|  |
| --- |
| Documentation()  OnRun()  SyncSalesPrice("Start Date" : Date;"End Date" : Date)  DealerInfo.RESET;  DealerInfo.SETRANGE(Active, TRUE);  //DealerInfo.SETFILTER("company Database", '<>%1', '');  //DealerInfo.SETFILTER("company Database",'%1', 'DEVTRADERS');  IF DealerInfo.FINDSET THEN  REPEAT  Item.RESET;  IF Item.FINDSET THEN REPEAT  SalesPrice.RESET;  SalesPrice.SETRANGE("Starting Date", "Start Date","End Date");  IF SalesPrice.FINDSET THEN  REPEAT  ConnectSQL(DealerInfo);  ReadSalesPrice(DealerInfo, SalesPrice);  IF SQLDataReader.HasRows THEN BEGIN  DisconnectSQL;  UpdateSalesPrice(DealerInfo, SalesPrice);  END ELSE BEGIN  DisconnectSQL;  InsertSalesPrice(DealerInfo, SalesPrice);  END;  DisconnectSQL;  UNTIL SalesPrice.NEXT =0;  UNTIL Item.NEXT = 0;  UNTIL DealerInfo.NEXT =0;  //MESSAGE('Item sales price sync done.');  LOCAL UpdateSalesPrice(DealerInfo : Record "Dealer Information";SalesPrice : Record "Sales Price")  CLEAR(commandtext);  ConnectSQL(DealerInfo);  CASE SalesPrice."Sales Type" OF  SalesPrice."Sales Type"::Customer:  SalesTypeVar := 0;  SalesPrice."Sales Type"::"Customer Price Group":  SalesTypeVar := 1;  SalesPrice."Sales Type"::"All Customers":  SalesTypeVar := 2;  SalesPrice."Sales Type"::Campaign:  SalesTypeVar := 3;  END;  IF SalesPrice."Ending Date" <> 0D THEN  EndDate := FORMAT(SalesPrice."Ending Date")  ELSE  EndDate := FORMAT('1753-01-01 00:00:00.000');  commandtext := UpdateTxt + '[' + DealerInfo."Company Name" + '$Sales Price' + ']' + SpaceTxt + SetTxt +  '[Unit Price] = @UnitPrice,' +  '[Item No\_] = @itemNo,' +  '[Sales Type] = @SalesType,' +  '[Sales Code] = @SalesCode,' +  '[Starting Date] = @StartingDate,' +  '[Ending Date] = @EndingDate,' +  '[Variant Code] = @VariantCode,' +  '[Minimum Quantity] = @MinimumQuantity,' +  '[Unit of Measure Code] = @UnitOfMeasureCode,' +  '[Currency Code] = @CurrencyCode' +  SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SalesPrice."Item No.") +''''+ SpaceTxt+ AndText+  SpaceTxt + '[Sales Type]' + ' = ''' + FORMAT(SalesTypeVar) +''''+ SpaceTxt + AndText+  SpaceTxt + '[Sales Code]' + ' = ''' + FORMAT(SalesPrice."Sales Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SalesPrice."Variant Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Minimum Quantity]' + ' = ''' + FORMAT(SalesPrice."Minimum Quantity") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Unit of Measure Code]' + ' = ''' + FORMAT(SalesPrice."Unit of Measure Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Starting Date]' + ' = ''' + FORMAT(SalesPrice."Starting Date") +'''' ;    { commandtext := UpdateTxt + '[' + 'DEVMOTORSPVTLTD$Sales Price' + ']' + SpaceTxt + SetTxt +  '[Unit Price] = @UnitPrice,' +  '[Item No\_] = @itemNo,' +  '[Sales Type] = @SalesType,' +  '[Sales Code] = @SalesCode,' +  '[Starting Date] = @StartingDate,' +  '[Ending Date] = @EndingDate,' +  '[Variant Code] = @VariantCode,' +  '[Minimum Quantity] = @MinimumQuantity,' +  '[Unit of Measure Code] = @UnitOfMeasureCode,' +  '[Currency Code] = @CurrencyCode' +  SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SalesPrice."Item No.") +''''+ SpaceTxt+ AndText+  SpaceTxt + '[Sales Type]' + ' = ''' + FORMAT(SalesTypeVar) +''''+ SpaceTxt + AndText+  SpaceTxt + '[Sales Code]' + ' = ''' + FORMAT(SalesPrice."Sales Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SalesPrice."Variant Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Minimum Quantity]' + ' = ''' + FORMAT(SalesPrice."Minimum Quantity") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Unit of Measure Code]' + ' = ''' + FORMAT(SalesPrice."Unit of Measure Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Starting Date]' + ' = ''' + FORMAT(SalesPrice."Starting Date") +'''' ;  }    SetupSQLCommand;  SQLCommand.Parameters.AddWithValue('@UnitPrice',SalesPrice."Unit Price");  SQLCommand.Parameters.AddWithValue('@itemNo',FORMAT(SalesPrice."Item No."));  SQLCommand.Parameters.AddWithValue('@SalesType',SalesTypeVar);  SQLCommand.Parameters.AddWithValue('@SalesCode',FORMAT(SalesPrice."Sales Code"));  SQLCommand.Parameters.AddWithValue('@StartingDate',FORMAT(SalesPrice."Starting Date"));  SQLCommand.Parameters.AddWithValue('@EndingDate',EndDate);  SQLCommand.Parameters.AddWithValue('@VariantCode',FORMAT(SalesPrice."Variant Code"));  SQLCommand.Parameters.AddWithValue('@MinimumQuantity',SalesPrice."Minimum Quantity");  SQLCommand.Parameters.AddWithValue('@UnitOfMeasureCode',FORMAT(SalesPrice."Unit of Measure Code"));  SQLCommand.Parameters.AddWithValue('@CurrencyCode',FORMAT(SalesPrice."Currency Code"));  SQLCommand.ExecuteNonQuery;  LOCAL InsertSalesPrice(DBINfo : Record "Dealer Information";SalesPrice : Record "Sales Price")  CLEAR(commandtext);  ConnectSQL(DBINfo);  CASE SalesPrice."Sales Type" OF  SalesPrice."Sales Type"::Customer:  SalesTypeVar := 0;  SalesPrice."Sales Type"::"Customer Price Group":  SalesTypeVar := 1;  SalesPrice."Sales Type"::"All Customers":  SalesTypeVar := 2;  SalesPrice."Sales Type"::Campaign:  SalesTypeVar := 3;  END;  IF SalesPrice."Ending Date" <> 0D THEN  EndDate := FORMAT(SalesPrice."Ending Date")  ELSE  EndDate := FORMAT('1753-01-01 00:00:00.000');  commandtext := InsertTxt + '['+ DBINfo."Company Name" +'$Sales Price'+ ']' + SpaceTxt + '([Unit Price], [Item No\_],[Sales Type],[Sales Code], [Starting Date], '+  '[Ending Date],[Variant Code], [Minimum Quantity], [Unit of Measure Code], [Currency Code],[Price Includes VAT],[Allow Invoice Disc\_],' +  '[VAT Bus\_ Posting Gr\_ (Price)],[Allow Line Disc\_])'+ ValuesTxt + '(@UnitPrice, @itemNo, @SalesType, @SalesCode,' +  '@StartingDate, @EndingDate, @VariantCode, @MinimumQuantity, @UnitOfMeasureCode, @CurrencyCode, @PriceInclVat, @AllowInvDisc, @VatBusPos, @AllowLineDisc)';  {commandtext := InsertTxt + '[' + 'DEVMOTORSPVTLTD$Sales Price' +']' + SpaceTxt + '([Unit Price], [Item No\_],[Sales Type],[Sales Code], [Starting Date], '+  '[Ending Date],[Variant Code], [Minimum Quantity], [Unit of Measure Code], [Currency Code],[Price Includes VAT],[Allow Invoice Disc\_],' +  '[VAT Bus\_ Posting Gr\_ (Price)],[Allow Line Disc\_])'+ ValuesTxt + '(@UnitPrice, @itemNo, @SalesType, @SalesCode,' +  '@StartingDate, @EndingDate, @VariantCode, @MinimumQuantity, @UnitOfMeasureCode, @CurrencyCode, @PriceInclVat, @AllowInvDisc, @VatBusPos, @AllowLineDisc)';    }    SetupSQLCommand;  SQLCommand.Parameters.AddWithValue('@UnitPrice',SalesPrice."Unit Price");  SQLCommand.Parameters.AddWithValue('@itemNo',FORMAT(SalesPrice."Item No."));  SQLCommand.Parameters.AddWithValue('@SalesType',SalesTypeVar);  SQLCommand.Parameters.AddWithValue('@SalesCode',FORMAT(SalesPrice."Sales Code"));  SQLCommand.Parameters.AddWithValue('@StartingDate',FORMAT(SalesPrice."Starting Date"));  SQLCommand.Parameters.AddWithValue('@EndingDate',EndDate);  SQLCommand.Parameters.AddWithValue('@VariantCode',FORMAT(SalesPrice."Variant Code"));  SQLCommand.Parameters.AddWithValue('@MinimumQuantity',SalesPrice."Minimum Quantity");  SQLCommand.Parameters.AddWithValue('@UnitOfMeasureCode',FORMAT(SalesPrice."Unit of Measure Code"));  SQLCommand.Parameters.AddWithValue('@CurrencyCode',FORMAT(SalesPrice."Currency Code"));  SQLCommand.Parameters.AddWithValue('@PriceInclVat', SalesPrice."Price Includes VAT");  SQLCommand.Parameters.AddWithValue('@AllowInvDisc', SalesPrice."Allow Invoice Disc.");  SQLCommand.Parameters.AddWithValue('@AllowLineDisc', SalesPrice."Allow Line Disc.");  SQLCommand.Parameters.AddWithValue('@VatBusPos', FORMAT(SalesPrice."VAT Bus. Posting Gr. (Price)"));  SQLCommand.ExecuteNonQuery;  LOCAL ReadSalesPrice(DealerInfo : Record "Dealer Information";SalesPrice : Record "Sales Price")  ClearSQLCommand;  CASE SalesPrice."Sales Type" OF  SalesPrice."Sales Type"::Customer:  SalesTypeVar := 0;  SalesPrice."Sales Type"::"Customer Price Group":  SalesTypeVar := 1;  SalesPrice."Sales Type"::"All Customers":  SalesTypeVar := 2;  SalesPrice."Sales Type"::Campaign:  SalesTypeVar := 3;  END;  commandtext := ReadCommandTxt + '[' + DealerInfo."Company Name" + '$Sales Price' + ']' +  SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SalesPrice."Item No.") +''''+ SpaceTxt+ AndText+  SpaceTxt + '[Sales Type]' + ' = ''' + FORMAT(SalesTypeVar) +''''+ SpaceTxt + AndText+  SpaceTxt + '[Sales Code]' + ' = ''' + FORMAT(SalesPrice."Sales Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SalesPrice."Variant Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Minimum Quantity]' + ' = ''' + FORMAT(SalesPrice."Minimum Quantity") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Unit of Measure Code]' + ' = ''' + FORMAT(SalesPrice."Unit of Measure Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Starting Date]' + ' = ''' + FORMAT(SalesPrice."Starting Date") +'''' ;  {commandtext := ReadCommandTxt + '[' + 'DEVMOTORSPVTLTD$Sales Price' + ']' +  SpaceTxt + WhereTxt + '[Item No\_]' + ' = ''' + FORMAT(SalesPrice."Item No.") +''''+ SpaceTxt+ AndText+  SpaceTxt + '[Sales Type]' + ' = ''' + FORMAT(SalesTypeVar) +''''+ SpaceTxt + AndText+  SpaceTxt + '[Sales Code]' + ' = ''' + FORMAT(SalesPrice."Sales Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Variant Code]' + ' = ''' + FORMAT(SalesPrice."Variant Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Minimum Quantity]' + ' = ''' + FORMAT(SalesPrice."Minimum Quantity") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Unit of Measure Code]' + ' = ''' + FORMAT(SalesPrice."Unit of Measure Code") +'''' + SpaceTxt + AndText+  SpaceTxt + '[Starting Date]' + ' = ''' + FORMAT(SalesPrice."Starting Date") +'''' ;  }  SetupSQLCommand;  SQLDataReader := SQLCommand.ExecuteReader;  SQLDataReader.Read;  LOCAL ConnectSQL(DealerInfo : Record "Dealer Information")  SQLConnectionMgt.SetupSQLConnection(SQLConnection, DealerInfo."company Database");  LOCAL DisconnectSQL()  SQLConnectionMgt.CloseSQLConnection(SQLConnection);  LOCAL ClearSQLCommand()  CLEAR(commandtext);  LOCAL SetupSQLCommand()  SQLConnectionMgt.SetupSQLCommand(SQLConnection,SQLCommand,commandtext,SQLCommandType::Text);  LOCAL HasSalesPrice() : Boolean |