1. update语句的第二种写法

update

(select \* from iesk\_stockAccept where orgID = 'xx')

set billNo = billNo||'aa'

1. 相关子查询和非相关子查询

能单独运行的子查询即是非相关子查询，反之则是相关子查询

1. 多值查询

update user set (birthday,degree) =

(

--- 子查询

);

1. Sql语句执行顺序 on只是用来定义连接条件的，不会过滤结果集
2. from
3. join on
4. where
5. group by
6. having
7. select
8. order by
9. fetch first
10. null值的处理

尽量不要让有null值的列出现，因为它破坏了二值(true和false)逻辑结构

1. 日期和时间(只有加减操作)

三种格式 Time hh:mm:ss Date yyyy-mm-dd TimeStamp yyyy-mm-dd hh:mm:ss.zzzzzz

示例：values DATE(‘2012-08-13’) + 1 DAY ; 注意空格

语法和人类英语相当

1. 数据类型转换
2. 直接用数据类型作为函数 如 values INTEGER('12.3')
3. cast函数 如 values cast('12.8' as INTEGER)

注意：转换时可能会发生截断

1. 好好利用执行计划，分析sql语句执行效率
2. values语句

定义临时集合：如 values (1,2),(3,4) 一个括号为一行，括号内一个逗号为一列

这个临时集合可排序、分组，可使用聚合函数

1. 公共表表达式with

with temp(billNo,busDate) as

(select a.billNo,a.busDate from iesk\_stockAccept a)

select temp.billNo from temp

注意语法，另外没有第三行不能执行，和values不同

1. 四属性表设计，行转列

一些技巧：

01 减少数据库访问次数，因为数据库连接是昂贵的

02 避免在有索引的字段上使用函数，这会使索引失效

03 排序操作放在最后

PL/SQL

01、声明变量

DECLARE <variable-name> <data-type> <DEFAULT constant>

declare x integer default 0;

02、变量赋值

A、set x=10; set x=(select sum(attr) from table);

B、values 10 into x;

C、select sum(attr) into x from table;

03、存储过程

create procedure 过程名()

begin

--- 变量声明，赋值，执行语句

end!

调用：call过程名()

04、游标(Cursor)

有点像Java中的List类，用来定义一个集合，并可以遍历这个集合。

声明游标：

DECLARE <cursor-name> CURSOR [WITHOUT HOLD | WITH HOLD]

[WITHOUT RETURN | WITH RETURN TO CALLER | WITH RETURN TO CLIENT]

FOR <select-statement>

HOLD，语句commit之后游标仍然处于打开状态。rollback后游标都会关闭。

return to caller，游标的结果集返回给调用者；return to client，游标的结果集返回给客户端，如果有嵌套，跳出嵌套。