

Manual
Recurso Code Implementation.
Converter Campo Tipo Data
FullCopyConvert Data

Sumário

Bem-vindo ao FullCopyConvert Data! Estas instruções irão guiá-lo para utilizar o recurso Code Implementation (Implementação de Código). Para realizar o tratamento para um campo do tipo data no destino.

Sobre o FullCopyConvert Data.....	3
Contato.....	3
1 – Code Implementation - Inicio. Convertendo para Campo Data.....	4

Sobre o FullCopyConvert Data.

FullCopyConvert é uma ferramenta especializada em conversão e migração de dados, oferece uma forma confortável e fácil de converter suas informações de um banco de dados para outro. Com o FullCopyConvert você poderá converter as informações das seguintes bases de dados disponíveis.



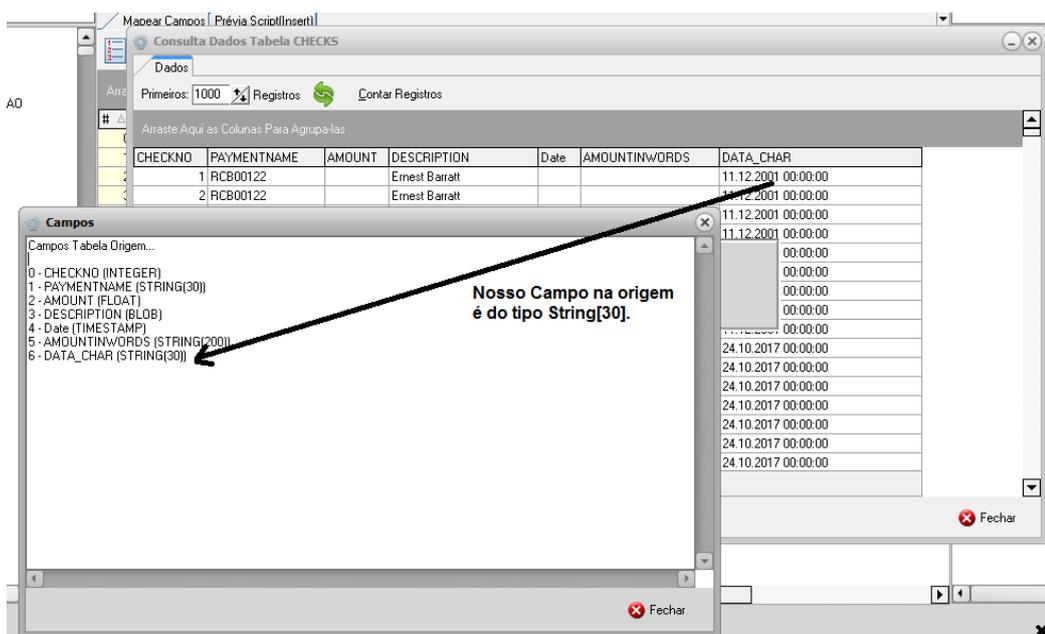
Contato.

O suporte do FullCopyConvert Data é feito através de e-mail. É necessário apenas enviar um e-mail para suporte@fullcopyconvert.com.br relatando o problema o ocorrido ou mesmo uma dúvida.

1 – Code Implementation - Inicio. Convertendo para Campo Data

O **FullCopyConvert** conta com um recurso de **implementação de código pascal**. Onde você poderá utilizar funções nativas do FullCopyConvert ou mesmo poderá criar suas próprias funções. Este recurso é bastante útil principalmente nas importações de arquivos **csv, excel, paradox e access entre outros tipos de bases**, pois conta com pouquíssimas funções para realização de tratamentos dos dados. Neste tutorial iremos demonstrar como realizar a conversão de um tipo de dados que na origem é uma data, porém esta data está armazenada em um campo do tipo “String”, e queremos converter esse campo para um campo do tipo Data no destino.

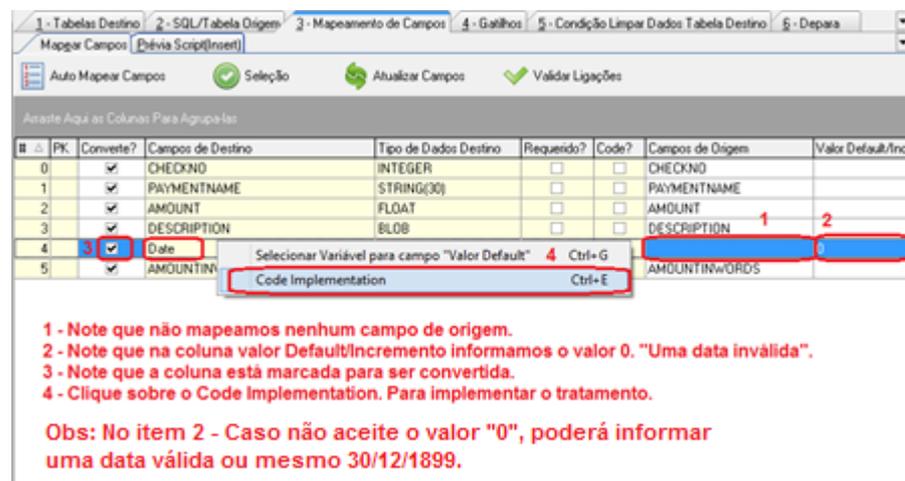
Cenário: Origem:



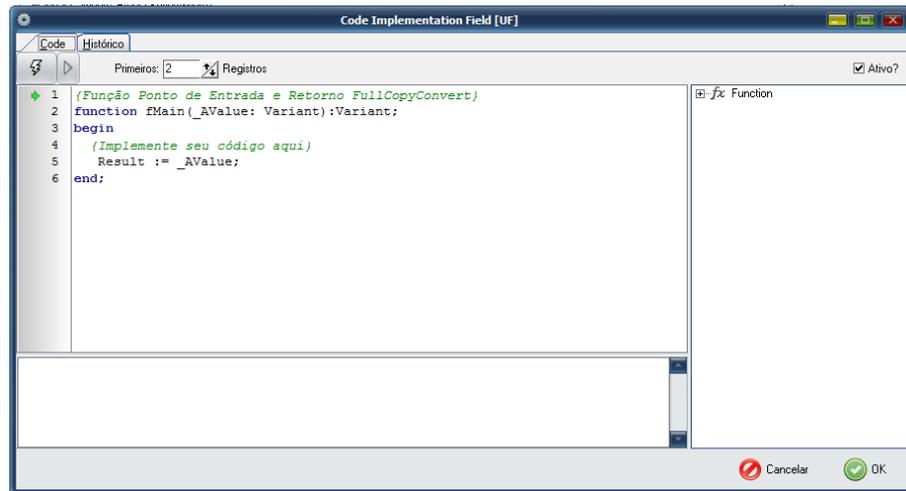
Cenário: Destino

#	Δ	PK	Converte?	Campos de Destino	Tipo de Dados Destino	Requerido?
0			<input checked="" type="checkbox"/>	CHECKNO	INTEGER	<input type="checkbox"/>
1			<input checked="" type="checkbox"/>	PAYMENTNAME	STRING(30)	<input type="checkbox"/>
2			<input checked="" type="checkbox"/>	AMOUNT	FLOAT	<input type="checkbox"/>
3			<input checked="" type="checkbox"/>	DESCRIPTION	BLOB	<input type="checkbox"/>
4			<input checked="" type="checkbox"/>	Date	DATE	<input type="checkbox"/>
5			<input checked="" type="checkbox"/>	AMOUNTINWORDS	STRING(200)	<input type="checkbox"/>

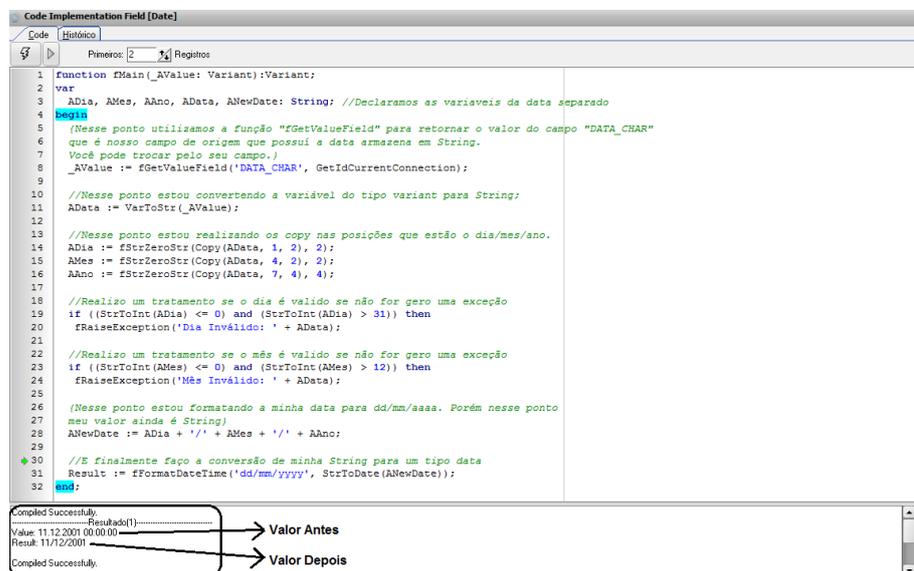
1. Agora vamos ver um pouco na prática como iremos utilizar esse recurso no FullCopyConvert. Abra o FullCopyConvert e **vá até a 4° (quarta etapa)**. Selecione uma tabela que deseja trabalhar e depois de informar os dados da Aba 2 – SQL/Tabela Origem, **clique na aba 3 – Mapeamento de Campos**. Localize o campo que deseja realizar um tratamento e clique com o botão direito do mouse e selecione a opção **“Code Implementation”** ou mesmo pressione “Ctrl + E”. Como demonstra imagem abaixo: **Leia as anotações da imagem**.



2. Após seguir as orientações especificadas anteriormente note que irá abrir a seguinte tela. E já irá retornar uma função chamada **“fMain”**. Essa função é o ponto de entrada de cada coluna que se deseja realizar o tratamento. A função **“fMain”** **não pode ser alterada e não poderá acrescentar mais parâmetros de entrada** nesta função. A função **“fMain”** serve como um ponto de entrada do FullCopyConvert. A cada passada de registro nesta determinada coluna o FullCopyConvert irá realizar a chamada da função fMain e irá retornar os dados que da própria função gerar. É muito importante que exista o ponto de retorno da função no caso **Result**. Veja abaixo a função original sem alteração na mesma.



3. Agora vamos ver como nossa função ficou. **Veja e analise os comentários.**



4. No exemplo da tela acima podemos ver que fizemos alguns tratamentos. Segue a função para ser copiada e modificada por vocês como desejarem.

```
function fMain(_AValue: Variant):Variant;
var
  ADia, AMes, AAno, AData, ANewDate: String; //Declaramos as variaveis da data separado
begin
  {Nesse ponto utilizamos a função "fGetValueField" para retornar o valor do campo "DATA_CHAR"
  que é nosso campo de origem que possui a data armazenada em String.
  Você pode trocar pelo seu campo.}
  _AValue := fGetValueField("DATA_CHAR", GetIdCurrentConnection);

  //Nesse ponto estou convertendo a variável do tipo variant para String;
  AData := VarToStr(_AValue);

  //Nesse ponto estou realizando os copy nas posições que estão o dia/mes/ano.
  ADia := fStrZeroStr(Copy(AData, 1, 2), 2);
  AMes := fStrZeroStr(Copy(AData, 4, 2), 2);
  AAno := fStrZeroStr(Copy(AData, 7, 4), 4);

  //Realizo um tratamento se o dia é valido se não for gero uma exceção
  if ((StrToInt(ADia) <= 0) and (StrToInt(ADia) > 31)) then
    fRaiseException('Dia Inválido: ' + AData);

  //Realizo um tratamento se o mês é valido se não for gero uma exceção
  if ((StrToInt(AMes) <= 0) and (StrToInt(AMes) > 12)) then
    fRaiseException('Mês Inválido: ' + AData);

  {Nesse ponto estou formatando a minha data para dd/mm/aaaa. Porém nesse ponto
  meu valor ainda é String}
  ANewDate := ADia + '/' + AMes + '/' + AAno;

  //E finalmente fapo a conversão de minha String para um tipo date
  Result := fFormatDateTime('dd/mm/yyyy', StrToDate(ANewDate));
end;
```

//Nesse ponto estou realizando os copy nas posições que estão o dia/mes/ano.

ADia := fStrZeroStr(Copy(AData, 1, 2), 2);

AMes := fStrZeroStr(Copy(AData, 4, 2), 2);

AAno := fStrZeroStr(Copy(AData, 7, 4), 4);

//Realizo um tratamento se o dia é valido se não for gero uma exceção

if ((StrToInt(ADia) <= 0) and (StrToInt(ADia) > 31)) then

fRaiseException('Dia Inválido: ' + AData);

//Realizo um tratamento se o mês é valido se não for gero uma exceção

if ((StrToInt(AMes) <= 0) and (StrToInt(AMes) > 12)) then

fRaiseException('Mês Inválido: ' + AData);

{Nesse ponto estou formatando a minha data para dd/mm/aaaa. Porém nesse ponto meu valor ainda é String}

ANewDate := ADia + '/' + AMes + '/' + AAno;

//E finalmente faço a conversão de minha String para um tipo data

Result := fFormatDateTime('dd/mm/yyyy', StrToDate(ANewDate));

end;

OBS: A máscara 'dd/mm/yyyy' você deve especificar como seu banco de dados irá aceitar a inserção dos dados. Talvez em alguns bancos tenha que inverter a máscara para 'mm/dd/yyyy'. Fica a critério de vocês analisarem e realizarem a devida correção.

Veja abaixo com será gerado nosso script na prévia após os tratamentos.

Registrar Ferramentas Visão Ajuda

Etapa 4 de 6 - Ligações Tabelas de Destino e Origem/SQL

Verifique as Ligações das Tabelas Destino e Origem/SQL a Serem Processadas e Clique em Avançar

1 - Tabelas Destino 2 - SQL/Tabela Origem 3 - Mapeamento de Campos 4 - Gatilhos 5 - Condição Limpar Dados Tabela Destino 6 - Depara

Mapear Campo Prévia Script(Insert)

Gerar Prévia Limpar Prévia Primeiros: 10 Registros

Log Prévia Script(Insert)... Thread(0)
Iniciando Prévia... Thread(0)

Carregando Registros para Tabela de Destino: CHECKS Thread (0)
Carregando Registros para Depara Tabela de Destino: CHECKS Thread (0)
Aguarde Contando Registros Tabela (CHECKS)... Thread (0)
Data/Hora Início Prévia Tabela: CHECKS 24/10/2017 20:21:40 Thread (0)

Inserindo Registros Tabela CHECKS(Base Destino) Thread (0)
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(1, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(2, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(3, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(4, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(5, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(6, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(7, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(8, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(9, 'RCB00122', NULL, _DESCRIPTION, '11/12/2001', NULL);
INSERT INTO CHECKS (CHECKNO, PAYMENTNAME, AMOUNT, DESCRIPTION, 'Date', AMOUNTINWORDS) VALUES(10, 'RCB00122', NULL, _DESCRIPTION, '24/10/2017', NULL);

Total de Script's Geradas: 10 Para Tabela CHECKS Thread (0)
Total de Script's Ignorados 0 Para Tabela CHECKS Thread (0)
Data/Hora Término Prévia Tabela: CHECKS 24/10/2017:20:21:40 Thread (0)
TERMINO PROCESSO DATA/HORA(24/10/2017 20:21:40) THREAD(0)

Dúvidas entre em contato.

Questões gerais

info@fullcopyconvert.com.br

Vendas questões relacionadas

registro@fullcopyconvert.com.br

Apoiar

Problemas com o uso de nossos programas ou questões simplesmente técnicas?

suporte@fullcopyconvert.com.br

Requisito	Limite Versão Trial	Limite Versão Pro	Limites Trial	Limites Pro
Conversão de Registros	Sim	Não	1000 Registros Por Tabela. 5 Tabelas por Vez.	Não Há
Conversão Tabela	Sim	Não	5 Por Vez	Não Há
Criação Índices	Sim	Não	5 Por Vez	Não Há
Criação Foreign Key	Sim	Não	5 Por Vez	Não Há
Criação Projetos	Não	Não	Não Há	Não Há
De/Para	Não	Não	Não Há	Não Há
Bulk Insert	Sim	Não	0 Registros	Não Há
Agendamento	Não	Não	1000 Registros Por Tabela. 5 Tabelas por Vez.	Não Há
Gatilhos	Não	Não	Não Há	Não Há
Exportação para Script	Sim	Não	1000 Registros Por Tabela. 5 Tabelas por Vez.	Não Há