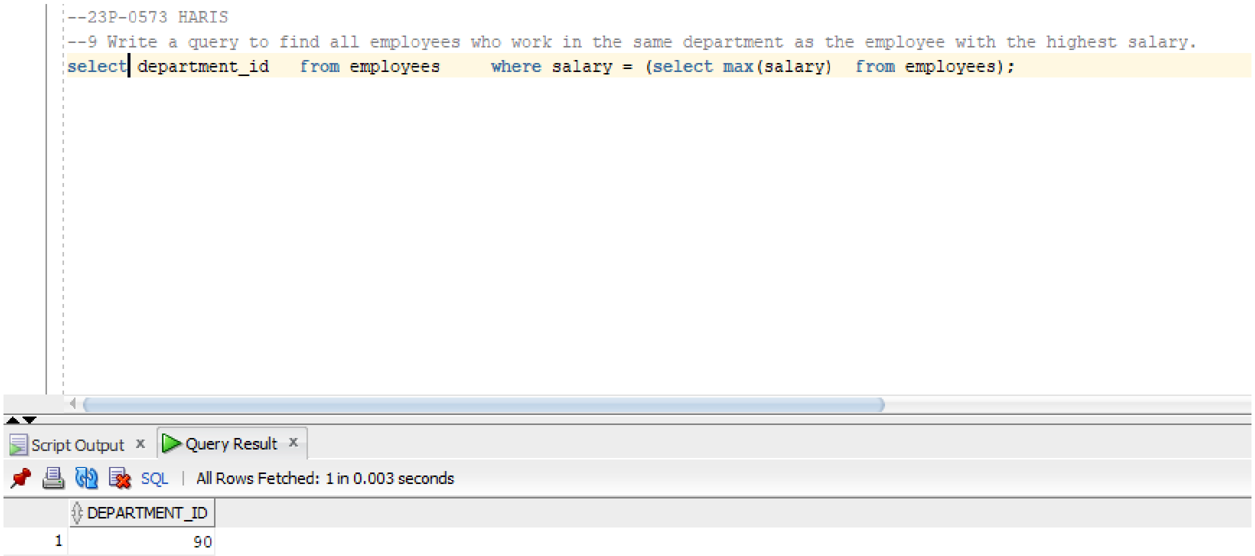


```
--23P-0573 HARIS
--8 Display the first name, last name, and job ID of employees whose job ID is the same as that of the employee with ID 200.
select first_name ,last_name  ,job_id   from employees where job_id=  (select job_id  from employees where employee_id='200');
```

All Rows Fetched: 1 in 0.001 seconds

	FIRST_NAME	LAST_NAME	JOB_ID
1	Jennifer	Whalen	AD_ASST

--9 Write a query to find all employees who work in the same department as the employee with the highest salary. select department_id from employees where salary = (select max(salary) from employees);

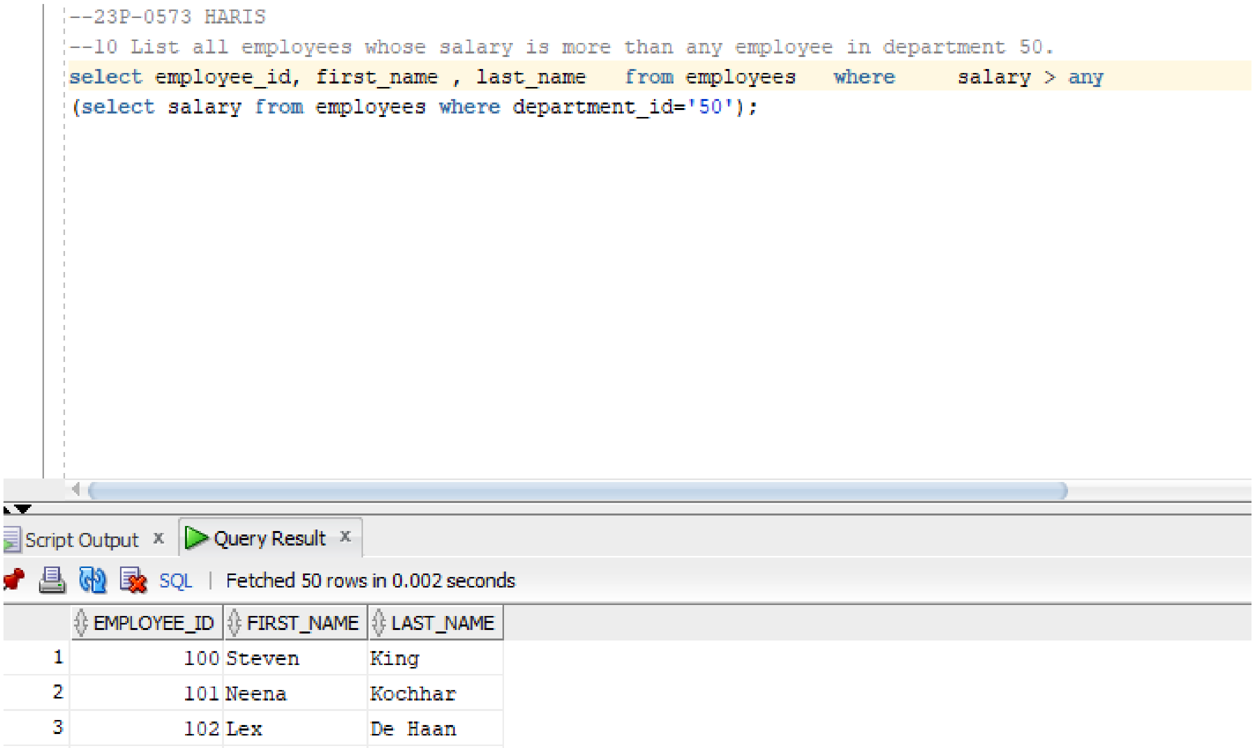


```
--23P-0573 HARIS
--9 Write a query to find all employees who work in the same department as the employee with the highest salary.
select department_id   from employees     where salary = (select max(salary)  from employees);
```

All Rows Fetched: 1 in 0.003 seconds

	DEPARTMENT_ID
1	90

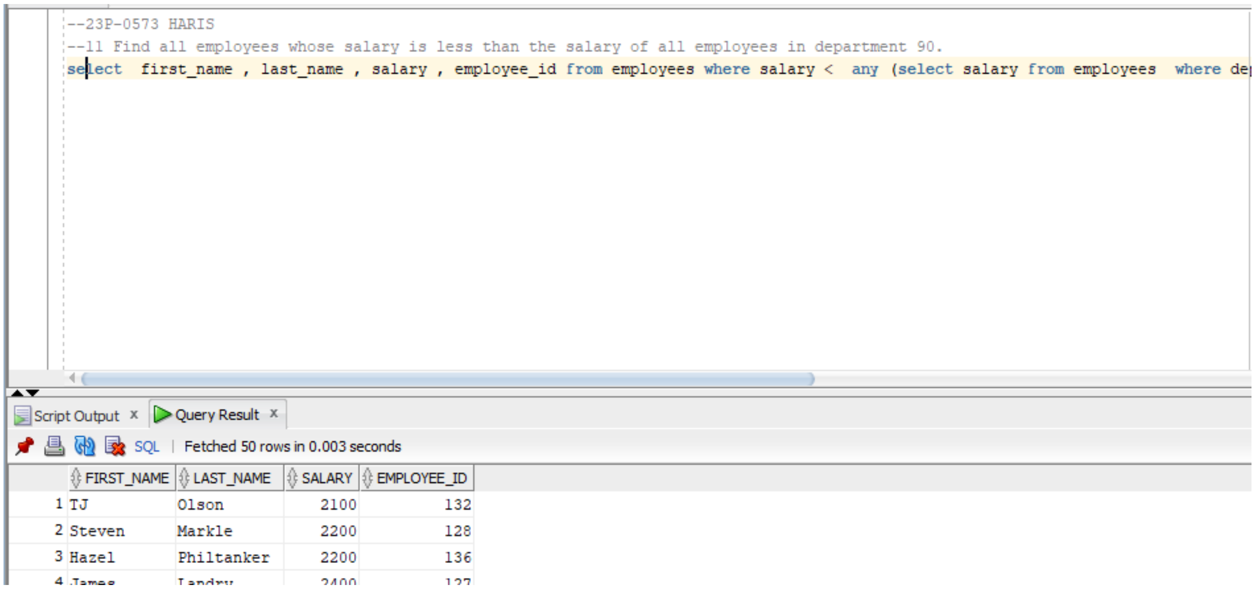
--10 List all employees whose salary is more than any employee in department 50. select employee_id, first_name , last_name from employees where salary > any (select salary from employees where department_id='50');



```
--23P-0573 HARIS
--10 List all employees whose salary is more than any employee in department 50.
select employee_id, first_name , last_name   from employees   where     salary > any
(select salary from employees where department_id='50');
```

```
Fetched 50 rows in 0.002 seconds

   EMPLOYEE_ID FIRST_NAME LAST_NAME
1          100 Steven     King
2          101 Neena      Kochhar
3          102 Lex        De Haan
```

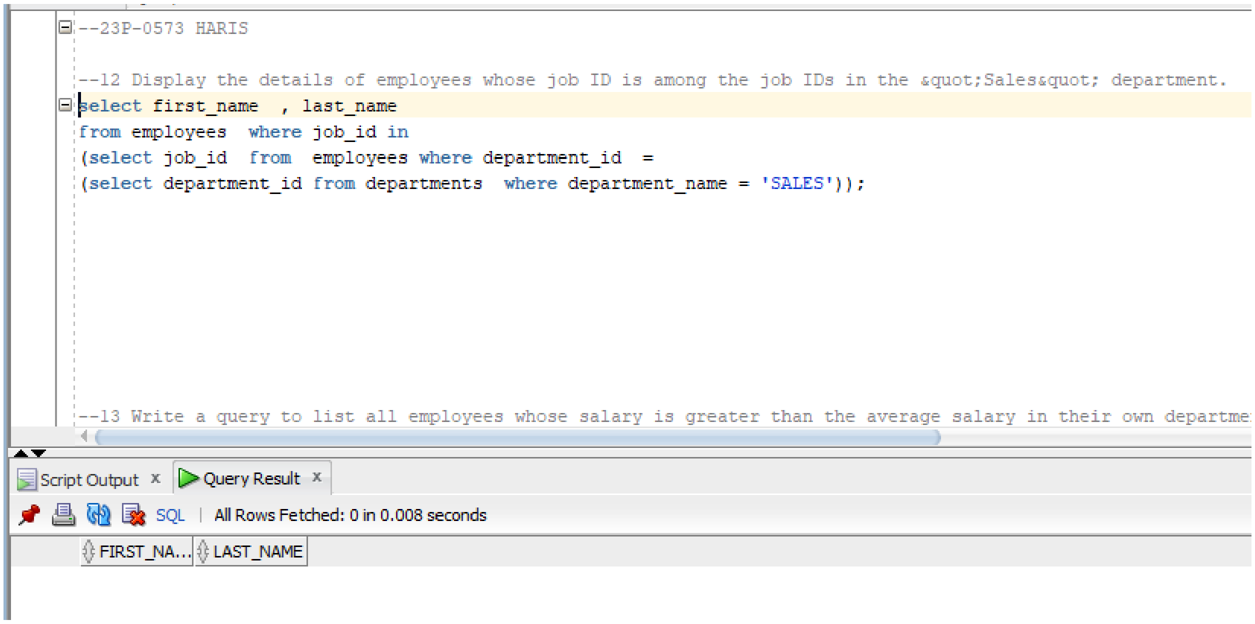
--11 Find all employees whose salary is less than the salary of all employees in department 90. select first_name , last_name , salary , employee_id from employees where salary < any (select salary from employees where department_id ='90');



```
--23P-0573 HARIS
--11 Find all employees whose salary is less than the salary of all employees in department 90.
select  first_name , last_name , salary , employee_id from employees where salary <  any (select salary from employees  where de
```

```
Fetched 50 rows in 0.003 seconds

  FIRST_NAME LAST_NAME  SALARY EMPLOYEE_ID
1 TJ         Olson        2100         132
2 Steven     Markle       2200         128
3 Hazel      Philtanker   2200         136
4 James      Landry       2400         127
```

--12 Display the details of employees whose job ID is among the job IDs in the "Sales" department. select first_name , last_name from employees where job_id in (select job_id from employees where department_id = (select department_id from departments where department_name = 'SALES'));

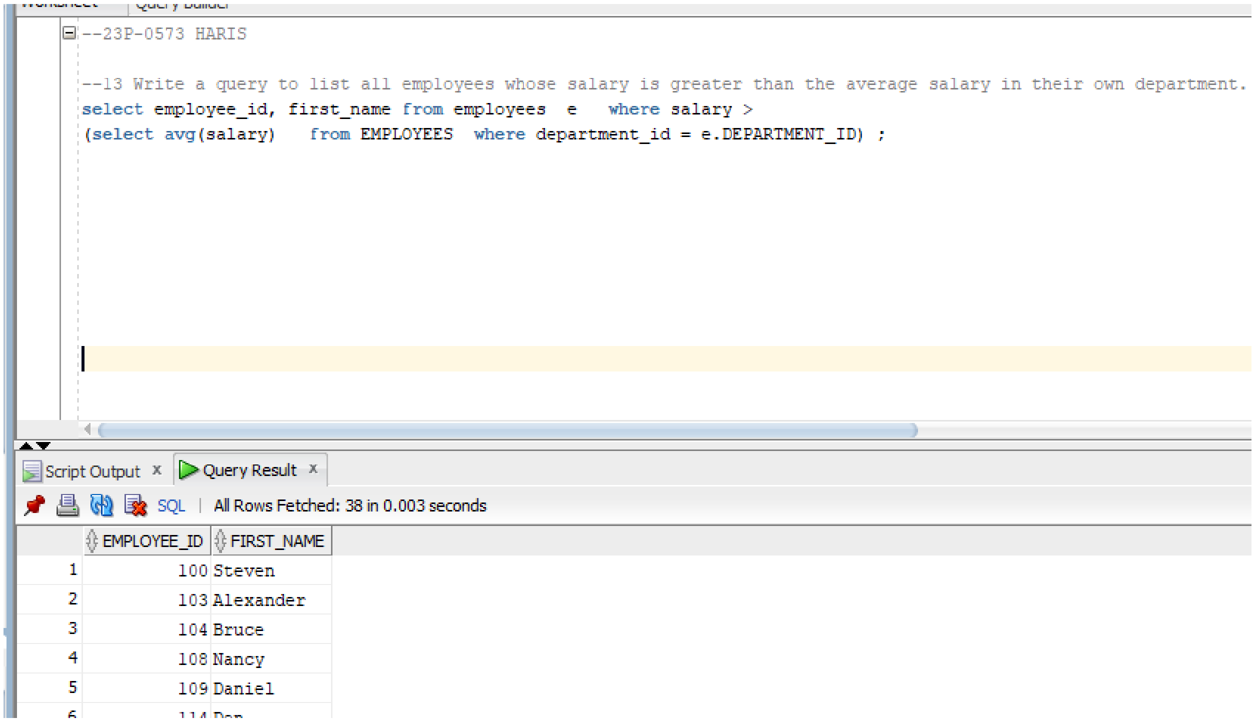


```
--23P-0573 HARIS

--12 Display the details of employees whose job ID is among the job IDs in the &quot;Sales&quot; department.
select first_name  , last_name
from employees  where job_id in
(select job_id  from  employees where department_id  =
(select department_id from departments  where department_name = 'SALES'));







--13 Write a query to list all employees whose salary is greater than the average salary in their own departme
```

All Rows Fetched: 0 in 0.008 seconds

```
FIRST_NA... LAST_NAME
```

--13 Write a query to list all employees whose salary is greater than the average salary in their own department. select employee_id, first_name from employees e where salary > (select avg(salary) from EMPLOYEES where department_id = e.DEPARTMENT_ID) ;

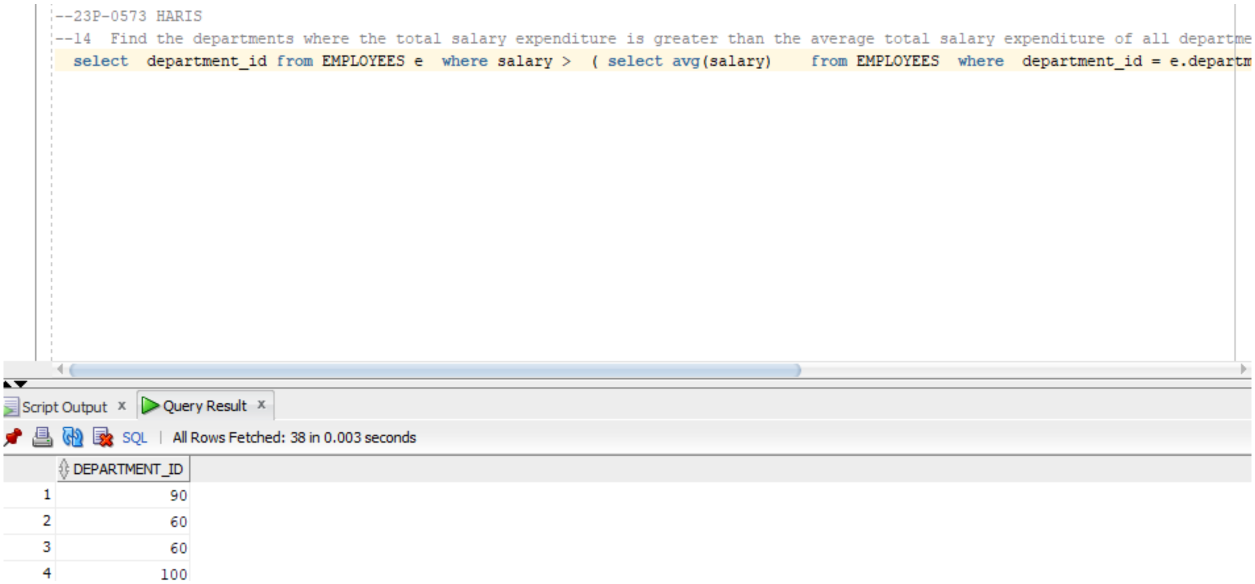


```
--23P-0573 HARIS

--13 Write a query to list all employees whose salary is greater than the average salary in their own department.
select employee_id, first_name from employees  e   where salary >
(select avg(salary)   from EMPLOYEES  where department_id = e.DEPARTMENT_ID) ;
```

All Rows Fetched: 38 in 0.003 seconds

	EMPLOYEE_ID	FIRST_NAME
1	100	Steven
2	103	Alexander
3	104	Bruce
4	108	Nancy
5	109	Daniel
6	114	Den

--14 Find the departments where the total salary expenditure is greater than the average total salary expenditure of all departments. select department_id from EMPLOYEES e where salary > (select avg(salary) from EMPLOYEES where department_id = e.department_id);

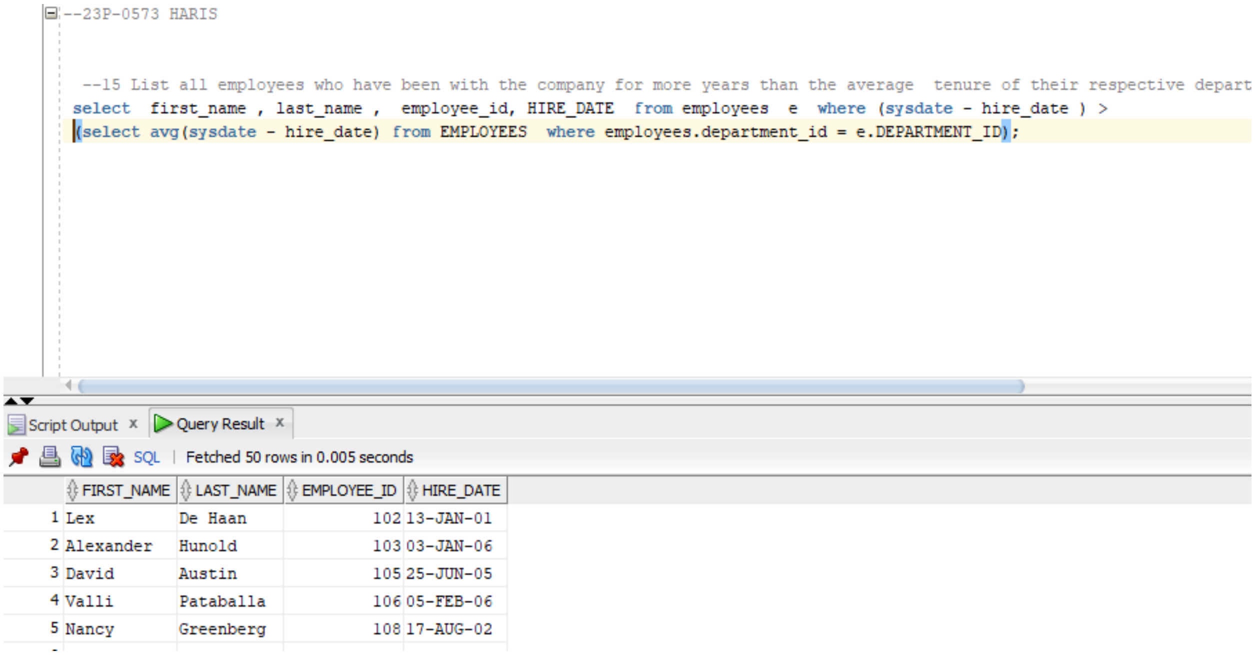


```
--23P-0573 HARIS
--14  Find the departments where the total salary expenditure is greater than the average total salary expenditure of all departme
 select  department_id from EMPLOYEES e  where salary >  ( select avg(salary)    from EMPLOYEES  where  department_id = e.departm
```

All Rows Fetched: 38 in 0.003 seconds

	DEPARTMENT_ID
1	90
2	60
3	60
4	100

--15 List all employees who have been with the company for more years than the average tenure of their respective department. select first_name , last_name , employee_id, HIRE_DATE from employees e where (sysdate - hire_date) > (select avg(sysdate - hire_date) from EMPLOYEES where employees.department_id = e.DEPARTMENT_ID);

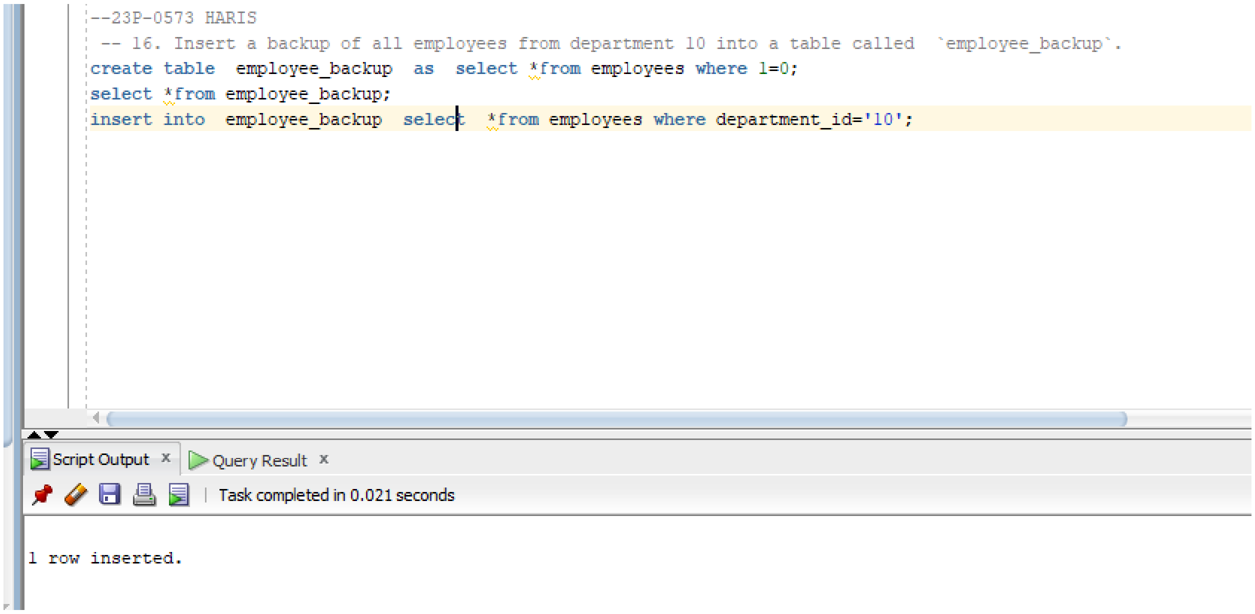


```
--23P-0573 HARIS

 --15 List all employees who have been with the company for more years than the average  tenure of their respective depart
select  first_name , last_name ,  employee_id, HIRE_DATE  from employees  e  where (sysdate - hire_date ) >
(select avg(sysdate - hire_date) from EMPLOYEES  where employees.department_id = e.DEPARTMENT_ID);
```

Fetched 50 rows in 0.005 seconds

	FIRST_NAME	LAST_NAME	EMPLOYEE_ID	HIRE_DATE
1	Lex	De Haan	102	13-JAN-01
2	Alexander	Hunold	103	03-JAN-06
3	David	Austin	105	25-JUN-05
4	Valli	Pataballa	106	05-FEB-06
5	Nancy	Greenberg	108	17-AUG-02

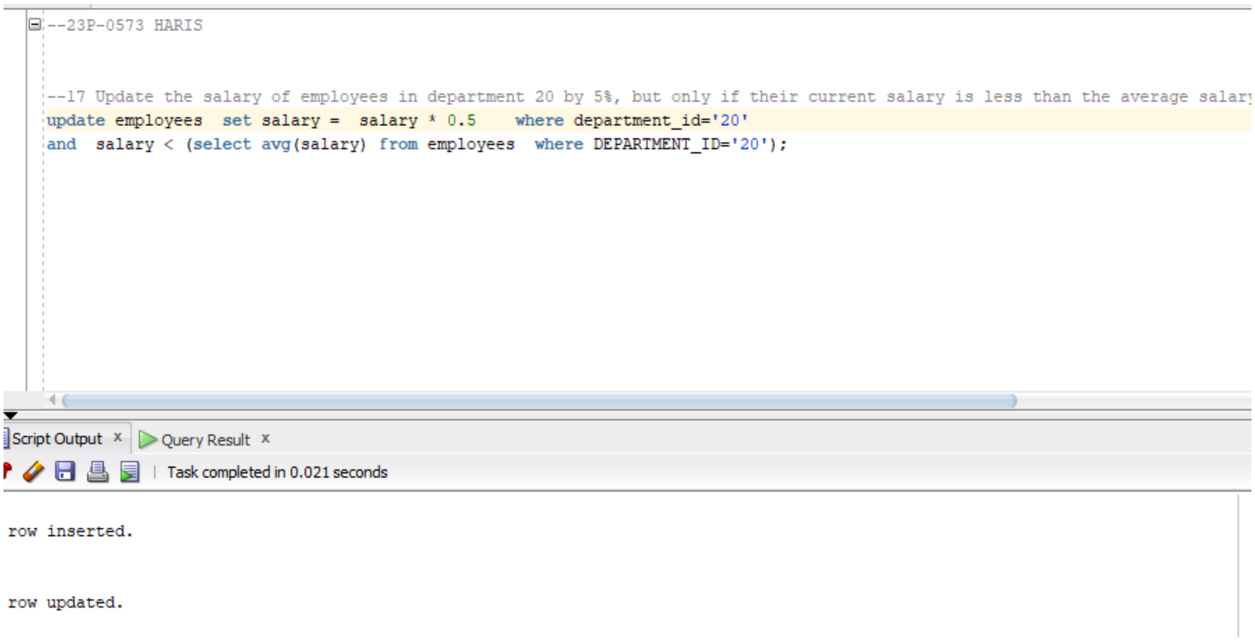
-- 16. Insert a backup of all employees from department 10 into a table called `employee_backup`. create table employee_backup as select *from employees where 1=0; select *from employee_backup; insert into employee_backup select *from employees where department_id='10';



```
--23P-0573 HARIS
 -- 16. Insert a backup of all employees from department 10 into a table called  `employee_backup`.
create table  employee_backup  as  select *from employees where 1=0;
select *from employee_backup;
insert into  employee_backup  select  *from employees where department_id='10';
```

```
1 row inserted.
```

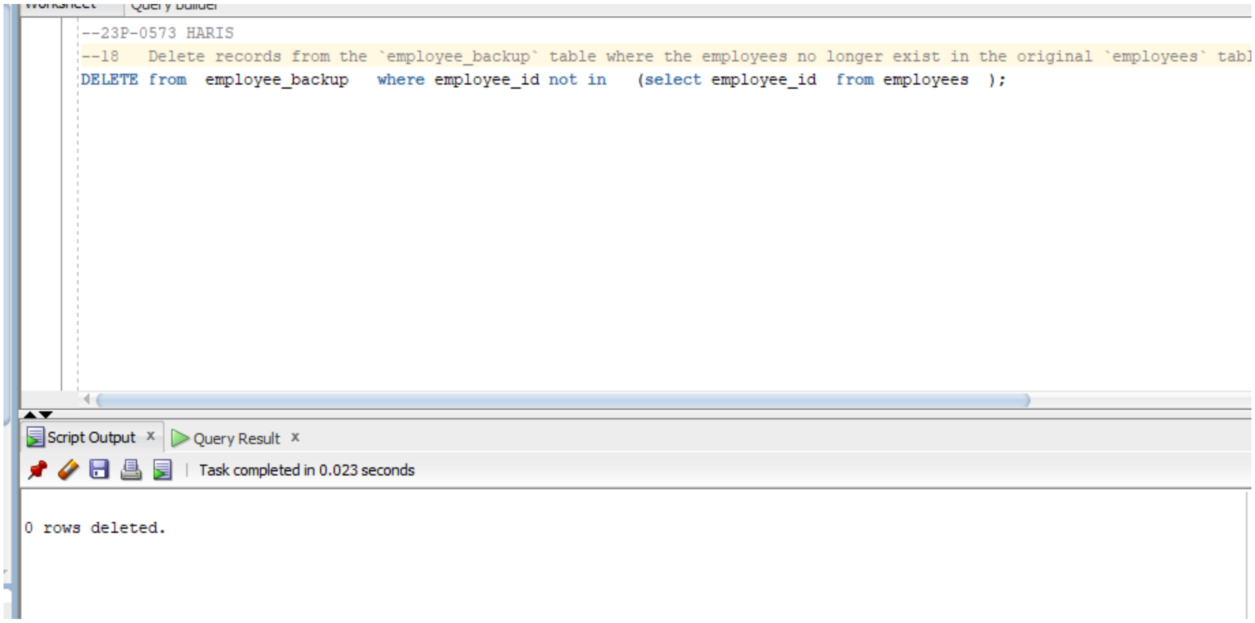
--17 Update the salary of employees in department 20 by 5%, but only if their current salary is less than the average salary of department 20. update employees set salary = salary * 0.5 where department_id='20' and salary < (select avg(salary) from employees where DEPARTMENT_ID='20');



```
--23P-0573 HARIS

--17 Update the salary of employees in department 20 by 5%, but only if their current salary is less than the average salary
update employees  set salary =  salary * 0.5    where department_id='20'
and  salary < (select avg(salary) from employees  where DEPARTMENT_ID='20');
```

```
row inserted.


row updated.
```

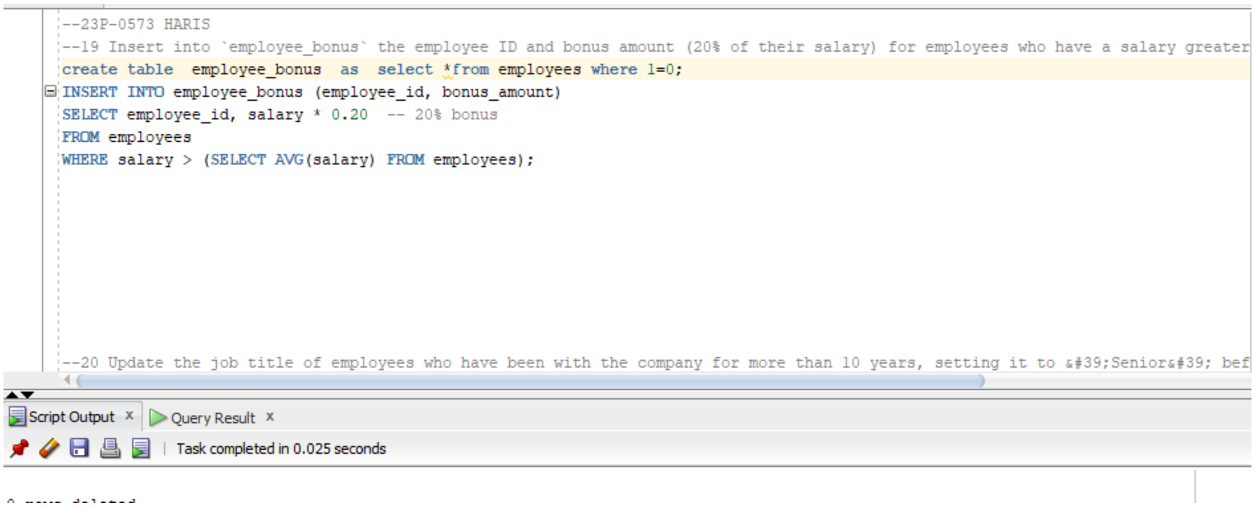
--18 Delete records from the `employee_backup` table where the employees no longer exist in the original `employees` table. DELETE from employee_backup where employee_id not in (select employee_id from employees);



```
--23P-0573 HARIS
--18  Delete records from the `employee_backup` table where the employees no longer exist in the original `employees` tabl
DELETE from  employee_backup   where employee_id not in   (select employee_id  from employees  );
```

```
0 rows deleted.
```

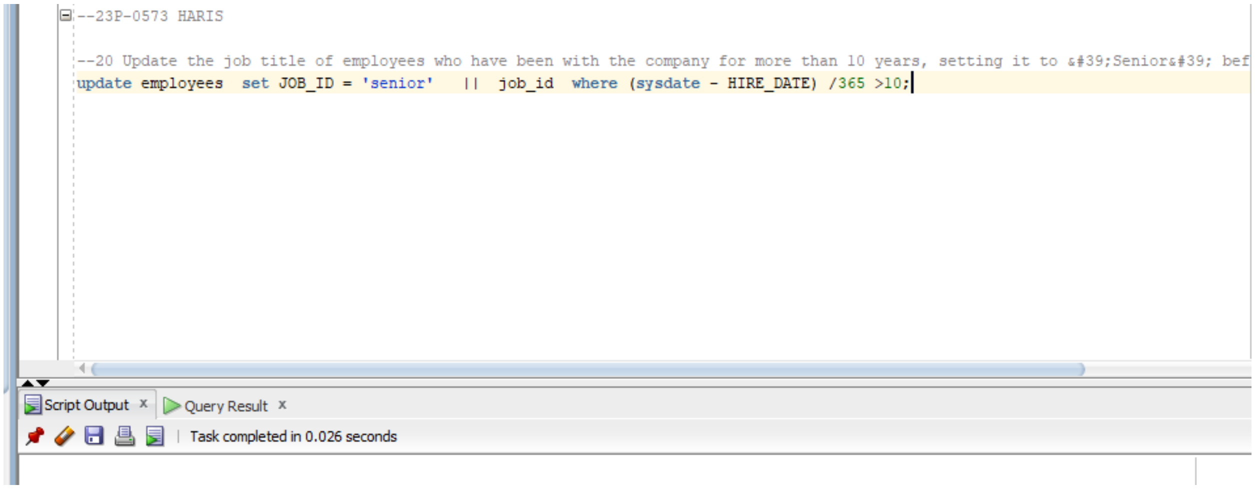
--19 Insert into `employee_bonus` the employee ID and bonus amount (20% of their salary) for employees who have a salary greater than the average salary of the company. create table employee_bonus as select *from employees where 1=0; INSERT INTO employee_bonus (employee_id, bonus_amount) SELECT employee_id, salary * 0.20 -- 20% bonus FROM employees WHERE salary > (SELECT AVG(salary) FROM employees);



```
--23P-0573 HARIS
--19 Insert into `employee_bonus` the employee ID and bonus amount (20% of their salary) for employees who have a salary greater
create table  employee_bonus  as  select *from employees where 1=0;
INSERT INTO employee_bonus (employee_id, bonus_amount)
SELECT employee_id, salary * 0.20  -- 20% bonus
FROM employees
WHERE salary > (SELECT AVG(salary) FROM employees);

--20 Update the job title of employees who have been with the company for more than 10 years, setting it to &#39;Senior&#39; bef
```

```
0 rows deleted
```

--20 Update the job title of employees who have been with the company for more than 10 years, setting it to 'Senior' before their current job title. update employees set JOB_ID = 'senior' || job_id where (sysdate - HIRE_DATE) /365 >10;



```
--23P-0573 HARIS

--20 Update the job title of employees who have been with the company for more than 10 years, setting it to &#39;Senior&#39; bef
update employees  set JOB_ID = 'senior'   ||  job_id  where (sysdate - HIRE_DATE) /365 >10;
```

```
Task completed in 0.026 seconds
```

This queury showing error i try to resolve but still showing issue .