

CORES E TAMANHOS

MSS V5.0

Primavera

Rev.1 2019-09-26

SYSDEV MSS

MOBILE SALES SYSTEM

www.sysdevmss.com

Índice

RESUMO	2
MSART	3
SQL 1	3
SQL 3	3
MSGRH	4
MSGRV	8

RESUMO

A configuração no driver MSS, para utilizar os artigos com cores e tamanhos do Primavera, obriga a alterações de mapeamentos de algumas tabelas já em utilização, assim como obriga a mapear as tabelas MSGRH e MSGRV.

Na tabela de artigos, vai ser necessário efetuar alterações nos mapeamentos já existentes, e vai ser necessário um novo mapeamento.

Assim, o primeiro mapeamento servirá para ir buscar todos os artigos incluindo o artigo pai dos artigos com cores e tamanhos.

O terceiro mapeamento vai buscar todos os artigos, com a cor e tamanho correspondente.

MSART

SQL 1

Campo	Descrição	Tabela ERP	Campo ERP
ARTTRD	Tipo de recolha de dados	%OPTIONS%	Campo a Validar: Artigo.TratamentoDim SE=1 ENTAO 'G' SENAO 'S'
ARTTGR	Tem grelha	%OPTIONS%	Campo a Validar: Artigo.TratamentoDim SE=1 ENTAO 'S' SENAO 'N'

Filtro
Artigo.ArtigoAnulado = 0 and Artigo.TratamentoDim<>2

SQL 3

Igual ao SQL 1, só alterando os seguintes dados:

Mapeamentos

Campo	Descrição	Tabela ERP	Campo ERP
ARTTRD	Tipo de recolha de dados	%DEFAULT%	'F'
ARTTGR	Tem grelha	%DEFAULT%	'N'

Filtro
Artigo.ArtigoAnulado = 0 and ISNULL(Artigo.ArtigoPai, '') <> '' and ISNULL(Artigo.RubDim1, '') <> '' and ISNULL(Artigo.RubDim2, '') <> ''

MSGRH

SQL em modo Avançado

SQL

```

DECLARE @DESTAM VARCHAR(25)
DECLARE @DESHOR VARCHAR(25)
DECLARE @DESVAR VARCHAR(25)
DECLARE @MSGRH TABLE (
    REF VARCHAR(60) not null,
    DESCR VARCHAR(100) NOT NULL,
    DESCRV VARCHAR(100) NOT NULL,
    CC1 VARCHAR (30) NOT NULL,
    DC1 VARCHAR (100) NOT NULL,
    CC2 VARCHAR (30) NOT NULL,
    DC2 VARCHAR (100) NOT NULL,
    CC3 VARCHAR (30) NOT NULL,
    DC3 VARCHAR (100) NOT NULL,
    CC4 VARCHAR (30) NOT NULL,
    DC4 VARCHAR (100) NOT NULL,
    CC5 VARCHAR (30) NOT NULL,
    DC5 VARCHAR (100) NOT NULL,
    CC6 VARCHAR (30) NOT NULL,
    DC6 VARCHAR (100) NOT NULL,
    CC7 VARCHAR (30) NOT NULL,
    DC7 VARCHAR (100) NOT NULL,
    CC8 VARCHAR (30) NOT NULL,
    DC8 VARCHAR (100) NOT NULL,
    CC9 VARCHAR (30) NOT NULL,
    DC9 VARCHAR (100) NOT NULL,
    CC10 VARCHAR (30) NOT NULL,
    DC10 VARCHAR (100) NOT NULL,
    CC11 VARCHAR (30) NOT NULL,
    DC11 VARCHAR (100) NOT NULL,
    CC12 VARCHAR (30) NOT NULL,
    DC12 VARCHAR (100) NOT NULL,
    CC13 VARCHAR (30) NOT NULL,
    DC13 VARCHAR (100) NOT NULL,
    CC14 VARCHAR (30) NOT NULL,
    DC14 VARCHAR (100) NOT NULL,
    CC15 VARCHAR (30) NOT NULL,
    DC15 VARCHAR (100) NOT NULL,
    CC16 VARCHAR (30) NOT NULL,
    DC16 VARCHAR (100) NOT NULL,
    CC17 VARCHAR (30) NOT NULL,
    DC17 VARCHAR (100) NOT NULL,
    CC18 VARCHAR (30) NOT NULL,
    DC18 VARCHAR (100) NOT NULL,

```

```

CC19 VARCHAR (30) NOT NULL,
DC19 VARCHAR (100) NOT NULL,
CC20 VARCHAR (30) NOT NULL,
DC20 VARCHAR (100) NOT NULL,
CC21 VARCHAR (30) NOT NULL,
DC21 VARCHAR (100) NOT NULL,
CC22 VARCHAR (30) NOT NULL,
DC22 VARCHAR (100) NOT NULL,
CC23 VARCHAR (30) NOT NULL,
DC23 VARCHAR (100) NOT NULL,
CC24 VARCHAR (30) NOT NULL,
DC24 VARCHAR (100) NOT NULL,
CC25 VARCHAR (30) NOT NULL,
DC25 VARCHAR (100) NOT NULL,
CC26 VARCHAR (30) NOT NULL,
DC26 VARCHAR (100) NOT NULL,
CC27 VARCHAR (30) NOT NULL,
DC27 VARCHAR (100) NOT NULL,
CC28 VARCHAR (30) NOT NULL,
DC28 VARCHAR (100) NOT NULL,
CC29 VARCHAR (30) NOT NULL,
DC29 VARCHAR (100) NOT NULL,
CC30 VARCHAR (30) NOT NULL,
DC30 VARCHAR (100) NOT NULL,
DAC VARCHAR (20) NOT NULL)

```

```

DECLARE ref_cursor CURSOR FOR

```

```

SELECT art.ArtigoPai, art.RubDim2, Id.Descricao, art.Dim2, art.Dim1
FROM artigo art WITH (NOLOCK)
JOIN LinhasDimensao Id WITH (NOLOCK) ON Id.Dimensao = art.Dim2 and Id.RubDim = art.RubDim2
where art.ArtigoPai <> " and art.TratamentoDim=2 and art.RubDim1 <> " and art.RubDim2 <> "
GROUP BY art.ArtigoPai, art.RubDim2, Id.Descricao, art.Dim2, art.Dim1, Id.Ordem
ORDER BY art.ArtigoPai, Id.Ordem

```

```

OPEN ref_cursor
FETCH NEXT FROM ref_cursor INTO @REF, @TAM, @DESTAM, @DESHOR, @DESVER
SET @SQL = "
SET @REFT = "
SET @POSICAO = 1
WHILE @@FETCH_STATUS = 0
BEGIN
    IF (@REFT <> @REF)
    BEGIN
        SET @REFT = @REF
    END
END

```

```

        IF @SQL = "
            SET @SQL = 'SELECT ''' + @REFT + ''', ''' + @DESHOR + ''', ''' + @DESVR + ''''
        ELSE
        BEGIN
            WHILE @POSICAO <= 30
            BEGIN
                SET @SQL = @SQL + ', ''', ''''
                SET @POSICAO = @POSICAO + 1
            END

        SET @POSICAO = 1
        SET @SQL = @SQL + ', ''''

        INSERT INTO @MSGRH exec (@SQL)
        SET @SQL = 'SELECT ''' + @REFT + ''', ''' + @DESHOR + ''', ''' + @DESVR + ''''
        END
        END

        IF @TAM <> "
        BEGIN
            --SET @SQL = @SQL + ', ''' + @TAM + ''', ''' + @DESHOR + ''' + @TAM + ''' + @TAM + ''''
            SET @SQL = @SQL + ', ''' + @TAM + ''', ''' + @DESTAM + ''' + @TAM + ''''
            SET @POSICAO = @POSICAO + 1
        END

        FETCH NEXT FROM ref_cursor INTO @REF, @TAM, @DESTAM, @DESHOR, @DESVR

    END
    CLOSE ref_cursor;
    DEALLOCATE ref_cursor;

    IF @REFT <> "
    BEGIN
        IF @POSICAO <= 30
        BEGIN
            WHILE @POSICAO <= 30
            BEGIN
                SET @SQL = @SQL + ', ''', ''''
                SET @POSICAO = @POSICAO + 1
            END

        SET @POSICAO = 1
        SET @SQL = @SQL + ', ''''

        INSERT INTO @MSGRH EXEC (@SQL)
        END
        END

```

```
SELECT * FROM @MSGRH  
DELETE @MSGRH
```


MSGRV

SQL em modo Avançado

SQL

```

DECLARE @POSICAO INT
DECLARE @INC INT
DECLARE @SQL VARCHAR(MAX)
DECLARE @REF VARCHAR(60)
DECLARE @REFT VARCHAR(60)
DECLARE @TAM VARCHAR(25)
DECLARE @COR VARCHAR(25)
DECLARE @CORT VARCHAR(25)
DECLARE @DESVR VARCHAR(25)
DECLARE @MSGRV TABLE (
    REF VARCHAR(60) not null,
    DESCR VARCHAR(100) NOT NULL,
    DESCRV VARCHAR(100) NOT NULL,
    SEQ INT NOT NULL,
    CC1 VARCHAR (1) NOT NULL,
    CC2 VARCHAR (1) NOT NULL,
    CC3 VARCHAR (1) NOT NULL,
    CC4 VARCHAR (1) NOT NULL,
    CC5 VARCHAR (1) NOT NULL,
    CC6 VARCHAR (1) NOT NULL,
    CC7 VARCHAR (1) NOT NULL,
    CC8 VARCHAR (1) NOT NULL,
    CC9 VARCHAR (1) NOT NULL,
    CC10 VARCHAR (1) NOT NULL,
    CC11 VARCHAR (1) NOT NULL,
    CC12 VARCHAR (1) NOT NULL,
    CC13 VARCHAR (1) NOT NULL,
    CC14 VARCHAR (1) NOT NULL,
    CC15 VARCHAR (1) NOT NULL,
    CC16 VARCHAR (1) NOT NULL,
    CC17 VARCHAR (1) NOT NULL,
    CC18 VARCHAR (1) NOT NULL,
    CC19 VARCHAR (1) NOT NULL,
    CC20 VARCHAR (1) NOT NULL,
    CC21 VARCHAR (1) NOT NULL,
    CC22 VARCHAR (1) NOT NULL,
    CC23 VARCHAR (1) NOT NULL,
    CC24 VARCHAR (1) NOT NULL,
    CC25 VARCHAR (1) NOT NULL,
    CC26 VARCHAR (1) NOT NULL,
    CC27 VARCHAR (1) NOT NULL,
    CC28 VARCHAR (1) NOT NULL,
    CC29 VARCHAR (1) NOT NULL,

```

```
CC30 VARCHAR (1) NOT NULL,
TES VARCHAR (1) NOT NULL)
```

```
DECLARE ref_cursor CURSOR FOR
```

```
SELECT art.ArtigoPai, art.RubDim1, art.RubDim2, Id.Descricao
FROM artigo art WITH (NOLOCK)
JOIN LinhasDimensao Id WITH (NOLOCK) ON Id.Dimensao = art.Dim1 and Id.RubDim = art.RubDim1
JOIN LinhasDimensao Id2 WITH (NOLOCK) ON Id2.Dimensao = art.Dim2 and Id2.RubDim = art.RubDim2
where art.ArtigoPai <> " and art.TratamentoDim=2 and art.RubDim1 <> " and art.RubDim2 <> "
GROUP BY art.ArtigoPai, art.RubDim1, art.RubDim2, Id.Descricao, Id2.Ordem, Id.Ordem
ORDER BY art.ArtigoPai, art.RubDim1, Id2.Ordem, Id.Ordem
```

```
OPEN ref_cursor
```

```
FETCH NEXT FROM ref_cursor INTO @REF, @COR, @TAM, @DESVER
```

```
SET @SQL = "
```

```
SET @REFT = "
```

```
SET @CORT = "
```

```
SET @POSICAO = 1
```

```
WHILE @@FETCH_STATUS = 0
```

```
BEGIN
```

```
    IF (@REFT <> @REF)
```

```
    BEGIN
```

```
        SET @REFT = @REF
```

```
        SET @CORT = @COR
```

```
        SET @INC = 1
```

```
        IF @SQL = "
```

```
            SET @SQL = 'SELECT ''' + @REFT + ''', ''' + @CORT + ''', ''' + @DESVER + ''', ' +
RTRIM(CAST(@INC AS CHAR(3)))
```

```
        ELSE
```

```
        BEGIN
```

```
            WHILE @POSICAO <= 30
```

```
            BEGIN
```

```
                SET @SQL = @SQL + ', "N"'
```

```
                SET @POSICAO = @POSICAO + 1
```

```
            END
```

```
SET @POSICAO = 1
```

```
SET @SQL = @SQL + ', ""'
```

```
INSERT INTO @MSGRV exec (@SQL)
```

```
SET @SQL = 'SELECT ''' + @REFT + ''', ''' + @CORT + ''', ''' + @DESVER + ''', ' + RTRIM(CAST(@INC
AS CHAR(3)))
```

```
END
```

```
END
```

```
IF (@CORT <> @COR)
```

```

BEGIN
    SET @CORT = @COR

    WHILE @POSICAO <= 30
    BEGIN
        SET @SQL = @SQL + ', "N"'
        SET @POSICAO = @POSICAO + 1
    END

    SET @POSICAO = 1
    SET @INC = @INC + 1
    SET @SQL = @SQL + ', ""'

    INSERT INTO @MSGRV exec (@SQL)
    SET @SQL = 'SELECT ''' + @REFT + ''', ''' + @CORT + ''', ''' + @DESVER + ''', ' + RTRIM(CAST(@INC
    AS CHAR(3)))
    END

    IF @TAM <> ''
    BEGIN
        SET @SQL = @SQL + ', "S"'
        SET @POSICAO = @POSICAO + 1
    END
    FETCH NEXT FROM ref_cursor INTO @REF, @COR, @TAM, @DESVER
END
CLOSE ref_cursor;
DEALLOCATE ref_cursor;

IF @REFT <> ''
BEGIN
    WHILE @POSICAO <= 30
    BEGIN
        SET @SQL = @SQL + ', "N"'
        SET @POSICAO = @POSICAO + 1
    END
    SET @SQL = @SQL + ', ""'

    --PRINT @SQL
    INSERT INTO @MSGRV EXEC (@SQL)
END

SELECT * FROM @MSGRV ORDER BY REF

DELETE @MSGRV

```