

1. 游标

2. 存储过程

1. 游标

相当于指针, 通过游标PL/SQL程序可以一次处理查询结果集中的一行, 并可以对该行数据执行特定操作

a. 显示游标: 显式游标用于处理返回多行的查询。

步骤如下:

- i. 声明游标: `cursor cur_name [(in_param [, in_param]...)] [return return_type] is select_sentence;`
- ii. 打开游标: `open cur_name [(in_param [, in_param]...)];`
- iii. 读取游标: `fetch cur_name into (variable);`
- iv. 关闭游标: `close cur_name`

a. 隐式游标: 在 PL/SQL 程序中执行DML SQL 语句时自动创建隐式游标, 名字默认叫sql, 感觉隐式就是普通的 PL/SQL (特别是配合for使用), 无形中使用, 因为打开, 读取等操作Oracle系统自动完成

a. REF 游标: REF 游标用于处理运行时才能确定的动态 SQL 查询的结果

<https://www.2cto.com/database/201501/371435.html>

<http://www.cnblogs.com/sc-xx/archive/2011/12/03/2275084.html>

2. 存储过程

存储过程相当于java中函数, 将一些操作集合起来. 存储过程报错在数据库中, 可以被重复使用, (也正因为如此, 存储过程不便迁移), 存储过程是已经编译好的代码, 所有被引用时, 效率非常高 (这也是项目中喜欢用存储过程的原因)

语法:

```
create [ or replace ] procedure pro_name [ (in_param in/out param_type [ ,  
in_param in/out param_type ]...) ] is|as  
begin  
  plsql_sentences;  
  [ exception ]  
  [ do_something_sentences; ]  
end [ pro_name ]
```

调用:

```
begin  
  pro_name [ (in_param [ , in_param]...) ];  
end;
```

函数必须有返回值, 这是与存储过程的最大的不同, 也说明了函数不如存储过程灵活