# ORDER BY clause

ORDER BY clause is used to arrange the records either in ascending order or descending order.

Syntax:

```
SELECT column_name
FROM table_name
ORDER BY column_name ASC/DESC;
```

## Characteristics of ORDER BY clause

> ORDER BY clause executes after SELECT

clause.
> We can pass multiple columns inside ORDER BY clause.
> By default, the table will be arranged according to Primary key in ascending order.
> By default, ORDER BY clause takes ASC as a argument.

══════════════════════════════════════════════════════════════════════════════════

77. WAQTD THE DETAILS OF EMP ACCORDING TO MAX SAL HOLDERS TO MIN HOLDERS.

```
SELECT *
FROM EMP
```

```
ORDER BY SAL DESC;
```

78. WAQTD THE EMP FNAME, LNAME ACCORDING TO ALPHABETICAL ORDER OF FNAME.

```
SELECT FNAME,LNAME
FROM EMP
ORDER BY FNAME ASC;
```

79. WAQTD THE NUMBER OF EMPS WORKING AS SALESMAN,MANAGER, DISPATCHER, ACCOUNTANT IN EVERY DEPT IF THE DEPT CONTAINS AVG SALARY MORE THAN 32000. ARRANGE THE RECORDS ACCORDING TO NUMBER OF EMPS.

```
SELECT COUNT(*),DNO
FROM EMP
WHERE JOB IN
('SALESMAN','MANAGER','DISPATCHER','ACCOUNTANT
')
GROUP BY DNO
HAVING AVG(SAL)>32000
ORDER BY COUNT(*);
```

══════════════════════════════════════════════
═══════════════════════════════════

LIMIT and OFFSET

════════════════

LIMIT is used to return the specified number of rows from the given table.

OFFSET is used to ignore or skip the specified number of rows from the given table.

<Note: By default, the table will be arranged according to Primary Key).>

==============================================
===================================

80. WAQTD THE FIRST 5 RECORDS FROM EMP TABLE.

```
SELECT *
FROM EMP
LIMIT 5;
```

81. WAQTD THE TOP 10 RECORDS FROM THE EMP TABLE.

```
SELECT *
FROM EMP
LIMIT 10;
```

82. WAQTD THE TOP 3 RECORDS FROM THE EMP TABLE.

```
SELECT *
FROM EMP
LIMIT 3;
```

83. WAQTD THE FIRST RECORD FROM THE EMP TABLE.

```
SELECT *
FROM EMP
LIMIT 1;
```

84. WAQTD THE SECOND RECORD FROM THE EMP TABLE.

```
SELECT *
FROM EMP
LIMIT 1 OFFSET 1;
```

85. WAQTD THE 5TH RECORD FROM EMP TABLE.

```
SELECT *
```

```
FROM EMP
LIMIT 1 OFFSET 4;
```

86. WAQTD THE 7TH AND 8TH RECORD FROM EMP TABLE.

```
SELECT *
FROM EMP
LIMIT 2 OFFSET 6;
```

87. WAQTD THE 10TH RECORD FROM EMP TABLE.

```
SELECT *
FROM EMP
LIMIT 1 OFFSET 9;
```

88. WAQTD THE LAST 3 RECORDS FROM EMP TABLE.

```
SELECT *
FROM EMP
ORDER BY EID DESC
LIMIT 3;
```

89. WAQTD THE LAST RECORD FROM EMP TABLE.

```
SELECT *
FROM EMP
ORDER BY EID DESC
LIMIT 1;
```

90. WAQTD THE LAST 5TH RECORD FROM EMP TABLE.

```
SELECT *
FROM EMP
ORDER BY EID DESC
LIMIT 1 OFFSET 4;
```

91. WAQTD THE LAST 10 RECORDS FROM EMP TABLE.

```
SELECT *
FROM EMP
ORDER BY EID DESC
LIMIT 10;
```

92. WAQTD THE 2ND MAXIMUM SALARY FROM EMP

TABLE.

```
SELECT SAL
FROM EMP
ORDER BY SAL DESC
LIMIT 1 OFFSET 1;
```

93. WAQTD THE 2ND MINIMUM SALARY FROM EMP TABLE.

```
SELECT DISTINCT SAL
FROM EMP
ORDER BY SAL ASC
LIMIT 1 OFFSET 1;
```

94. WAQTD THE 5TH MAX SALARY.

```
SELECT DISTINCT SAL
FROM EMP
ORDER BY SAL DESC
LIMIT 1 OFFSET 4;
```

95. WAQTD THE 7TH AND 8TH MIN SALARY.

```
SELECT DISTINCT SAL
FROM EMP
ORDER BY SAL
LIMIT 2 OFFSET 6;
```

## Character Functions

1. UPPER(): This function is used to convert the string values to UPPER case.
Syntax: SELECT UPPER('pentagon');

2. LOWER(): This function is used to convert the string values to lower case.
Syntax: SELECT LOWER('PENTAGON');

3. LENGTH(): This function is used to obtain the length of the given string.
Syntax: SELECT LENGTH('PENTAGON');

4. REVERSE(): This function is used to reverse

the given string.
Syntax: SELECT REVERSE('pentagon');

5. CONCAT(): This function is used to combine the given strings.
Syntax: CONCAT('str1','str2',...,'str n');

96. WAQTD THE EMP FULLNAME FROM THE EMP TABLE.

```
FULLNAME
--------
Siddarth Patil
```

```
SELECT CONCAT(FNAME,' ',LNAME) FULLNAME
FROM EMP;
```

97. WAQTD THE FULLNAME. ARRANGE THE RECORDS IN ALPHABETICAL ORDER.

```
SELECT CONCAT(FNAME,' ',LNAME) FULLNAME
FROM EMP
ORDER BY FULLNAME;
```

6. SUBSTR(): This function is used to obtain the part of the givens string.

Syntax:

```
SUBSTR('original_sal',position[,length]);
```

98. WAQTD THE FIRST 3 CHARACTERS OF THE FNAME.

```
SELECT SUBSTR(FNAME,1,3) SUBNAME
FROM EMP;
```

OR

```
SELECT LEFT(FNAME,3)
FROM EMP;
```

99. WAQTD THE LAST 4 CHARACTERS OF THE FNAME.

```
SELECT SUBSTR(FNAME,-4) SUBNAME
```

```
FROM EMP;

OR

SELECT RIGHT(FNAME,4) SUBNAME
FROM EMP;
```

═════════════════════════════════════════════════════════════════════════════════

7. LEFT(): This function is used to obtain the left part of the string.

Syntax: LEFT('string',no_of_characters);

8. RIGHT(): This function is used to obtain the right part of the string.

```
Syntax: RIGHT('string',no_of_characters);
```

100. WAQTD THE DETAILS OF EMP WHOSE FNAME STARTS WITH S WITHOUT USING LIKE OPERATOR.

```
SELECT *
FROM EMP
WHERE SUBSTR(FNAME,1,1)='S';
```

OR

```
SELECT *
FROM EMP
```

```
WHERE LEFT(FNAME,1)='S';
```

101. WAQTD THE FNAME,LNAME OF THE EMP WHOSE FNAME STARTS WITH VOWELS.

```
SELECT FNAME,LNAME
FROM EMP
WHERE LEFT(FNAME,1)IN('A','E','I','O','U');
```

102. WAQTD THE FNAME,LNAME WHOSE LNAME ENDS WITH VOWELS.

```
SELECT FNAME,LNAME
FROM EMP
WHERE RIGHT(LNAME,1)IN('A','E','I','O','U');
```

103. WAQTD THE FIRST HALF OF THE FNAME.

```
SELECT SUBSTR(FNAME,1,LENGTH(FNAME)/2)
FIRST_HALF
FROM EMP;
```

OR

```
SELECT LEFT(FNAME,LENGTH(FNAME)/2) FIRST_HALF
FROM EMP;
```

104. WAQTD THE SECOND HALF OF THE FNAME.

```
SELECT SUBSTR(FNAME,LENGTH(FNAME)/2+1)
```

```
SECOND_HALF
FROM EMP;
```

105. WAQTD THE FIRST HALF OF THE CHARACTERS IN UPPER CASE AND SECOND HALF IN LOWER REVERSE CASE

```
SIDDhtra
```

```
SELECT
CONCAT(UPPER(SUBSTR(FNAME,1,LENGTH(FNAME)/2)),
LOWER(REVERSE(SUBSTR(FNAME,LENGTH(FNAME)/2+1))
)) FIRSTeman
FROM EMP;
```

9. REPLACE(): This function is used to substitute substring with a new string.

Syntax: REPLACE('string','substr','new_str');

106. Requirement.

MALAYALAM   A → 4

```
SELECT
LENGTH('HARSH')-LENGTH(REPLACE('HARSH','H',''))
);
```

107. WAQTD THE FNAME,LNAME OF THE EMP WHOSE FNAME CONTAINS EXACTLY 1 H.

```
SELECT FNAME,LNAME
FROM EMP
WHERE
LENGTH(FNAME)-LENGTH(REPLACE(lower(FNAME),'h',
''))=1;
```

108. WAQTD THE FNAME,LNAME OF THE EMP IF THE EMP FNAME CONTAINS EXACTLY 2 A.

```
SELECT FNAME,LNAME
FROM EMP
```

```
WHERE
LENGTH(FNAME)-LENGTH(REPLACE(lower(FNAME),'a',
''))=2;
```

═════════════════════════════════════════════════════════════════════════════════

NUMBER FUNCTIONS

════════════════

ABS(): This function is used to convert the negative numbers into positive numbers.

Syntax: ABS(m);

Ex: SELECT ABS(-20);

MOD(); This function is used to obtain the remainder of the given two numbers.

Syntax: MOD(m,n);
Ex: SELECT MOD(4,2);

ROUND(): This function is used to round off the given floating values.
Syntax: ROUND(m,n);
Ex: SELECT ROUND(4.79);

CEIL(): This function is used to obtain the next integer value from the given decimal number.

FLOOR(): This function is used to obtain the current integer value from the given decimal number.

POW(): This function is used to obtain the power of the given number. It takes the integer value and the degree value as a argument.
Ex: SELECT POW(3,4);

109. WAQTD THE DETAILS OF EMP WHO HAVE EVEN EID.

```
SELECT *
FROM EMP
WHERE MOD(EID,2)=0;
```

110. WAQTD THE AVG SALARY OBTAINED BY EVERY DEPT. ROUND OFF THE VALUES TO 100'S PLACE.

```
SELECT ROUND(AVG(SAL),-2) AVG_SAL,DNO
FROM EMP
GROUP BY DNO;
```

==============================================

===================================

DATE FUNCTIONS
==============

SYSDATE(): This function is used to obtain the current date and time.

CURDATE(): This function is used to obtain the current date.

DATEDIFF(): This function is used to obtain the differences between two dates.
Syntax: SELECT DATEDIFF(CURDATE(),'2024-09-01');

DATE_ADD(): This function is used to add the time duration based on the interval.
Syntax: DATE_ADD(date_value, INTERVAL time_dur);

DATE_SUB(): This function is used to subtract the time duration based on the interval.

Syntax: DATE_SUB(date_value, INTERVAL time_dur);

DATE_FORMAT(): This function is used to convert the date values into individual characters based on Format Model.
Syntax:
DATE_FORMAT(date_value,'format_model');

YEAR(): This function is used to obtain the year of the given date value.

MONTH(): This function is used to obtain the month of the given date value.