

Created by Ahsan Arif

## MAX Function

---

The MAX function returns the maximum value of an expression.

The syntax for the MAX function is:

```
SELECT MAX(expression )  
FROM tables  
WHERE predicates;
```

### Simple Example

For example, you might wish to know the maximum salary of all employees.

```
SELECT MAX(salary) as "Highest salary"  
FROM employees;
```

In this example, we've aliased the max(salary) field as "Highest salary". As a result, "Highest salary" will display as the field name when the result set is returned.

### Example using GROUP BY

In some cases, you will be required to use a GROUP BY clause with the MAX function.

For example, you could also use the MAX function to return the name of each department and the maximum salary in the department.

```
SELECT department, MAX(salary) as "Highest salary"  
FROM employees  
GROUP BY department;
```

Because you have listed one column in your SELECT statement that is not encapsulated in the MAX function, you must use a GROUP BY clause. The department field must, therefore, be listed in the GROUP BY section.

## Frequently Asked Questions

---

**Question:** I'm trying to pull some info out of a table. To simplify, let's say the table (report\_history) has 4 columns:

user\_name, report\_job\_id, report\_name, report\_run\_date.

Each time a report is run in Oracle, a record is written to this table noting the above info. What I am trying to do is pull from this table when the last time each distinct report was run and who ran it last.

**My initial query:**

```
SELECT report_name, max(report_run_date)
FROM report_history
GROUP BY report_name
```

runs fine. However, it does not provide the name of the user who ran the report.

Adding user\_name to both the select list and to the group by clause returns multiple lines for each report; the results show the last time each person ran each report in question. (i.e. User1 ran Report 1 on 01-JUL-03, User2 ran Report1 on 01-AUG-03). I don't want that....I just want to know who ran a particular report the last time it was run.

Any suggestions?

**Answer:** This is where things get a bit complicated. The SQL statement below will return the results that you want:

```
SELECT rh.user_name, rh.report_name, rh.report_run_date
FROM report_history rh,
  (SELECT max(report_run_date) as maxdate, report_name
   FROM report_history
   GROUP BY report_name) maxresults
WHERE rh.report_name = maxresults.report_name
AND rh.report_run_date= maxresults.maxdate;
```

Let's take a few moments to explain what we've done.

First, we've aliased the first instance of the report\_history table as rh.

Second, we've included two components in our FROM clause. The first is the table called report\_history (aliased as rh). The second is a select statement:

```
(SELECT max(report_run_date) as maxdate, report_name
 FROM report_history
 GROUP BY report_name) maxresults
```

We've aliased the max(report\_run\_date) as *maxdate* and we've aliased the entire result set as *maxresults*.

Now, that we've created this select statement within our FROM clause, Oracle will let us join these results against our original report\_history table. So we've joined the report\_name and report\_run\_date fields between the tables called *rh* and *maxresults*. This allows us to retrieve the report\_name, max(report\_run\_date) as well as the user\_name.

---

**Question:** I need help in an SQL query. I have a table in Oracle called *orders* which has the following fields: order\_no, customer, and amount.

I need a query that will return the customer who has ordered the highest total amount.

**Answer:** The following SQL should return the customer with the highest total amount in the orders table.

```
select query1.* from
  (SELECT customer, Sum(orders.amount) AS total_amt
   FROM orders
   GROUP BY orders.customer) query1,

  (select max(query2.total_amt) as highest_amt
   from (SELECT customer, Sum(orders.amount) AS total_amt
        FROM orders
        GROUP BY orders.customer) query2) query3
where query1.total_amt = query3.highest_amt;
```

This SQL statement will summarize the total orders for each customer and then return the customer with the highest total orders. This syntax is optimized for Oracle and may not work for other database technologies.

---

**Question:** I'm trying to retrieve some info from an Oracle database. I've got a table named *Scoring* with two fields - Name and Score. What I want to get is the highest score from the table and the name of the player.

**Answer:** The following SQL should work:

```
SELECT Name, Score
FROM Scoring
WHERE Score = (select Max(Score) from Scoring);
```

---

**Question:** I need help in an SQL query. I have a table in Oracle called *cust\_order* which has the following fields: OrderNo, Customer\_id, Order\_Date, and Amount.

I would like to find the customer\_id, who has Highest order count.

I tried with following query.

```
SELECT MAX(COUNT(*)) FROM CUST_ORDER GROUP BY CUSTOMER_ID;
```

This gives me the max Count, But, I can't get the CUSTOMER\_ID. Can you help me please?

**Answer:** The following SQL should return the customer with the highest order count in the cust\_order table.

```
select query1.* from
  (SELECT Customer_id, Count(*) AS order_count
   FROM cust_order
   GROUP BY cust_order.Customer_id) query1,

  (select max(query2.order_count) as highest_count
   from (SELECT Customer_id, Count(*) AS order_count
        FROM cust_order
        GROUP BY cust_order.Customer_id) query2) query3
where query1.order_count = query3.highest_count;
```

This SQL statement will summarize the total orders for each customer and then return the customer with the highest order count. This syntax is optimized for Oracle and may not work for other database technologies.