INSTITUTO DO EMPREGO
E FORMAÇÃO PROFISSIONAL

AEP

cesae digital

Centro para o Desenvolvimento
de Competências Digitais

PROGRAMAÇÃO SQL

Views e Stored Procedures

Helder Rodrigo Pinto

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- TCL – Transaction Control Language
- Information Schema
- DCL – Data Control Language
- Indexes
- Views
- Stored Procedures
- Triggers

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- TCL – Transaction Control Language
- Information Schema
- DCL – Data Control Language
- Indexes
- Views
- Stored Procedures
- Triggers

TRANSACTIONS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Transações são sequências de operações realizadas como uma única unidade lógica de trabalho.

- Se uma transação for bem-sucedida, todas as modificações de dados são confirmadas e tornam-se parte permanente da base de dados.
- Se a transação falha, todas as modificações devem ser desfeitas.

TRANSACTIONS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- COMMIT:
  - Aplica permanentemente todas as mudanças realizadas durante a transação atual
- ROLLBACK
  - Permite desfazer todas as alterações efetuadas desde o último COMMIT ou ROLLBACK.
- SAVEPOINT
  - Permite guardar temporariamente um ponto onde voltar, quando necessário, com recurso a um ROOLBACK.

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

```
INSERT INTO Departamento (idDepto, nome, idGerente)
VALUES(10, 'Marketing', 3);

UPDATE Funcionario
SET salario = salario * 1.05
WHERE idDepto = 5;

COMMIT;

DELETE FROM Funcionario;

ROLLBACK;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

MySQL

```
INSERT into Aluno values(5,'Raul');
commit;
UPDATE Aluno set name='Marco' where id='5';
savepoint A;
INSERT into Aluno values(6,'Tiago');
savepoint B;
INSERT into Aluno values(7,'Bruno');
savepoint C;
SELECT * from Aluno;
rollback to B;
SELECT * from Aluno;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

SQLServer

```
INSERT into Aluno values(5,'Raul');
commit;
UPDATE Aluno set name='Marco' where id='5';
SAVE TRANSACTION A;
INSERT into Aluno values(6,'Tiago');
SAVE TRANSACTION B;
INSERT into Aluno values(7,'Bruno');
SAVE TRANSACTION C;
SELECT * from Aluno;
rollback TRANSACTION B;
SELECT * from Aluno;
```

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- TCL – Transaction Control Language
- Information Schema
- DCL – Data Control Language
- Indexes
- Views
- Stored Procedures
- Triggers

INFORMATION SCHEMA

cesae digital
Centro para o Desenvolvimento de Competências Digitais

```
SELECT *
FROM INFORMATION_SCHEMA.TABLES;

SELECT *
FROM INFORMATION_SCHEMA.TABLES
where table_type = 'BASE TABLE';
```

INFORMATION SCHEMA

cesae digital
Centro para o Desenvolvimento de Competências Digitais

```
SELECT *
FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS
```

```
SELECT *
FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS
WHERE TABLE_NAME = 'Artigo'
```

INFORMATION SCHEMA

cesae digital
Centro para o Desenvolvimento de Competências Digitais

```
SELECT TABLE_SCHEMA, TABLE_NAME, COLUMN_NAME,
CONSTRAINT_NAME
FROM INFORMATION_SCHEMA.KEY_COLUMN_USAGE
WHERE REFERENCED_TABLE_SCHEMA IS NOT NULL;
```

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- TCL – Transaction Control Language
- Information Schema
- DCL – Data Control Language
- Indexes
- Views
- Stored Procedures
- Triggers

DCL – Data Control Language

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Usado para definir o controlo de acesso aos dados
    - **Controlo de Acesso**: permite controlar o acesso aos dados numa base de dados.
    - **Segurança de Dados**: Permite gerir permissões, garantindo que apenas utilizadores autorizados possam aceder ou modificaros os dados.

DCL – Data Control Language

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- **GRANT**
  - usado para conceder permissões a utilizadofes ou grupos.
  - `GRANT SELECT ON table TO user;`

- **REVOKE**
  - Usado para retirar permissões previamente concedidas.
  - `REVOKE SELECT ON table FROM user;`

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- TCL – Transaction Control Language
- Information Schema
- DCL – Data Control Language
- Indexes
- Views
- Stored Procedures
- Triggers

Indexes

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Usados para acelerar e otimizar consultas para encontrar rapidamente a informação necessária.
- Funcionam como índices em livros.
- No entanto, diminuem a performance em operações de manipulação de dados.

Indexes

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Criar um índice:

```
CREATE INDEX idx_nome ON Clientes (nome);
```

- Criar um índice para garantir que não haja duplicados:

```
CREATE UNIQUE INDEX idx_email_unico ON Clientes
(email);
```

- Criar um índice expressão:

```
CREATE INDEX idx_nome_lower ON Clientes
(LOWER(nome));
```

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- TCL – Transaction Control Language
- Information Schema
- DCL – Data Control Language
- Indexes
- Views
- Stored Procedures
- Triggers

VIEW

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- É uma tabela virtual baseada no conjunto de resultados de uma instrução SQL.

- Contém linhas e colunas de uma ou mais tabelas mas apresenta os dados como se viessem de uma única tabela.

VIEW

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- ESTRUTURA

```
CREATE VIEW nome AS
SELECT coluna1, coluna2, ...
FROM tabela
WHERE condição;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Fazer pesquisas à Base de Dados

```
CREATE VIEW Lista_Actual_Clientes AS
SELECT nome, telef
FROM Cliente
WHERE estado='activo'
```

VIEW

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Executar uma VIEW

```
SELECT *
FROM  Lista_Actual_Clientes;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Fazer pesquisas à Base de Dados

```
CREATE VIEW Total_Produtos_Encomendados_Por_Produto AS
select p.descricao, sum(l.quantidade) AS Total
from LinhaEncomenda l, Produto p
where l.codProduto=p.codProduto
group by p.descricao
```

VIEW

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Atualizar uma VIEW

MySQL

```
CREATE OR REPLACE VIEW  Lista_Actual_Produtos AS
SELECT descrição, preco
FROM Produto
WHERE estado='1'
```

VIEW

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Apagar uma VIEW

SQLServer

```
IF OBJECT_ID ('Lista_Actual_Produtos', 'V')
IS NOT NULL
DROP VIEW Lista_Actual_Produtos GO
```

*V – View

*U – Tabela

Entre outras...

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- TCL – Transaction Control Language
- Information Schema
- DCL – Data Control Language
- Indexes
- Views
- Stored Procedures
- Triggers

STORED PROCEDURE

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- É uma coleção de comandos em SQL
- Agrupa tarefas repetitivas, aceita parâmetros de entrada e retorna um valor de saída.
- Pode reduzir o tráfego na rede, visto que os comandos são executados diretamente no servidor e cria mecanismos de segurança entre a manipulação dos dados.

STORED PROCEDURE

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Estrutura e SQL Server

SQLServer

```
CREATE PROCEDURE nome(
@variável/eis tipo_de_dados )
AS
   comando
GO
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Mostra o nº de habitantes de um dado Continente

SQLServer

```
CREATE PROCEDURE numHabitContinente(
@in_continente VARCHAR(20) )
AS
   SELECT nome, nhab FROM continente
   WHERE nome = @in_continente;
GO
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Executar Stored Procedure numHabitContinente em SQL Server

SQLServer

```
USE bd;
GO
EXEC numHabitContinente
      @in_continente = 'Europa';
GO
```

STORED PROCEDURE

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Estrutura em MySQL

MySQL

```
DELIMITER //
CREATE PROCEDURE nome
(IN  variável/eis tipo_de_dados)
BEGIN
      comando
END //
DELIMITER ;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Mostra o nº de habitantes de um dado Continente

MySQL

```
DELIMITER //
CREATE PROCEDURE numHabitContinente
(IN in_continente VARCHAR(20) )
BEGIN
  SELECT nome, nhab FROM continente
  WHERE nome = in_continente;
END //
DELIMITER ;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Executar Stored Procedure numHabitContinente em MySQL

MySQL

```
CALL numHabitContinente('Europa');
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Permite inserir um registo de um novo cliente

```
DELIMITER //
CREATE PROCEDURE insereCliente
(IN in_nomeCliente VARCHAR(200),
 IN in_telef INT,
 IN in_email VARCHAR(200) )
BEGIN
        INSERT INTO Cliente (nome, telef, mail)
        VALUES (in_nomeCliente, in_telef, in_email);
END //
DELIMITER ;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Executar Stored Procedure insereCliente

```
CALL insereCliente('Helder', '912345678', 'helder@mail.pt');
CALL insereCliente(‘André’, '912345678', ‘andre@mail.pt');
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Pesquisa os dados de um dado Cliente (pelo nome)

```
DELIMITER //
CREATE PROCEDURE pesqFacturasDeClientePorTelef
(IN in_telef VARCHAR(20) )
BEGIN
  SELECT c.nome, f.data FROM Cliente c, Factura f
  WHERE c.id=f.cliente and c.telef = in_telef;
  ORDER BY f.data
END //
DELIMITER ;
```

EXEMPLO

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Executar Stored Procedure pesqClientesPorNome

```
CALL pesqFacturasDeClientePorTelef('912345678');
```

Variáveis

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Insere um determinado produto com o respetivo preço atual (assumindo que guardamos o histórico de preços)

```
create procedure InsereProduto_Preco(
      @nome varchar(50),
      @preco decimal(10,2) ) as
begin
      insert into produtos values (@nome)
      Declare @idProd int --criar a variável
      set @idProd = (select top(1)id from produtos order by id
desc)
      insert into precos values (@idProd, @preco, GETDATE())
end
```

Variáveis

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Executar Stored Procedure InsereProduto_Preco

```
Exec InsereProduto_Preco @nome='prego',
@preco='0.3';
```

Listas

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Insere vários produtos e respetiva quantidade numa venda

```
CREATE TYPE TipoItemVenda AS TABLE(ProdutoID INT, Quantidade INT);

create PROCEDURE InserirVenda (@Itens TipoItemVenda READONLY) AS
BEGIN
	DECLARE @venda INT;
	INSERT INTO venda VALUES (GETDATE());
	set @venda = (select top(1) id from venda order by id desc);

	INSERT INTO linhaVenda (venda, produto, qtd)
	SELECT @venda, ProdutoID, Quantidade FROM @Itens;
END;
```

Listas

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Cria a lista de produtos e quantidade

```
DECLARE @ItensVenda TipoItemVenda;

INSERT INTO @ItensVenda (ProdutoID, Quantidade) VALUES (1, 1);
INSERT INTO @ItensVenda (ProdutoID, Quantidade) VALUES (2, 5);
```

- Executar Stored Procedure InserirVenda

```
EXEC InserirVenda @Itens = @ItensVenda;
```

NOTAS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- `CONCAT('%', _name , '%');`
- `CURRENT_DATE();`
- `YEAR("2020-01-01");`

OBJECTIVOS

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- TCL – Transaction Control Language
- Information Schema
- DCL – Data Control Language
- Indexes
- Views
- Stored Procedures
- Triggers

Triggers

cesae digital
Centro para o Desenvolvimento de Competências Digitais

- Recurso de programação SQL executado sempre que o evento associado ocorrer.

Triggers

Atualiza Stock ao vender

cesae digital
Centro para o Desenvolvimento de Competências Digitais

```
CREATE TRIGGER tgr_inserir_venda
ON venda
FOR INSERT
AS
BEGIN
    DECLARE @quant  int, @produto int
    SELECT @quant = qtd, @produto = prod FROM INSERTED

    UPDATE produto SET stock = stock - @quant
    WHERE id = @produto
END
GO
```

Triggers

Atualiza Stock ao vender

cesae digital
Centro para o Desenvolvimento de Competências Digitais

(mais eficiente)

```
CREATE TRIGGER tgr_inserir_venda
ON venda FOR INSERT
AS
BEGIN
    UPDATE produto SET stock = stock - i.quant FROM
inserted i
    WHERE id = i.produto
END
GO
```

Triggers

Previne que a data da venda seja superior à data atual

cesae digital
Centro para o Desenvolvimento de Competências Digitais

```
CREATE TRIGGER VerificaDataVenda ON venda INSTEAD OF INSERT AS
BEGIN
	IF (SELECT data_venda FROM inserted) > GETDATE()
	BEGIN
		RAISERROR ('A data da venda não pode ser superior à data atual.', 16,
1);
	END
	ELSE
	BEGIN
		INSERT INTO venda SELECT data_venda FROM inserted;
	END
END
```

Triggers

previne o registo do mesmo produto numa venda e verifica se há stock antes de registar uma venda

cesae digital
Centro para o Desenvolvimento de Competências Digitais

```
create TRIGGER VerificaStock_e_Previne_ProdutoDuplicado ON linhaVenda INSTEAD OF INSERT AS
BEGIN
	DECLARE @qtd int, @stock int, @venda INT, @produto INT, @ErrorMsg VARCHAR(MAX) = 'Não há stock
suficiente para o produto: ';
	SELECT @qtd = i.qtd, @stock = s.stock, @venda = i.venda, @produto = i.produto
	FROM inserted i INNER JOIN stock s ON i.produto = s.produto
	where s.produto = i.produto
	if (@stock<@qtd)
	begin
		RAISERROR (@ErrorMsg, 16, 1);
		SELECT @ErrorMsg = @ErrorMsg + CAST(produto AS VARCHAR(MAX)) from inserted
		INSERT INTO notificacoes VALUES (@ErrorMsg);
		RETURN;
	END
```

Triggers

previne o registo do mesmo produto numa venda e verifica se há stock antes de registar uma venda

cesae digital
Centro para o Desenvolvimento de Competências Digitais

```
        else
        begin
                IF EXISTS (SELECT * FROM linhaVenda WHERE venda = @venda AND produto = @produto)
                BEGIN
                        RAISERROR ('Um produto não pode ser inserido mais de uma vez na mesma venda.', 16,
1);
                        RETURN;
                END
                ELSE
                BEGIN
                        INSERT INTO linhaVenda
                        SELECT venda, produto, qtd FROM inserted;
                END
        end
END
```

cesae digital

Centro para o Desenvolvimento
de Competências Digitais

INSTITUTO DO EMPREGO
E FORMAÇÃO PROFISSIONAL

AEP

www.cesae.pt