

UFCD 0787- Criar Utilizadores

- Objetivo:**
- Compreender como criar utilizadores numa base de dados.
 - Aplicar comandos SQL para atribuir e retirar permissões.
 - Compreender a importância do controlo de acessos na segurança dos dados

Recorda

Comando	Função	Exemplo
CREATE USER	Cria um novo utilizador.	CREATE USER joao IDENTIFIED BY 'senha123';
GRANT	Concede permissões a um utilizador.	GRANT SELECT ON escola.alunos TO joao;
REVOKE	Retira permissões previamente concedidas.	REVOKE INSERT ON escola.alunos FROM joao;
DROP USER	Apaga o utilizador.	DROP USER joao;

Tipos comuns de permissões:

SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, ALL PRIVILEGES

Analisa a seguinte situação

1. **Base de dados "Escola"**, com as seguintes tabelas:
 - Alunos (id, nome, turma)
 - Professores (id, nome, disciplina)
 - Notas (id_aluno, id_professor, nota)
2. O **Administrador (root)** quer criar utilizadores com diferentes níveis de acesso:

Utilizador	Função	Permissões necessárias
professor1	Lançar e consultar notas	SELECT e INSERT na tabela Notas
secretaria	Consultar alunos e professores	SELECT nas tabelas Alunos e Professores
direcao	Acesso total a todas as tabelas	ALL PRIVILEGES

Exercícios Práticos

Nota: Podes testar estes comandos num SGBD como MySQL, MariaDB.

◆ Exercício 1 — Criar utilizadores

1. Cria os três utilizadores indicados acima, cada um com uma palavra-passe à tua escolha.
2. `CREATE USER professor1 IDENTIFIED BY '_____';`
3. `CREATE USER secretaria IDENTIFIED BY '_____';`
4. `CREATE USER direcao IDENTIFIED BY '_____';`

◆ Exercício 2 — Atribuir permissões

2. Atribui as permissões de acordo com a função de cada utilizador.
 - professor1 → SELECT e INSERT na tabela Notas
 - secretaria → SELECT em Alunos e Professores
 - direcao → todas as permissões sobre a base de dados Escola
3. `GRANT _____ ON Escola.Notas TO professor1;`
4. `GRANT _____ ON Escola.Alunos TO secretaria;`
5. `GRANT _____ ON Escola.Professores TO secretaria;`
6. `GRANT _____ ON Escola.* TO direcao;`

◆ Exercício 3 — Testar acessos

3. Liga-te à base de dados com o utilizador professor1 e tenta:
 - Inserir uma nota (deve funcionar)
 - Apagar uma nota (deve **falhar**)
 - Regista o resultado abaixo:
4. Inserção: _____
5. Eliminação: _____

♦ **Exercício 4 — Retirar permissões**

4. O administrador decide que secretaria já não pode consultar os professores.
5. Escreve o comando SQL para retirar essa permissão:
6. REVOKE _____ ON Escola.Professores FROM secretaria;

♦ **Exercício 5 — Apagar um utilizador**

5. O utilizador professor1 saiu da escola. Remove-o do sistema.
6. DROP USER _____;

4. Questões de Reflexão

1. Porque é importante criar utilizadores com diferentes permissões numa base de dados?

2. O que aconteceria se todos os utilizadores tivessem **ALL PRIVILEGES**?

3. Explica a diferença entre os comandos GRANT e REVOKE.