

Change buttons names and add functions

Summary:

On the invoicing form, I have 24 hidden buttons that could be activated by other forms, the name of that form is "FrmBilling", these buttons are named BtnCustom1 to BtnCustom24, they can be setup using 2 forms, the first one is to show the location of these 24 buttons on the billing form (FrmCustomButtons) and the second is to rename the buttons and change the text on these 24 buttons and assign a product or functions to them (FrmCustomButtonsSetup).

FrmBilling:

The screenshot displays the 'FrmBilling' invoicing form. At the top, it shows 'STATION #: 105' and 'ORDER #: ' followed by a green button labeled 'Delete this order'. Below this is a navigation bar with links: Backorder, Products, Customers, Inventory, Internet sites, Tools, History, Salespersons, and Help. The status 'NEW ORDER' is indicated on the right. The main form is divided into several sections: 'CUSTOMER / SOCIETY' with fields for address, phone, and delivery options; 'CASHIER INFO' showing '5 Marie-Josée Larocque'; and a large table for line items with columns for DELETE, EDIT, Order #, Product ID, Description, Quantity, Unit price, Extended price, Discount rate, Discount amount, Line total, and Delivery date. To the right of the table is a vertical list of 24 custom buttons, labeled BtnCustom1 through BtnCustom24. At the bottom, there is a row of icons for various functions: New order, All categories, Add a customer, Print the order, Payment, View Coupon, Issue gift certificate, Apply a coupon, View a certificate, and Search a deposit. The bottom right corner shows a summary of the order: Sub total: 0.00, GST: 0.00, QST: 0.00, Tax 3: 0.00, and a bolded TOTAL: 0.00.

AS you know, in many businesses, they want to be able to customize the billing form to meet their needs, I wish that they could be able to do it themselves, this is the reason behind the 2 forms (FrmCustomButtons and FrmCustomButtonsSetup).

There is 2 steps in this, the first step is to be able to rename the name and the text of the buttons, the second step is to assign a product or a function to them.

FrmCustomButtons:

As you can see, this form is to show the location of the buttons on the billing form, its size is 70% of the original form, however, the 24 custom buttons are functional, and I did put some code on them that open the second form (FrmCustomButtonsSetup)

Here is the code on the first 3 buttons (BtnCustom1 to BtnCustom3)

```
Public Class FrmCustomButtons
    Private ReadOnly SQL As New SQLControl
    Private Sub BtnCustom1_Click(sender As Object, e As EventArgs) Handles BtnCustom1.Click
        Try
            Me.ButtonChosenTB.Text = Me.BtnCustom1.Name.ToString
            Me.ButtonChosenTextTB.Text = Me.BtnCustom1.Text.ToString
            Dim FormCustomButtonsSetup As New FrmCustomButtonsSetup
            FormCustomButtonsSetup.Show()
            FormCustomButtonsSetup.ButtonChosenTB.Text = ButtonChosenTB.Text
            FormCustomButtonsSetup.ButtonChosenTextTB.Text = ButtonChosenTextTB.Text
            If SQL.HasException(True) Then Exit Sub
        Catch ex As Exception
            MsgBox(ex.Message)
            System.IO.File.AppendAllText("C:\AutoCashRegister\Documents\log.txt", ex.ToString &
vbNewLine & vbNewLine)
```

```

End Try
End Sub

Private Sub BtnCustom2_Click(sender As Object, e As EventArgs) Handles BtnCustom2.Click
    Try
        Me.ButtonChosenTB.Text = Me.BtnCustom2.Name.ToString
        Me.ButtonChosenTextTB.Text = Me.BtnCustom2.Text.ToString
        Dim FormCustomButtonsSetup As New FrmCustomButtonsSetup
        FormCustomButtonsSetup.Show()
        FormCustomButtonsSetup.ButtonChosenTB.Text = ButtonChosenTB.Text
        FormCustomButtonsSetup.ButtonChosenTextTB.Text = ButtonChosenTextTB.Text
        If SQL.HasException(True) Then Exit Sub
    Catch ex As Exception
        MsgBox(ex.Message)
        System.IO.File.AppendAllText("C:\AutoCashRegister\Documents\log.txt", ex.ToString &
vbNewLine & vbNewLine)
    End Try
End Sub

Private Sub BtnCustom3_Click(sender As Object, e As EventArgs) Handles BtnCustom3.Click
    Try
        Me.ButtonChosenTB.Text = Me.BtnCustom3.Name.ToString
        Me.ButtonChosenTextTB.Text = Me.BtnCustom3.Text.ToString
        Dim FormCustomButtonsSetup As New FrmCustomButtonsSetup
        FormCustomButtonsSetup.Show()
        FormCustomButtonsSetup.ButtonChosenTB.Text = ButtonChosenTB.Text
        FormCustomButtonsSetup.ButtonChosenTextTB.Text = ButtonChosenTextTB.Text
        If SQL.HasException(True) Then Exit Sub
    Catch ex As Exception
        MsgBox(ex.Message)
        System.IO.File.AppendAllText("C:\AutoCashRegister\Documents\log.txt", ex.ToString &
vbNewLine & vbNewLine)
    End Try
End Sub
End
Class

```

FrmCustomButtonsSetup:

Setup one custom button

☒ Assign to a product
☐ Assign to a function
☐ Print a report

The button above is exactly this size in the invoicing module!

Button width: 100
Button height: 123

Change the button description in the invoicing module:

BtnCustom1 *

 Saved

** Do not exceed the width of the field above*

If a user wants to rename this button to a product name, it will choose from the radio button "Assign to a product" and the code will be in a later document, for the moment I just want to rename these buttons not only on the FrmCustomButtons, but also on the FrmBilling and when they are renamed, then they become visible in the FrmBilling.

This is the code on the FrmCustomButtonsSetup:

```
Public Class FrmCustomButtonsSetup
    Private ReadOnly SQL As New SQLControl
    Private ReadOnly BaseFormClass As New BaseFormClass
    Private Sub FrmCustomButtonsSetup_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Try
            If Application.OpenForms().OfType(Of FrmCustomButtons).Any Then
                ButtonChosenTB.Text = FrmCustomButtons.ButtonChosenTB.Text
                ButtonChosenTextTB.Text = FrmCustomButtons.ButtonChosenTextTB.Text
            End If
            LblButtonSizeInInvoicing.Text = ("The button above is exactly this size in the
invoicing module!")
            If SQL.HasException(True) Then Exit Sub
        Catch ex As Exception
            MsgBox(ex.Message)
            System.IO.File.AppendAllText("C:\AutoCashRegister\Documents\log.txt", ex.ToString &
vbNewLine & vbNewLine)
```

```

End Try
End Sub

Public Function IsFormOpen(ByVal frm As Form) As Boolean
    If Application.OpenForms.OfType(Of Form).Contains(frm) Then
        Return True
    Else
        Return False
    End If
End Function

Private Sub CheckButtonFormat()
    Try
        If ButtonChosenTB.Text = "BtnCustom1" Then
            WidthTB.Text = 100
            HeightTB.Text = 123
        ElseIf ButtonChosenTB.Text = "BtnCustom2" Then
            WidthTB.Text = 100
            HeightTB.Text = 123
        ElseIf ButtonChosenTB.Text = "BtnCustom3" Then
            WidthTB.Text = 100
            HeightTB.Text = 123
        ElseIf ButtonChosenTB.Text = "BtnCustom4" Then
            WidthTB.Text = 100
            HeightTB.Text = 123
        ElseIf ButtonChosenTB.Text = "BtnCustom5" Then
            WidthTB.Text = 100
            HeightTB.Text = 123
        ElseIf ButtonChosenTB.Text = "BtnCustom6" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom7" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom8" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom9" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom10" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom11" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom12" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom13" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom14" Then
            WidthTB.Text = 142
            HeightTB.Text = 36
        ElseIf ButtonChosenTB.Text = "BtnCustom15" Then
            WidthTB.Text = 142
            HeightTB.Text = 100
        ElseIf ButtonChosenTB.Text = "BtnCustom16" Then
            WidthTB.Text = 155
            HeightTB.Text = 31
        ElseIf ButtonChosenTB.Text = "BtnCustom17" Then
            WidthTB.Text = 155
            HeightTB.Text = 31
        ElseIf ButtonChosenTB.Text = "BtnCustom18" Then

```

```

        WidthTB.Text = 155
        HeightTB.Text = 31
    ElseIf ButtonChosenTB.Text = "BtnCustom19" Then
        WidthTB.Text = 155
        HeightTB.Text = 31
    ElseIf ButtonChosenTB.Text = "BtnCustom20" Then
        WidthTB.Text = 71
        HeightTB.Text = 58
    ElseIf ButtonChosenTB.Text = "BtnCustom21" Then
        WidthTB.Text = 71
        HeightTB.Text = 58
    ElseIf ButtonChosenTB.Text = "BtnCustom22" Then
        WidthTB.Text = 71
        HeightTB.Text = 58
    ElseIf ButtonChosenTB.Text = "BtnCustom23" Then
        WidthTB.Text = 71
        HeightTB.Text = 58
    ElseIf ButtonChosenTB.Text = "BtnCustom24" Then
        WidthTB.Text = 71
        HeightTB.Text = 58
    End If
    If SQL.HasException(True) Then Exit Sub
Catch ex As Exception
    MsgBox(ex.Message)
    System.IO.File.AppendAllText("C:\AutoCashRegister\Documents\log.txt", ex.ToString &
vbNewLine & vbNewLine)
End Try
End Sub

Private Sub ButtonChosenTB_TextChanged(sender As Object, e As EventArgs) Handles
ButtonChosenTB.TextChanged
    Try
        CheckButtonFormat()
        Dim BtnCustom0 As New Button
        Me.Controls.Add(BtnCustom0)
        With BtnCustom0
            .BackColor = Color.FromArgb(64, 0, 64)
            .ForeColor = Color.White
            .Text = Me.ButtonChosenTB.Text
            .TextAlign = ContentAlignment.BottomCenter
            .Width = Me.WidthTB.Text
            .Height = Me.HeightTB.Text
            .Font = New Font("Microsoft Sans Serif", 8, FontStyle.Bold)
        End With
        ButtonChosenTextTB.Width = Me.WidthTB.Text
        If SQL.HasException(True) Then Exit Sub
    Catch ex As Exception
        MsgBox(ex.Message)
        System.IO.File.AppendAllText("C:\AutoCashRegister\Documents\log.txt", ex.ToString &
vbNewLine & vbNewLine)
    End Try
End Sub

```