Assignment 5_Shayan 27027

Question 1)

```
--Shayan 27027
/*  For every employee, display their ID, first name and last name together
in one column separated by a space, and another column that shows
- 'Low Earner' if their salary is less than equal to 5000.
   If Salary is greater than 5000, but lower than equal to 1000, then 'Middle Earner'.
   Otherwise 'High Earner' except for where we have null or 0 value in salary
   column.*/
select *
from OEHR_EMPLOYEES

Select EMPLOYEE_ID,FIRST_NAME + ' ' + LAST_NAME as Employee,
       case when salary <= 5000 then 'LOW-EARNER'
            when salary > 5000 and salary <= 10000 then 'MIDDLE-EARNER'
            when salary = 0 or salary is Null then 'Null'
            else 'HIGH-EARNER'
       end as Salary_category
from OEHR_EMPLOYEES


select *
from OEHR_EMPLOYEES
```

100 %

Results | Messages

	EMPLOYEE_ID	Employee	Salary_category
1	100	Steven King	HIGH-EARNER
2	101	Neena Kochhar	HIGH-EARNER
3	102	Lex De Haan	HIGH-EARNER
4	103	Alexander Hunold	MIDDLE-EARNER
5	104	Bruce Ernst	MIDDLE-EARNER
6	105	David Austin	LOW-EARNER
7	106	Valli Pataballa	LOW-EARNER
8	107	Diana Lorentz	LOW-EARNER
9	108	Nancy Greenberg	HIGH-EARNER
10	109	Daniel Faviet	MIDDLE-EARNER
11	110	John Chen	MIDDLE-EARNER

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 107 rows

Question 2)

```
--Shayan 27027
/* Find the names of employees who earn more than the average salary of the company.
Arrange the result by the highest salary, and if more than one person has the same
salary, you should give preference to the length of "First Name" column.
So, two people with same salary, the first record would be of the person who has
more letters in his/her first name.*/

SELECT FIRST_NAME, SALARY
FROM OEHR_EMPLOYEES
WHERE SALARY > (SELECT AVG(SALARY)
                FROM OEHR_EMPLOYEES)
ORDER BY SALARY DESC, LEN(FIRST_NAME) DESC;
```

100 %

Results | Messages

	FIRST_NAME	SALARY
1	Steven	24000
2	Neena	17000
3	Lex	17000
4	John	14000
5	Karen	13500
6	Michael	13000
7	Shelley	12000
8	Alberto	12000
9	Nancy	12000
10	Lisa	11500
11	Gerald	11000

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 51 row

Question 3)

```sql
--Shayan 27027
/*  Write a SQL query to find details of those employees where the salary falls
within the range of the smallest salary and 7000 (both inclusive).
Sort the results by highest salary to lowest salary.
Put the results in a table and show results.*/

select *
into #new_table
from OEHR_EMPLOYEES
where salary between (select min(Salary)
                      from OEHR_EMPLOYEES) and 7000
order by salary desc

select *
from #new_table

--Shayan 27027
```

100 %

Results | Messages

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	104	Bruce	Ernst	BERNST	590.423.4568	2011-11-10	IT_PROG	6000	NULL	103	60
2	105	David	Austin	DAUSTIN	590.423.4569	2017-12-15	IT_PROG	4800	NULL	103	60
3	106	Valli	Pataballa	VPATABAL	590.423.4560	2018-07-28	IT_PROG	4800	NULL	103	60
4	107	Diana	Lorentz	DLORENTZ	590.423.5567	2019-07-30	IT_PROG	4200	NULL	103	60
5	113	Luis	Popp	LPOPP	515.124.4567	2020-05-28	FI_ACCOUNT	6900	NULL	108	100
6	115	Alexander	Khoo	AKHOO	515.127.4562	2015-11-07	PU_CLERK	3100	NULL	114	30
7	116	Shelli	Baida	SBAIDA	515.127.4563	2018-06-15	PU_CLERK	2900	NULL	114	30
8	117	Sigal	Tobias	STOBIAS	515.127.4564	2018-01-13	PU_CLERK	2800	NULL	114	30
9	118	Guy	Himuro	GHIMURO	515.127.4565	2019-05-07	PU_CLERK	2600	NULL	114	30
10	119	Karen	Colmenares	KCOLMENA	515.127.4566	2020-01-30	PU_CLERK	2500	NULL	114	30
11	123	Shanta	Vollman	SVOLLMAN	650.123.4234	2018-04-01	ST_MAN	6500	NULL	100	50

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 63 rows

Question 4)

```sql
--Shayan 27027
/* Using subqueries, display all the records of employee table, display
another column that gives the average salary of the whole company in each cell,
and create another column which compares the employee's salary with the company
average salary.
If employee salary is greater than company average, this column should show
"Above Average", otherwise should show below average.*/

select *,
       (select Avg(salary)
        from OEHR_EMPLOYEES) as AVG,
        case when salary > (select Avg(salary)
                            from OEHR_EMPLOYEES) then 'Above-Average'
             else 'Below-Average'
         end as Salary_Category
from OEHR_EMPLOYEES


--Shayan 27027
```

100 %

Results | Messages

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	AVG	Salary_Category
1	100	Steven	King	SKING	515.123.4567	2007-12-07	AD_PRES	24000	NULL	NULL	90	6461	Above-Average
2	101	Neena	Kochhar	NKOCHHAR	515.123.4568	2010-03-13	AD_VP	17000	NULL	100	90	6461	Above-Average
3	102	Lex	De Haan	LDEHAAN	515.123.4569	2013-07-05	AD_VP	17000	NULL	100	90	6461	Above-Average
4	103	Alexander	Hunold	AHUNOLD	590.423.4567	2010-06-25	IT_PROG	9000	NULL	102	60	6461	Above-Average
5	104	Bruce	Ernst	BERNST	590.423.4568	2011-11-10	IT_PROG	6000	NULL	103	60	6461	Below-Average
6	105	David	Austin	DAUSTIN	590.423.4569	2017-12-15	IT_PROG	4800	NULL	103	60	6461	Below-Average
7	106	Valli	Pataballa	VPATABAL	590.423.4560	2018-07-28	IT_PROG	4800	NULL	103	60	6461	Below-Average
8	107	Diana	Lorentz	DLORENTZ	590.423.5567	2019-07-30	IT_PROG	4200	NULL	103	60	6461	Below-Average
9	108	Nancy	Greenberg	NGREENBE	515.124.4569	2015-02-06	FI_MGR	12000	NULL	101	100	6461	Above-Average
10	109	Daniel	Faviet	DFAVIET	515.124.4169	2015-02-05	FI_ACCOUNT	9000	NULL	108	100	6461	Above-Average
11	110	John	Chen	JCHEN	515.124.4269	2018-03-20	FI_ACCOUNT	8200	NULL	108	100	6461	Above-Average

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 107 rows

Question_5)

```sql
--Shayan 27027
/*Using Subqueries, from the department's table,
show only the departments that do not have any employee.*/

--by subqueires (all department ids not found in employee)
SELECT DEPARTMENT_ID, DEPARTMENT_NAME
FROM OEHR_DEPARTMENTS
WHERE DEPARTMENT_ID Not IN ( SELECT Distinct DEPARTMENT_ID
                             FROM OEHR_EMPLOYEES
                             where DEPARTMENT_ID is not null)
```

50 %

Results | Messages

	DEPARTMENT_ID	DEPARTMENT_NAME
1	120	Treasury
2	130	Corporate Tax
3	140	Control And Credit
4	150	Shareholder Services
5	160	Benefits
6	170	Manufacturing
7	180	Construction
8	190	Contracting
9	200	Operations
10	210	IT Support
11	220	NOC

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 16 rows

Question_6)

```sql
--Shayan 27027
/* Find employees who were hired after the hiring date of the latest employee in
IT department.*/

-- finding all employees
select *
from OEHR_EMPLOYEES
where HIRE_DATE > (select max(hire_Date)
                   from OEHR_EMPLOYEES as a
                   inner join OEHR_DEPARTMENTS as b
                   on a.DEPARTMENT_ID=b.DEPARTMENT_ID
                   where DEPARTMENT_NAME = 'IT')
```

150 %

Results | Messages

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	113	Luis	Popp	LPOPP	515.124.4567	2020-05-28	FI_ACCOUNT	6900	NULL	108	100
2	119	Karen	Colmenares	KCOLMENA	515.127.4566	2020-01-30	PU_CLERK	2500	NULL	114	30
3	124	Kevin	Mourgos	KMOURGOS	650.123.5234	2020-05-07	ST_MAN	5800	NULL	100	50
4	128	Steven	Markle	SMARKLE	650.124.1434	2020-08-28	ST_CLERK	2200	NULL	120	50
5	132	TJ	Olson	TJOLSON	650.124.8234	2019-09-30	ST_CLERK	2100	NULL	121	50
6	135	Ki	Gee	KGEE	650.127.1734	2020-06-02	ST_CLERK	2400	NULL	122	50
7	136	Hazel	Philtanker	HPHILTAN	650.127.1634	2020-07-28	ST_CLERK	2200	NULL	122	50
8	148	Gerald	Cambrault	GCAMBRAU	011.44.1344.619268	2020-04-05	SA_MAN	11000	0.300000011920929	100	80
9	149	Eleni	Zlotkey	EZLOTKEY	011.44.1344.429018	2020-07-20	SA_MAN	10500	0.200000002980232	100	80
10	155	Oliver	Tuvault	OTUVAULT	011.44.1344.486508	2020-05-14	SA_REP	7000	0.150000005960464	145	80
11	163	Danielle	Greene	DGREENE	011.44.1346.229268	2019-09-08	SA_REP	9500	0.150000005960464	147	80

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 26 rows

Question_7)

```sql
--Shayan 27027
/*Using subqueries, find the departments where the average salary of department is greater
than the company's overall average salary.*/

SELECT d.DEPARTMENT_ID, DEPARTMENT_NAME, AVG(SALARY) AS avg_salary
FROM OEHR_EMPLOYEES e
inner join OEHR_DEPARTMENTS d
ON e.DEPARTMENT_ID = d.DEPARTMENT_ID
GROUP BY d.DEPARTMENT_ID, d.DEPARTMENT_NAME
HAVING AVG(SALARY) > (
    SELECT AVG(SALARY)
    FROM OEHR_EMPLOYEES
);
```

150 %

Results | Messages

	DEPARTMENT_ID	DEPARTMENT_NAME	avg_salary
1	20	Marketing	9500
2	40	Human Resources	6500
3	70	Public Relations	10000
4	80	Sales	8955
5	90	Executive	19333
6	100	Finance	8600
7	110	Accounting	10150

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 7 rows

Question_8)

```sql
--Shayan 27027
/* List the employees who have a job title that is not present in the
job history table.*/

select * from OEHR_EMPLOYEES
select * from OEHR_JOB_HISTORY

SELECT e.EMPLOYEE_ID, e.FIRST_NAME, e.LAST_NAME, e.JOB_ID
FROM OEHR_EMPLOYEES e
WHERE e.JOB_ID NOT IN (
    SELECT Distinct JOB_ID
    FROM OEHR_JOB_HISTORY
);
```

150 %

Results | Messages

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	JOB_ID
1	100	Steven	King	AD_PRES
2	101	Neena	Kochhar	AD_VP
3	102	Lex	De Haan	AD_VP
4	108	Nancy	Greenberg	FI_MGR
5	109	Daniel	Faviet	FI_ACCOUNT
6	110	John	Chen	FI_ACCOUNT
7	111	Ismael	Sciarra	FI_ACCOUNT
8	112	Jose Manuel	Urman	FI_ACCOUNT
9	113	Luis	Popp	FI_ACCOUNT
10	114	Den	Raphaely	PU_MAN
11	115	Alexander	Khoo	PU_CLERK

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 43 rows

Question_9)

```sql
--Shayan 27027
/* Find the names of employees who have been in the company longer than the average tenure
of employees in their department.*/
select first_name, last_name, a.DEPARTMENT_ID, DATEDIFF(DAY, HIRE_DATE, GETDATE()) as dif,
       avgTenure
from OEHR_EMPLOYEES  as a
inner join (SELECT department_id, AVG(DATEDIFF(Day, HIRE_DATE, GETDATE())) AS AvgTenure
            FROM OEHR_EMPLOYEES
            GROUP BY DEPARTMENT_ID) as b
on a.DEPARTMENT_ID = b.department_id
where DATEDIFF(DAY, HIRE_DATE, GETDATE()) > avgTenure
```

150 %

Results | Messages

	first_name	last_name	DEPARTMENT_ID	dif	avgTenure
1	Michael	Hartstein	20	2997	2723
2	Den	Raphaely	30	3434	2537
3	Alexander	Khoo	30	3272	2537
4	Winston	Taylor	50	2290	2265
5	Nandita	Sarchand	50	3018	2265
6	Alexis	Bull	50	2628	2265
7	Kelly	Chung	50	2514	2265
8	Jennifer	Dilly	50	2454	2265
9	Sarah	Bell	50	3010	2265
10	Britney	Everett	50	2617	2265
11	Matthew	Weiss	50	2845	2265

Query executed successfully. DESKTOP-S0NQ4VV\SQLEXPRESS ... DESKTOP-S0NQ4VV\Tesla ... HR_DATA 00:00:00 50 rows

Question_10)

```sql
--Shayan 27027
/*Using subquery, show only the employee record with the 3rd highest salary.
If two people have same 3rd highest salary, show the one that has the lowest difference
between his/her employee ID and the company's average salary.
(Yes, you read it write. I want you to compare employee ID with company's average salary.
Just for fun)*/

-- Showing employee record with just 3rd high salary
Select *
from OEHR_EMPLOYEES
where salary in (select top 1*        -- just 3rd high salaries
                 from (select Distinct top 3 salary
                       from OEHR_EMPLOYEES
                       group by salary
                       order by salary desc) as a
                 order by salary)
order by ((Select avg(salary)         -- the one with lowest difference
           from OEHR_EMPLOYEES) - EMPLOYEE_ID)
```

0 %

Results | Messages

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
145	John	Russell	JRUSSEL	011.44.1344.429268	2017-03-23	SA_MAN	14000	0.400000005960464	100	80