SQL

Neste capítulo se encontram os padrões de código adotados para SQL e alguns exemplos ilustrativos, caso mesmo com os exemplos descritos, ainda restem dúvidas, não hesite em perguntar

- Considerações Iniciais
- Exemplo: CASE
- Exemplo: DELETE
- Exemplo: Modificador DISTINCT
- Exemplo: UPDATE
- Palavras Reservadas
- Dúvidas Frequentes
- Exemplo: INSERT
- Exemplo: FIRST 1

Considerações Iniciais

Introdução

Para rotinas em SQL, deve-se observar as seguintes considerações:

1. **Palavras reservadas** devem ser escritas **integralmente em letras maiúsculas**.
2. Nomes de **tabelas e campos** devem utilizar a notação **PascalCase**, mantendo **siglas em letras maiúsculas**.
3. Os **aliases de tabelas** devem ser escritos **integralmente em letras maiúsculas** e utilizar abreviações **curtas e significativas** da tabela de origem.
4. **Parâmetros** devem possuir o **prefixo "P"** e utilizar a notação **PascalCase**.
5. Para facilitar a implementação e a leitura das cláusulas, as **palavras reservadas devem ser alinhadas de forma que exista uma coluna contínua separando-as dos campos e tabelas**, conforme o exemplo abaixo:

```
    SELECT SUM(NF.ValorNF) AS ValorNF,
           CTE.CodigoCTE
      FROM NotaFiscal AS NF
INNER JOIN ConhecimentoCTE AS CTE
        ON CTE.CodigoNF = NF.CodigoNF
     WHERE NF.EmpresaID = :PNotaFiscalEmpresaID
       AND CTE.EmpresaID = :PConhecimentoCTEEmpresaID
  GROUP BY CTE.CodigoCTE
 UNION ALL
    SELECT (SELECT CodigoXML
              FROM ArquivoXML) AS CodigoXML,
           RELCTE.CodigoCTE
      FROM DocumentoXML AS DOCXML
INNER JOIN (SELECT CodigoCTE
              FROM RelacaoCTE) AS RELCTE
        ON RELCTE.CodigoCTE = DOCXML.CodigoCTE
     WHERE DOCXML.EmpresaID = :PDocumentoXMLEmpresaID
  ORDER BY 1
```

Perceba que, na cláusula do exemplo acima, as linhas 4 e 14 possuem as **maiores palavras reservadas/expressões**. Dessa forma, **as demais palavras reservadas recebem a adição de espaços em branco à esquerda**, para que **todas as expressões terminem na mesma coluna**. Assim, forma-se uma **coluna contínua ao longo de toda a cláusula**, separando as

palavras reservadas à esquerda dos campos e tabelas à direita.

Exemplo: CASE

Sem expressão no CASE

```
SELECT T1.Campo1, T1.Campo2
  FROM Tabela1 T1
 WHERE T1.Campo2 = :PInformacao1
   AND T1.Campo3 = :PInformacao2
   AND CASE
         WHEN
           T1.Campo4 IS NOT NULL
         THEN
           T1.Campo4 = :PInformacao3
         ELSE
           T1.Campo5 = :PInformacao3
       END
```

Com expressão no CASE

```
SELECT T1.Campo1, T1.Campo2
  FROM Tabela1 T1
 WHERE T1.Campo2 = :PInformacao1
   AND T1.Campo3 = :PInformacao2
   AND CASE COALESCE(T1.Campo4, "")
         WHEN
           ""
         THEN
           T1.Campo5 ||
           SUBSTRING(T1.Campo6 FROM 1 FOR 1) ||
           SUBSTRING(T1.Campo6 FROM 3 FOR 5)
         ELSE
           T1.Campo7
       END ApelidoDoCampo
```

Com CASE no SELECT

```
SELECT
  CASE
    WHEN
      EXISTS (SELECT 1
        FROM Tabela1
          WHERE Campo1 = :PCampo1
            AND Campo2 = :PCampo2)
      AND EXISTS (SELECT 1
                  FROM Tabela1
                 WHERE Campo1 = :PCampo1
                   AND Campo2 = :PCampo2)
    THEN
    "True"
    ELSE
    "False"
  END AS Resultado
  FROM RDB$DATABASE;
```

Exemplo: DELETE

```
DELETE
  FROM TabelaExemplo
 WHERE Abrev = :PAbrev
   AND Tipo = :PTipo
   AND Categoria = :PCategoria
```

> As comparações realizadas nas linhas 3-5 poderiam ter seus operadores alinhados na mesma coluna, entretanto essa não é uma prática recomendada para cláusulas mais extensas.

Exemplo: Modificador DISTINCT

```
SELECT DISTINCT T.Campo1, T.Campo2, T.Campo3
                T.Campo4, T.Campo5, T.Campo6
           FROM Tabela T
     INNER JOIN OutraTabela O
             ON O.Campo1 = T.Campo7
          WHERE T.Campo1 = :Param1
            AND T.Campo8 = :Param2
```

Exemplo: UPDATE

```
UPDATE TabelaExemplo
  SET Numero = :PNumero,
      Nome = :PNome
WHERE Categoria = :PCategoria
  AND Tipo = :PTipo
```

> As comparações realizadas nas linhas 2-5 poderiam ter seus operadores alinhados na mesma coluna, entretanto essa não é uma prática recomendada para cláusulas mais extensas

Palavras Reservadas

As palavras reservadas em SQL devem ser escritas com todas as letras maiúsculas:

ACTIVE	DOUBLE	NUMERIC
ADD	DROP	OF
AND	EDIT	ON
ALL	ELSE	ONLY
ALTER	END	OPEN
ANY	EXECUTE	OR
AS	EXISTS	ORDER
ASC	FILTER	OUTER
ASCENDING	FIRST	PERCENT
AT	FLOAT	PLAN
AVG	FOR	POSITION
BEFORE	FOREIGN	PRECISION
BEGIN	FROM	PREPARE
BETWEEN	FULL	PRIMARY
BLOB	FUNCTION	PROCEDURE
BY	GENERATOR	PUBLIC
CASE	GEN_ID	REAL
CAST	GROUP	RIGHT
CHAR	HAVING	ROWS
COALESCE	HOUR	SELECT
COLLATE	IF	SET

COLUMN	IN	SIZE
COMMIT	INACTIVE	SOME
CONTINUE	INDEX	SQL
COUNT	INNER	SUM
CREATE	INSERT	TABLE
CURRENT	INTEGER	THEN
CURRENT_DATE	INTO	TRIGGER
CURRENT_ROLE	IS	TRUNCATE
CURRENT_TIME	JOIN	TYPE
CURRENT_TIMESTAMP	KEY	UNION
CURRENT_TRANSACTION	LAST	UPDATE
CURRENT_USER	LEFT	UPDATING
CURSOR	LENGTH	USE
DATABASE	LIKE	USER
DATE	LONG	USING
DAY	MAX	VALUE
DEC	MERGE	VALUES
DECIMAL	MIN	VIEW
DECLARE	MINUTE	WHEN
DEFAULT	MONTH	WHERE
DELETE	NAMES	WHILE
DESC	NO	WITH
DESCENDING	NOT	YEAR
DISTINCT	NULL	YEARDAY
DO	NULLIF	

Dúvidas Frequentes

Espaço dedicado para a inclusão de dúvidas comuns durante o desenvolvimento em SQL. Caso sua dúvida não esteja aqui, talvez seja interessante adicioná-la.

Quando utilizar parâmetros, QuotedStr ou aspas duplas?

Parâmetros:

Utilizar principalmente para as condições da cláusula

```
SELECT *
  FROM Tabela
 WHERE Campo1 = :PParametro1
   AND Campo2 = :PParametro2
   AND Campo3 = :PParametro3
```

QuotedStr:

Utilizar quando for preciso concatenar uma variável string na consulta

```
SELECT Campo1, Campo2, Campo3
  CASE
    WHEN
('      Campo4 = QuotedStr(LVariavel)');
    THEN
      Campo4
    ELSE
      Campo5
  END ApelidoDoCampo
  FROM Tabela
 WHERE Campo1 IS NOT NULL
```

Aspas duplas:

Utilizar quando uma string fixa for inserida na cláusula

```
SELECT Campo1, Campo2, Campo3
  FROM Tabela
```

```
WHERE Campo1 = "NFE"
```

A utilização da palavra reservada "AS" é obrigatória para apelidar campos ou tabelas?

Qual a diferença entre a utilização de um UNION e UNION ALL?

Qual JOIN devo utilizar na cláusula?

Exemplo: INSERT

INSERT com SELECT

```
INSERT INTO Tabela(Abrev, Nome, Numero)
    SELECT Abrev, Nome, Numero
      FROM TabelaExemplo
  GROUP BY 1, 2, 3
```

INSERT

```
INSERT INTO Tabela(Abrev, Nome, Numero)
    VALUES (:PAbrev, :PNome, :PNumero)
```

Exemplo: FIRST 1

```
SELECT FIRST 1 P.NumPedCli,
              IP.NItemPED
         FROM NotaPed AS N
   INNER JOIN Pedidos AS P
           ON P.Abrev = N.Abrev
          AND P.Modelo = N.ModeloPed
          AND P.NumPedido = N.NumPedido
   INNER JOIN ItensPe AS IP
           ON IP.Abrev = P.Abrev
          AND IP.Modelo = P.Modelo
          AND IP.NumPedido = P.NumPedido
          AND IP.Item = N.ItemPedido
        WHERE N.Abrev = :PAbrev
          AND N.ModeloNF = :PModeloNF
          AND N.NumLancto = :PNumLancto
          AND N.ItemNF = :PItemNF
```