



Programming
Your Future

Oracle SQL开发基础



Programming Your Future

课程结构

内容	课时 (H)
第一章 Oracle数据库基础	2.5
第二章 编写简单的查询语句	1.5
第三章 限制数据和对数据排序	2
第四章 单行函数	4
第五章 多表查询	4
第六章 分组函数	3
第七章 子查询	4
第八章 数据操作与事务控制	4
第九章 表和约束	4
第十章 其他数据库对象	4

第二章 编写简单的查询语句

目标:

本章旨在向学员介绍:

- 1) 基本查询语句及书写规则
- 2) 别名的使用方法
- 3) 关键字的使用方法

时间: 学时

教学方法:

本章要点

- SQL语言简介
- 基本查询语句
- SQL语句的书写规则
- 算术表达式的使用
- 空值（NULL）的应用
- 列别名的使用
- 连接运算符的使用
- DISTINCT关键字的用法
- SQL*PLUS/iSQL*PLUS命令的介绍

第二章 编写简单的查询语句

编写简单的查询语句:

2.1 案例环境及**SQL**语言的概念

2.2 基本**SELECT**语句

2.3 **SQL*PLUS/iSQL*PLUS**命令的介绍

2.1.1 课程案例环境简介 1/2

- EMPLOYEES（员工信息表）
 - 主要有employee_id（员工编号）、last_name（姓）、job_id(职位)、salary（工资）等。
- JOBS（职位信息表）
 - 主要有job_id(职位)、job_title（职位全称）等。
- JOB_GRADES（工资级别表）
 - 主要有grade_level（工资级别）、lowest_salary（最低工资）、highest_salary（最高工资）等。

2.1.1 课程案例环境简介 2/2

- departments(部门信息表)
 - 主要包括department_id（部门编号）、department_name(部门名称)、location_id（位置编号）等。
- locations（位置信息表）
 - 主要包括location_id(位置编号)、street_adress（地址）、city（城市）等。

2.1.2 SQL语言简介 1/2

- SQL称结构化查询语言 (Structured Query Language)
- SQL 是操作和检索关系型数据库的标准语言。已在Oracle、DB2等数据库管理系统上得到了广泛应用。
- 使用SQL语句，程序员和数据库管理员可以完成如下的任务：
 - 改变数据库的结构
 - 更改系统的安全设置
 - 增加用户对数据库或表的许可权限
 - 在数据库中检索需要的信息
 - 对数据库的信息进行更新

2.1.2 SQL语言简介 2/2

SQL语句分类

- DQL语句（数据查询语言）
 Select
- DML语句（数据操作语言）
 Insert / Update / Delete / Merge
- DDL语句（数据定义语言）
 Create / Alter / Drop / Truncate
- DCL语句（数据控制语言）
 Grant / Revoke
- TCL语句事务控制语句
 Commit / Rollback / Savepoint

第二章 编写简单的查询语句

编写简单的查询语句:

2.1 案例环境及SQL语言的概念

2.2 基本SELECT语句

2.3 SQL*PLUS/iSQL*PLUS命令的介绍

2.2.1 基本SELECT语句语法 1/3

- 基本查询语句语法:

```
SELECT  *|[DISTINCT] 列名|表达式 [别名]
[,...]}
FROM    表名;
```

- “*” 号的使用
- 在查询语句中查找特定的列

2.2.1 基本SELECT语句语法 2/3

- 查询公司所有部门的信息。

```
SELECT *  
FROM departments;
```

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500
90	Executive	100	1700
100	Finance	108	1700
110	Accounting	205	1700
200	Operations		1700

已选择9行。

2.2.1 基本SELECT语句语法 3/3

- 查询公司所有部门的信息。

```
SELECT department_id,  
       department_name,manager_id,location_id  
FROM departments;
```

2.2.2 在查询语句中查找特定的列

```
SELECT department_name, location_id  
FROM departments;
```

DEPARTMENT_NAME	LOCATION_ID
Administration	1700
Marketing	1800
Shipping	1500
IT	1400
Sales	2500
Executive	1700
Finance	1700
Accounting	1700
Operations	1700

已选择9行。

2.2.3 SQL语句的书写规则 1/2

- SQL语句相关概念：
 - 关键字（**Keyword**）：SQL语言保留的字符串，在自己的语法使用。例如，**SELECT** 和**FROM** 是关键字。
 - 语句（**statement**）：一条完整的SQL命令。例如，**SELECT * FROM departments;**是一条语句。
 - 子句（**clause**）：部分的SQL语句，通常是由关键字加上其他语法元素构成。例如，**SELECT ***是子句，**FROM departments**也是子句。

2.2.3 SQL语句的书写规则 2/2

- 不区分大小写。也就是说SELECT, select, Select, 执行时效果是一样的。
- 可以单行来书写,也可以书写多行,建议分多行书写,增强代码可读性。通常以子句分行。
- 关键字不可以缩写、分开以及跨行书写。如SELECT不可以写成SEL或SELE CT等形式。
- 每条语句需要以分号(;)结尾。
- 关键字大写,其他语法元素(如列名、表名等)小写。
- 代码适当缩进。

2.2.4 算术表达式的使用 1/2

- 算术运算符：+，-，*，/
- 算术表达式中优先级规则：
 - 先算乘除，后算加减。
 - 同级操作符由左到右依次计算。
 - 括号中的运算优先于其他运算符。
- 对NUMBER型数据可以使用算数操作符创建表达式(+ - * /)
- 对DATE型数据可以使用部分算数操作符创建表达式 (+ -)

2.2.4 算术表达式的使用 2/2

优先级示例

```
SELECT employee_id, last_name, salary, salary+400  
FROM employees;
```

```
SELECT employee_id, last_name, salary, 400+salary*12  
FROM employees;
```

```
SELECT employee_id, last_name, salary, (400+salary)*12  
FROM employees;
```

2.2.5空值（NULL）的应用

- **NULL**：表示未定义的，未知的。空值不等于零或空格。任意类型都可以支持空值。
- 空值（**NULL**）在算术表达式中的使用
 - 包括空值的任何算术表达式都等于空
 - 包括空值的连接表达式(||)等于与空字符串连接，也就是原来的字符串

```
SELECT last_name, salary, (400+salary)*12+(400+salary)
      *12*commission_pct
FROM employees;
```

2.2.6列标题的显示规则

- Sql*plus的默认显示
 - Date和character 型数据左对齐
 - Numeric 型数据右对齐
 - 列标题默认显示为大写
- iSql*plus的默认显示
 - 列标题缺省居中对齐
 - 列标题默认显示为大写

2.2.7使用列别名的方法 1/2

- 列别名基本书写方法有两种方式：
 - 第一种方式：列名 列别名
 - 第二种方式：列名 **AS** 列别名
- 以下三种情况，列别名两侧需要添加双引号（" "）：
 - 列别名中包含有空格
 - 列别名中要求区分大小写
 - 列别名中包含有特殊字符

2.2.7 使用列别名的方法 2/2

```
SELECT employee_id id, last_name as employee_name,  
       salary "Salary", (400+salary)*12 "Annual Salary"  
FROM   employees;
```

ID	EMPLOYEE_NAME	Salary	Annual Salary
100	King	24000	292800
101	Kochhar	17000	208800
102	De Haan	17000	208800

已选择26行。

2.2.8 连接运算符的使用

- 采用双竖线（||）来做连接运算符，更清楚地表达实际意思。

```
SELECT first_name||' '||last_name||"'s phone number is  
      '||phone_number "employee Phone number"  
FROM employees;
```

2.2.9 DISTINCT关键字的用法

- DISTINCT取消重复行

```
SELECT DISTINCT department_id  
FROM employees;
```

```
SELECT DISTINCT department_id, job_id  
FROM employees;
```

第二章 编写简单的查询语句

编写简单的查询语句:

2.1 案例环境及SQL语言的概念

2.2 基本SELECT语句

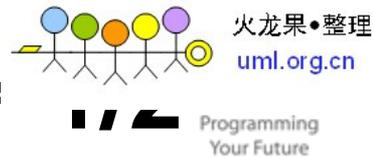
2.3 SQL*PLUS/iSQL*PLUS命令的介绍



2.3.1 SQL语句与SQL*PLUS命令的区别

- SQL语句与SQL*PLUS/iSQL*PLUS命令有以下主要区别：
 - SQL语句是开发语言，而SQL*PLUS/iSQL*PLUS是Oracle使用的工具。
 - SQL语句直接访问Oracle数据库，并返回结果；而SQL*PLUS/iSQL*PLUS命令是在返回结果上进行处理，如显示格式等。
 - SQL*PLUS/iSQL*PLUS命令只是使每个客户端环境有所不同，不会直接访问数据库。
 - SQL语句不可以缩写，而SQL*PLUS/iSQL*PLUS命令可以缩写。
 - SQL*PLUS/iSQL*PLUS命令结尾可以不加分号（;）。

2.3.2 DESC[RIBE]命令和SET命令



- DESC[RIBE]命令：显示表结构
 - DESC employees
- SET命令：设置环境变量
 - 语法：SET 系统变量 值
 - SET HEADING {ON|OFF}：控制是否显示列标题。默认是 ON
- 设置列的显示长度：
 - 对于字符字段，使用col HIREDATE format a100;
 - 对于数字字段，使用col DEPTNO format 999; 表示显示3位数字

2.3.2DESC[RIBE]命令和SET命令

```
SET HEADING OFF
```

```
col EMAIL format a20
```

```
select EMPLOYEE_ID,FIRST_NAME,LAST_NAME,EMAIL  
from employees;
```

```
SET HEADING ON
```

本章小结

- 基本的SQL查询语句的构成。
- 课程案例环境的设计。
- 查询语句的书写。
- 如何在查询语句中查找特定的列
- SQL语句的书写规则
- 算术表达式的使用
- 空值（NULL）的应用
- 列别名的使用
- 连接运算符的使用
- DISTINCT关键字的用法
- SQL*PLUS/iSQL*PLUS命令的介绍

练习

- 1.查询员工表中所有员工的信息。
- 2.查询员工表中员工的员工号、姓名、每个员工涨工资100元以后的年工资（按12个月计算）。
- 3.查询员工first_name和last_name，要求结果显示为“姓last_name名first_name”格式。
- 4.查询所有员工所从事的工作有哪些类型（要求去掉重复值）。

谢谢