Oracle中如何查询某个字段值包含小写字母？

如表info\_oper中oper\_no这个字段包含‘AAqq83’这样的字段

查询包含小写的所有数据：

Select oper\_no from info\_oper where regexp\_like(oper\_no, ’[[:lower:]]’);

OR

Select oper\_no from info\_oper where regexp\_like(oper, ’[a-z]’);

查询包含大写的所有数据：

Select oper\_no from info\_oper where regexp\_like(oper\_no, ‘[[:upper:]]’);

OR

Select oper\_no from info\_oper where regexp\_like(oper\_no, ‘[A-Z]’);

SqlServer获取插入行的id：

insert into tbale1(name) OUTPUT inserted.ID values(“emy”);

Oracle获取插入行的id：

Insert into table1(name) values (“emy”) RETURN ID INTO newID; newID 为自定义的字段。

Oracle实现两个字段数据的拼接的两种方式：

一、Select a.id ||’ ’||b.value from A a, B b where a.id = b.id;

最终实现两个字段的结果以中间留空格的形式来呈现。

二、CONCAT函数：

SELECT CONCAT(字段A, 字段B) FROM 表名。

Oracle中（+）的使用：

（+）在Oracle中表示外连接，并且总是放在非主表的一方。

例如：select A.a, B.a from A,B where A.b = B.b(+);

等价于select A.a, B.a from A,B from A LEFT JOIN B ON A.b=B.b;(即左连接)

数据表的连接有：

内连接：只有两个表相匹配的行才能在结果集中出现

外连接：包括左连接、右连接、全外连接。

有（+）的一方可以为空，即副表。

Oracle中 :

变量绑定：是指在sql语句的条件中使用变量而不是常量。

比如sharepool里有两个sql语句，

Select \* from tab1 where col1 = 1;

Select \* from tab1 where col1 = 2;

对Oracle数据库来说，这是两条完全不同的SQL，对这两条语句都需要进行hard parse。因为Oracle会根据SQL语句的文本去计算每个字符在内存里的hash值，因此虽然上述两个SQL只有一个字符不一样，Oracle根据hash算法在内存中得到的hash地址就不一样，所以Oracle就会认为这是两个完全不同的语句。而如果将上述的SQL改写成：select \* from tab1 where col1 =：val1；然后通过对变量var1 的赋值去查询，那么Oracle对这条语句第一次会进行hard parse，以后就只进行soft parse。假设某条语句被重复执行了几十万次，那么使用bind var 带来的好处就是巨大的。一个应用程序如果bind var 使用不充分，那么几乎一定会伴随着严重的性能问题。

绑定变量是相对文本变量来讲的，所谓文本变量是指在SQL直接书写查询条件，这样的SQL在不同条件下需要反复解析，绑定变量是指使用变量来代替直接书写条件，查询bind var 在运行时传递，然后绑定执行。优点是减少硬解析，降低CPU的争用，节省shared\_pool。缺点是不能使用histogram，SQL优化比较困难。

Oracle數據庫：no listener錯誤

查看文件listener.ora。已收藏相應網頁（記得學習查看）