

Gerenciamento em Banco de Dados

CRIAÇÃO de USUÁRIOS e PRIVILÉGIOS

Gerenciamento em Banco de Dados

Definições – USUÁRIO e PRIVILÉGIO

- **USUÁRIO (schema):** pessoa previamente cadastrada que deseja acessar o Oracle.
- **PRIVILÉGIO:** Autorização para que o usuário acesse e manipule um objeto de banco de dados. Exemplo: um usuário pode ter o privilégio de selecionar tabelas, porém não modificá-las. Existem dois tipos de privilégio:
 - ◆ **PRIVILÉGIOS DE SISTEMA:** permissão de executar determinada ação em objetos de banco de dados. Existem mais de cem tipos de privilégios associados a ações de banco de dados. O nome do privilégio é praticamente o nome da ação que ele executa.

Gerenciamento em Banco de Dados

Definições – USUÁRIO e PRIVILÉGIO

PRIVILÉGIO DE OBJETO: direito de executar uma determinada ação em um objeto específico, por exemplo, o direito de incluir uma linha em uma tabela. Eles não se aplicam a todos os objetos de banco de dados. Quando um usuário cria um objeto como uma tabela, esta só pode ser visualizada por ele. Para que outro usuário possa ter acesso a ela, é necessário que o proprietário conceda privilégios a ele ou um papel que irá acessar a tabela.

Gerenciamento em Banco de Dados

USUÁRIOS

- Os usuários podem ser autenticados no banco de dados através de três métodos diferentes:
 - ◆ Autenticação através do banco de dados.
 - ◆ Autenticação através do sistema operacional.
 - ◆ Autenticação através da rede.
- Quando um usuário cria um novo objeto no banco de dados (uma tabela, por exemplo), este fará parte de um schema (esquema) que tem o mesmo nome do usuário.
- Um usuário de banco de dados somente poderá ser criado pelo DBA ou por outro usuário com o privilégio de sistema `CREATE USER`. Além de informar o nome e a senha, é possível determinar, também, quais tablespaces estarão disponíveis e, até mesmo, quanto espaço de armazenamento o novo usuário poderá utilizar em cada tablespace.

Gerenciamento em Banco de Dados

CRIAÇÃO DE USUÁRIOS

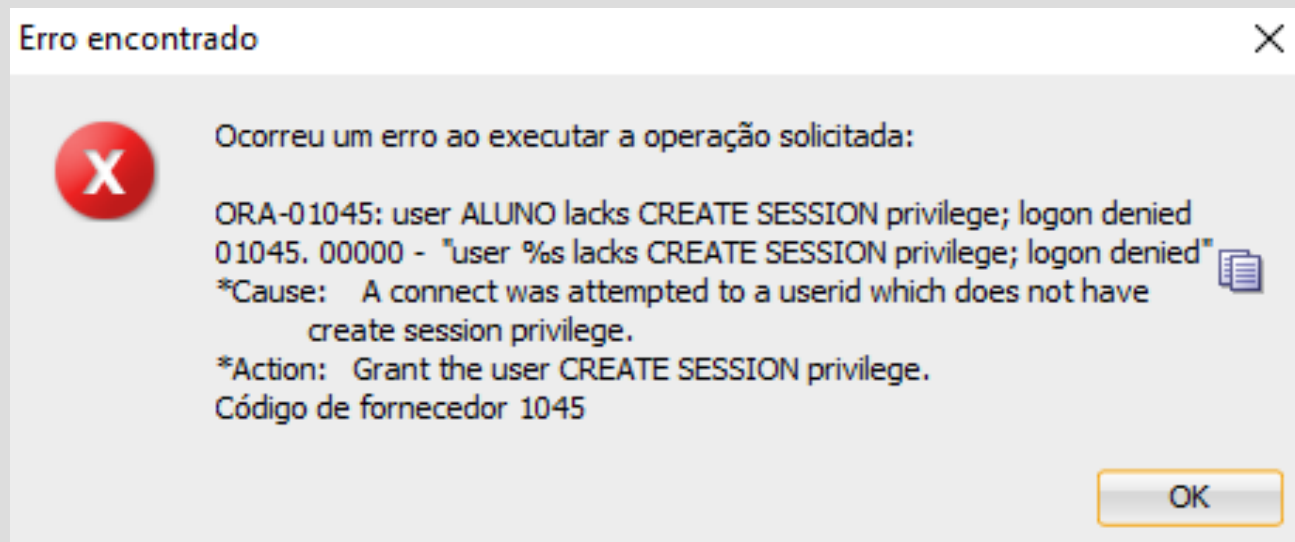
- Sintaxe

```
CREATE USER nome_usuario  
IDENTIFIED [BY senha | externally]  
DEFAULT TABLESPACE nome_tablespace  
TEMPORARY TABLESPACE nome_tablespace_temporaria  
QUOTA numero_inteiro [K | M] ON nome_tablespace  
PROFILE nome_profile  
PASSWORD EXPIRE  
ACCOUNT [LOCK | UNLOCK]
```

Gerenciamento em Banco de Dados

CRIAÇÃO DE USUÁRIOS

- Exemplos:
- Criação do usuário aluno com senha uninove.
 - ◆ **CREATE USER aluno IDENTIFIED BY uninove;**
- Conectando-se com usuário aluno recém-criado



Ocorre um erro, pois o usuário não possui privilégio CREATE SESSION atribuído a ele.

Gerenciamento em Banco de Dados

CRIAÇÃO DE USUÁRIOS

- **Exemplos:**
- **Criação do usuário aluno2 com senha uninove2, com senha expirada, conta liberada, nos tablespaces padrão de usuário e temporário.**

```
CREATE USER aluno2 IDENTIFIED BY uninove2  
ACCOUNT UNLOCK  
PASSWORD EXPIRE  
DEFAULT TABLESPACE USERS  
TEMPORARY TABLESPACE TEMP  
QUOTA UNLIMITED ON USERS;
```

Gerenciamento em Banco de Dados

CRIAÇÃO DE USUÁRIOS

OU

```
CREATE USER aluno2 IDENTIFIED BY uninove2  
DEFAULT TABLESPACE USERS  
PASSWORD EXPIRE  
QUOTA UNLIMITED ON USERS;
```

OBS: TABLESPACE DEFAULT padrão é o SYSTEM.

OBS 2: Note que se o comando QUOTA for omitido temos que o valor padrão é igual a zero, ou seja, o usuário em questão não terá permissão para CRIAR nenhum objeto no TABLESPACE DEFAULT.

Gerenciamento em Banco de Dados

CRIAÇÃO DE USUÁRIOS

- Ao logar via SQL Developer obtemos o erro abaixo:




- Para trocar (resetar) a senha via SQL Developer é necessário baixar o Oracle Instant Client e proceder como no link abaixo:
 - ◆ <http://phxcharger.blogspot.com.br/2014/03/reset-expired-password-using-sql.htm>

Gerenciamento em Banco de Dados

CRIAÇÃO DE USUÁRIOS

- Via SQL*PLUS a troca de senha é automática:

 Executar Linha de Comandos SQL

```
SQL*Plus: Release 11.2.0.2.0 Production on Qua Set 28 17:45:52 2016  
Copyright (c) 1982, 2010, Oracle. All rights reserved.
```

```
SQL> conn aluno2/uninove2  
ERROR:  
ORA-28001: the password has expired
```

```
Alterando senha para aluno2  
Nova senha:  
Redigite a nova senha:  
ERROR:  
ORA-01045: user ALUNO2 lacks CREATE SESSION privilege; logon denied
```

```
Senha alterada  
SQL> _
```

Gerenciamento em Banco de Dados

ALTERAÇÃO DE USUÁRIOS

- **ALTER USER**
- **Exemplos**
- **Limitando a quota para 100M no TABLESPACE USERS**
 - ◆ **ALTER USER aluno2 QUOTA 100M ON USERS;**
- **Sem limitação de quotas no TABLESPACE USERS**
 - ◆ **ALTER USER aluno2 QUOTA UNLIMITED on USERS;**
 - ◆ **Obs: Note que é necessário quota no tablespace DEFAULT para efetuar operações como criação de tabelas.**
- **Alterando o tablespace padrão (dados). Note que o tablespace TBSUNINOVE tem que existir para que o comando não apresente erros**
 - ◆ **ALTER USER aluno2 DEFAULT TABLESPACE TBSUNINOVE;**

Gerenciamento em Banco de Dados

ALTERAÇÃO DE USUÁRIOS

- **ALTER USER**
- **Exemplos**
- **Alterando a senha do usuário, sem expiração da mesma**
 - ◆ **ALTER USER aluno2 identified by uninove;**
- **Fazer com que a senha fique expirada**
 - ◆ **ALTER USER aluno2 password expire;**

Gerenciamento em Banco de Dados

ELIMINANDO USUÁRIOS

- **DROP USER nome_do_usuario [CASCADE];**
- **Caso a opção CASCADE seja informada eliminará não só o usuário como todos os objetos criados por ele.**
- **Caso a opção CASCADE seja omitida temos duas possibilidades:**
 - ◆ **Apagará o usuário pois o mesmo não é 'proprietário' de nenhum objeto, ou seja, não existe nenhum objeto que foi criado por ele;**
 - ◆ **Erro caso haja ao menos um objeto que foi criado por ele**

Gerenciamento em Banco de Dados

CONSULTANDO USUÁRIOS

- **SELECT USERNAME, ACCOUNT_STATUS FROM DBA_USERS;**
- **Observe, a seguir, os diferentes status que uma conta pode apresentar:**

OPEN: a conta está disponível para uso.

LOCKED: a conta foi bloqueada pelo DBA.

EXPIRED: a senha expirou. O usuário deverá redefini-la.

EXPIRED & LOCKED: a conta foi bloqueada e a senha expirou.

Gerenciamento em Banco de Dados

DBA – logando-se como outro usuário

- **SELECT NAME, PASSWORD FROM SYS.USER\$ WHERE NAME = 'ALUNO';**

	NAME	PASSWORD
1	ALUNO	AEEFA4FFE0CCD99A

- **Trocar a senha para algo temporário**
ALTER USER ALUNO IDENTIFIED BY SENHA_TEMP;
- **Efetua os testes necessários**
- **Volta a senha original do usuário em questão**
- **ALTER USER ALUNO IDENTIFIED BY VALUES 'AEEFA4FFE0CCD99A';**

Gerenciamento em Banco de Dados

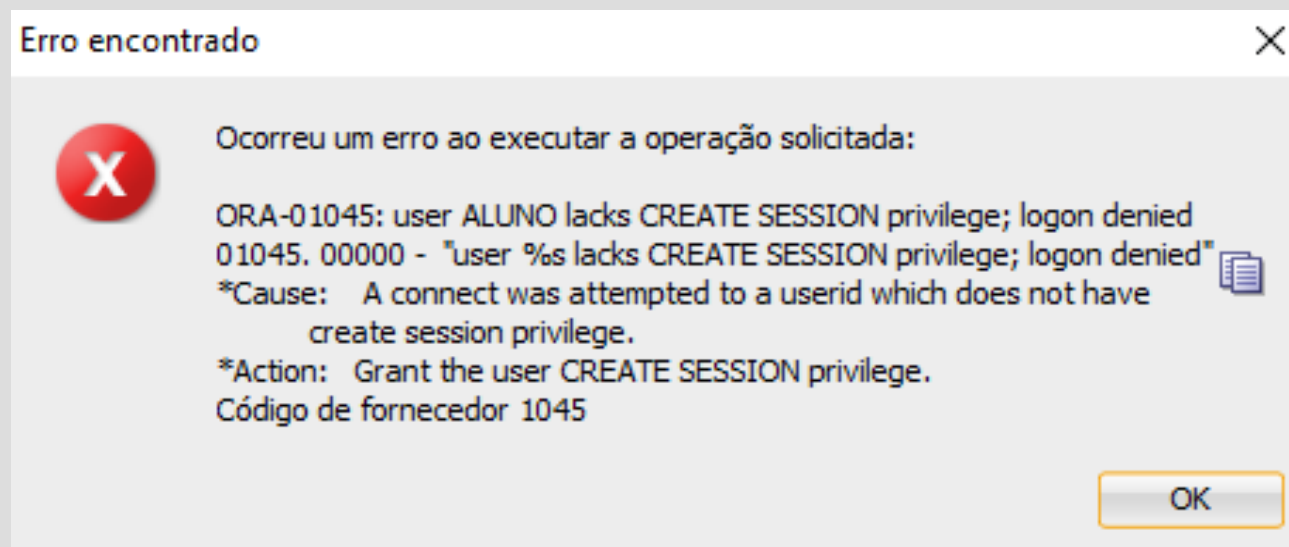
PRIVILÉGIOS DE SISTEMA

- Listar privilégios de sistema
 - ◆ **SELECT * FROM SYSTEM_PRIVILEGE_MAP;**
- Concedendo privilégios (GRANT) a um usuário
 - ◆ **GRANT CREATE TABLE, CREATE VIEW TO nome_usuario;**
GRANT CREATE TABLE, CREATE VIEW TO aluno;
- Concedendo privilégios (GRANT) a todos usuários
 - ◆ **GRANT CREATE TABLE, CREATE VIEW TO PUBLIC;**

Gerenciamento em Banco de Dados

PRIVILÉGIOS DE SISTEMA

- Note que nos exemplos anteriores criamos usuários porém os mesmos não conseguiam se logar ao Oracle pois obtinham o erro abaixo:



- Para resolver este problema devemos dar permissão para que o usuário em questão crie uma sessão
 - ◆ **GRANT CREATE SESSION TO aluno;**

Gerenciamento em Banco de Dados

PRIVILÉGIOS DE SISTEMA

- **Pode-se, também, conceder privilégios a determinado usuário e permitir que ele, por sua vez, também conceda para outros os mesmos privilégios que está recebendo. Para que isto seja possível, deve-se acrescentar a opção `WITH ADMIN OPTION` ao comando `GRANT`.**
 - ◆ **`GRANT CREATE TABLE TO ALUNO WITH ADMIN OPTION;`**
- **Se a permissão do usuário `ALUNO` para conceder seus privilégios a outros for revogada, os usuários aos quais ele concedeu os privilégios continuarão a retê-los (não serão revogados).**

Gerenciamento em Banco de Dados

PRIVILÉGIOS DE SISTEMA

- **Revogando Privilégios (REVOKE)**
 - ◆ **REVOKE CREATE VIEW FROM ALUNO;**
- **A tabela a seguir apresenta as visões de dicionário de dados relacionadas aos privilégios de sistema.**

Visão	Descrição
DBA_SYS_PRIVS	Privilégios de sistema atribuídos a usuários e papéis.
SESSION_PRIVS	Privilégios de sistema concedidos ao usuário na sessão atual diretamente ou através de um papel.
ROLE_SYS_PRIVS	Privilégios de sistema concedidos ao usuário na sessão atual através de um papel.

- **SELECT * FROM ROLE_SYS_PRIVS; -- USUÁRIO LOGADO**
- **SELECT * FROM SESSION_PRIVS; -- USUÁRIO LOGADO**
- **SELECT * FROM DBA_SYS_PRIVS WHERE GRANTEE = 'NOME_USUARIO';**

Gerenciamento em Banco de Dados

PRIVILÉGIOS DE OBJETOS

- **Privilégio de objeto é o direito de realizar um tipo específico de ação sobre determinados objetos de um banco de dados (por exemplo: tabelas, visões e procedures) que não fazem parte do esquema do usuário (que não foram criados pelo usuário).**
- **Utilizar GRANT (conceder priv) e REVOKE (revogar priv)**
- **Os privilégios de objetos podem ser concedidos a todos os usuários do banco de dados através do grupo especial PUBLIC.**
- **Pode-se conceder privilégios de objetos a determinado usuário e permitir que ele, por sua vez, também conceda-os para outros. Para que isto seja possível, deve-se acrescentar a opção WITH GRANT OPTION ao comando GRANT.**

Gerenciamento em Banco de Dados

PRIVILÉGIOS DE OBJETOS

- **GRANT SELECT ON nome_tabela TO nome_usuario [WITH GRANT OPTION];**
- **Exemplos:**
 - ◆ **GRANT SELECT ON tabTESTE TO PUBLIC;**
 - ◆ **GRANT SELECT ON tabTESTE TO ALUNO;**
 - ◆ **GRANT SELECT ON tabTESTE TO ALUNO WITH GRANT OPTION;**
- **Porém, diferente do que acontece quando se concedem permissões de sistema, se os privilégios de objetos do usuário forem revogados, serão também revogados os privilégios de todos os usuários da cadeia, isto é, que receberam seus privilégios através deste usuário (ALUNO, no exemplo acima)**

Gerenciamento em Banco de Dados

PRIVILÉGIOS DE OBJETOS

- **Lista de privilégios de objeto mais comuns**

Privilégio de objeto	Descrição
ALTER	Alterar uma definição de tabela.
INDEX	Criar um índice em uma tabela.
SELECT	Ler linhas em uma tabela ou visão.
INSERT	Inserir linhas em uma tabela ou visão.
UPDATE	Atualizar linhas em uma tabela ou visão.
DELETE	Excluir linhas de uma tabela ou visão.
EXECUTE	Executar uma procedure ou função.

Gerenciamento em Banco de Dados

PRIVILÉGIOS DE OBJETOS

- Quando necessário é possível ainda restringir acesso a colunas de uma tabela.

Exemplo:

Funcionário		
ID	Nome	Salário
1001	ANTONIO ALVARES	4500,00
1002	BEATRIZ BERNARDES	3800,00
1003	CLAUDIO CARDOSO	2900,00

- **GRANT UPDATE (NOME) ON FUNCIONARIO TO ALUNO;**

O usuário **ALUNO** terá permissão somente para alterar a coluna **NOME** da tabela **FUNCIONARIO**. As demais colunas não estarão disponíveis para alteração.

Gerenciamento em Banco de Dados

PRIVILÉGIOS DE OBJETOS

- Para revogar privilégios usar REVOKE
 - ◆ REVOKE UPDATE ON FUNCIONARIO FROM ALUNO;
 - ◆ REVOKE SELECT ON FUNCIONARIO FROM ALUNO;
- A tabela a seguir apresenta as visões de dicionário de dados relacionadas aos privilégios de objetos.

Visão	Descrição
DBA_TAB_PRIVS	Privilégios de tabelas concedidos a usuários e papéis.
DBA_COL_PRIVS	Privilégios de colunas concedidos ao usuário na sessão atual, diretamente ou através de um papel.
ROLE_TAB_PRIVS	Privilégios de tabelas concedidos ao usuário na sessão atual através de um papel.

- SELECT * FROM DBA_TAB_PRIVS WHERE GRANTEE = 'NOME_USUARIO';
- SELECT * FROM DBA_COL_PRIVS WHERE GRANTEE = 'NOME_USUARIO';
- SELECT * FROM ROLE_TAB_PRIVS; -- USUÁRIO LOGADO

Gerenciamento em Banco de Dados

Criação de tabelas por usuários diferentes

- Suponha que o usuário (schema) **ALUNO** tenha criado a tabela **TBL_CADASTRO** e que em seguida tenha dado permissão de consulta a esta tabela ao usuário **PROFESSOR**
- O usuário **PROFESSOR** deverá sempre se referir a tabela **TBL_CADASTRO** criada pelo usuário **ALUNO** como **ALUNO.TBL_CADASTRO**.
- Exemplo:
 - ◆ **SELECT * FROM ALUNO.TBL_CADASTRO;**
 - ◆ **DELETE FROM ALUNO.TBL_CADASTRO;**
- Note ainda que posso ter diferentes tabelas com o mesmo nome, desde que criados por usuários diferentes. Ex: **ALUNO.TBL_CADASTRO**, **PROFESSOR.TBL_CADASTRO**