

Aula 07 – User management

01 – Verificar qual usuário está conectado ao banco de dados

```
SQL> show user
```

02 – Listar todos os usuários do banco de dados

```
SQL> select username, account_status from dba_users;
```

03 – Criar um novo usuário no banco de dados

```
SQL> create user pedro identified by pedro123;
```

04 – Listar todos os usuários do banco para confirmar a criação do novo usuário

```
SQL> select username, account_status from dba_users;
```

05 – Abrir uma nova sessão no Linux e abrir o console SQL do Oracle (como usuário oracle)

```
$ sqlplus / as sysdba
```

```
SQL> connect pedro/pedro123
```

Receberemos a mensagem de que o usuário não pode se conectar ao banco de dados.

06 – Criar a permissão de criação de sessão para o usuário Pedro

```
SQL> grant create session to pedro;
```

07 – Listar os tablespace alocados para os usuários

```
SQL> select default_tablespace, temporary_tablespace from  
dba_users;
```

08 – Tentar listar alguma tabela de sistema

```
SQL> select * from tab;
```

Como podemos ver não tivemos nenhum resultado, pois o novo usuário não tem permissão para acessar os dados da tabela;

09 – Verificar qual default e temporary tablespaces estão associados ao usuário Pedro

```
SQL> select default_tablespace, temporary_tablespace from  
dba_users where username=' PEDRO' ;
```

Como podemos ver USER_DBLOJA e TEMP group são os tablespaces associados ao usuário Pedro.

10 – Vamos associar o tablespace TBS1 como sendo o default tablespace do Pedro

```
SQL> alter user pedro default tablespace TBS1;
```

11 – Vamos verificar se o default tablespace do pedro foi alterado

```
SQL> select default_tablespace, temporary_tablespace from  
dba_users where username=' PEDRO' ;
```

NOTA: Agora, os objetos de banco de dados que o pedro criar no banco de dados serão armazenados neste tablespace, mas será que ele consegue guardar dados no tablespace? A resposta é NÃO, pois precisamos determinar qual a quota de armazenamento o pedro tem direito.

12 – Definir a quota de armazenamento de dados para o pedro

```
SQL> alter user pedro quota 5m on TBS1;
```

13 – Consultar a quota total e disponível de armazenamento para o usuário pedro

```
SQL> select username, tablespace_name, bytes, max_bytes from  
dba_ts_quotas;
```

14 – Vamos criar o usuário ana, mas agora, queremos que a ana troque a sua senha no seu primeiro login.

```
SQL> create user ana identified by ana123 password expire;
```

```
SQL> grant create session to ana;
```

15 – Excluir o usuário ana

```
SQL> drop user ana;
```

NOTA: Quando apagamos um usuário, o Oracle apagará somente o usuário. Todos os objetos de banco de dados criados por este usuário não serão apagados. Caso seja necessário apagar também os objetos de banco de dados do usuário devemos acrescentar o parâmetro “**cascade**”. Por exemplo: **drop user ana cascade;**

Não é possível excluir um usuário que esteja conectado ao banco de dados, então efetue logoff deste usuário antes de excluí-lo.

16 – O usuário pedro esqueceu a senha e precisamos recuperá-la

```
SQL> alter user pedro identified by trocar123 password expire;
```

Podemos trocar a senha do usuário sem o “Password expire”, mas o ideal é permitirmos que o próprio usuário escolha a sua senha.

17 – O usuário pedro saiu de férias e queremos bloquear a sua conta

```
SQL> alter user pedro account lock;
```

18 – Desbloquear a conta do usuário

```
SQL> alter user pedro account unlock;
```

USER PRIVILEGES - SYSTEM

NOTA: Criar um usuário ana com default tablespace em TBS1 com quota de 5m.

01 – Vamos tentar criar uma tabela no banco de dados

```
SQL> create table cliente (id number, nome varchar2(100));
```

Receberemos uma mensagem de que não há privilégios suficientes para esta operação.

02 – É possível o próprio usuário consultar seus privilégios

```
SQL> select * from session_privs;
```

03 – Permitir que o usuário pedro possa criar tabelas (acessar sessão como SYS)

```
SQL> grant create table to pedro;
```

04 – Consultar novamente os privilégios do usuário

```
SQL> select * from session_privs;
```

05 – Tentar criar a tabela novamente

```
SQL> create table cliente (id number, nome varchar2(100));
```

Agora a tabela será criada.

06 – Consultar em qual tablespace a tabela foi criada

```
SQL> select tablespace_name, table_name from dba_tables where  
table_name = 'CLIENTE';
```

07 – Será que o pedro pode dar permissão de criação de tabela para a ana?

```
SQL> grant create table to ana;
```

Como podemos ver, não. Pedro não tem privilégios para dar permissão para outro usuário.

08 – Vamos dar privilégios administrativos para o pedro, para que ele possa dar permissões para outros usuários

```
SQL> grant create table to pedro with admin option;
```

09 – Agora pedro poderá dar permissão de criação de tabela para ana. Vamos ver?

```
SQL> grant create table to ana;
```

Agora a permissão foi criada para a ana. Não esquecer de definir o tablespace e a quota para a ana.

10 – Criar a tabela vendas com o usuário ana. Inserir um ou dois registros e comitar

```
SQL> create table vendas(id number);
```

```
SQL> insert into vendas values(1);
```

```
SQL> commit;
```

11 – Será que o usuário pedro tem acesso aos dados cadastrados na tabela vendas?

```
SQL> conn pedro;
```

```
SQL> select * from vendas;
```

Como podemos ver, o pedro não tem permissão de selecionar dados na tabela vendas, que pertence a ana.

12 – Como podemos permitir que o pedro tenha acesso a tabela vendas da ana?

```
SQL> grant select any table to pedro;
```

13 – Tentar selecionar os dados novamente como usuário pedro

```
SQL> conn pedro;
```

```
SQL> select * from vendas;
```

14 – Consultar os privilégios do pedro

```
SQL> select * from session_privs;
```

USER PRIVILEGES - OBJECT

01 – Criar uma tabela como usuário SYS

```
SQL> create table cliente(id number);
```

02 – Inserir pelo menos dois registros;

03 – Conectar como pedro e tentar listar o conteúdo da tabela clientes

```
SQL> select * from sys.cliente;
```

Estando como pedro, se não informarmos o sys.cliente o Oracle database buscará a tabela clientes no Schema do pedro.

Mesmo assim, receberemos uma mensagem de erro informando que a tabela não existe.

04 – Vamos permitir que o usuário pedro possa usar o comando SELECT na tabela cliente do usuário SYS. Conecte-se como SYS e rode o comando abaixo

```
SQL> grant select on sys.cliente to pedro;
```

05 – Acessar como usuário pedro e tentar rodar o select novamente

```
SQL> select * from sys.cliente;
```

06 – Agora conseguimos realizar a consulta

07 – Agora, o pedro deve excluir o registro de id 2

```
SQL> delete from sys.cliente where id=2;
```

Recebemos uma mensagem de erro informando que o pedro não tem privilégios suficientes para executar a operação de exclusão.

08 – Vamos, então, permitir que o pedro possa excluir registros da nossa tabela

```
SQL> grant delete on sys.cliente to pedro
```

09 – Agora, conectados como pedro, executamos o comando de exclusão novamente

```
SQL> delete from sys.cliente where id=2;
```

```
SQL> commit;
```

10 – Será que o pedro pode dar permissão de seleção e exclusão para a ana?

```
SQL> grant select, delete on sys.cliente to ana;
```

Recebemos a mensagem dizendo que o pedro não possui privilégios suficientes para isso.

11 – Vamos nos conectar como usuário SYS e permitir que o pedro também possa dar privilégios de select e delete para outros usuários, inclusive para a ana

```
SQL> grant select, delete on sys.cliente to pedro with grant option;
```

REVOKE PRIVILEGES - SYSTEM

01 – Vamos ver os privilégios do pedro

```
SQL> select * from session_privs;
```

02 – Como revogar o privilégio de criar tabela do pedro?

```
SQL> revoke create table from pedro;
```

03 – Consultar, novamente, os privilégios do pedro

```
SQL> select * from session_privs;
```

04 – Conectar-se como ana e consultar os privilégios da ana

```
SQL> conn ana
```

```
SQL> select * from session_privs;
```

REVOKE PRIVILEGES - OBJECT

01 – Conecte-se como SYS e retire os privilégios de delete e select do pedro na tabela cliente

```
SQL> revoke select, delete on sys.cliente from pedro;
```

02 – Conecte-se como pedro e tente listar a tabela cliente com o comando select

```
SQL> select * from sys.cliente;
```

03 – Agora, vamos testar se a ana também perdeu os privilégios. Conecte-se como ana e tente consultar a tabela cliente com o comando select.

```
SQL> conn ana;
```

```
SQL> select * from sys.cliente;
```

Como podemos ver, a ana também perdeu os privilégios de select e delete da tabela sys.cliente