

# 软件开发魔典

Oracle数据库从入门到项目实践  
(超值版)

软件开发魔典  
——从入门到项目实践

项目  
—— 实践

基础知识  
核心应用  
高级应用  
行业应用  
项目实践

清华大学出版社



## 本课时学习重点

- 数据库安全策略概述
- 用户账户管理
- 用户权限管理
- 数据库角色管理
- 概要文件的管理
- 资源限制与口令管理
- 锁定与解锁用户

## 数据库安全策略概述

- 安全性是评估一个数据库的重要指标，Oracle数据库从3个层次上采取安全控制策略：
  - (1) 系统安全性
  - (2) 数据安全性
  - (3) 网络安全性
- 数据库的安全可以从以下几个方面进行管理：
  - (1) 用户账户管理
  - (2) 权限和角色管理
  - (3) 数据口令加密管理
  - (4) 表空间设置和配额管理
  - (5) 用户资源限制管理



## 用户账户管理

- 预定义用户
- 用户的安全属性
- 用户的登录方式
- 新建普通用户
- 修改用户信息
- 查询用户信息
- 删除用户账户

## 预定义用户

- 在oracle创建时创建的用户，称为预定义用户，根据作用不同，预定义用户分为3类：
- 1 管理员用户
- 2.示例方案用户
- 3. 内置用户



## 用户的安全属性

- 在创建用户时，必须使用安全属性对用户进行限制，用户的安全属性主要包括用户名、用户身份认证、默认表空间、临时表空间、表空间配额、概要文件等。

## 用户的登录方式

- 在Oracle中，用户登录数据库的主要方式有3种：
- (1) 密码验证方式：把验证密码放在Oracle数据库中，这是最常用的验证方式，同时安全性也比较高。
- (2) 外部验证方式：这种验证的密码通常与数据库所在的操作系统的密码一致。
- (3) 全局验证方式：这种验证方式也不是把密码放在Oracle数据库中，也是不常用的验证方式。

## 新建普通用户

- 在Oracle数据库中，创建用户时需要特别注意的是：用户的密码必须以字母开头。用户可以使用**CREATE USER**语句创建用户。语法规则如下：
- **CREATE USER** username IDENTIFIED BY password
- OR EXTERNALLY AS certificate\_DN
- OR GLOBALLY AS directory\_DN
- [DEFAULT TABLESPACE tablespace]
- [TEMPORARY TABLESPACE tablespace| tablespace\_group\_name]
- [QUOTA size|UNLIMITED ON tablespace]
- [PROFILE profile]
- [PASSWORD EXPIRE]
- [ACCOUNT LOCK|UNLOCK]



## 修改用户信息

- 在Oracle数据库中，可以使用ALTER USER语句修改用户信息。具体使用的语法规则如下：
- ALTER USER username IDENTIFIED
- { BY password[REPLACE old\_password]
- | EXTERNALLY [AS certificate\_DN]
- | GLOBALLY [AS directory\_DN]}
- [DEFAULT TABLESPACE tablespace]
- [TEMPORARY TABLESPACE tablespace| tablespace\_group\_name]
- [QUOTA size|UNLIMITED ON tablespace]
- [PROFILE profile]
- [PASSWORD EXPIRE]
- [ACCOUNT LOCK|UNLOCK]

## 查询用户信息

- 在Oracle中，包含用户信息的数据字典如表所示。

视图名称	说明
DBA_USERS	包含数据库的所有用户的详细信息（15项）
ALL_USERS	包含数据库所有用户的用户名、用户ID和用户创建时间（3项）
USER_USERS	包含当前用户的详细信息（10项）
DBA_TS_QUOTAS	包含所有用户的表空间配额信息
USER_TS_QUOTAS	包含当前用户的表空间配额信息
V\$SESSION	包含用户会话信息
V\$SESSTAT	包含用户会话统计信息

## 删除用户账户

- 在Oracle数据库中，可以使用**DROP USER**语句删除用户账户，具体的语法规则如下：
- **DROP USER username [CASCADE];**
- 参数介绍如下：
- **username**: 为用户的名称；
- **CASCADE**: 为关键字，是可选参数，如果要删除的用户中没有任何数据库对象，可以省略**CASCADE**关键字。



## 用户权限管理

- 系统权限概述
- 系统权限授予
- 系统权限收回
- 对象权限概述
- 对象权限授予
- 对象权限收回
- 查看用户权限

## 系统权限概述

- 在Oracle中，一共有200多项系统权限，用户可通过数据字典 `system_privilege_map` 获得所有的系统权限。

## 系统权限授予

- 在Oracle中，必须是拥有GRANT权限的用户才可以执行GRANT语句。授予系统权限的语法如下：
- GRANT system\_privilege
- [ALL PRIVILEGES TO {user IDENTIFIED BY password|role}]
- [WITH ADMIN OPTION]

## 系统权限收回

- 只有数据库管理员才能收回系统权限，而且撤销系统权限的前提是当前的用户已经存在要撤销的系统权限。收回系统权限的语法规则如下：
- REVOKE system\_privilege
- FROM user|role

## 对象权限概述

- 对象权限是指对某个特定模式对象的操作权限。数据库模式对象所有者拥有该对象的所有对象权限，对象权限的管理实际上是对象所有者对其他用户操作该对象的权限管理。
- 在Oracle数据库中，不同类型的对象具有不同的对象权限，而有的对象并没有对象权限，只能通过系统权限进行管理，如簇、索引、触发器、数据库链接等。



## 对象权限授予

- 在Oracle数据库中，用户可以直接访问同名Schema下的数据库对象，如果需要访问其它Schema下的数据库对象，就需要具有相应的对象权限。授予对象权限的语法规则如下：
- GRANT object\_privilege|ALL
- ON schema.object
- TO user|role
- [WITH ADMIN OPTION]
- [WITH THE GRANT ANY OBJECT]

## 对象权限收回

- 使用REVOKE语句不仅可以收回系统权限，还可以收回对象权限。具体的语法规则如下：
- REVOKE object\_privilege|ALL
- ON schema.object
- FROM user|role
- [CASCADE CONSTRAINTS]
- [CASCADE CONSTRAINTS]选项表示该用户授予其他用户的权限也一并收回。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/267044121056006143>