

2009

*Angel's Touch Massage Invoice*



Jeanne N Asberry

5/6/2009

## *TABLE OF CONTENTS*

### **Part One**

Introduction.....	3
Login Form .....	4
Main User Interface .....	4-5
Shortcuts, Default Values, and Data Entry Ranges .....	5-7
Menu Options.....	7-9
Errors.....	9
Reports .....	10
User Backups .....	11
Help.....	11

### **Part Two**

Source Code .....	12-21
Mission Statement.....	22
Program Flowchart and Pseudocode.....	23-24

## *INTRODUCTION*

This documentation package contains documents and demonstrates some of the features of an invoice program for Angel's Touch Massage Therapy.

Angel's Touch Invoice program covers forms and reports to assist the company in tracking calculations of daily financial transactions. This program is designed to be user-friendly for the customer and employee. The program is ADA<sup>1</sup> compliant; it utilizes font and color dialog boxes. Another feature to aid the user is its contextual menu and tooltips.

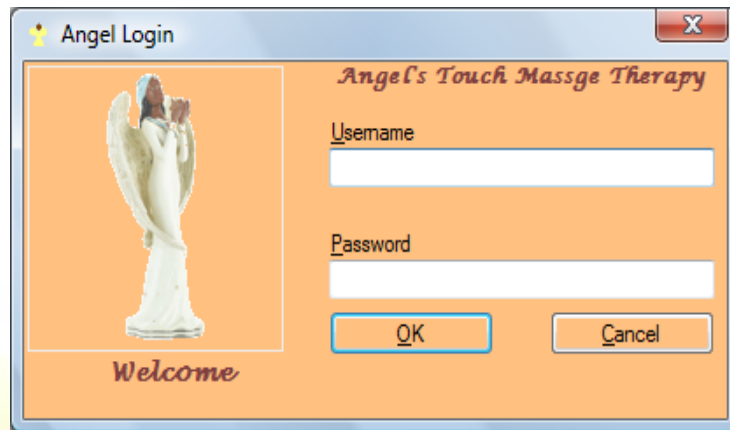
Provided in this documentation package are information on the main database user-interface, out of range dialog boxes, about box, splash screen, login form, source code, program flowchart, and pseudocode. A mission, vision, and goal statement are also included in this package.



---

<sup>1</sup> Americans with Disabilities Act of 1990

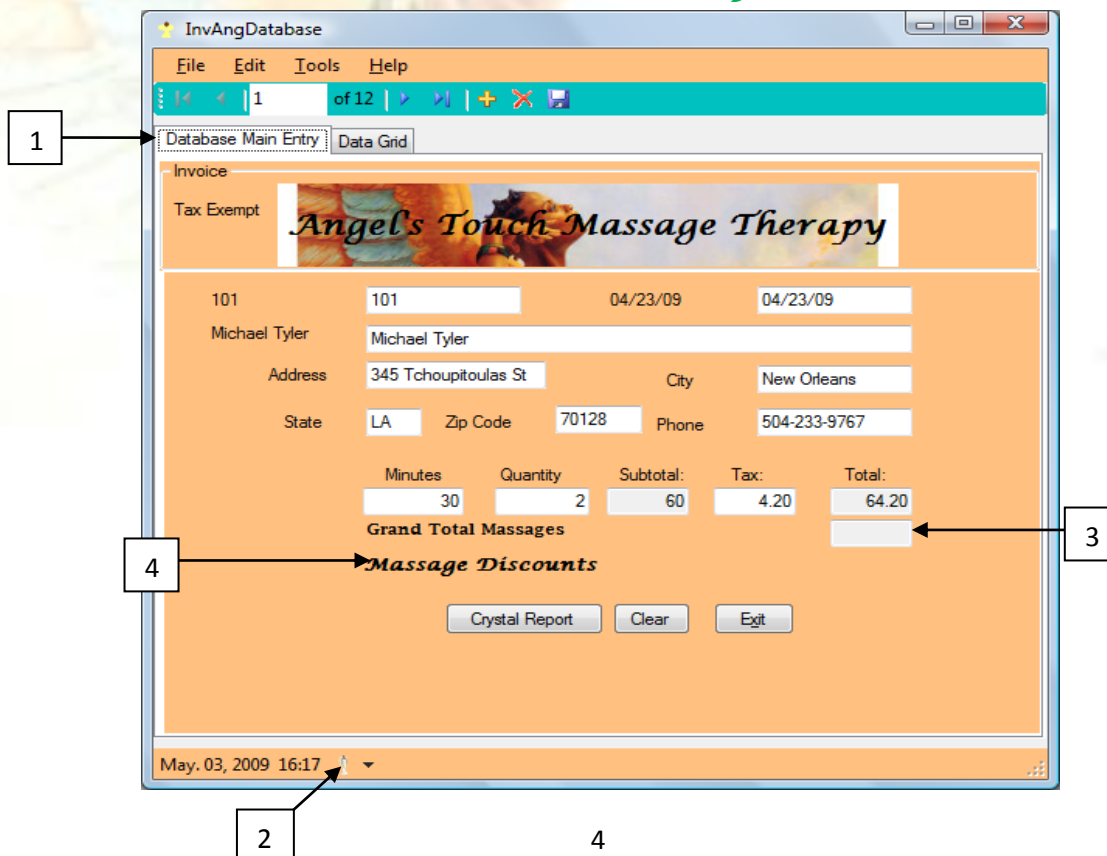
## LOGIN FORM



The image shows a Windows-style dialog box titled "Angel Login". On the left side, there is a vertical image of a white angel statue. Below the image, the word "Welcome" is written in a cursive font. On the right side, the text "Angel's Touch Massage Therapy" is displayed in a decorative font. Below this text are two input fields: "Username" and "Password". At the bottom right, there are two buttons: "OK" and "Cancel".

This login form was so designed that the user would have three attempts to enter the password. If user supplies the correct password, the main database form would open. If the password entered is incorrect, then a dialog appears informing the user to contact the administrator. By utilizing the login feature, this will provide security to the program. Since the company's name is Angel's Touch, a picture box with an angel has been selected for the form.

## Main User Interface



The image shows the main user interface of a program titled "InvAngDatabase". The interface has a menu bar with "File", "Edit", "Tools", and "Help". Below the menu bar is a toolbar with navigation icons and a status bar showing "1 of 12". The main area is divided into several sections:

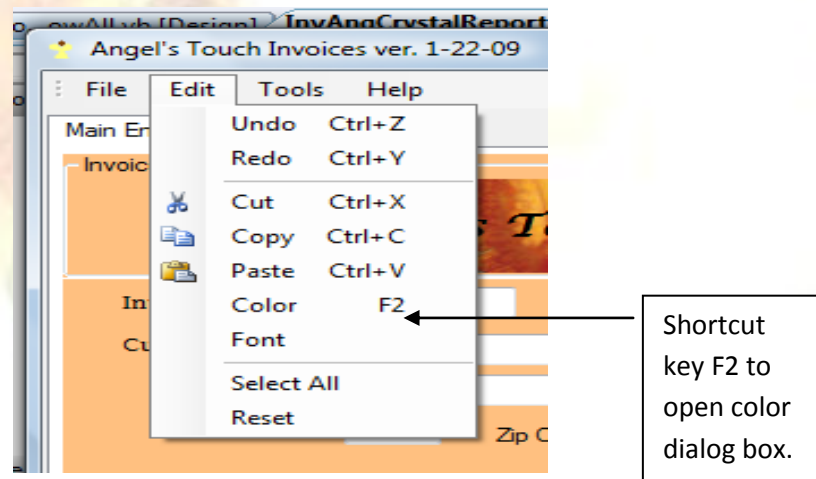
- Database Main Entry:** A tabbed area with "Data Grid" selected.
- Invoice:** A section with a header "Angel's Touch Massage Therapy" and a "Tax Exempt" checkbox.
- Customer Information:** Fields for "101", "Michael Tyler", "Address" (345 Tchoupitoulas St), "City" (New Orleans), "State" (LA), "Zip Code" (70128), and "Phone" (504-233-9767).
- Summary Table:** A table with columns "Minutes", "Quantity", "Subtotal:", "Tax:", and "Total:". The values are: Minutes: 30, Quantity: 2, Subtotal: 60, Tax: 4.20, Total: 64.20.
- Grand Total Messages:** A field with a value of 64.20.
- Message Discounts:** A section with a "Crystal Report" button.
- Buttons:** "Clear" and "Exit" buttons.
- Status Bar:** Shows "May. 03, 2009 16:17".

Numbered callouts (1, 2, 3, 4) point to specific elements: 1 points to the "Database Main Entry" tab, 2 points to the status bar, 3 points to the "Grand Total Messages" field, and 4 points to the "Message Discounts" section.

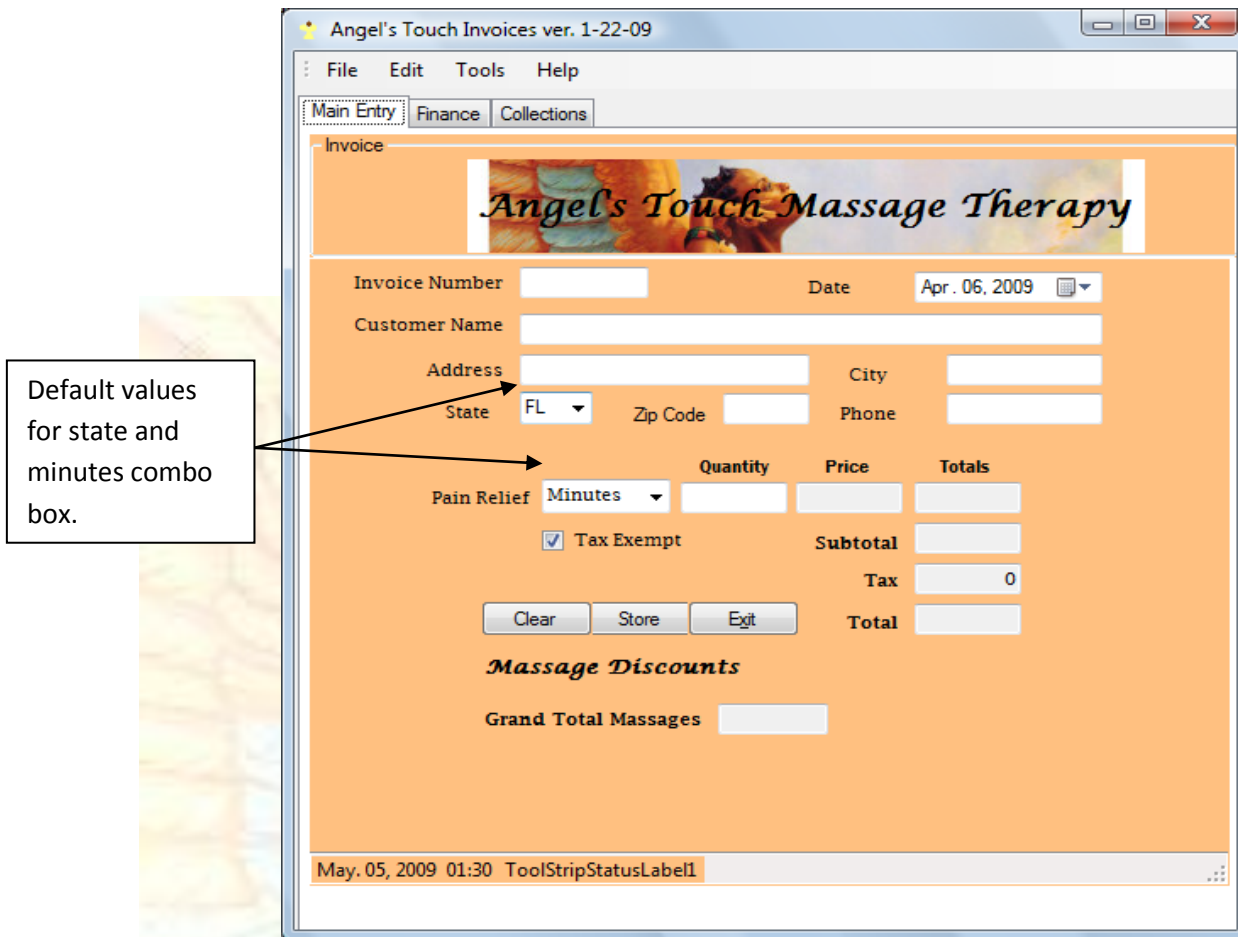
This is the main entry form. First I would like to call attention to the tab control which has to be set first on the form. It contains a collection to add other tabs. The first tab listed here is the Database Main Entry tab. The other tab is the data grid view, it list the records in rows and columns. Second is the status strip item located at the bottom of the screen. The arrow points to a separator of an angel.

Third arrow draws attention to the grand total messages textbox. This textbox displays an accumulator of messages when the clear button is clicked. Box number four points to the label message discounts. A user enters minutes and quantity, then click the [Tab] key through to the total textbox; a message will display based on the dollar amount. The higher the dollar amount the more of a perk the customer will receive.

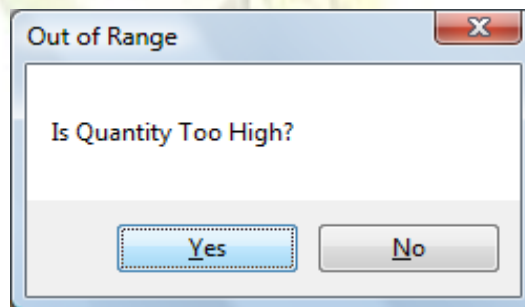
## SHORTCUTS, DEFAULT VALUES, AND DATA ENTRY RANGES



The picture above is a capture of the Edit menu showing standard items. Beside the color dialog box is code to open the color dialog box when the [F2] key is pressed.



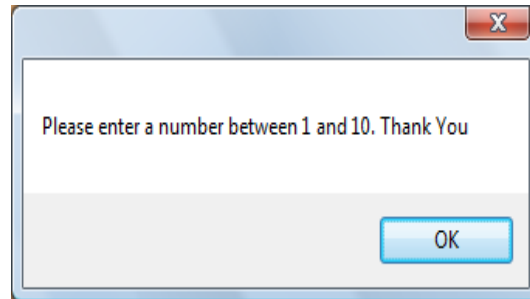
The arrows above point to the default values for the state and minutes combo box. When the user clicks the clear button the state combo box will default to FL and minutes combo box to Minutes. Tax exempt also has a default value of checked, so that tax becomes zero.



This is a dialog box for the Out of Range test when user enters quantity. The question simply verifies if the user is entering a quantity that is over what the

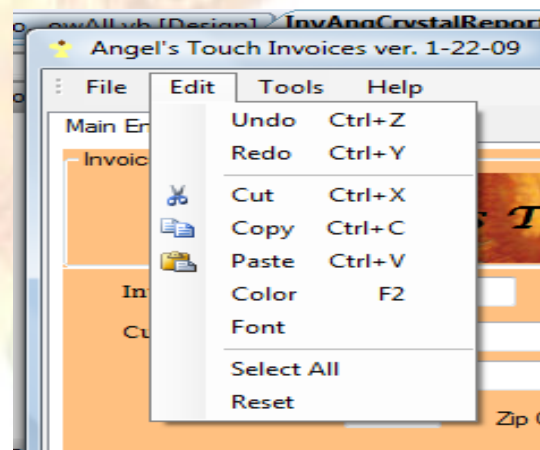


program has established. If the user clicks yes, then the dialog box will disappear and focus will be set to quantity. This will allow the user to reenter the quantity. If the user clicks no, then the program will simply continue with the calculation of minutes and quantity.

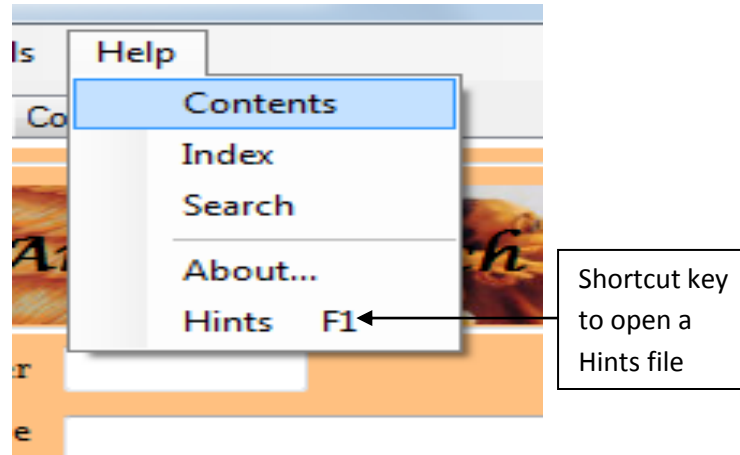


This is another dialog box corresponding to the Out of Range. If the user enters a number higher than 10, this box will appear telling the user to enter a number within the specified range.

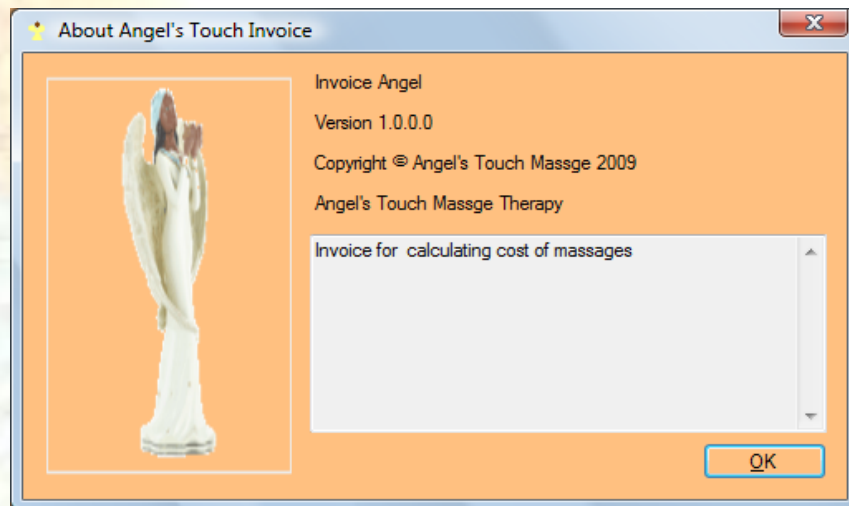
## *MENU OPTIONS*



Listed in this menu are the standard menu items. The newly added items are color, font, and reset. When user clicks on Color or [F2] key, a color dialog box will appear. This will allow the user to change the colors in textboxes. The font box will appear when user clicks on Font allowing the user to change the font style. Reset will allow the user to return the font and color to the defaults.



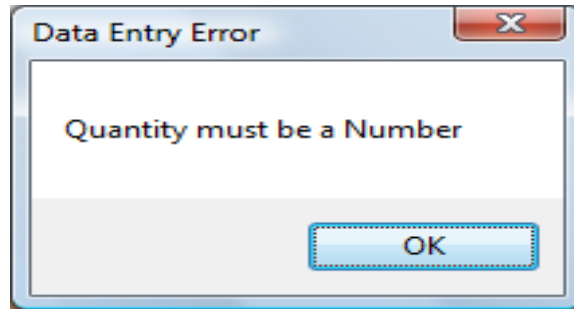
On the Help menu are listed the predefined contents. A text file of Hints is set to open on the [F1] key or when user clicks on the word Hints.



The About box is shown above; it gives information about the program such as the version number, copyright and how the program will be utilized. It can be accessed through the main form by clicking on Help>About menu strip.



## ERRORS




This Data Entry Error for the quantity textbox is one way of catching an error for user inputs. In the event the user forgets to enter a quantity, this dialog box appears telling the user to input a number. When the user clicks the [OK] button, focus is returned to the quantity textbox allowing the user to key a number.

User must enter a number between one and ten. If the user does not the program is designed to prompt them for a quantity. The user will not be able to tab pass this field.

The invoice program has a try catch feature built into the code on an overflow procedure. If user enters a string of long data, the error message will appear telling the user to back away from the computer. This message will show after the Quantity must be a number dialog box above.

Other possible errors which can occur is that the user will forget to enter the minutes causing a zero total. User can also forget to click on tax exempt label at top of screen, so customer will be charged tax. If the save button on the navigation toolbar is not clicked, a record will not be saved. User will have to reenter records. The best way to update records is not through the Visual Basic screen, use Access.

## REPORTS



*Angel's Touch Massage  
Therapy Invoice*

<u>Customer Name</u>	<u>Invoice Date</u>	<u>Minutes</u>	<u>Quantity</u>	<u>Subtotal</u>
Michael Tyler	04/23/09	30	2	\$60.00
Charlie Wilson	09/12/08	45	1	\$45.00
Ivana Trump	01/03/09	60	3	\$180.00
Winnette Lang	09/14/09	120	1	\$120.00
Alvin Mabe	12/1/08	90	2	\$180.00
Makissa Ader	6/3/09	90	2	\$180.00
KeVon Lang	3/6/08	30	6	\$180.00
Steven Joss	11/17/09	75	10	\$750.00
Denzel Washington	12/24/09	120	2	\$240.00
LeShanice Jones	6/27/08	30	1	\$30.00
Kishon Asbery	1/07/09	30	1	\$30.00
Halle Berry	2/10/09	75	2	\$150.00
<b>Record Count</b>				<b>\$2,145.00</b>

Short Date: 5/6/2009      Page 1 of 1      Grand Total

This is a copy of a Crystal Report. The report is for a list of twelve massage records. It contains a record count, grand total, page number, and date. The fields shown are Customer Name, Invoice Date, Minutes, Quantity, and Subtotal.



*Angel's Touch Massage  
Therapy Invoice*

<u>Customer Name</u>	Michael Tyler
<u>Invoice Date</u>	04/23/09
<u>Minutes</u>	30
<u>Quantity</u>	2
<u>Subtotal</u>	60.00

Shown above is a screen capture of a Crystal Report for one record. A copy of the record can be placed in a file to maintain a tracking system. Another copy can be given to the customer as receipt of services.

### *USER BACKUPS*

Backups should be maintained on a hard drive dedicated to only the invoice program and CDs (maybe three different CDs). These three copies need to be distributed to the appropriate persons. When an update such as a customer record being deleted or added takes place; the program should be backed up to hard disk. All invoice records should be accompanied by a paper copy. This will be used as another form of backup.

### *HELP*

Helpful Hints: 1) Be patient with the program. 2) Enter numbers where numbers are appropriate. 3) Use [Tab] key for the calculations as there is no [Calculate] button. 4) If further help is needed, contact the system administrator or help desk.

## SOURCE CODE

Public Class frmInvAngDatabase

'Title: Angel's Touch Massage

'Created:January 12, 2009

'Author: Jeanne N Asberry

'Description: Invoice for Mother's Company

Dim DiscountPrice As Integer

'string array stores state abbreviations

Dim arrayStates As String() = {"AR", "LA", "NC", "SC", "VA", "SC", "TN",  
"KY", "FL", "MS", "GA", "AL" }

Dim colFormtextboxes As New Collection()

Dim ctrlFormtextboxes As Control

Private Sub InvAngBindingNavigatorSaveItem\_Click(ByVal sender As  
System.Object, ByVal e As System.EventArgs) Handles

InvAngBindingNavigatorSaveItem.Click

Me.Validate()

Me.InvAngBindingSource.EndEdit()

Me.TableAdapterManager.UpdateAll(Me.InvAngDataSet)

End Sub

#Region "Inputs"

Private Sub frmInvAngDatabase\_Load(ByVal sender As System.Object, ByVal  
e As System.EventArgs) Handles MyBase.Load

'TODO: This line of code loads data into the 'InvAngDataSet.InvAng' table.  
You can move, or remove it, as needed.

Me.InvAngTableAdapter.Fill(Me.InvAngDataSet.InvAng)

Timer1.Enabled = True

MenuStrip1.Show() 'command to make menu visible

Array.Sort(arrayStates) 'sort an array

'build collection of textboxes

With colFormtextboxes

.Add(txtDbInvoiceNumber)

.Add(txtDbInvoiceDate)

.Add(txtDbCustomerName)

.Add(txtDbAddress)

.Add(txtDbCity)

```

.Add(txtDbState)
.Add(txtDbZipCode)
.Add(txtDbPhone)
.Add(txtDbMinutes)
.Add(txtDbQuantity)
.Add(txtDbTotal)
.Add(txtDbSubtotal)
.Add(txtDbTax)
.Add(txtDbTotal)
.Add(txtDbGrandTotal)

```

End With

End Sub

' If you check for non-numeric you can check as they enter each character  
' so they see their error immediately

```

Private Sub txtDbQuantity_Leave(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtDbQuantity.Leave
    If Not IsNumeric(txtDbQuantity.Text) Then
        MessageBox.Show("Quantity must be a Number", "Data Entry Error",
MessageBoxButtons.OK)
        txtDbQuantity.Focus()
    Else
        Dim Response As Windows.Forms.DialogResult
        If Cdbl(txtDbQuantity.Text) > 10 Then ' Out of Range test for Quantity
            Response = MessageBox.Show("Is Quantity Too High?", "Out of
Range", MessageBoxButtons.YesNo)
            If Response = Windows.Forms.DialogResult.Yes Then
                MessageBox.Show("Please enter a number between 1 and 10. Thank
You")
                txtDbQuantity.Focus()
            Exit Sub
            End If
        End If
    End If
    Call Calc()
End Sub
Private Sub txtDbTotal_Leave(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtDbTotal.Leave
    txtDbTotal.Text = (txtDbTotal.Text)
End Sub

```



```

Private Sub txtDbTotal_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtDbTotal.LostFocus
    DiscountPrice = CInt(CDec(Val(txtDbTotal.Text)))
    Select Case DiscountPrice ' If more than one match, only 1st is executed
        Case Is > 450 ' Constants are separated by commas
            lblDbMessageDiscounts.Text = "50% Discount" :
lblDbMessageDiscounts.ForeColor = Color.DarkGreen
        Case 300 To 449 ' Need Is with relational operator such as >
            lblDbMessageDiscounts.Text = "40% Discount" :
lblDbMessageDiscounts.ForeColor = Color.Firebrick
        Case 225 To 299 ' Use To with Ranges
            lblDbMessageDiscounts.Text = "Entitled to one free message" :
lblDbMessageDiscounts.ForeColor = Color.Brown
        Case 150 To 224
            lblDbMessageDiscounts.Text = "Receive 10 minutes more at regular
price of message" : lblDbMessageDiscounts.ForeColor = Color.Goldenrod
        Case Else ' Execute if all others not matched
            lblDbMessageDiscounts.Text = "Sorry, you get no perks" :
lblDbMessageDiscounts.ForeColor = Color.Black
    End Select
End Sub
#End Region

```

```

#Region "Calculations"

```

```

Private Sub Calc()
    Const TaxRate As Decimal = CDec(0.07)
    'assign user input values User(calculations)
    Try
        txtDbSubtotal.Text = CStr(Val(txtDbMinutes.Text) *
Val(txtDbQuantity.Text)) 'calculation for Subtotal
        txtDbSubtotal.Text = FormatNumber(txtDbSubtotal.Text, 2)
'formatting subtotal to two decimal places with dollar sign
        txtDbTax.Text = CStr(Val(txtDbSubtotal.Text) * TaxRate)
'calculation for tax
        txtDbTax.Text = FormatNumber(txtDbTax.Text, 2)
        Debug.WriteLine("txtDbTax=" & txtDbTax.Text)

        txtDbTotal.Text = CStr(Val(txtDbSubtotal.Text) + Val(txtDbTax.Text))
'calculation for overall total
    End Try
End Sub

```



```

        txtdbTotal.Text = FormatNumber(txtdbTotal.Text, 2)
'formatting final total to two decimal places
        Debug.WriteLine("txtdbTotal=" & txtdbTotal.Text)
    Catch ex As OverflowException
        MsgBox("Back away from the computer")
    End Try
End Sub

```

```

Private Sub lblTaxExempt_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles lblTaxExempt.Click
    If lblTaxExempt.Visible = True Then 'calculation for tax exempt box
        txtdbTax.Text = CStr(0)
        txtdbTotal.Text = txtdbSubtotal.Text
    Else
        Call Calc()
    End If
End Sub
#End Region

```

```

#Region "Buttons"

```

```

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnClear.Click
    'remove text displayed in Textbox or Combobox
    txtdbGrandTotal.Text = CStr(Val(txtdbGrandTotal.Text) +
Val(txtdbTotal.Text)) 'accumulator for all total messages
    txtdbInvoiceNumber.Clear()
    txtdbCustomerName.Clear()
    txtdbAddress.Clear()
    txtdbCity.Clear()
    txtdbState.Text = "FL"
    txtdbZipCode.Clear()
    txtdbPhone.Clear()
    txtdbMinutes.Text = "Minutes"
    txtdbQuantity.Clear()
    txtdbSubtotal.Clear()
    txtdbTax.Clear()
    txtdbTotal.Clear()
    txtdbInvoiceNumber.Focus()
    'remove assigned values from labels

```

```
lblDbInvoiceNumber.Text = "Invoice Number"  
lblDbInvoiceDate.Text = "Date"  
lblDbCustomerName.Text = "Customer Name"  
lblDbAddress.Text = "Address"  
lblDbCity.Text = "City"
```

```
lblDbState.Text = "State"  
lblDbZipCode.Text = "Zip Code"  
lblDbPhone.Text = "Phone"  
lblDbQuantity.Text = "Quantity"
```

```
lblDbSubtotal.Text = "Subtotal"  
lblDbTax.Text = "Tax"  
lblDbTotal.Text = "Total"  
lblDbGrandTotal.Text = "Grand Total Messages"
```

```
End Sub
```

```
Private Sub btnDbExit_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles btnDbExit.Click
```

```
End ' code to program Exit button
```

```
End Sub
```

```
Private Sub btnDbCrystalReport_Click(ByVal sender As System.Object, ByVal  
e As System.EventArgs) Handles btnDbCrystalReport.Click
```

```
InvAngCrystalReportShowAll.Show()
```

```
End Sub
```

```
#End Region
```

```
#Region "Menu"
```

```
Private Sub dbColorToolStripMenuItem1_Click(ByVal sender As  
System.Object, ByVal e As System.EventArgs) Handles  
dbColorToolStripMenuItem1.Click
```

```
'changes the color of font in a textboxes
```

```
ColorDialog1.ShowDialog()
```

```
For Each ctrlFormtextbox In colFormtextboxes
```

```
    NewToolStripMenuItem.ForeColor = ColorDialog1.Color
```

```
Next
```

```
txtDbInvoiceNumber.ForeColor = ColorDialog1.Color
```

```
txtDbCustomerName.ForeColor = ColorDialog1.Color
```

```
txtDbInvoiceDate.ForeColor = ColorDialog1.Color
```

```
txtDbAddress.ForeColor = ColorDialog1.Color
```

```
txtDbCity.ForeColor = ColorDialog1.Color
txtDbState.ForeColor = ColorDialog1.Color
txtDbZipCode.ForeColor = ColorDialog1.Color
txtDbPhone.ForeColor = ColorDialog1.Color
txtDbMinutes.ForeColor = ColorDialog1.Color
txtDbQuantity.ForeColor = ColorDialog1.Color
txtDbSubtotal.ForeColor = ColorDialog1.Color
txtDbTax.ForeColor = ColorDialog1.Color
txtDbTotal.ForeColor = ColorDialog1.Color
txtDbGrandTotal.ForeColor = ColorDialog1.Color
```

End Sub

```
Private Sub dbFontToolStripMenuItem1_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles dbFontToolStripMenuItem1.Click
```

'changes the size of font in the textboxes

```
FontDialog1.ShowDialog()
```

```
For Each ctrlFormtextbox In colFormtextboxes
```

```
    NewToolStripMenuItem.Font = FontDialog1.Font
```

```
Next
```

```
txtDbInvoiceNumber.Font = FontDialog1.Font
```

```
txtDbInvoiceDate.Font = FontDialog1.Font
```

```
txtDbCustomerName.Font = FontDialog1.Font
```

```
txtDbAddress.Font = FontDialog1.Font
```

```
txtDbCity.Font = FontDialog1.Font
```

```
txtDbState.Font = FontDialog1.Font
```

```
txtDbZipCode.Font = FontDialog1.Font
```

```
txtDbPhone.Font = FontDialog1.Font
```

```
txtDbMinutes.Font = FontDialog1.Font
```

```
txtDbQuantity.Font = FontDialog1.Font
```

```
txtDbSubtotal.Font = FontDialog1.Font
```

```
txtDbTax.Font = FontDialog1.Font
```

```
txtDbTotal.Font = FontDialog1.Font
```

```
txtDbGrandTotal.Font = FontDialog1.Font
```

End Sub

```
Private Sub dbAboutToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
dbAboutToolStripMenuItem.Click
```

'displays information about program

```
frmInvAboutBox.Show()
```

End Sub

Private Sub dbHintsToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles dbHintsToolStripMenuItem.Click

Process.Start("Notepad", "/Hints.text")

'opens a file for helpful hint in Notepad

End Sub

Private Sub dbResetToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles dbResetToolStripMenuItem.Click

'resets a font and color to original settings

Dim myfont As New Font("Georgia", 8, FontStyle.Regular)

txtdbInvoiceNumber.Font = myfont

txtdbInvoiceDate.Font = myfont

txtdbCustomerName.Font = myfont

txtdbAddress.Font = myfont

txtdbCity.Font = myfont

txtdbState.Font = myfont

txtdbZipCode.Font = myfont

txtdbPhone.Font = myfont

txtdbMinutes.Font = myfont

txtdbQuantity.Font = myfont

txtdbSubtotal.Font = myfont

txtdbTax.Font = myfont

txtdbTotal.Font = myfont

txtdbGrandTotal.Font = myfont

txtdbInvoiceNumber.ForeColor = Color.Black

txtdbInvoiceDate.ForeColor = Color.Black

txtdbCustomerName.ForeColor = Color.Black

txtdbAddress.ForeColor = Color.Black

txtdbCity.ForeColor = Color.Black

txtdbState.ForeColor = Color.Black

txtdbZipCode.ForeColor = Color.Black

txtdbPhone.ForeColor = Color.Black

txtdbMinutes.ForeColor = Color.Black

txtdbQuantity.ForeColor = Color.Black

txtdbSubtotal.ForeColor = Color.Black

txtdbTax.ForeColor = Color.Black

txtdbTotal.ForeColor = Color.Black

txtdbGrandTotal.ForeColor = Color.Black

End Sub

#End Region

```
Private Sub Timer1_Tick(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
    ssdbclock.Text = Format(Now(), "MMM. dd, yyyy HH:mm")
'format the time and date in status strip
End Sub
```

#Region "Context"

```
Private Sub FontToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles FontToolStripMenuItem.Click
'changes the size of font in the textboxes
FontDialog1.ShowDialog()
For Each ctrlFormtextbox In colFormtextboxes
    NewToolStripMenuItem.Font = FontDialog1.Font
Next
txtdbInvoiceNumber.Font = FontDialog1.Font
txtdbInvoiceDate.Font = FontDialog1.Font
txtdbCustomerName.Font = FontDialog1.Font
txtdbAddress.Font = FontDialog1.Font
txtdbCity.Font = FontDialog1.Font
txtdbState.Font = FontDialog1.Font
txtdbZipCode.Font = FontDialog1.Font
txtdbPhone.Font = FontDialog1.Font
txtdbMinutes.Font = FontDialog1.Font
txtdbQuantity.Font = FontDialog1.Font
txtdbSubtotal.Font = FontDialog1.Font
txtdbTax.Font = FontDialog1.Font
txtdbTotal.Font = FontDialog1.Font
txtdbGrandTotal.Font = FontDialog1.Font
End Sub
```

```
Private Sub ColorToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ColorToolStripMenuItem.Click
'changes the color of font in a textboxes
ColorDialog1.ShowDialog()
For Each ctrlFormtextbox In colFormtextboxes
    NewToolStripMenuItem.ForeColor = ColorDialog1.Color
Next
txtdbInvoiceNumber.ForeColor = ColorDialog1.Color
```



```
txtDbInvoiceDate.ForeColor = ColorDialog1.Color
txtDbCustomerName.ForeColor = ColorDialog1.Color
txtDbAddress.ForeColor = ColorDialog1.Color
txtDbCity.ForeColor = ColorDialog1.Color
txtDbState.ForeColor = ColorDialog1.Color
txtDbZipCode.ForeColor = ColorDialog1.Color
txtDbPhone.ForeColor = ColorDialog1.Color
txtDbMinutes.ForeColor = ColorDialog1.Color
txtDbQuantity.ForeColor = ColorDialog1.Color
txtDbSubtotal.ForeColor = ColorDialog1.Color
txtDbTax.ForeColor = ColorDialog1.Color
txtDbTotal.ForeColor = ColorDialog1.Color
txtDbGrandTotal.ForeColor = ColorDialog1.Color
```

**End Sub**

**Private Sub** ResetToolStripMenuItem\_Click(**ByVal** sender **As** System.Object, **ByVal** e **As** System.EventArgs) **Handles** ResetToolStripMenuItem.Click

'resets a font and color to original settings

**Dim** myfont **As New** Font("Georgia", 8, FontStyle.Regular)

```
txtDbInvoiceNumber.Font = myfont
```

```
txtDbCustomerName.Font = myfont
```

```
txtDbAddress.Font = myfont
```

```
txtDbCity.Font = myfont
```

```
txtDbState.Font = myfont
```

```
txtDbZipCode.Font = myfont
```

```
txtDbPhone.Font = myfont
```

```
txtDbMinutes.Font = myfont
```

```
txtDbQuantity.Font = myfont
```

```
txtDbSubtotal.Font = myfont
```

```
txtDbTax.Font = myfont
```

```
txtDbTotal.Font = myfont
```

```
txtDbGrandTotal.Font = myfont
```

```
txtDbInvoiceNumber.ForeColor = Color.Black
```

```
txtDbCustomerName.ForeColor = Color.Black
```

```
txtDbAddress.ForeColor = Color.Black
```

```
txtDbCity.ForeColor = Color.Black
```

```
txtDbState.ForeColor = Color.Black
```

```
txtDbZipCode.ForeColor = Color.Black
```

```
txtDbPhone.ForeColor = Color.Black
```



```
txtdbMinutes.ForeColor = Color.Black  
txtdbQuantity.ForeColor = Color.Black  
txtdbSubtotal.ForeColor = Color.Black  
txtdbTax.ForeColor = Color.Black  
txtdbTotal.ForeColor = Color.Black  
txtdbGrandTotal.ForeColor = Color.Black
```

```
End Sub
```

```
#End Region
```

```
End Class
```



## MISSION, VISION, AND GOAL

### Mission

*The purpose of Angel's Touch Massage Company is to promote its family-oriented, energetic, ethical and caring environment for everyone young and older. We believe in customer service and the customer's relaxation needs are our number one priority. Our focus is on the body to facilitate holistic healing.*

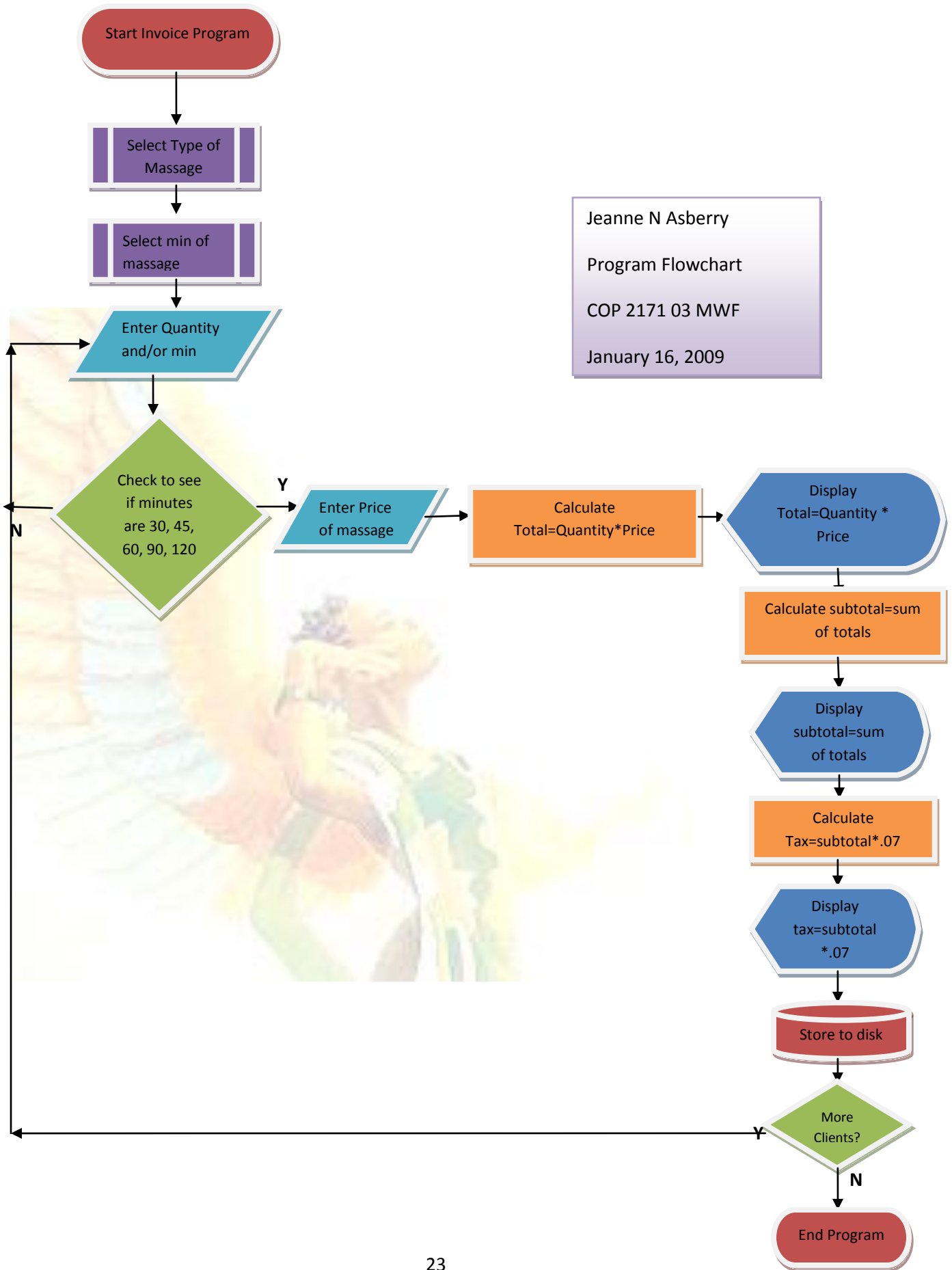
### Vision

*Angel's Touch vision is to gain website presence to reach a larger range of customers. We want to promote health, healing, and diversified medicine.*

### Goal

*The goal of Angel's Touch is to build a trusting company where customers feel comfortable, healthy, and welcome.*





Jeanne N Asberry  
 Program Flowchart  
 COP 2171 03 MWF  
 January 16, 2009

## Pseudocode

**Begin Invoice Program**

**Select Type of Massage**

**Select number of minutes 30, 45, 60, and 90**

**Enter other minutes and/or quantity**

**Check to see that minutes are valid. If minutes are not 30, 60, 45, 90 or 120 then reenter.**

**If yes then enter price.**

**Calculate client charges. Total: Multiply quantity times price. Check that client charges are valid.**

**Display total to monitor (add: Printout Client Charges)**

**Calculate client Subtotal: Add all totals**

**Display Subtotal to monitor (add: Printout of subtotals)**

**Calculate client Tax: Multiply subtotal times 7%**

**Display Tax to monitor**

**Save to Disk**

**Are there more client invoices to process? If yes Enter new quantity. If no end the program**

**End Invoice Program**