

Rick's Auto Repair Program – RAuto .exe

Problem Statement:

Purpose - To provide Rick's Auto Repair the ability to calculate customer repair costs.

Output:

Display –

1. Customer Name
2. Labor Cost
3. Parts Cost (including tax)
4. Total Cost

Input:

Keyboard –

1. Customer Name
2. Hours Worked
3. Cost of Parts and Supplies

User Interface – Form Layout with sample input and output:

The screenshot shows a Windows-style application window titled "Rick's Auto Repair". The interface includes three input fields: "Please Enter Customer Name" (containing "Chuck Rion Chevrolet and Jeep Dealership"), "Please Enter Hours Worked" (containing "3.5"), and "Please Enter the Cost of Parts and Supplies" (containing "23.55"). There are three buttons: "Calc Cost", "Clear Data", and "Quit". Below the input fields is a text area displaying the following bill:

```
*** Rick's Automobile Repair Bill ***
Customer:   Chuck Rion Chevrolet and Jeep Dealership
Labor Cost: $122.50
Parts Cost:  $24.73
=====
Total Cost:  $147.23

Come Again -- Chuck Rion Chevrolet and Jeep Dealership

*** We Appreciate Your Business ***
```

Rules:

Customer Name shall be no longer than 44 characters.
Hours Worked shall be a non-negative number.
Program to remain running until operator terminates.
Program shall ensure user really wants to quit.

Formulas:

Labor Rate is \$35.00 per hour.
Tax Rate is 5%.
Labor Cost = Hours Worked * Labor Rate.
Parts Cost (including tax) = Cost of Parts + (Cost of Parts * .05)
Total Cost = Parts Cost (including tax) + Labor Cost

Data Dictionary:

Constant Names:	Type	Description
SNG_LABOR_RATE	Number	The current hourly wage rate
SNG_TAX_RATE	Number	The current tax rate

Variable Names:	Type	Description
intAnswerToQuit	Number	Result of answer to message box [6 is YES] [7 is NO]
sngHoursWorked	Number	Hours of work expended for customer
sngLaborCost	Number	Labor cost charged to customer
sngPartsCost	Number	Parts and Supplies cost charged to customer
sngPartsW_Tax	Number	Cost of parts and supplies charged to cust. with tax added
sngTotalCost	Number	Total cost of parts, labor, and tax being charged to customer
strCustomer	40 bytes String	Customer name (maximum of 40 characters)

File Names:	Description
RAuto.exe	Executable program file (Rick's Auto Repair program)
RAutoDocument.doc	Documentation file
RAutoMainForm.frm	Visual Basic Source code file (startup form – only form in project)
RAutoProject.vbp	Visual Basic Project file
RAutoProject.vbw	Visual Basic Project workspace

Sample Data:

Input: Customer Name: **John Doe**
Input: Hours Worked: **3.5**
Input: Cost of Parts and Supplies: **23.55**
Output: Labor Cost: **122.50**
Output: Parts Cost: **24.73**
Output: Total Cost - Customer Bill: **147.23**

Pseudo-code for Rick's Auto Repair:

Begin program (Get Input of Customer Data by User)

Emphasize instruction to user for text entry of name

Get customer name

Emphasize instruction to user for numeric entry of hours

Get number of hours worked

Emphasize instruction to user for numeric entry of parts and supplies cost

Get cost of parts and supplies

Sub Button Calculate on Click (Main Module – click event to calculate data)

Clear Data Display Area
Assign Customer Name to variable
Assign Hours Worked to variable
Assign Parts and Supplies Cost to variable
Calculate Labor Cost
Calculate Parts Cost (including tax)
Calculate Total Cost
Display result of calculations to Data Display Area

End Sub

Sub Button Clear on Click (click event to clear text boxes and data printout area)

Set Customer Name Text box to NULL
Set Hours Worked Text box to NULL
Set Parts and Supplies Text box to NULL
Clear Data Display area

End Sub

Sub Button Quit on Click (click event to end program)

Check if User really wants to quit
If so then **End program**
Else SetFocus to Customer Name Text box

End Sub

Form Data:

```
VERSION 5.00
Begin VB.Form frmRAutoMain
    Caption           = "Rick's Auto Repair"
    ClientHeight      = 4905
    ClientLeft        = 3615
    ClientTop         = 3990
    ClientWidth       = 7050
    LinkTopic         = "Form1"
    ScaleHeight       = 4905
    ScaleWidth        = 7050
Begin VB.PictureBox picDisplayData
    AutoRedraw        = -1 'True
BeginProperty Font
    Name              = "MS Sans Serif"
    Size              = 8.25
    Charset           = 0
    Weight            = 700
    Underline         = 0 'False
    Italic            = 0 'False
    Strikethrough     = 0 'False
EndProperty
    Height            = 2535
    Left              = 120
    ScaleHeight       = 2475
    ScaleWidth        = 6795
    TabIndex          = 6
    TabStop           = 0 'False
    Top               = 2160
    Width             = 6855
End
Begin VB.CommandButton cmdQuitProgram
    Caption           = "&Quit"
    Height            = 375
    Left              = 5160
    TabIndex          = 5
    TabStop           = 0 'False
    Top               = 1560
    Width             = 1215
End
Begin VB.CommandButton cmdClearData
    Caption           = "Cl&ear Data"
    Height            = 375
    Left              = 3840
    TabIndex          = 4
    Top               = 1560
    Width             = 1215
End
Begin VB.CommandButton cmdCalcTotalCost
    Caption           = "&Calc Cost"
    Height            = 375
    Left              = 4440
    TabIndex          = 3
    Top               = 960
End
```

```
        Width          = 1215
    End
    Begin VB.TextBox txtGetPartsCost
        BeginProperty Font
            Name          = "MS Sans Serif"
            Size          = 8.25
            Charset       = 0
            Weight        = 700
            Underline     = 0 'False
            Italic        = 0 'False
            Strikethrough = 0 'False
        EndProperty
        Height          = 375
        Left           = 1920
        TabIndex       = 2
        Top            = 1440
        Width          = 975
    End
    Begin VB.TextBox txtGetHoursOfLabor
        BeginProperty Font
            Name          = "MS Sans Serif"
            Size          = 8.25
            Charset       = 0
            Weight        = 700
            Underline     = 0 'False
            Italic        = 0 'False
            Strikethrough = 0 'False
        EndProperty
        Height          = 375
        Left           = 1920
        TabIndex       = 1
        Top            = 840
        Width          = 735
    End
    Begin VB.TextBox txtGetCustName
        BeginProperty Font
            Name          = "MS Sans Serif"
            Size          = 8.25
            Charset       = 0
            Weight        = 700
            Underline     = 0 'False
            Italic        = 0 'False
            Strikethrough = 0 'False
        EndProperty
        Height          = 375
        Left           = 1920
        TabIndex       = 0
        Top            = 240
        Width          = 5055
    End
    Begin VB.Label lblSayDATA
        Caption          = "Please Enter the Cost of Parts and Supplies"
        Height          = 735
        Index           = 2
        Left           = 120
        TabIndex       = 9
    End
```

```
        Top           = 1440
        Width        = 1695
    End
    Begin VB.Label lblSayDATA
        Caption       = "Please Enter Hours Worked"
        Height        = 495
        Index         = 1
        Left          = 120
        TabIndex      = 8
        Top           = 720
        Width         = 1455
    End
    Begin VB.Label lblSayDATA
        Caption       = "Please Enter Customer Name"
        Height        = 495
        Index         = 0
        Left          = 120
        TabIndex      = 7
        Top           = 120
        Width         = 1455
    End
End
Attribute VB_Name = "frmRAutoMain"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
'*****
'*****
' Rick Shuster
' CST-431 Project 1 12 June 1999
' Professor Rion
' Project Purpose: Plan and implement introductory programming
'                   assignment using Visual Basic 6.0
' Project: Auto Repair shop cost calculator
' Files: RAutoMain.frm
'        RAutoProject.vbp
'*****
'*****

Option Explicit
Const SNG_LABOR_RATE As Single = 35
Const SNG_TAX_RATE As Single = 0.05

Private Sub cmdCalcTotalCost_Click()
'*****
' Main module- Assigns values from text boxes to variables
' Calculates Customer Bill and displays results in Picture box
'*****
'Dimension Variables
    Dim strCustomer As String * 44
    Dim sngHoursWorked As Single
    Dim sngPartsCost As Single
    Dim sngLaborCost As Single
    Dim sngPartsW_Tax As Single
    Dim sngTotalCost As Single
```

```
'Assign Data to Variables
    strCustomer = Left$(txtGetCustName, 44)
    sngHoursWorked = Val(txtGetHoursOfLabor)
    sngPartsCost = Val(txtGetPartsCost)

'Calculate Result
    sngLaborCost = sngHoursWorked * SNG_LABOR_RATE
    sngPartsW_Tax = sngPartsCost + (sngPartsCost * SNG_TAX_RATE)
    sngTotalCost = sngLaborCost + sngPartsW_Tax

'Display Results
    picDisplayData.Cls
    picDisplayData.Print Tab(20); "*** Rick's Automobile Repair Bill ***"
    picDisplayData.Print
    picDisplayData.Print "Customer: ", strCustomer
    picDisplayData.Print
    picDisplayData.Print "Labor Cost:", Format$(sngLaborCost, "Currency")
    picDisplayData.Print "Parts Cost:", Format$(sngPartsW_Tax, "Currency")
    picDisplayData.Print , "======"
    picDisplayData.Print "Total Cost:", Format$(sngTotalCost, "Currency")
    picDisplayData.Print
    picDisplayData.Print Tab(4); "Come Again -- " & strCustomer
    picDisplayData.Print
    picDisplayData.Print Tab(20); "*** We Appreciate Your Business ***"
    txtGetCustName.SetFocus
End Sub
```

```
Private Sub cmdCalcTotalCost_GotFocus()
'*****
'Bold and UN-Bold user direction labels
'*****
    lblSayDATA(0).FontBold = False
    lblSayDATA(1).FontBold = False
    lblSayDATA(2).FontBold = False
End Sub
```

```
Private Sub cmdClearData_Click()
'*****
'Clear data from Input and Output Data area for next calculation
' Then sets focus to customer name text entry box
'*****
    txtGetCustName = ""
    txtGetHoursOfLabor = ""
    txtGetPartsCost = ""
    picDisplayData.Cls
    txtGetCustName.SetFocus
End Sub
```

```
Private Sub cmdClearData_GotFocus()
'*****
'Bold and UN-Bold user direction labels
'*****
```

```
    lblSayDATA(0).FontBold = False
    lblSayDATA(1).FontBold = False
    lblSayDATA(2).FontBold = False
End Sub
```

```
Private Sub cmdQuitProgram_Click()
'*****
' Quit Rick's Auto Repair Application
' Gives user choice to really quit or go back to program
'*****
    Dim intAnswerToQuit As Integer
    intAnswerToQuit = MsgBox("Are you sure you want to quit", 4)
    If intAnswerToQuit = 6 Then
        End
    End If
End Sub
```

```
Private Sub cmdQuitProgram_GotFocus()
'*****
'Bold and UN-Bold user direction labels
'*****
    lblSayDATA(0).FontBold = False
    lblSayDATA(1).FontBold = False
    lblSayDATA(2).FontBold = False
End Sub
```

```
Private Sub txtGetCustName_GotFocus()
'*****
'Bold and UN-Bold user direction labels
'*****
    lblSayDATA(0).FontBold = True
    lblSayDATA(1).FontBold = False
    lblSayDATA(2).FontBold = False
End Sub
```

```
Private Sub txtGetHoursOfLabor_GotFocus()
'*****
'Bold and UN-Bold user direction labels
'*****
    lblSayDATA(0).FontBold = False
    lblSayDATA(1).FontBold = True
    lblSayDATA(2).FontBold = False
End Sub
```

```
Private Sub txtGetHoursOfLabor_LostFocus()
'*****
'Ensure user does NOT enter a negative number by making it positive
'*****
    Dim intStrLen As Integer
    Dim strMyString As String
    strMyString = txtGetHoursOfLabor.Text
```



```
intStrLen = Len(strMyString)

If Left$(strMyString, 1) = "-" Then
    txtGetHoursOfLabor = Right$(strMyString, (intStrLen - 1))
End If
End Sub

Private Sub txtGetPartsCost_GotFocus()
'*****
'Bold and UN-Bold user direction labels
'*****
    lblSayDATA(0).FontBold = False
    lblSayDATA(1).FontBold = False
    lblSayDATA(2).FontBold = True
End Sub

Private Sub txtGetPartsCost_LostFocus()
'*****
'Ensure user does NOT enter a negative number by making it positive
'*****
Dim intStrLen As Integer
Dim strMyString As String
strMyString = txtGetPartsCost.Text
intStrLen = Len(strMyString)

If Left$(strMyString, 1) = "-" Then
    txtGetPartsCost.Text = Right$(strMyString, (intStrLen - 1))
End If
End Sub
```