

Tips to write good SQL

ORACLE

Copyright © Oracle Corporation, 2004. All rights reserved.

Objectives

After completing this lesson, you should be able to do the following:

- **Tip to write SQL using index**
- **Tip to write efficient SQL**

Tips to write SQL using index

- **Don't use NOT EQUAL Operators**

Index can tell you what is there in a table but not what is not in a table and the index is used when optimizer_goal is FIRST_ROWS

Select * from emp where empno != 0;

Tips to write SQL using index

- Don't use Function

Once using function, optimizer cannot use the index unless functional index

Select * from emp where substr(ename, 1, 1) = 'K';

Tips to write SQL using index

- **Don't compare mismatched data type**

Select * from emp where empno = '7788';

•

Tips to write SQL using index

- **Use EXIST function instead of IN function**

Select product_id, qty from product where product_id = 167 and item_no in (select item_no from items); =====>

Select product_id, qty from product a where product_id = 167 and exists (select 'x' from items b where b.item_no = a.item_no);

- **Use hint to instruct optimizer to use better execution plan**

Tips to write SQL using index

ORACLE

Copyright © Oracle Corporation, 2004. All rights reserved.

Others Tips

- **Early elimination of candidate rows**

By early elimination of candidate rows, there is less operation in the later stage as well as resource.

Assume that invoice_lines is large table

Select v.vendor_num, l.invoice_num, sum (l.amount) from vendors v, invoices l,
invoice_lines l where

v.vendor_name = 'ACME' and l.vendor_num = v.vendor_num and l.vendor_num =
l.vendor_num and l.invoice_num = l.invoice_num and

i.paid = 'N' group by v.vendor_num, l.invoice_num order by l.invoice_num;

=====è

Select v.vendor_num, l.invoice_num, sum (l.amount) from vendors v, invoices l,
invoice_lines l where

v.vendor_name = 'ACME' and l.vendor_num = v.vendor_num and l.paid = 'N' and
l.vendor_num = l.vendor_num and l.invoice_num = l.invoice_num

Group by v.vendor_num = l.invoice_num order by l.invoice_num;

Others Tips

- **Minimize the number of throwaway rows**

- Compare the number of rows from the two input row sources with the number of rows from the join operation. If both input row sources have more rows than the join operation has, you have identified a throwaway of rows.
- Example

Rows	Operation
100	NESTED LOOPS
45	TABLE ACCESS (...) OF our_outer_table
4530	TABLE ACCESS (...) OF our_inner_table

- **Select the table with less number of records as driving table**

Other Tips

ORACLE

Copyright © Oracle Corporation, 2004. All rights reserved.