# Store Procedure

## Variáveis

SELECT \* FROM e1

DECLARE @T NVARCHAR

SELECT @T=C1 FROM e1 WHERE C2='b'

SELECT @T

SET @T = (SELECT C1 FROM e1 WHERE C2='b')

SELECT @T

### Variáveis tabela

DECLARE @t TABLE (

SPID INT

,Status VARCHAR(1000) NULL

,Login SYSNAME NULL

,HostName SYSNAME NULL

,BlkBy SYSNAME NULL

,DBName SYSNAME NULL

,Command VARCHAR(1000) NULL

,CPUTime INT NULL

,DiskIO INT NULL

,LastBatch VARCHAR(100) NULL

,ProgramName VARCHAR(1000) NULL

,SPID2 INT

,REQUESTID INT

);

INSERT INTO @t EXEC sp\_who2

SELECT \* FROM @t

## Controle de fluxo

### Bloco = BEGIN .... END

### IF

IF *Condição* *Comando*

ELSE é do IF mais próximo

IF 1=2 IF 2=3 PRINT '2=3' ELSE PRINT '1=2'

### WHILE

WHILE boolean\_expression

{ sql\_statement | statement\_block | BREAK | CONTINUE }

DECLARE @x INT = 0

WHILE @x < 10 BEGIN

PRINT @x

SET @x += 1

END

DECLARE @x INT = 0

WHILE 1=1 BEGIN

PRINT @x

SET @x += 1

IF @x>10 BREAK

END

## Cursores

DECLARE @x NVARCHAR

DECLARE c CURSOR FOR SELECT c1 FROM e1

OPEN c

WHILE 1 = 1 BEGIN

FETCH NEXT FROM c INTO @x

IF @@FETCH\_STATUS <> 0

BREAK

PRINT @x

END

CLOSE c

DEALLOCATE c

## Tratamento de Erros

BEGIN TRY

{ sql\_statement | statement\_block }

END TRY

BEGIN CATCH

[ { sql\_statement | statement\_block } ]

END CATCH

-- deve ter ; no comando anterior

;THROW 51000, 'Mensagem de erro', 11

erro deve ser > 50000

RAISERROR (N'Done with reputation changes at %s', 0, 1, @Now) WITH NOWAIT

Diferença entre RAISERROR e TROW:

<https://learn.microsoft.com/pt-br/sql/t-sql/language-elements/throw-transact-sql?view=sql-server-ver16&f1url=%3FappId%3DDev15IDEF1%26l%3DPT-BR%26k%3Dk(THROW_TSQL)%3Bk(sql13.swb.tsqlresults.f1)%3Bk(sql13.swb.tsqlquery.f1)%3Bk(MiscellaneousFilesProject)%3Bk(DevLang-TSQL)%26rd%3Dtrue#differences-between-raiserror-and-throw>

PRINT <-> RAISERROR:

<https://learn.microsoft.com/pt-br/sql/t-sql/language-elements/print-transact-sql?view=sql-server-ver16&f1url=%3FappId%3DDev15IDEF1%26l%3DPT-BR%26k%3Dk(PRINT_TSQL)%3Bk(sql13.swb.tsqlresults.f1)%3Bk(sql13.swb.tsqlquery.f1)%3Bk(MiscellaneousFilesProject)%3Bk(DevLang-TSQL)%26rd%3Dtrue#remarks>

### severidade:

<https://learn.microsoft.com/pt-br/sql/relational-databases/errors-events/database-engine-error-severities?view=sql-server-ver16>

BEGIN TRY

-- Generate a divide-by-zero error.

SELECT 1/0;

END TRY

BEGIN CATCH

SELECT

ERROR\_NUMBER() AS ErrorNumber,

ERROR\_SEVERITY() AS ErrorSeverity,

ERROR\_STATE() AS ErrorState,

ERROR\_PROCEDURE() AS ErrorProcedure,

ERROR\_LINE() AS ErrorLine,

ERROR\_MESSAGE() AS ErrorMessage;

END CATCH;

GO

## Documentação

-- =============================================

-- Author: <Author,,Name>

-- Create date: <Create Date, ,>

-- Description: <Description, ,>

-- =============================================

-- =============================================

-- Author: Marcos Paulo Ferreira Rebello

-- Create date: 2019-10-22

-- Description: Recria a tabela Z\_IPREJUN\_Estat\_Beneficio\_t (usada para localizar benefícios)

-- 2023-06-12 - Data de expedição do ato

-- 2023-05-09 - Data de publicação do ato

-- 2022-10-20 - Incluído N\_TIPENCBEN

-- 2022-10-19 - Incluído Se\_Complementacao

-- 2021-11-24 - Consulta de benefícios refeita, limitada a aposentadorias e pensões, e campos alterados

-- 2021-11-29 - Alteração para incluir quebras de cotas

-- 2022-05-17 - Alteração para incluir descrição do cargo

-- =============================================