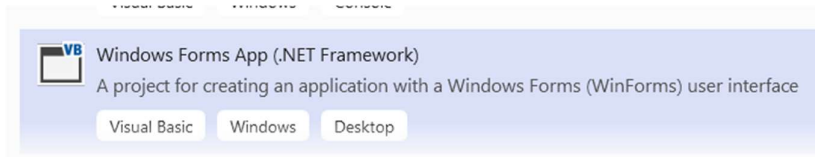
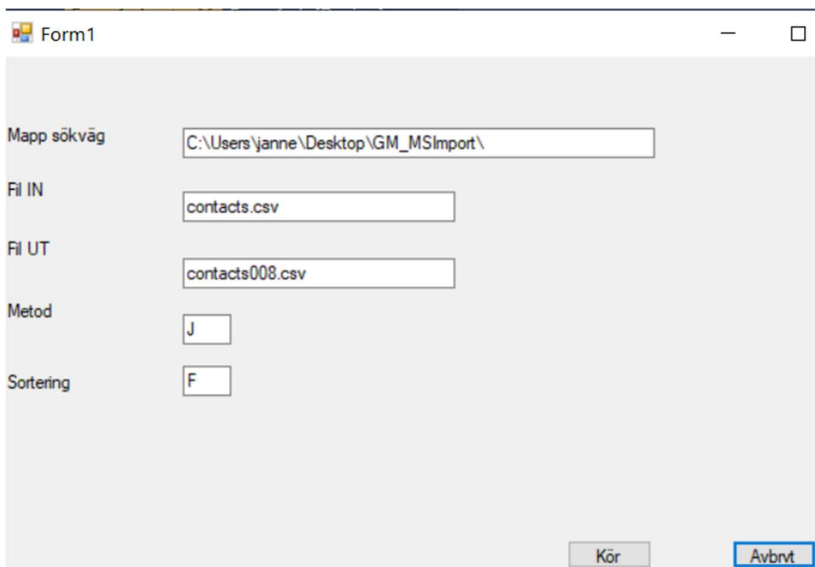


Projektet är skapat baserat på



Formuläret som används ser ut så här



Nedan finner du aktuell kod för hela Form1.vb modulen

```
Imports System.IO
Imports System.Text
```

```
Public Class Form1
```

```
    '*****
    'Struktur i huvuddrag för de arbetande procedurerna
    ' Public Sub WriteCsvFile(ArgRunType As String, ArgFileNameIn As String, ArgFileNameOut As String)
    '     Public Function isOpenCol(Argstr As String) As Boolean
    '     Public Function CvtHeader(ArgRunType As String, ArgStr As String) As String
    '         Public Function CvtHeader2(ArgRunType As String, ArgStr As String) As String
    '     Public Function CvtLine_L(ArgStr As String) As String
    '     Public Function CvtLine_J(ArgStr As String) As String
    '         Public Function GetComId(ArgCol As String, ArgIndex As Integer) As String
    '     Public Function CleanComCol(ArgCol As String)
    ' 
```

```
    '*****
```

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    'Jannes miljö för tester med eget material
```

```
    TextBox1.Text = "C:\Users\janne\Desktop\GM_MSImport\" 'mapp (sökväg)
    TextBox2.Text = "contacts.csv" 'IN-fil exporterad från GM
    TextBox3.Text = "contacts008.csv" 'UT-fil att ta vidare till Outlook import
    TextBox4.Text = "J" 'Layout på GM filen; L= Lennart, J= Janne
    TextBox5.Text = "F" 'Sort: O=Oförändrat, E=Efternamn, F=Förnamn
```

```
End Sub
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Close()
End Sub
```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
```

```
    Call WriteCsvFile(TextBox4.Text, TextBox5.Text, TextBox1.Text & TextBox2.Text, TextBox1.Text &
    TextBox3.Text)
```

```
End Sub
```

```
Public Sub WriteCsvFile(ArgRunType As String, ArgSort As String, ArgFileNameIn As String, ArgFileNameOut
As String)
```

```
    '*****
    'Stöd inför import av Kontakter från GM till Mina Kontakter i Outlook
    'Läs en .csv fil, modifiera innehållet, skriv till en .csv fil
    ' Modifiera rubrikraden för att underlätta mappningen
    ' Omfördela placering av telefonnummer (för Janne)
    ' Ta hand om "radbyte speciellt. Kod som är vbLf används som postavskiljare
    ' men samma kod används om man i någon kolumn använt Radbyte tangenten
    ' logiken störs av ; (semikolon) och , (komma) och " (citatt)
    '*****
```

```
    Dim myfilepath As String = ArgFileNameOut
    Dim myappendingoption As Boolean = False 'True: append text to file, False: overwrite
    Dim sw As StreamWriter = New StreamWriter(myfilepath, myappendingoption,
System.Text.Encoding.Default)
    Dim stringReader As String 'indata variabel för hel rad
    Dim stringReader2 As String
```

```

Dim wRows As Integer

'*****
'   Bara ett tfn nummer per kolumn
'   Ta bort dubblettnummer orsakade av exporten
'*****

stringReader2 = ""

Using MyReader As New Microsoft.VisualBasic.
    FileIO.TextFieldParser(ArgFileNameIn)
    Dim fileReader As System.IO.StreamReader
    fileReader =
    My.Computer.FileSystem.OpenTextFileReader(ArgFileNameIn)

    While Not MyReader.EndOfData
        stringReader = fileReader.ReadLine()

        If stringReader = Nothing Then Exit While

        stringReader = stringReader.Replace(vbLf, "") 'ta bort radbrytning/ postavskiljare

        wRows += 1

        If wRows > 1 Then 'särbehandla radbrytning/postavgränsare; inne i vissa kolumner

            Do While isOpenCol(stringReader) 'öppen om kolumn startar med " men utan samma avslut
                stringReader2 = fileReader.ReadLine() '????? CrL
                stringReader = stringReader & stringReader2.Replace(vbLf, "")
            Loop

        End If

        Select Case wRows
            Case 1
                stringReader = CvtHeader(ArgRunType, ArgSort, stringReader) 'hantera rubrikraden
            Case > 1
                Select Case ArgRunType
                    Case "L"
                        stringReader = ChgSortOrder(stringReader, ArgSort) 'Hantera sortering
                        stringReader = CvtLine_L(stringReader) 'datarader Lennart
                    Case "J"
                        stringReader = ChgSortOrder(stringReader, ArgSort) 'Hantera sortering
                        stringReader = CvtLine_J(stringReader) 'datarader Janne
                    Case Else
                End Select
            End Select
            stringReader = stringReader & vbCrLf 'lägg till postavskiljare
            sw.Write(stringReader) 'skriv till fil
        End While

    End Using

    sw.Close()

End Sub
Public Function isOpenCol(Argstr As String) As Boolean
'*****
'   Är en kolumn öppen
'   om fält börjar med "" måste den avslutas på samma sätt
'   räkna förekomster
'*****

```

```

Const cTest1 = ", ""
Const cTest2 = """, "
Dim ix0, ix1
Dim wAnt1 As Integer
Dim wAnt2 As Integer

ix0 = 1
ix1 = InStr(ix0, Argstr, cTest1)

Do While ix1 > 0
    wAnt1 += 1
    ix0 = ix1 + Len(cTest1)
    ix1 = InStr(ix0, Argstr, cTest1)
Loop

ix0 = 1
ix1 = InStr(ix0, Argstr, cTest2)

Do While ix1 > 0
    wAnt2 += 1
    ix0 = ix1 + Len(cTest2)
    ix1 = InStr(ix0, Argstr, cTest2)
Loop

If wAnt1 > wAnt2 Then
    Return True
Else
    Return False
End If

```

End Function

```
Public Function CvtHeader(ArgRunType As String, ArgSort As String, ArgStr As String) As String
```

```

'*****
'byt ut texter i kolumnrubriken för att sedan kunna
'använda standardmappning
'*****

```

```

Dim wStr As String = ArgStr
Dim ix0, ix1
Dim wNameFirst As Boolean = False

```

```

Const cName = "Name," 'måste särbehandlas som 1a kolumn
Const cGivenName = ",Given Name,"
Const cFamilyName = ",Family Name,"
Const cBirthday = ",Birthday,"
Const cEmail = ",E-mail 1 - Value,"
Const cEmail2 = ",E-mail 2 - Value,"
Const cEmail3 = ",E-mail 3 - Value,"
Const cPhone1 = ",Phone 1 - Value,"
Const cPhone2 = ",Phone 2 - Value,"
Const cPhone3 = ",Phone 3 - Value,"
Const cPhone4 = ",Phone 4 - Value,"
Const cOccup = ",Occupation,"
Const cNotes = ",Notes,"
Const cGroup = ",Group Membership,"
Const cStreet = ",Address 1 - Street,"
Const cCity = ",Address 1 - City,"
Const cPostcode = ",Address 1 - Postal Code,"
Const cCountry = ",Address 1 - Country,"

```

```

Const cNamn = "Namn,"
Const cForNamn = ",Förnamn,"
Const cEfterNamn = ",Efternamn,"
Const cFodelsedag = ",Födelsedag,"
Const cEpost = ",E-postadress,"
Const cEPost2 = ",E-postadress 2,"
Const cEPost3 = ",E-postadress 3,"
Const cTfn1 = ",Mobiltelefon,"
Const ctfn2 = ",Telefon 1,"
Const ctfn3 = ",Hem,"
Const ctfn4 = ",Telefon 2,"
Const cYrke = ",Yrke,"
Const cAnteck = ",Anteckningar,"
Const cGrupp = ",Kategorier,"
Const cGata = ",Postadress,"
Const cOrt = ",Ort,"
Const cPostnr = ",Postnr,"
Const cLand = ",Land/Region,"

' byt ut text i en kolumn åt gången
' postens längd ändras efter varje kolumnbyte

ix0 = 1

Do While ix0 < Len(wStr)
    ix1 = 0
    ix0 = 1

    If Not wNameFirst Then                'särbehandla Name som 1a kolumn
        wNameFirst = True
        ix1 = InStr(ix0, wStr, cName, CompareMethod.Text)
        If ix1 > 0 Then
            wStr = Mid(wStr, 1, ix1 - 1) & cNamn & Mid(wStr, ix1 + Len(cName))
            Continue Do
        End If
    End If

    ix1 = InStr(ix0, wStr, cGivenName, CompareMethod.Text) 'särbehandla m.a.a.sortering
    If ix1 > 0 Then
        wStr = Mid(wStr, 1, ix1 - 1) & cForNamn & Mid(wStr, ix1 + Len(cGivenName))
        Continue Do
    End If

    ix1 = InStr(ix0, wStr, cFamilyName, CompareMethod.Text) 'särbehandla m.a.a.sortering
    If ix1 > 0 Then
        wStr = Mid(wStr, 1, ix1 - 1) & cEfterNamn & Mid(wStr, ix1 + Len(cFamilyName))
        Continue Do
    End If

    ix1 = InStr(ix0, wStr, cBirthday, CompareMethod.Text)
    If ix1 > 0 Then
        wStr = Mid(wStr, 1, ix1 - 1) & cFodelsedag & Mid(wStr, ix1 + Len(cBirthday))
        Continue Do
    End If

    ix1 = InStr(ix0, wStr, cEmail, CompareMethod.Text)
    If ix1 > 0 Then
        wStr = Mid(wStr, 1, ix1 - 1) & cEpost & Mid(wStr, ix1 + Len(cEmail))
        Continue Do
    End If

```

```

ix1 = InStr(ix0, wStr, cEmail2, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cEPost2 & Mid(wStr, ix1 + Len(cEmail2))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cPhone1, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cTfn1 & Mid(wStr, ix1 + Len(cPhone1))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cPhone2, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cTfn2 & Mid(wStr, ix1 + Len(cPhone2))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cPhone3, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cTfn3 & Mid(wStr, ix1 + Len(cPhone3))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cPhone4, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cTfn4 & Mid(wStr, ix1 + Len(cPhone4))
    Continue Do
End If

' Fält som bara används hos L

ix1 = InStr(ix0, wStr, cEmail3, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cEPost3 & Mid(wStr, ix1 + Len(cEmail3))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cOccup, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cYrke & Mid(wStr, ix1 + Len(cOccup))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cNotes, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cAnteck & Mid(wStr, ix1 + Len(cNotes))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cGroup, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cGrupp & Mid(wStr, ix1 + Len(cGroup))
    Continue Do
End If

```

```

ix1 = InStr(ix0, wStr, cStreet, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cGata & Mid(wStr, ix1 + Len(cStreet))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cCity, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cOrt & Mid(wStr, ix1 + Len(cCity))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cPostcode, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cPostnr & Mid(wStr, ix1 + Len(cPostcode))
    Continue Do
End If

ix1 = InStr(ix0, wStr, cCountry, CompareMethod.Text)
If ix1 > 0 Then
    wStr = Mid(wStr, 1, ix1 - 1) & cLand & Mid(wStr, ix1 + Len(cCountry))
    Continue Do
End If

Exit Do

```

Loop

```

wStr = CvtHeader2(ArgRuntype, wStr)
CvtHeader = wStr 'returnera hela raden

```

End Function

```

Public Function CvtHeader2(ArgRuntype As String, ArgStr As String) As String
    '*****
    'skapa texter i kolumnrubriken för att sedan kunna
    'använda standardmappning
    'Om möjligt ha en lösning som automatiskt fortplantar sig till
    'CvtLiine proceduren (public constant?/ separat modul i så fall)
    'Å andra sidan kan nya kolumner variera för L/J
    '*****
    'Verkar OK måste säkerställa att lika många kolumner läggs till i CvtLine!!!!
    '*****
    Dim wStr As String = ArgStr

    Const cImpNbr = ",Importnr."
    Const cUpdDate = ",Importdat."

    ' lägg till kolumner i slutet
    ' postens längd ändras efter varje kolumntilllägg

    Select Case ArgRuntype
        Case "L" 'Lennart
            wStr = wStr.Replace(vbLf, "") & cImpNbr & cUpdDate
        Case "J" 'Janne
            wStr = wStr.Replace(vbLf, "") & cImpNbr & cUpdDate
    End Select

    CvtHeader2 = wStr 'returnera hela raden

```

End Function

```

Public Function ChgSortOrder(ArgStr As String, ArgSort As String)
'*****
'Behandla "sortering" d.v.s. data i Namn - kolumnen
' O = Oförändrat
' E = Efternamn + Förnamn
' F = Förnamn + Efternamn
'Värdet i Namn kolumner skall ändras
'*****
Dim wStr, wStr2, wStr3 As String
Dim ix1, ix2
Const cComma = ","
Const cSpace = " "

Select Case ArgSort
Case "F"
    wStr2 = GetDelNamn(ArgStr, 2)
    wStr3 = GetDelNamn(ArgStr, 4)
    ix1 = 1
    ix2 = InStr(ix1, ArgStr, cComma)
    If ix2 > 0 Then
        wStr = wStr2 & cSpace & wStr3 & (Mid(ArgStr, ix2))
    Else
        wStr = ArgStr
    End If

Case "E"
    wStr2 = GetDelNamn(ArgStr, 2)
    wStr3 = GetDelNamn(ArgStr, 4)
    ix1 = 1
    ix2 = InStr(ix1, ArgStr, cComma)
    If ix2 > 0 Then
        wStr = wStr3 & cSpace & wStr2 & (Mid(ArgStr, ix2))
    Else
        wStr = ArgStr
    End If

Case Else
    wStr = ArgStr

End Select

ChgSortOrder = wStr

End Function

Public Function GetDelNamn(ArgStr As String, ArgCol As Integer) As String

'Förnamn i kolumn 2; mellan 1a o 2a kommatecknet
'Efternamn i kolumn 4; mellan 3e o 4e kommatecknet

Dim ix1, ix2
Dim wStr As String
Const cComma = ","
Const cEmpty = ""

Select Case ArgCol
Case 2
    ix1 = InStr(1, ArgStr, cComma)
    ix2 = InStr(ix1 + 1, ArgStr, cComma)
    If ix1 = 0 Or ix2 = 0 Then Return cEmpty
    wStr = Mid(ArgStr, ix1 + 1, ix2 - ix1 - 1)

```



```

    Return wStr
Case 4
    ix1 = InStr(1, ArgStr, cComma)
    ix1 = InStr(ix1 + 1, ArgStr, cComma)
    ix1 = InStr(ix1 + 1, ArgStr, cComma)
    ix2 = InStr(ix1 + 1, ArgStr, cComma)
    If ix1 = 0 Or ix2 = 0 Then Return cEmpty
    wStr = Mid(ArgStr, ix1 + 1, ix2 - ix1 - 1)
    Return wStr
Case Else
    Return ArgStr
End Select

```

End Function

Public Function CvtLine_L(ArgStr As String) As String

```

'*****
'I vissa av de kolumner där rubrik (ev. ändrats skall innehållet modifieras
'I någon kolumn har exporten lagt fler nummer, ibland även samma nummer
'Det finns 3 kolumner som påverkas. till slut skall de bara finnas ett nr
' i varje kolumn. Konsekvensen i nuläget blir att man tappar uppgiften
'Måste lösas om sådana fall dyker upp. FN. SKRIVS en signal om tappet
' i debug listan så att man uppmärksammas på problemet
'*****

```

```

Dim wStr As String = ArgStr
Dim wStr2, wStr3 As String
Dim MailWrkText(29) As String           'arbets array; 30 eleem
Dim MailKolText(3) As String           'ett element för varje kolumn
Dim MailKolNbr = New Integer() {31, 33, 35} 'OBS nollräkning/ 0-3

```

```

Const cComma = ","
Const cEmpty = ""
Const cAvgrNew = "///"
Dim wMaxMailCol As Integer = MailKolText.Length - 1
Dim wMaxMailWrk As Integer = MailWrkText.Length - 1

```

Static Dim ImpNbr As Integer

```
Dim ix0, ix1, ix2, ix3, ix10, ix12, ix13
```

```
Dim ColNbr As Integer
```

```

For ix1 = 0 To wMaxMailCol - 1
    MailKolText(ix1) = cEmpty           'initiera
Next

```

```

'*****
'Behandla kolumner med mail värden
'*****

```

```

ix0 = 1
ColNbr = 1

```

```

Do While ix0 < Len(wStr)
    ix1 = InStr(ix0, wStr, cComma, CompareMethod.Text) 'ix1 början på kolumn sök
    ix2 = 0

    If ix1 > 0 Then                                     'ix1 början på kolumn hittad
        ix0 = ix1 + 1
        ColNbr += 1

        For ix3 = 0 To wMaxMailCol - 1
            If ColNbr = MailKolNbr(ix3) Then           'skall kolumnen bearbetas?

```

```

                ix2 = InStr(ix1 + 1, wStr, cComma, CompareMethod.Text) 'ix2 slut på kolumn
                wStr2 = Mid(wStr, ix1 + 1, ix2 - ix1 - 1) 'hämta kolumnnehåll
                If wStr2 <> "" Then
                    MailKolText(ix3) = CleanComCol(wStr2)
                End If
            Exit For
        End If
    Next

Else
    Exit Do

End If

Loop
'*****
'Omfördela vissa kolumner
'det finns bara plats för 3 Mailnr
'vill man ha endast 1 i varje kolumn måste man banta
'använder 30?? arbetskolumner just in case
' ix10 anger kolumn MailKolText
' ix12 anger förekomst #
' ix13 anger kolumn MailWrkText
' Placer ensamma adresser i klol 1 o 2 resten i 3an
'*****
ix10 = 0

Do While ix10 <= wMaxMailCol - 1

    ix13 = 0 'index i MailWrkText tabell (0-29)
    ix12 = 1

    wStr3 = GetComId(MailKolText(ix10), ix12) 'hämta ett av flera nummer

    Do While wStr3 <> cEmpty

        For ix13 = 0 To wMaxMailWrk - 1 ' placera i hjälptabell

            If MailWrkText(ix13) = wStr3 Then 'finns redan
                Exit For
            End If

            If MailWrkText(ix13) = cEmpty Then 'lägg till
                MailWrkText(ix13) = wStr3
                If ix13 > wMaxMailCol Then Debug.Print(ix13.ToString & "/" & wStr3) 'hamnar utanför
                Exit For
            End If

        Next

        ix12 += 1
        wStr3 = GetComId(MailKolText(ix10), ix12) 'hämta ett av flera nummer
    Loop

    ix10 += 1
Loop

'*****
'Hantera överskott
'*****
ix2 = wMaxMailCol - 1 ' plats för överskottet = sista elementet

```

```

For ix1 = ix2 + 1 To wMaxMailWrk - 1
    If MailWrkText(ix1) > cEmpty Then
        If MailWrkText(ix2) > cEmpty Then
            MailWrkText(ix2) = MailWrkText(ix2) & cAvgrNew & MailWrkText(ix1)
        Else
            MailWrkText(ix2) = MailWrkText(ix1)
        End If
    Else
        Exit For
    End If
End If
Next

For ix10 = 0 To wMaxMailCol - 1      'tappar ev fler Mailnr
    MailKolText(ix10) = MailWrkText(ix10)
Next

'*****
'Återlägg skapade värden
'*****
For ix3 = 0 To wMaxMailCol - 1
    If MailKolText(ix3) <> cEmpty Then
        ColNbr = 1
        ix0 = 1
        Do While ColNbr < MailKolNbr(ix3)
            ColNbr += 1
            ix1 = InStr(ix0, wStr, cComma, CompareMethod.Text)      'ix1 början på kolumn
            If ix1 > 0 Then ix0 = ix1 + 1
        Loop
        ix2 = InStr(ix1 + 1, wStr, cComma, CompareMethod.Text)      'ix2 slut på kolumn
        wStr = Mid(wStr, 1, ix1) & MailKolText(ix3) & Mid(wStr, ix2)
    End If
Next

ImpNbr += 1
CvtLine_L = wStr & cComma & ImpNbr.ToString & cComma & Now.Date

End Function
Public Function CvtLine_J(ArgStr As String) As String
'*****
'I vissa av de kolumner där rubrik (ev. ändrats skall innehållet modifieras
'I någon kolumn har exporten lagt fler nummer, ibland äeven samma nummer
'Det finns 4 kolumner som påverkas. till slut skall de bara finnas ett nr
' i varje kolumn. Konsekvensen i nuläget blir att man tappar uppgiften
'Måste lösas om sådana fall dyker upp. FN. SKRIVS en signal om tappet
' i debug listan så att man uppmärksammas på problemet
'*****
Const cComma = ","
Const cEmpty = ""
Static Dim ImpNbr As Integer
Dim wStr As String = ArgStr
Dim wStr2 As String
Dim wStr3 As String
Dim TfnWrkText(29) As String      'arbets array; 30 eleem
Dim TfnKolText(3) As String      'ett element för varje kolumn
Dim TfnKolNbr = New Integer() {35, 37, 39, 41}      'OBS nollräkning/ 0-3
Dim wMaxTfnlCol As Integer = TfnKolText.Length - 1
Dim wMaxTfnWrk As Integer = TfnWrkText.Length - 1

Dim ix0, ix1, ix2, ix3, ix10, ix12, ix13

```

```

Dim ColNbr As Integer
'deklarera array OBS 0 räkning

For ix1 = 0 To wMaxTfnlCol - 1
    TfnKolText(ix1) = cEmpty           'initiera
    'TfnKolNbr(ix1) += Len(cMoreCol)   'justera för tillagda kolumner
Next

'*****
'Behandla kolumner med telefonvärden
'*****
ix0 = 1
ColNbr = 1

Do While ix0 < Len(wStr)
    ix1 = InStr(ix0, wStr, cComma, CompareMethod.Text) 'ix1 början på kolumn sök
    ix2 = 0

    If ix1 > 0 Then 'ix1 början på kolumn hittad
        ix0 = ix1 + 1
        ColNbr += 1

        For ix3 = 0 To wMaxTfnlCol - 1
            If ColNbr = TfnKolNbr(ix3) Then 'skall kolumnen bearbetas?
                ix2 = InStr(ix1 + 1, wStr, cComma, CompareMethod.Text) 'ix2 slut på kolumn
                wStr2 = Mid(wStr, ix1 + 1, ix2 - ix1 - 1) 'hämta kolumninhåll
                If wStr2 <> "" Then
                    TfnKolText(ix3) = CleanComCol(wStr2)
                End If
            End If
        Next

    Else
        Exit Do

    End If

Loop

'*****
'Omfördela vissa kolumner
'det finns bara plats för 4 tfnnr
'vill man ha endast 1 i varje kolumn måste man banta
'använder 30?? arbetskolumner just in case
' ix10 anger kolumn TfnKolText
' ix12 anger förekomst #
' ix13 anger kolumn TfnWrkText
'*****
ix10 = 0

Do While ix10 <= wMaxTfnlCol - 1

    ix13 = 0           'index i TfnWrkText tabell (0-29)
    ix12 = 1

    wStr3 = GetComId(TfnKolText(ix10), ix12) 'hämta ett av flera nummer

    Do While wStr3 <> cEmpty

        For ix13 = 0 To wMaxTfnWrk - 1 ' placera i hjälptabell

```

```

        If TfnWrkText(ix13) = wStr3 Then      'finns redan
            Exit For
        End If

        If TfnWrkText(ix13) = cEmpty Then    'lägg till
            TfnWrkText(ix13) = wStr3
            If ix13 > wMaxTfnlCol Then Debug.Print(ix13.ToString & "/" & wStr3) 'hamnar utanför
            Exit For
        End If

    Next

    ix12 += 1
    wStr3 = GetComId(TfnKolText(ix10), ix12)    'hämta ett av flera nummer
Loop

    ix10 += 1
Loop

For ix10 = 0 To wMaxTfnlCol - 1      'tappar ev fler tfnnr
    TfnKolText(ix10) = TfnWrkText(ix10)
Next

'*****
'Återlägg skapade värden
'*****
For ix3 = 0 To wMaxTfnlCol - 1
    If TfnKolText(ix3) <> cEmpty Then
        ColNbr = 1
        ix0 = 1
        Do While ColNbr < TfnKolNbr(ix3)
            ColNbr += 1
            ix1 = InStr(ix0, wStr, cComma, CompareMethod.Text)    'ix1 början på kolumn
            If ix1 > 0 Then ix0 = ix1 + 1
        Loop
        ix2 = InStr(ix1 + 1, wStr, cComma, CompareMethod.Text)    'ix2 slut på kolumn
        wStr = Mid(wStr, 1, ix1) & TfnKolText(ix3) & Mid(wStr, ix2)
    End If
Next

ImpNbr += 1
CvtLine_J = wStr & cComma & ImpNbr.ToString & cComma & Now.Date

End Function
Public Function GetComId(ArgCol As String, ArgIndex As Integer) As String
'*****
'plocka ett av flera nummer som kan finnas i samma fält
'numren skiljs åt av särskild kod (avgränsare)
'*****

Const cAvgrNew = "///"      'har ersatt gamla värdet (se procedur CleanComCol)
Const cEmpty = ""
Dim ix0, ix1, ix2, ix3, ix9
Dim wStr As String
Dim wStop As Integer
Dim wStart As Integer
Dim wIndex As Integer

If ArgCol = cEmpty Then Return ArgCol

wStr = cEmpty

```

```

' beräkna antal möjliga index baserat på antal avgränsningsvärden
ix0 = 0
ix1 = 1
wIndex = 1
Do While ix1 > 0
    ix1 = 0
    ix1 = InStr(ix0 + 1, ArgCol, cAvgrNew, CompareMethod.Text)      '1a avgr
    If ix1 > 0 Then
        ix0 = ix1
        wIndex += 1
    Else
        Exit Do
    End If
Loop

If wIndex < ArgIndex Then    'fel om begärt index > vad som är möjligt
    Return wStr
End If

ix0 = 0
ix1 = 0
ix2 = 0

For ix3 = 1 To ArgIndex
    ix1 = InStr(ix0 + 1, ArgCol, cAvgrNew, CompareMethod.Text)      '1a avgr
    If ix1 > 0 Then
        ix2 = InStr(ix1 + 1, ArgCol, cAvgrNew, CompareMethod.Text) '2a avgr
        If ix2 > 0 Then
            ix0 = ix2
        End If
        ix9 = ix0
        ix0 = ix1
    End If
Next

'Bestäm position för start/stop för begärt index

ix0 = 0
ix1 = 0
ix2 = 0

If ArgIndex = 1 Then      '1a index
    ix1 = InStr(1, ArgCol, cAvgrNew, CompareMethod.Text)
    wStart = 1
    If ix1 = 0 Then
        wStop = Len(ArgCol)
    Else
        wStop = ix1 - 1
    End If
Else      'övriga index
    ix0 = 0
    For ix3 = 1 To ArgIndex - 1
        ix1 = InStr(ix0 + 1, ArgCol, cAvgrNew, CompareMethod.Text)      '1a avgr
        If ix1 > 0 Then
            ix0 = ix1
            '    If ix3 = ArgIndex Then
            'ix0 = ix1
            ix2 = InStr(ix1 + 1, ArgCol, cAvgrNew, CompareMethod.Text)      '2a avgr
            '    End If
        End If
    Next

```

```

Next

wStart = ix1 + Len(cAvgrNew)
If ix2 = 0 Then
    wStop = Len(ArgCol)
Else
    wStop = ix2 - ix1 - Len(cAvgrNew)
End If

End If

wStr = Mid(ArgCol, wStart, wStop)
GetComId = wStr

End Function
Public Function CleanComCol(ArgCol As String)
    *****
    'städa upp vissa kolumner
    *****
    Const cAvgr = "::::"
    Const cAvgrNew = "///"      'ersatt värde för att bli tydligare för ögat
    Const cSpace = " "         'blanksteg

    Dim wStr As String = ArgCol
    Dim wStr2 As String
    Dim wStr3 As String
    Dim wStr4 As String

    Dim wNbr As String

    Dim ix0, ix1, ix2, ix9
    *****
    ' ta bort blanksteg
    *****
    ix0 = 1
    Do While ix0 < Len(wStr)
        ix1 = InStr(ix0, wStr, cSpace, CompareMethod.Text)
        If ix1 > 0 Then
            ix0 = ix1 + 1
            wStr = Mid(wStr, 1, ix1 - 1) & Mid(wStr, ix0)
        Else
            Exit Do
        End If
    Loop
    *****
    ' byt avgränsare
    *****
    ix0 = 1
    Do While ix0 < Len(wStr)
        ix1 = InStr(ix0, wStr, cAvgr, CompareMethod.Text)
        If ix1 > 0 Then
            ix0 = ix1 + 1
            wStr = Mid(wStr, 1, ix1 - 1) & cAvgrNew & Mid(wStr, ix1 + Len(cAvgr))
        Else
            Exit Do
        End If
    Loop

    *****
    ' behandla bara om mer än ett nummer
    *****

```

```

ix1 = InStr(1, wStr, cAvgrNew, CompareMethod.Text)
If ix1 = 0 Then Return wStr

'*****
' ta bort flera förekomster av samma nummer som det FÖRSTA
'*****
wNbr = Mid(wStr, 1, ix1 - 1)
wStr2 = cAvgrNew & wNbr
ix0 = Len(wNbr)

Do While ix0 < Len(wStr)
    ix1 = InStr(ix0 + 1, wStr, wStr2, CompareMethod.Text)
    If ix1 > 0 Then
        wStr3 = Mid(wStr, 1, ix1 - 1)
        wStr4 = Mid(wStr, ix1 + Len(wStr2))
        wStr = wStr3 & wStr4
    Else
        Exit Do
    End If
Loop

'*****
' ta bort flera förekomster av samma nummer som det NÄSTA i tur
'*****
ix9 = 1 ' anger antal behandlade nr 1 = första i förra slingan

ix0 = 1
ix1 = InStr(ix0, wStr, cAvgrNew, CompareMethod.Text) ' 1a avgränsare
If ix1 = 0 Then Return wStr ' Fanns bara ett nr

ix2 = InStr(ix1 + Len(cAvgrNew), wStr, cAvgrNew, CompareMethod.Text) ' 2a avgränsare
If ix2 = 0 Then Return wStr ' Fanns bara 2 nr och de var olika

ix9 += 1
wNbr = Mid(wStr, ix1 + Len(cAvgrNew), ix2 - ix1 - Len(cAvgrNew))
wStr2 = cAvgrNew & wNbr
ix0 = ix2

Do While ix0 < Len(wStr)
    ix1 = InStr(ix0, wStr, wStr2, CompareMethod.Text)
    If ix1 > 0 Then
        wStr3 = Mid(wStr, 1, ix1 - 1)
        wStr4 = Mid(wStr, ix1 + Len(wStr2))
        wStr = wStr3 & wStr4
    Else
        Exit Do
    End If
Loop

CleanComCol = wStr

End Function

End Class

```