

# ASSIGNMENT : 06

E-ID	FNAME	LNAME	HIRE-DATE	JOB-ID	SAL	DEPT-ID
PK	NOT NULL	NOT NULL	NOT NULL	NOT NULL	NOT NULL	should be greater than or equal to 10

1. Create a table EMP with following structure.

EMP-ID	FIRST-NAME	LAST-NAME	HIRE-DATE	JOB-ID	SAL	DEPT-ID
198	Donald	Connell	21-Jun-99	SH-CLERK	2600	50
199	Douglas	Grant	13-Jan-98	SH-CLERK	3000	50
200	Jennifer	Whalen	17-Sep-87	AD-ASST	4400	10
201	Michael	Kertzstein	14-Jan-99	IT-PROG	6000	20
202	Pat	Fay	25-Oct-89	AC-MGR	6500	20
203	Susan	Mavris	26-Nov-76	AD-VP	7500	40
204	Hermann	Baer	23-Aug-95	AD-PRES	9500	90
205	Shelley	Higgins	24-Feb-98	AC-MGR	2300	60
206	William	Gitz	12-Mar-01	IT-PROG	5000	60
100	Steven	King	15-JUN-02	AD-ASST	8456	100
101	Neena	Kochhar	10-Jul-03	SH-CLERK	3400	30

- CREATE TABLE EMP (

```

    EMP-ID      INT(5) PRIMARY KEY,
    FIRST-NAME   VARCHAR(30) NOT NULL,
    LAST-NAME    VARCHAR(30) NOT NULL,
    HIRE-DATE    DATE NOT NULL,
    JOB-ID       VARCHAR(20) NOT NULL,
    SAL          DECIMAL(10,2) CHECK (SAL >= 10),
    DEPT-ID     INT(5) NOT NULL
);

```

2. Display the Fname of all employees in ascending order. Give appropriate alias name to the column.

- ```
SELECT FIRST_NAME AS Employee_Name  
FROM EMP  
ORDER BY FIRST_NAME ASC;
```

| Employee - Name |
|-----------------|
| Donald          |
| Douglas         |
| Hermann         |
| Jennifer        |
| Michael         |
| Neena           |
| Pat             |
| Shelly          |
| Steven          |
| Susan           |
| William         |

3. Display the Fname of all employees in descending order. Give appropriate alias name to the column.

- ```
SELECT FIRST_NAME AS EMPLOYEE_NAME  
FROM EMP  
ORDER BY FIRST_NAME DESC;
```

EMPLOYEE_NAME	EMPLOYEE_NAME
William	Hermann
Susan	Douglas
Steven	
Shelly	Donald.
Pat	
Neena	
Michael	
Jennifer	

4. Display the hire date of all employees in ascending order.

- ```
SELECT HIRE-DATE  
FROM EMP  
ORDER BY HIRE-DATE ASC;
```

| HIRE-DATE  |
|------------|
| 1901-03-12 |
| 1902-06-15 |
| 1903-07-10 |
| 1976-11-26 |
| 1987-09-17 |
| 1989-10-25 |
| 1995-08-23 |
| 1998-01-13 |
| 1998-02-24 |
| 1999-01-14 |
| 1999-06-21 |

5. Display the employee details whose name starts with either J or M. Sort the result by employee lname.

- ```
SELECT * FROM EMP  
WHERE FIRST-NAME LIKE 'J%' OR FIRST-NAME LIKE 'M%'  
ORDER BY LAST-NAME;
```

EMP-ID	FIRST-NAME	LAST-NAME	HIRE-DATE	JOB-ID	SAL	DEPT-ID
201	Michael	Hartstein	1999-01-19	IT-PROG	6000	20
200	Jennifer	Whalen	1987-09-17	AD-ASST	4400	10

6. Find the highest, lowest, average and sum of all employees. Give alias names to all the columns as 'MAX', 'Min', 'Avg', 'Sum'.

- ```
SELECT MAX(SAL) AS Max, MIN(SAL) AS Min, AVG(SAL)  
AS Avg, SUM(SAL) AS Sum  
FROM EMP;
```

| Max  | Min  | Avg         | Sum   |
|------|------|-------------|-------|
| 9500 | 2300 | 5377.818182 | 59156 |

7. Find the highest, lowest, average and sum of salary of all employees of each individual job type.

- SELECT JOB-ID, MAX(SAL) AS Max, MIN(SAL) AS Min, AVG(SAL) AS Avg, SUM(SAL) AS Sum  
FROM EMP  
GROUP BY JOB-ID;

8. Display the number of people under each job.

- SELECT JOB-ID, COUNT(\*) AS Num-Employees.  
FROM EMP  
GROUP BY JOB-ID;

| JOB-ID   | Num-Employees |
|----------|---------------|
| AD-ASST  | 2             |
| SH-CLERK | 3             |
| IT-PROG  | 2             |
| AC-MGR   | 2             |
| AD-VP    | 1             |
| AD-PRES  | 1             |

9. Display the number of managers in the company without listing their emp-id or names.

- SELECT COUNT(\*) AS Total-Managers  
FROM EMP

WHERE JOB-ID LIKE '%.MGR%' OR JOB-ID LIKE '%.PRES.'  
OR JOB-ID LIKE '%.VP%';

|                |
|----------------|
| Total-Managers |
| 4              |

16. Find the difference between the highest and lowest salaries.
- `SELECT (MAX(SAL) - MIN(SAL)) AS Salary-Difference  
FROM EMP;`

Salary-Difference

7200

11. Display the maximum and average salary of the engineers.

- `SELECT MAX(SAL), AVG(SAL)  
FROM EMP  
WHERE JOB-ID = 'IT-PROG';`

| MAX(SAL) | AVG(SAL) |
|----------|----------|
| 6000     | 5500     |

12. Display the employee name that is first and the employee name that is last in the alphabetized list of all employees.

- `SELECT MIN(FIRST-NAME) AS First-Employee,  
MAX(FIRST-NAME) AS Last-Employee  
FROM EMP;`

| First-Employee | Last-Employee |
|----------------|---------------|
| Donald         | William       |

13. Display the date when the first employee was hired and last date of hiring the employees. Rename the column as 'First Hire - Date' and 'Last. Hire - Date'.

- `SELECT MIN(HIRE-DATE) AS "First Hire - Date",  
MAX(HIRE-DATE) AS "Last Hire - Date"  
FROM EMP;`

| First Hire - Date | Last Hire - Date |
|-------------------|------------------|
| 1901-03-12        | 1999-06-21       |

(14) Display the maximum and average salary of clerks.

- ```
SELECT MAX(SAL), AVG(SAL)
FROM EMP
WHERE JOB-ID LIKE '%CLERK';
```

MAX(SAL)	AVG(SAL)
3400	3000

(15) Display the dept no and the salary of the lowest paid employee for each department. Give an alias name to the minimum salary column.

- ```
SELECT DEPT-ID, MIN(SAL) AS Min-Salary
FROM EMP
GROUP BY DEPT-ID;
```

(16) Display the dept no. and the salary of the lowest paid employee for each department. Exclude any groups where the minimum salary is 3000 or less. Sort the output in descending order of salary.

- ```
SELECT DEPT-ID, MIN(SAL) AS Min-Salary
FROM EMP
GROUP BY DEPT-ID
HAVING MIN(SAL) > 3000
ORDER BY Min-Salary DESC;
```

DEPT-ID	Min-Salary
90	9500.00
100	8956.00
40	7500.00
20	6000.00
10	4400.00
30	3400.00

17. Display the following for all rows of the table  
<first-name> whose designation is <job-id>  
gets <Salary> but wants to earn <3\*Salary>

- SELECT FIRST-NAME || ' whose designation is'  
|| JOB-ID ||  
' gets ' || SAL || ' but wants to earn ' || (SAL\*3)  
AS sentence  
FROM EMP;

SENTENCE
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1

18. Display the details of all the employee in  
the following format.

198, Donald, Connell, 21-Jun-99, SH-CLERK,  
2600, 50

- SELECT EMP\_ID || ', ' || FIRST-NAME || ', ' ||  
LAST-NAME || ', ' ||  
TO\_CHAR(HIRE\_DATE, 'DD-Mon-YY') || ', ' || JOB\_ID  
||

- ' , '11  
SAL 11 ', '11 DEPT\_ID AS EMP DETAILS  
FROM EMP;
19. Display today's date. Rename the column as TODAY.
- SELECT SYSDATE AS TODAY FROM DUAL;
20. Display the employee id, the month and the year in which the employees were hired.
- SELECT EMP\_ID, TO\_CHAR(HIRE\_DATE, 'DAY'),  
TO\_CHAR(HIRE\_DATE, 'YYYY')  
FROM EMP;
21. Display the employee name and the hire date of all employees in the format 'dd-month-yy'.
- SELECT FIRST\_NAME, EMP\_ID, TO\_CHAR(HIRE\_DATE, 'DD-Mon-YYYY')  
FROM EMP;
22. Display the employee\_id, months in which the employees were hired.
- SELECT FIRST\_NAME, EMP\_ID, TO\_CHAR(HIRE\_DATE, 'Month-DD-YYYY')  
FROM EMP;
23. Display the employee name, Employee id and the hire date of all employee in the format 'dd-mon-yyyy'.

• SELECT FIRST\_NAME, EMP\_ID, TO\_CHAR(HIRE\_DATE,  
'DD-Mon-YYYY')  
FROM EMP;

24. Display the employee name, employee id and  
the hire date of all employee in the format  
'month-dd-yyyy'.

• SELECT FIRST\_NAME, EMP\_ID, TO\_CHAR(HIRE\_DATE,  
'Month-DD-YYYY')  
FROM EMP;

25. Display the system year in full spelling.  
(Ex: Nineteen Ninety Nine for 1999).

• SELECT TO\_CHAR(SYSDATE, 'YYYYSP') AS  
YEAR-IN-WORDS  
FROM DUAL;

26. find date, 15 days after today's date, 15 days  
before today's date.

• SELECT SYSDATE + 15 AS After-15days,  
SYSDATE - 15 Before-15days  
FROM DUAL;

27. Ensure the domain constraint and entity  
integrity constraint from above the table.

• This script enforces domain constraints (NOT  
NULL, CHECK SAL > 10) and entity integrity  
constraint (PRIMARY KEY on EMP\_ID) for the EMP  
table.

28. Drop the domain constraint from the table.

- ALTER TABLE EMP DROP CONSTRAINT SYS\_C#;

29. Drop the primary key from the table .

- ALTER TABLE EMP DROP PRIMARY KEY ;

30 . Drop the created table .

- DROP TABLE EMP ;

OUTPUT ;

0 row(s) affected .