

DBMS LAB ASSIGNMENT - 3

EMPLOYEE Table Data:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
198	Connell	SH_CLERK	2600	2.5	124	50
199	Grant	SH_CLERK	2600	2.2	124	50
200	Whalen	AD_ASST	4400	1.3	101	10
201	Hartstein	IT_PROG	6000	nan	100	20
202	Fay	AC_MGR	6500	nan	210	20
203	Mavris	AD_VP	7500	nan	101	40
204	Baer	AD_PRES	3500	1.5	101	90
205	Higgins	AC_MGR	2300	nan	101	60
206	Gitz	IT_PROG	5000	nan	103	60
100	King	AD_ASST	8956	0.3	108	100
101	Kochar	SH_CLERK	3400	1.3	118	30

Q1. Display last_name, job_id, employee_id for each employee with employee_id appearing first.

Answer (SQL):

```
SELECT EMPLOYEE_ID, LAST_NAME, JOB_ID FROM EMPLOYEE;
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID
198	Connell	SH_CLERK
199	Grant	SH_CLERK
200	Whalen	AD_ASST
201	Hartstein	IT_PROG
202	Fay	AC_MGR
203	Mavris	AD_VP
204	Baer	AD_PRES
205	Higgins	AC_MGR
206	Gitz	IT_PROG
100	King	AD_ASST
101	Kochar	SH_CLERK

Q2. Display the details of all employees of department 60.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE DEPARTMENT_ID = 60;
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
205	Higgins	AC_MGR	2300	nan	101	60
206	Gitz	IT_PROG	5000	nan	103	60

Q3. Display the employee details of the employee whose last_name is King.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE LAST_NAME = 'King';
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
100	King	AD_ASST	8956	0.3	108	100

Q4. Display unique job_id from EMPLOYEE table. Give alias name to the column as JOB_TITLE.

Answer (SQL):

```
SELECT DISTINCT JOB_ID AS JOB_TITLE FROM EMPLOYEE;
```

Output:

JOB_TITLE
SH_CLERK
AD_ASST
IT_PROG
AC_MGR
AD_VP
AD_PRES

Q5. Display last_name, salary and salary increase of Rs300. Give the new column name as 'Increased Salary'.

Answer (SQL):

```
SELECT LAST_NAME, SALARY, SALARY+300 AS Increased_Salary FROM EMPLOYEE;
```

Output:

LAST_NAME	SALARY	Increased_Salary
Connell	2600	2900
Grant	2600	2900
Whalen	4400	4700
Hartstein	6000	6300
Fay	6500	6800
Mavris	7500	7800

Baer	3500	3800
Higgins	2300	2600
Gitz	5000	5300
King	8956	9256
Kochar	3400	3700

Q6. Display last_name, salary and annual compensation of all employees, plus a onetime bonus of Rs 100.

Answer (SQL):

```
SELECT LAST_NAME, SALARY, (SALARY*12+100) AS Annual_Compensation FROM EMPLOYEE;
```

Output:

LAST_NAME	SALARY	Annual_Compensation
Connell	2600	31300
Grant	2600	31300
Whalen	4400	52900
Hartstein	6000	72100
Fay	6500	78100
Mavris	7500	90100
Baer	3500	42100
Higgins	2300	27700
Gitz	5000	60100
King	8956	107572
Kochar	3400	40900

Q7. Display the details of those employees who get commission.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE COMM_PCT IS NOT NULL;
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
198	Connell	SH_CLERK	2600	2.5	124	50
199	Grant	SH_CLERK	2600	2.2	124	50
200	Whalen	AD_ASST	4400	1.3	101	10
204	Baer	AD_PRES	3500	1.5	101	90
100	King	AD_ASST	8956	0.3	108	100
101	Kochar	SH_CLERK	3400	1.3	118	30

Q8. Display the details of those employees who do not get commission.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE COMM_PCT IS NULL;
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
201	Hartstein	IT_PROG	6000	nan	100	20
202	Fay	AC_MGR	6500	nan	210	20
203	Mavris	AD_VP	7500	nan	101	40
205	Higgins	AC_MGR	2300	nan	101	60
206	Gitz	IT_PROG	5000	nan	103	60

Q9. Display the Employee_id, Department_id and Salary all employees whose salary is greater than 5000.**Answer (SQL):**

```
SELECT EMPLOYEE_ID, DEPARTMENT_ID, SALARY FROM EMPLOYEE WHERE SALARY > 5000;
```

Output:

EMPLOYEE_ID	DEPARTMENT_ID	SALARY
201	20	6000
202	20	6500
203	40	7500
100	100	8956

Q10. Display the Last_Name and Salary of all employees whose salary is between 4000 and 7000.**Answer (SQL):**

```
SELECT LAST_NAME, SALARY FROM EMPLOYEE WHERE SALARY BETWEEN 4000 AND 7000;
```

Output:

LAST_NAME	SALARY
Whalen	4400
Hartstein	6000
Fay	6500
Gitz	5000

Q11. Display the details of all employees whose salary is either 6000 or 6500 or 7000.**Answer (SQL):**

```
SELECT * FROM EMPLOYEE WHERE SALARY IN (6000,6500,7000);
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
201	Hartstein	IT_PROG	6000	nan	100	20

202	Fay	AC_MGR	6500	nan	210	20
-----	-----	--------	------	-----	-----	----

Q12. Display the details of all those employees who work either in department 10 or 20 or 30 or 50.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE DEPARTMENT_ID IN (10,20,30,50);
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
198	Connell	SH_CLERK	2600	2.5	124	50
199	Grant	SH_CLERK	2600	2.2	124	50
200	Whalen	AD_ASST	4400	1.3	101	10
201	Hartstein	IT_PROG	6000	nan	100	20
202	Fay	AC_MGR	6500	nan	210	20
101	Kochar	SH_CLERK	3400	1.3	118	30

Q13. Display the details of all employees whose salary is not equal to 5000.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE SALARY <> 5000;
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
198	Connell	SH_CLERK	2600	2.5	124	50
199	Grant	SH_CLERK	2600	2.2	124	50
200	Whalen	AD_ASST	4400	1.3	101	10
201	Hartstein	IT_PROG	6000	nan	100	20
202	Fay	AC_MGR	6500	nan	210	20
203	Mavris	AD_VP	7500	nan	101	40
204	Baer	AD PRES	3500	1.5	101	90
205	Higgins	AC_MGR	2300	nan	101	60
100	King	AD_ASST	8956	0.3	108	100
101	Kochar	SH_CLERK	3400	1.3	118	30

Q14. Display the details of all the CLERKS working in the organization.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE JOB_ID LIKE '%CLERK%';
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
-------------	-----------	--------	--------	----------	--------	---------------

198	Connell	SH_CLERK	2600	2.5	124	50
199	Grant	SH_CLERK	2600	2.2	124	50
101	Kochar	SH_CLERK	3400	1.3	118	30

Q15. Update the job_id's of the employees who earn more than 5000 to Grade_A. Display the table EMPLOYEE after updating.

Answer (SQL):

```
UPDATE EMPLOYEE SET JOB_ID = 'Grade_A' WHERE SALARY > 5000;
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
198	Connell	SH_CLERK	2600	2.5	124	50
199	Grant	SH_CLERK	2600	2.2	124	50
200	Whalen	AD_ASST	4400	1.3	101	10
201	Hartstein	Grade_A	6000	nan	100	20
202	Fay	Grade_A	6500	nan	210	20
203	Mavris	Grade_A	7500	nan	101	40
204	Baer	AD_PRES	3500	1.5	101	90
205	Higgins	AC_MGR	2300	nan	101	60
206	Gitz	IT_PROG	5000	nan	103	60
100	King	Grade_A	8956	0.3	108	100
101	Kochar	SH_CLERK	3400	1.3	118	30

Q16. Display the details of all those employees who are either CLERK or PROGRAMMER or ASSISTANT.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE JOB_ID LIKE '%CLERK%' OR JOB_ID LIKE '%PROG%' OR JOB_ID LIKE '%ASST%';
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
198	Connell	SH_CLERK	2600	2.5	124	50
199	Grant	SH_CLERK	2600	2.2	124	50
200	Whalen	AD_ASST	4400	1.3	101	10
201	Hartstein	IT_PROG	6000	nan	100	20
206	Gitz	IT_PROG	5000	nan	103	60
100	King	AD_ASST	8956	0.3	108	100
101	Kochar	SH_CLERK	3400	1.3	118	30

Q17. Display those employees whose designation is CLERK and salary is less than 3000.

Answer (SQL):

```
SELECT * FROM EMPLOYEE WHERE JOB_ID LIKE '%CLERK%' AND SALARY < 3000;
```

Output:

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY	COMM_PCT	MGR_ID	DEPARTMENT_ID
198	Connell	SH_CLERK	2600	2.5	124	50
199	Grant	SH_CLERK	2600	2.2	124	50

Q18. Display those employees Last_Name, Mgr_id from the EMPLOYEE table whose salary is above 3000 and work under Manager 101.

Answer (SQL):

```
SELECT LAST_NAME, MGR_ID FROM EMPLOYEE WHERE SALARY > 3000 AND MGR_ID=101;
```

Output:

LAST_NAME	MGR_ID
Whalen	101
Mavris	101
Baer	101