

## Chapter 1 Task

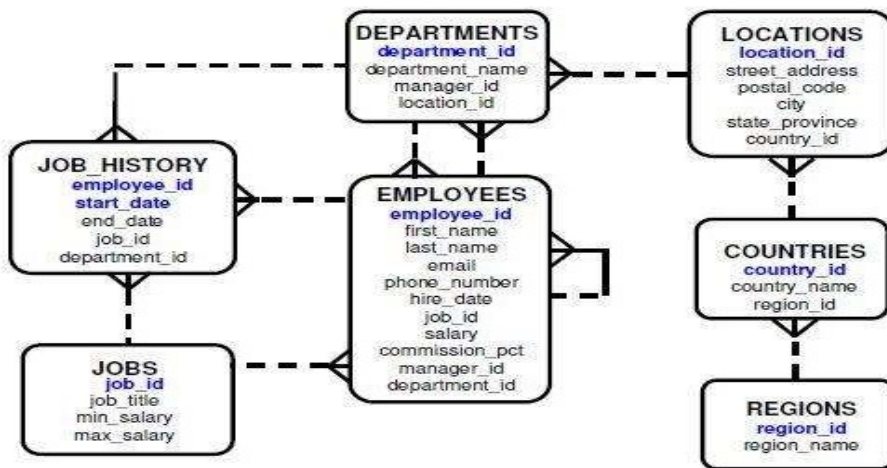
### Instructions:

-Study well and please attach Screen shots from the executable code in addition to sample of queries' results in your Answers.

- If you want to know any column name in a specific table you have to apply describe command: desc table\_name.

-HR schema:

### The Human Resources (HR) Schema



**Display the name concatenated with the job separated by a comma and space and names the column [Employee and Title] for all employees.**

## screenshots

```
1 select first_name||' '||last_name||' , '|| job_id "Employee and Title"
2 from hr.employees;
```

Employee and Title
William Gietz , AC_ACCOUNT

**Display the monthly and annual salary for all employees and  
rename the columns according to the time-period.**

## screenshots

### SQL Worksheet

```
1 select salary "Monthly Salary" , salary *12 "Annual Salary"  
2 from hr.employees;
```

Monthly Salary	Annual Salary
24000	288000
17000	204000
17000	204000

## Display all information about employees.

## screenshots

```

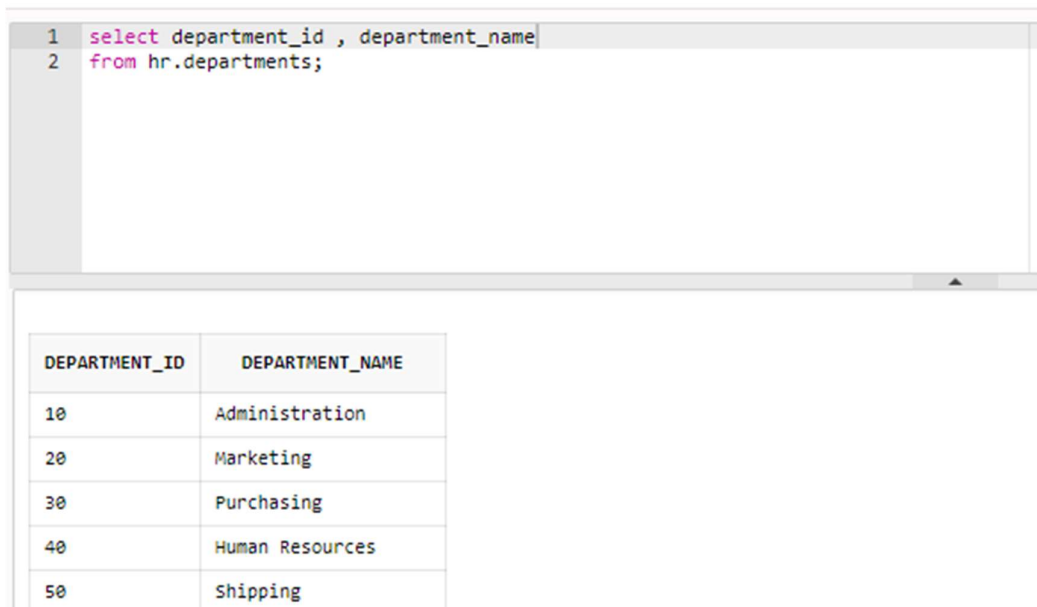
1 select *
2 from hr.employees;

```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES	24000	-	-	90
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	-	100	90
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	-	100	90
103	Alexander	Herold	AHEROLD	500.123.4567	23-JAN-05	IT_PROG	9000	-	103	90

**Display department number and department name for all departments.**

**screenshot**

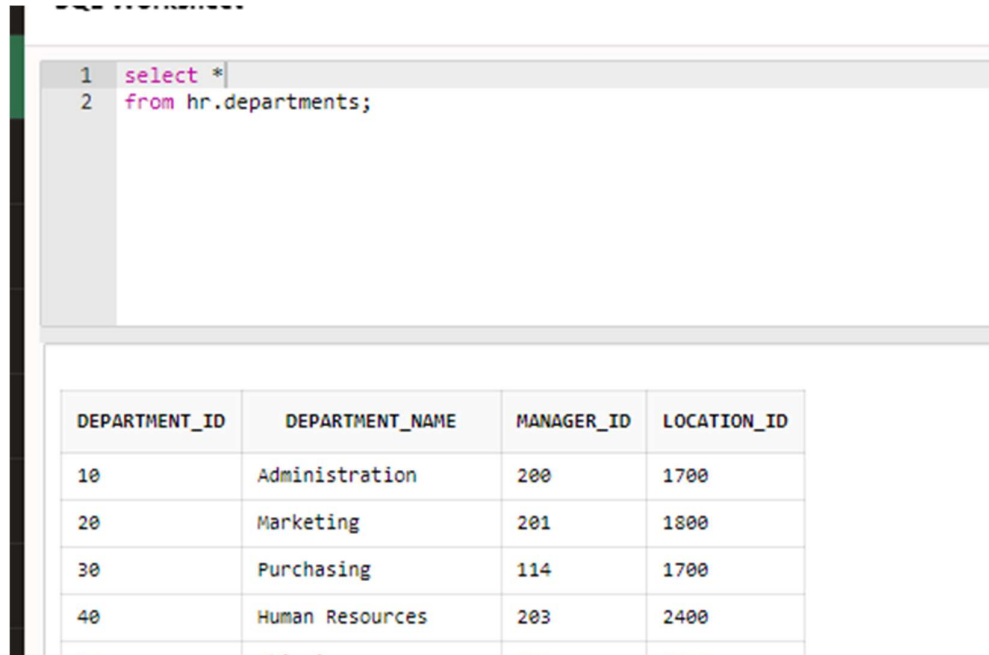


The screenshot shows a SQL query editor window with two lines of code. Below the editor is a table displaying the results of the query. The table has two columns: DEPARTMENT\_ID and DEPARTMENT\_NAME. The results are as follows:

DEPARTMENT_ID	DEPARTMENT_NAME
10	Administration
20	Marketing
30	Purchasing
40	Human Resources
50	Shipping

Display all information about departments.

## Screenshot



The screenshot shows a SQL query editor with two lines of code: `1 select *` and `2 from hr.departments;`. Below the editor, the results are displayed in a table with four columns: DEPARTMENT\_ID, DEPARTMENT\_NAME, MANAGER\_ID, and LOCATION\_ID. The table contains four rows of data representing different departments: Administration (ID 10, Manager 200, Location 1700), Marketing (ID 20, Manager 201, Location 1800), Purchasing (ID 30, Manager 114, Location 1700), and Human Resources (ID 40, Manager 203, Location 2400). The last row is partially obscured by a vertical bar on the left.

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
--	---	---	----

**Display unique salaries which the company offers.**

## Screenshots

### SQL Worksheet

```
1 select distinct salary  
2 from hr.employees;
```

SALARY
4200
7800
8000
6500
2200
3100

Display job title, minimum and maximum salaries for all jobs.

## Screenshots

```
1 select job_title, min_salary , max_salary  
2 from hr.jobs;
```

JOB_TITLE	MIN_SALARY	MAX_SALARY
President	20080	40000
Administration Vice President	15000	30000
Administration Assistant	3000	6000
Finance Manager	8200	16000
Accountant	4200	9000
Accounting Manager	8200	16000



For all regions, display **region id** concatenated with **[and its name is]** followed by **region name** and show the output as **Regions' details**.

## Screenshots

SQL Worksheet

```
1 select region_id || ' and its name is ' || region_name as "Regions' details"
2 from hr.regions;
```

Regions' details
1 and its name is Europe
2 and its name is Americas
3 and its name is Asia
4 and its name is Middle East and Africa

[Download CSV](#)  
4 rows selected.

Display unique values for both salaries and commissions, what are the numbers of rows in result?—compare that number with the same query but without unique values (check the salary= 4000 and commission=null).

## Code

```
1 select salary, COMMISSION_PCT
2 from hr.employees;
3
4 select count(salary), count(COMMISSION_PCT)
5 from hr.employees;
6
7
8 select distinct salary
9 from hr.employees;
10
11 select count(distinct salary)
12 from hr.employees;
13
14 select distinct COMMISSION_PCT
15 from hr.employees;
16 select count(unique COMMISSION_PCT)
17 from hr.employees;
18
19 select distinct salary , COMMISSION_PCT
20 from hr.employees;
21
```

```

1 select salary, COMMISSION_PCT
2 from hr.employees;
3

```

SALARY	COMMISSION_PCT
24000	-
17000	-
17000	-
9000	-
6000	-
4800	-
4800	-
4200	-
12008	-
9000	-
8200	-
7700	-
7800	-
6900	-
11000	-
3100	-
2900	-
2800	-
2600	-
2500	-
8000	-
8200	-
7900	-
6500	-
5800	-
7200	-

there is  
some repetition  
As you can see

```
3  
4 select count(salary), count(COMMISSION_PCT)  
5 from hr.employees;  
6  
7
```

COUNT(SALARY)	COUNT(COMMISSION_PCT)
107	35

[Download CSV](#)

Without the unique filter

```
7
8 select distinct salary
9 from hr.employees;
10
```

SALARY	
4200	
7800	
8000	
6500	
2200	
3100	
2900	No repetitions can be found
2600	
9600	
9000	
2400	
3600	
7200	
3400	
4400	
4800	
3200	
10500	
7500	
7000	
6800	
8800	
4100	
8300	
6000	
8200	

```
10  
11 select count(distinct salary)  
12 from hr.employees;  
13
```

COUNT(DISTINCTSALARY)
58

[Download CSV](#)

```

13
14 select distinct COMMISSION_PCT
15 from hr.employees;
16 select count(unique COMMISSION_PCT)
17 from hr.employees;
18

```

COMMISSION_PCT
.3
.4
-
.2
.35
.25
.1
.15

[Download CSV](#)  
 8 rows selected.

COUNT(UNIQUECOMMISSION_PCT)
7

[Download CSV](#)

```

18
19 select distinct salary , COMMISSION_PCT
20 from hr.employees;
21

```

SALARY	COMMISSION_PCT
9000	-
2600	-
7000	.25
6800	.1
10000	.2
9600	.2
7300	.15
8600	.2
3000	-
10000	-
17000	-
3100	-
2900	-
2700	-
3600	-
7000	.15
7200	.1
7400	.15
2200	-
13500	.3
10500	.2
8800	.25
4000	-
7800	-
6900	-
11000	-

salary is unique while commission is not  
Meaning that unique/distinct  
just work on one column at a time ★

I think that's what was in the question

**Star**  
or it means, for example, that  
4000 / -  
won't be repeated anymore .. just make the tuple unique

=====