<https://blog.csdn.net/lgdreamsky/article/details/11393999>

**Linux下的PRO\*C函数库**

PRO\*C是Oracle提供的应用程序专用开发工具，其宿主语言是C语言，能在C程序中嵌入SQL语句，进行数据库操作。这种嵌入式的SQL语句容易掌握，适合初学者。而且PRO\*C使用预编译技术，预编译器将源程序中的SQL语句转换为标准的Oracle库函数调用，从而生成C源程序，再经C编译器编译、链接后直接生成可执行文件，该预编译器是Oracle自带的。

使用PRO\*C必须要有Oracle提供的头文件和库函数，Oracle安装程序将这些文件安装在$ORACLE\_HOME/precomp目录下。

使用Oracle PRO\*C对Oracle数据库操作一般包括以下几个部分：

说明SQL通讯区：SQL通讯区用来记录执行每一个嵌入SQL语句的状态信息，通过在函数体外使用下面两个语句中的任一个实现：

1. #include **<sqlca.h>**
2. EXEC SQL INCLUDE sqlca;

声明宿主变量，即C语言变量：这些变量是应用程序与Oracle通信的桥梁，应用程序的输入数据通过C变量传递给Oracle，反之，Oracle的输出数据又通过C变量传递给应用程序。举例如下：

1. EXEC SQL BEGIN DECLARE SECTION;
2. char szUsername[16];
3. VARCHAR varPassword[16];
4. char \*szStmt1="CREATE TABLE USERS (USERNAME VARCHAR2(15) NOT NULL,
5. PASSWORD VARCHAR2(15) NOT NULL)";
6. char \*szStmt2= "SELECT PASSWORD FROM USERS WHERE USERNAME='chen'";
7. EXEC SQL END DECLARE SECTION;

值得注意的是：在SQL语句中使用C变量时，前面须加冒号，例如上面的变量应表示为：szUsername。其中，VARCHAR为C扩展数据类型，预编译时，PRO\*C预编译器将其扩展为一个C结构类型。

1. struct{
2. unsigned short len;
3. unsigned char arr[16];
4. }varNo;

在SQL语句中使用VARCHAR类型变量时，只要指出结构名称var Password就可以，但在C语句中使用VARCHAR类型变量时，必须具体说明所操作变量的结构元素名称是var Password.len还是var Password.arr。另外，如果用VARCHAR类型变量做函数参数的话，只能用指针形式。

连接数据库：

1. EXEC SQL CONNECT :username/password@DBname;

通过sqlca.sqlcode的值来判断连接数据库成是否功。

执行SQL语句：SQL语句可分为静态SQL语句和动态SQL语句。其中，静态SQL语句是在开发应用程序时就已经明确了的数据库操作，如：

1. EXEC SQL  SELECT password INTO :szPassword
2. FROM USERS
3. WHERE username=:szUsername;

动态SQL语句是在运行时由外部数据提供的，不能直接在C程序中嵌入SQL语句，但可以调用放在一个字符串变量里的SQL语句，最简单的方法是：EXEC SQL EXECUTE IMMEDIATE :szStmt1;但这样执行的SQL语句不能实现查询，实现查询可用下列方法：

1. EXEC SQL PREPARE select\_stmt FROM :szStmt2;
2. EXEC SQL EXECUTE select\_stmt INTO :szPassword;

如果不再需要已准备好的语句，应释放该语句，释放语句使用如下指令：

1. EXEC SQL DEALLOCATE PREPARE select\_stmt;

提交或回滚所做的数据库处理，并退出数据库。提交数据库处理使用如下指令：

1. EXEC SQL COMMIT WORK RELEASE;

滚回数据库操作使用如下指令：

1. EXEC SQL ROLLBACK WORK RELEASE;

语句中的RELEASE选项，用于关闭所有打开的游标，并断开与数据库服务器的连接。