10. SQL Queries - Scalar Functions

SQL scalar functions return a single value, based on the input value. The most useful scalar functions are the following:

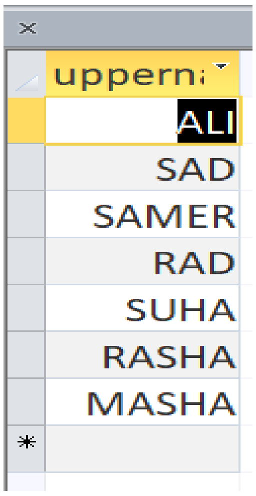
- **UCASE( )** - converts a field to upper case;
- **LCASE( )** - converts a field to lower case;
- **LEN( )** - returns the length of a text field;
- **ROUND( )** - rounds a numeric field to the number of decimals specified.

In the following paragraphs we will an example of each scalar functions.

1- UCASE( ) operation

Ex: Display the customer name field in upper case form?

Sol:- SELECT UCASE (cust_name) as uppername from Customer;

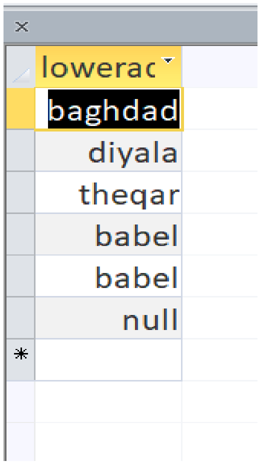


uppern:
ALI
SAD
SAMER
RAD
SUHA
RASHA
MASHA

2- LCASE( ) operation

Ex: Display the customer address field in lower case form where customer address is not basra?

Sol:- SELECT LCASE (cust_name) as loweraddress from Customer;

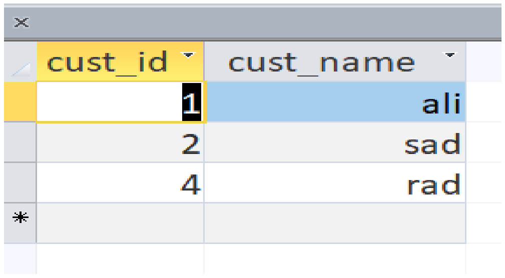


lowerac
baghdad
diyala
theqar
babel
babel
null

3- LEN( ) operation

Display the customer id and customer name where the length of customer name<=3 from the customer table?

Sol:- SELECT cust_id,cust_name from Customer where (LEN(cust_name) from Customer)<=3;

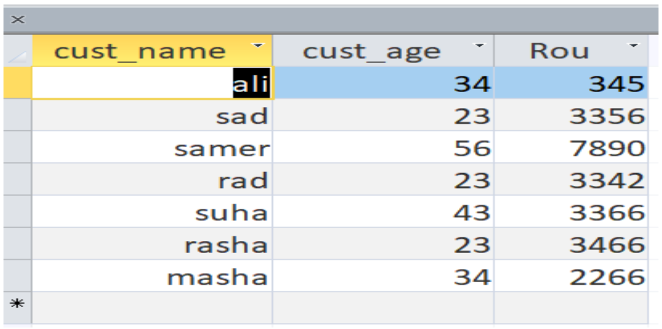


cust_id	cust_name
1	ali
2	sad
4	rad

4- ROUND( ) operation

Ex: Display the customer id and customer name where the length of customer name<=3 from the customer table?

Sol: SELECT cust_name, cust_age, ROUND(cust_mobile,2) AS Rou FROM Customer;



cust_name	cust_age	Rou
ali	34	345
sad	23	3356
samer	56	7890
rad	23	3342
suha	43	3366
rasha	23	3466
masha	34	2266

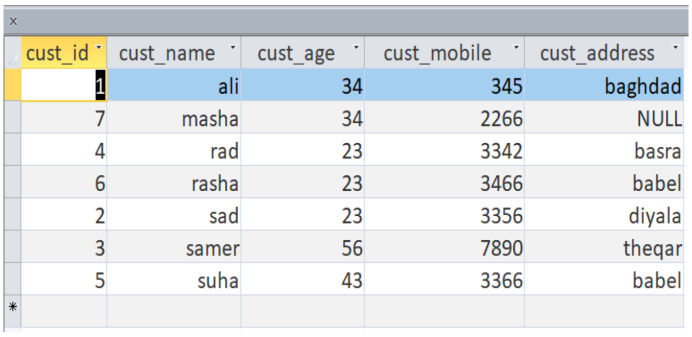
11. SQL Queries - Ordering Data

The ORDER BY keyword is used to sort the result-set by one or more columns. The ORDER BY keyword sorts the records in ascending order by default or using ASC keyword. To sort the records in a descending order, you can use the DESC keyword. Here we show two examples using the "Order By..." syntax.

The "Group By..." statement is used in conjunction with the aggregate functions to group the result-set by one or more columns.

EX1: display the content of customer table ORDERING by customer name?

Sol:- select *from Customer order by cust_name;

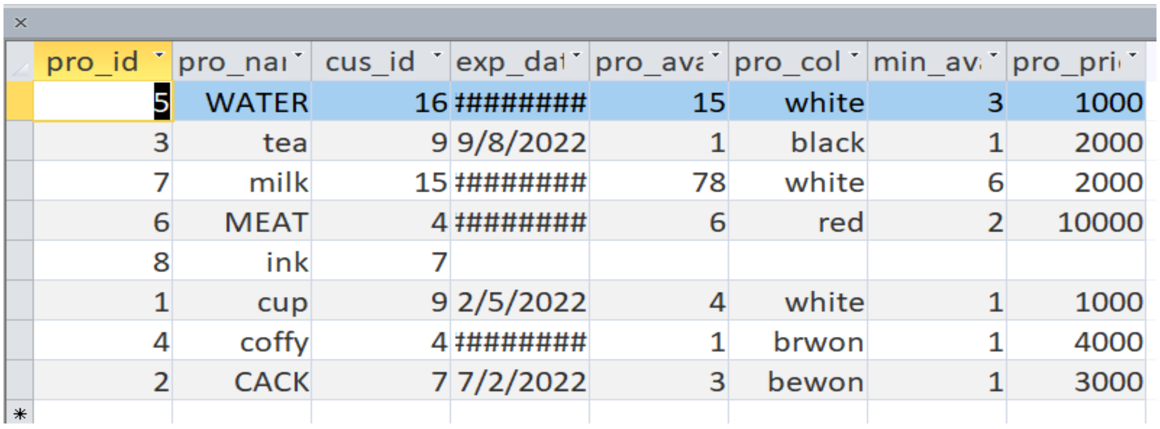


cust_id	cust_name	cust_age	cust_mobile	cust_address
1	ali	34	345	baghdad
7	masha	34	2266	NULL
4	rad	23	3342	basra
6	rasha	23	3466	babel
2	sad	23	3356	diyala
3	samer	56	7890	theqar
5	suha	43	3366	babel
*				

EX2:- In second example we use the same Order By operation but now using the inverse order.

Ordering the content of product table based on the descent product name order?

Sol:- SELECT *from product order by pro_name DESC;



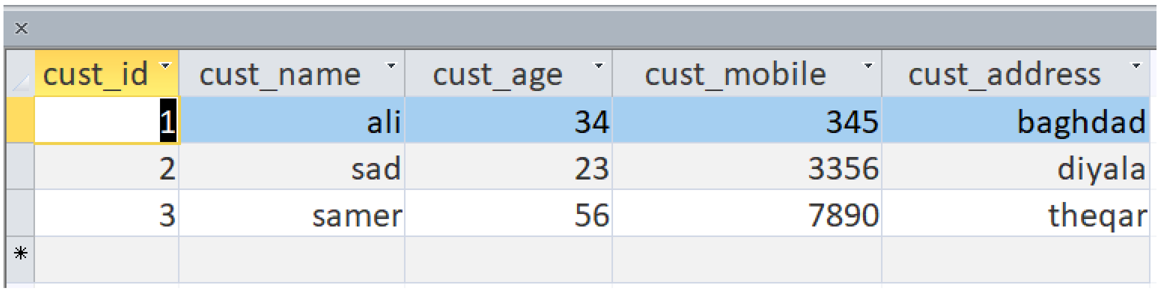
pro_id	pro_nai	cus_id	exp_dat	pro_ava	pro_col	min_av	pro_pri
5	WATER	16	########	15	white	3	1000
3	tea	9	9/8/2022	1	black	1	2000
7	milk	15	########	78	white	6	2000
6	MEAT	4	########	6	red	2	10000
8	ink	7					
1	cup	9	2/5/2022	4	white	1	1000
4	coffy	4	########	1	brwon	1	4000
2	CACK	7	7/2/2022	3	bewon	1	3000
*							

12. SQL Queries - Returning Top Elements

There is an useful statement in SQL that lets the user to return only the first n elements. This clause can be very useful on large tables with thousands of records, which may have deep impact on performance.

EX1: display the top three elements from customer table?

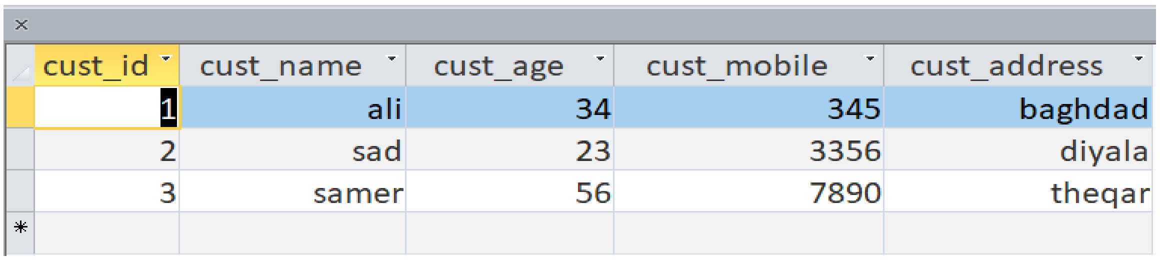
Sol:- SELECT top 3 *from Customer;



cust_id	cust_name	cust_age	cust_mobile	cust_address
1	ali	34	345	baghdad
2	sad	23	3356	diyala
3	samer	56	7890	theqar
*				

EX2:- Display the top three elements from customer table where the summation of the customer mobile column for the mobile number less than 3000 is more than 500?

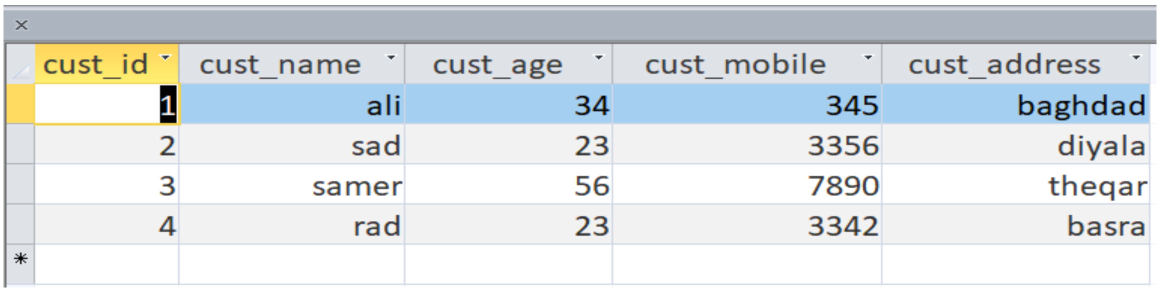
Sol: SELECT top 3 * from Customer where (select SUM(cust_mobile) as total from Customer where cust_mobile<3000)>500;



cust_id	cust_name	cust_age	cust_mobile	cust_address
1	ali	34	345	baghdad
2	sad	23	3356	diyala
3	samer	56	7890	theqar
*				

EX3:- Display the 4 elements from customer table where the summation of the customer mobile column for the mobile number less than 3000 is more than 500 and the maximum value of the Customer age column is less than 80?

Sol: SELECT top 4 * from Customer where (select AVG(cust_mobile) as AVER from Customer where cust_mobile<3000)>500 and (Select Max(cust_age) from Customer)< 80;



cust_id	cust_name	cust_age	cust_mobile	cust_address
1	ali	34	345	baghdad
2	sad	23	3356	diyala
3	samer	56	7890	theqar
4	rad	23	3342	basra
*				