

# PL/SQL 简明教程

翻译作品

屈庆磊

[quqinglei@icloud.com](mailto:quqinglei@icloud.com)

2013 年 7 月

## 目录

<b>1</b>	<b>关于此文档</b>	<b>2</b>
1.1	PL/SQL 简明教程	2
1.2	面向读者	2
1.3	先决条件	2
1.4	版权声明	2
<b>2</b>	<b>PL/SQL 总览</b>	<b>2</b>
2.1	简介	2
2.2	特性	3
2.3	优势	3
<b>3</b>	<b>基本语法</b>	<b>3</b>
3.1	范例	3
3.2	HelloWorld	4
3.3	标识符	4
3.4	分隔符	4
3.5	注释	5
3.6	PL/SQL 程序单元	5
<b>4</b>	<b>数据类型</b>	<b>6</b>

## 1 关于此文档

### 1.1 PL/SQL 简明教程

PL/SQL 是添加了过程特性的一门编程语言，是甲骨文公司于九十年代初期开发的对 SQL 的增强实现。

PL/SQL 是甲骨文数据库系统的三个关键点之一，其余的两个点为 SQL 本身，以及 JAVA。

本文档将会带给你一个对 PL/SQL 以及关系型数据库，比较容易理解的学习过程。

### 1.2 面向读者

本文档面向有熟练编程经验的专家，提供简单的学习方式。本文档会给你提供一个很容易理解的学习方式，当完成本文档的学习后，你有一个中级的能力。

### 1.3 先决条件

在看本书的时候，你需要对计算机科学有一定的了解，比如基本的软件概念，如什么是数据库，源代码，文本编辑器，程序运行方式，当然如果你懂 SQL 并且有其他语言的编程经验，那你可以比较轻松的学习了。

### 1.4 版权声明

© 本文档里的所有内容、图片皆为[tutorialspoint.com](http://tutorialspoint.com)所有。来自此网站的任何内容或者本文档的内容，在没有经过同意下不允许任何形式的再发行。如果不如此做则违法版权法。

此文档可能存在不准确或者错误，本网站并不承担任何担保，如果你找到了错误请联系[webmaster@tutorialspoint.com](mailto:webmaster@tutorialspoint.com)

## 2 PL/SQL 总览

本章节介绍 *PL/SQL* 的基本概念和定义

### 2.1 简介

PL/SQL 语言是甲骨文公司于二十世纪八十年代末发明的一种关系型数据库，下面是 PL/SQL 的一些可圈点的特点：

- 可移植、高性能的事务性处理语言
- 提供独立于操作系统的内建语言环境
- 可以直接通过命令行调用
- 可以直接通过外部程序调用
- 语法采用 ADA 和 Pascal 的语法格式
- 此语言不只是只适合甲骨文数据库，同样适用于 TimesTen 内存数据库以及 IBM DB2

## 2.2 特性

- 和 SQL 紧密关联
- 提供广泛的错误检查
- 提供很多的数据类型
- 提供结构化编程特性
- 提供函数以及过程调用
- 支持面向对象的编程
- 支持网页程序和服务器编程

## 2.3 优势

- SQL 是一种标准数据库语言，PL/SQL 和 SQL 紧密结合。支持静态和动态的 SQL。支持数据操纵、事务控制。动态 SQL 是允许嵌入 PL/SQL 块中的数据定义语句。
- 支持对代码块的一次整块发送，避免网络拥堵提供程序高性能。
- 提供程序员高效的操作
- 让你少花费很多时间考虑一些其他的東西
- 提供很高的安全性
- 提供对已定义包的直接调用
- 提供面向对象的编程
- 提供对网络程序和服务器程序的支持

# 3 基本语法

本章节描述一些基本的语法规则

## 3.1 范例

*PL/SQL* 是一种结构化语言，这意味着用它写的程序可以划分为很多程序块，每个块包含以下三个子块

**定义** 本块开始于关键字 **DECLARE**。是一个可选择的模块，定义所有的变量、游标、子程序、以及其他元素。

**可执行命令** 本块开始于 **BEGIN** 结束于 **END** 是一个强制性必须要有的模块。至少要有一行代码，哪怕是一个 **NULL** 命令。

**异常处理** 本块开始于关键字 **EXCEPTION** 此块是可选择的，可有可无。用来处理错误和异常。

每一个语句要以分号结束，PL/SQL 语句可以通过 **BEGIN END** 嵌套于其他语句里，下面是一个很基本的 PL/SQL 模块：

```
DECLARE
    <declarations section>
BEGIN
    <executable command(s)>
EXCEPTION
    <exception handling> END;
```

### 3.2 HelloWorld

```
DECLARE
    message varchar2(20) := 'Hello, World!';
BEGIN
    dbms_output.put_line(message);
END;
/
```

### 3.3 标识符

PL/SQL 标识符包含常量、变量、异常、过程、游标和一些保留字。标识符是由字符、特殊符号、美元符号等符号组成。标识符的长度不要大于 30 个字符。这东西我就不多说了，对于有编程经验的人意义不是很大，因为都是相通的。

### 3.4 分隔符

+, -, \*, / 加、减、乘、除

% 属性指标

' 字符串分隔符

. 元件选择

(,) 表达式或者是列表分割

: 主变量指示符

, 分隔符

" 引用标识符分隔符

= 关系操作符

@ 远程访问操作符

; 语句终结符

:= 赋值操作符

=> Association operator

字符串连接符

\*\* Exponentiation operator

<<, >> Label delimiter (begin and end)

/\*, \*/ 多行注释

-- 单行注释

.. 范围

<, >, <=, >= 关系操作符

<>, !=, =, = 不等号的其他版本

### 3.5 注释

废话少说，就是注释，注释就是把一些说明性的文档用某些约定的符号注释上，编译器会自动过滤掉此类语句，下面是个例子

```
DECLARE
    -- variable declaraton 其实这就是注释啦，此行不会执行
    message varchar2(20) := 'Hello, World!';
BEGIN
    /*
        PL/SQL executable statement(s)
        这里面就是多行注释，你可以写很多行说明的
    */
    dbms_output.put_line(message);
END;
/
```

当以上代码执行完后，会有以下输出：

Hello, World

PL/SQL procedure successfully completed.

### 3.6 PL/SQL 程序单元

一个 PL/SQL 程序单元可能是以下任何项目：其实这些都是废话，还是看实例比较好

- PL/SQL 块
- 函数
- 包
- 包体
- 过程

- 触发器
- 类型
- 类型块

以上所有的项目都会在下文的讲解中提到

4 数据类型

PL/SQL 变量、常量、参数必须拥有合法的数据类型，废话我就不翻译了

种类	说明
标量	单一值，无内部结构，如 NUMBER, DATE, BOOLEAN
大对象 (LOB)	比如文本、图像、视频剪辑、声音文件
混合变量	有内部结构的变量，如集合与记录
参考	指向其它数据的东东 (tbd)