Nama : Raden Sadiah Maharani

NIM : 2107126368

Basis Data Lanjut

Praktikum 3

1. Create one query that will convert 25-Dec-2004 into each of the following (you will have to convert 25-Dec-2004 to a date and then to character data):
   a) December 25th, 2004
   b) DECEMBER 25TH, 2004
   c) 25th december, 2004

→ a)
```
SELECT TO_CHAR
  (TO_DATE('25-Dec-2004','DD-MM-YYYY'),
  'fmMonth ddth, YYYY')
  AS "Date"
  FROM Dual;
```

Result :

Date
December 25th, 2004

b)
```
SELECT TO_CHAR
  (TO_DATE('25-Dec-2004','DD-MM-YYYY'),
  'MONTH DDTH, YYYY')
  AS "Date"
  FROM Dual;
```

Result :

Date
DECEMBER 25TH, 2004

c)
```
SELECT TO_CHAR
  (TO_DATE('25-Dec-2004','DD-MM-YYYY'),
  'ddth month, YYYY')
  AS "Date"
  FROM Dual;
```

Result :

Date
25th december , 2004

2. Convert JUNE192004 to a date using the fx format model.

→
```
SELECT TO_DATE('JUNE192004', 'fxMonthDDYYYY')
AS "Date"
FROM DUAL;
```

Result :

Date
06/19/2004

3. Not all Global Fast Foods staff members receive overtime pay. Instead of displaying a null value for these employees, replace null with zero. Include the employee's last name and overtime rate in the output. Label the overtime rate as "Overtime Status".

→
```
SELECT last_name, NVL (overtime_rate, 0) AS "Overtime
Status"
FROM f_staffs;
```

Result :

LAST_NAME	Overtime Status
Doe	10.25
Miller	0
Tuttle	0

4. For all null values in the specialty column in the DJs on Demand d_partners table, substitute "No Specialty." Show the first name and specialty columns only.

→
```
SELECT first_name,
NVL2(specialty, specialty, 'No Specialty')
AS "Specialty"
FROM d_partners
ORDER BY first_name;
```

Atau :

```
SELECT first_name,
COALESCE(specialty, 'No Specialty')
AS "Specialty"
FROM d_partners
ORDER BY first_name;
```

Result :

FIRST_NAME	Specialty
Allison	No Specialty
Jason	Hip Hop
Jennifer	All Types

5. Use the Oracle Database employees table and CASE expression to decode the departmen id. Display the department id, last name, salary and a column called "New Salary" whose value is based on the following conditions:
    - If the department id is 10 then 1.25 * salary
    - If the department id is 90 then 1.5 * salary
    - If the department id is 130 then 1.75 * salary
    - Otherwise, display the old salary

→
```
SELECT department_id "ID", last_name, salary,
CASE department_id
WHEN 10 THEN 1.25*salary
WHEN 90 THEN 1.5*salary
WHEN 130 THEN 1.75*salary
ELSE salary
END AS "New Salary"
FROM employees
ORDER BY department_id;
```

Result :

ID	LAST_NAME	SALARY	New Salary
10	Whalen	4400	5500
20	Fay	6000	6000
20	Hartstein	13000	13000
50	Rajs	3500	3500
50	Vargas	2500	2500
50	Mourgos	5800	5800
50	Matos	2600	2600
50	Davies	3100	3100
60	Ernst	6000	6000
60	Hunold	9000	9000
60	Lorentz	4200	4200
80	Taylor	8600	8600
80	Abel	11000	11000
80	Zlotkey	10500	10500
90	De Haan	17000	25500
90	King	24000	36000
90	Kochhar	17000	25500
110	Gietz	8300	8300
110	Higgins	12000	12000
-	Grant	7000	7000

6. Display the first name, last name, manager ID, and commission percentage of all employees in departments 80 and 90. In a 5th column called “Review”, again display the manager ID. If they don’t have a manager, display the commission percentage. If they don’t have a commission, display 99999.

→
```
SELECT first_name, last_name, manager_id,
commission_pct,
COALESCE(manager_id, commission_pct, 99999)
AS "Review"
FROM employees
WHERE department_id IN (80,90)
ORDER BY first_name;
```

Result :

FIRST_NAME	LAST_NAME	MANAGER_ID	COMMISSION_PCT	Review
Eleni	Zlotkey	100	.2	100
Ellen	Abel	149	.3	149
Jonathon	Taylor	149	.2	149
Lex	De Haan	100	-	100
Neena	Kochhar	100	-	100
Steven	King	-	-	99999