

```

Public Class formPayroll
    '-----
    '-          File Name : PayrollSystem_Dang.frm          -
    '-          Part of Project: PayrollSystem              -
    '-----
    '-          Written By: Binh D. Dang                   -
    '-          Written On: January 13th, 2022              -
    '-----
    '- File Purpose:                                         -
    -
    '- This file contains the main form for the entire appli- -
    '- cation. All user input is gathered on this form. The  -
    '- calculations of the payroll which are performed by the application -
    '- reside in this file as well and user can use it on the fly.
    '- Finally all generated output is created and put in a text file
    '- name payroll.txt                                     -
    '-----
    '- Program Purpose:                                     -
    '- The purpose of this program is to handle the calculations-
    '- of each employee's salary when input Wage and Hour   -
    '- company, then program can save the employee's name, ID,
    '- wages, and hours to the text file for later use.
    '- Program requires user to input name, id, wage, and hours.
    '- Then user will have option to calculate wages, save it,
    '- or cancel it when user doesn't want to save it in the txt file
    '-----
    '- Global Variable Dictionary (alphabetically):
    '- ctr - global integer for knowing what employee are in the track
    '- Employee - A user defined structure which will hold -
    '-          the employee's name and ID along with all numeric -
    '-          values for the project currently being          -
    '-          calculated.                                     -
    '- objMyStreamWriter - global Stream Writer for Write down to txt file
    '- path - global String that contains the path to create the txt file
    '-----
    'Global Constant
    '-rate - overtime rate

    '-----
    '--- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL
    STRUCTURES ---
    '--- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL
    STRUCTURES ---
    '--- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL STRUCTURES --- GLOBAL
    STRUCTURES ---
    '-----
    Structure Employee
        Dim name As String 'name of employee
        Dim ID As String 'ID of the employee
        Dim hour As String 'Hours employee work
        Dim wage As String 'Wage per hour of employee
    End Structure

```

```

'-----
'--- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES
'---
'--- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES
'---
'--- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES --- GLOBAL VARIABLES
'---
'-----
'
Dim path As String = "payroll.txt" ' global String that contains the path to create
the txt file
Dim employees() As Employee 'Structure variable array name employees() that contains
the necessary information
Dim objMyStreamWriter As System.IO.StreamWriter ' global Stream Writer for Write down
to txt file
Dim ctr As Integer = 0 'global integer for knowing what employee are in the track

'---- GLOBAL CONSTANT---
Const rate As Double = 1.5

'--- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS ---
'--- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS ---
'--- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS --- SUBPROGRAMS ---
'-----

Private Sub PayrollSystem_DANG_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
    '-----
    '-          Subprogram Name: PayrollSystem_DANG          -
    '-----
    '-          Written By: Binh Dang                          -
    '-          Written On: January 13, 2022                  -
    '-----
    '- Subprogram Purpose:                                     -
    '-                                                         -
    '- This subprogram started when the form is loaded and start
    '- checking if the payroll File is existed in the computer
    '- if the file not existed, create the file and start add the employee
    '- subprocedure.
    '- If the file existed, start reading the first employee using the
    '- ReadFile() procedure
    '-----
    '- Local Variable Dictionary (alphabetically):           -
    '- payrollFile- check if the file existed. if existed, it will contain the name
    path
    '- if not , it will be blank
    '-----
    Dim payrollFile = Dir(path)
    'If the file is not found it will create a text file
    'else open and read the File
    If payrollFile = "" Then
        objMyStreamWriter = System.IO.File.CreateText(path)
        objMyStreamWriter.Close()
        AddEmployee ()
    End If

```

```

Else
    ReadFile()
End If
End Sub

Private Sub btnCalc_Click(sender As Object, e As EventArgs) Handles btnCalc.Click
    '-----
    '- Subprogram Name: btnCalc -
    '-----
    '- Written By: Binh Dang -
    '- Written On: January 13, 2022 -
    '-----
    '- Subprogram Purpose: -
    '- -
    '- This subroutine extracts the values entered from the user-
    '- into the variable. Then calculate the wage and change the label
    '- start with 'the Employee is due'
    '-----
    '- Parameter Dictionary (in parameter order): -
    '- aCompanyBillInfo - The structure variable which contains -
    '- all information for the project from -
    '- which the program is creating a bill. -
    '-----
    '- Local Variable Dictionary (alphabetically): -
    '- h - input hours
    '- totalWage - multiply the wage with hours if the salary is checked
    '- then multiply it with 40
    '- w - input wage
    '-----
    Dim h As Double = CDbl(txtHours.Text)
    Dim w As Double = CDbl(txtWages.Text)
    Dim totalWage As Double

    'Alogrith check if the boc is checked, then calculate the wage
    If chkSalaried.Checked = True Then
        totalWage = 40.0 * w
    Else
        If (h <= 40 And h > 0) Then
            totalWage = w * h
        Else
            totalWage = w * 40 + (h - 40) * rate * w
        End If
    End If
    'Change the text lable to know how much the salary is
    lblWage.Text = "Employee is due " & FormatCurrency(totalWage)
End Sub

Private Sub btnAdd_Click(sender As Object, e As EventArgs) Handles btnAdd.Click
    '-----
    '- Subprogram Name: btnAdd -
    '-----
    '- Written By: Binh Dang -
    '- Written On: January 14, 2022 -
    '-----
    '- Subprogram Purpose: -

```

```

    '- -
    '- This subroutine is called whenever the user clicks the -
    '- "Add NEw Employee" button. This routine will call AddEmployee sub-
    '-----
    AddEmployee()
End Sub

Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles btnSave.Click
    '-----
    '- Subprogram Name: btnSave -
    '-----
    '- Written By: Binh Dang -
    '- Written On: January 14, 2022 -
    '-----
    '- Subprogram Purpose: -
    '- -
    '- This subroutine is called whenever the user clicks the -
    '- Save button. This routine will validate everything from the text boxes
    '- Then put to the note then save it if input right, else show an error message
    '- Later it will call the subprocedure Read File
    '-----
    '- Parameter Dictionary (in parameter order): -
    '- objMyStreamWriter - StreamWriter variable for writing to txt
    '-----
    '- Local Variable Dictionary (alphabetically): -
    '- (None) -
    '-----

    objMyStreamWriter = My.Computer.FileSystem.OpenTextFileWriter(path, True)
    'Validate the INput making sure that user input is right
    If IsNumeric(txtHours.Text) And IsNumeric(txtWages.Text) Then
        objMyStreamWriter.WriteLine(txtName.Text & ", " & txtID.Text & ", " &
txtHours.Text & ", " & txtWages.Text)
        objMyStreamWriter.Close()
        'When starting the program and add new employee, ctr will set to one
        If ctr > 0 Then
            ctr = employees.Count()
        End If
        ReadFile()
    Else
        MessageBox.Show("Input Error")
        objMyStreamWriter.Close()
        AddEmployee()
    End If
End Sub

Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles btnCancel.Click
    '-----
    '- Subprogram Name: btnCancel -
    '-----
    '- Written By: Binh Dang -
    '- Written On: January 14, 2022 -
    '-----
    '- Subprogram Purpose: -
    '- -

```

```

'- This subroutine is called whenever the user clicks the -
'- "Cancel" button. This routine will call ReadFile subprocedure
'-----
ReadFile()
End Sub

Private Sub btnNext_Click(sender As Object, e As EventArgs) Handles btnNext.Click
'-----
'- Subprogram Name: btnNext -
'-----
'- Written By: Binh Dang -
'- Written On: January 15, 2022 -
'-----
'- Subprogram Purpose: -
'- -
'- This subroutine is called whenever the user clicks the -
'- ">>" button. This routine will increase the counter to know -
'- what user it's on and then call ReadFile subprocedure
'-----
ctr = ctr + 1
ReadFile()
End Sub

Private Sub btnBack_Click(sender As Object, e As EventArgs) Handles btnBack.Click
'-----
'- Subprogram Name: btBack -
'-----
'- Written By: Binh Dang -
'- Written On: January 15, 2022 -
'-----
'- Subprogram Purpose: -
'- -
'- This subroutine is called whenever the user clicks the -
'- "<<" button. This routine will decrease the counter to know -
'- what user it's on and then call ReadFile subprocedure
'-----
ctr = ctr - 1
ReadFile()
End Sub

Sub ReadFile()
'-----
'- Subprogram Name: ReadFile -
'-----
'- Written By: Binh Dang -
'- Written On: January 15, 2022 -
'-----
'- Subprogram Purpose: -
'- -
'- This subroutine will hide the button save and cancel when the
'- employee is being read. Then it will turn the textbox to read only to not allow
'- user to edit the data field. Then it showed the the Back, Next and Add
employee button
'- Also when the new employee is being read, it also delete the label, so
'- user won't get confused by reading.

```

```

'- Then, the sub procedure will find the file with the path to read the file,
then read the employee structure
'-by reading it line by line,
'-Then, it will check the counter if it is in the right position else show the
message box
'-In case there are no employee is written in the file (first started), it move
to addEmployee
'-----
'- Parameter Dictionary (in parameter order): -
'- employees - This is the structure containing all -
'- of the current employee's information. -
'-----
'- Local Variable Dictionary (alphabetically): -
'- data - Get the information from splitting the data in the file line by line
'- line - The result from reading the input
'- n - the total number of employee
'-----

'When reading employeeemchange all txtBox to read only
txtName.ReadOnly = True
txtHours.ReadOnly = True
txtID.ReadOnly = True
txtWages.ReadOnly = True

'hide save and cancel button
btnSave.Visible = False
btnCancel.Visible = False

'show the back, next, and add employee button
btnBack.Visible = True
btnNext.Visible = True
btnAdd.Visible = True

'empty the label
lblWage.Text = ""

'This section put everything in the file to the employee structure
Dim data(3) As String
Dim line() As String
line = IO.File.ReadAllLines(path)
Dim n As Integer = line.Count - 1
ReDim employees(n)
For i As Integer = 0 To n
    data = line(i).Split(",")
    employees(i).name = data(0)
    employees(i).ID = data(1)
    employees(i).hour = data(2)
    employees(i).wage = data(3)
Next

'Check the ctr then make it show appropriate title and text!
If ctr <= n And ctr >= 0 Then
    txtName.Text = employees(ctr).name
    txtID.Text = employees(ctr).ID
    txtHours.Text = FormatNumber(employees(ctr).hour, 2)

```

```

        txtWages.Text = FormatNumber(employees(ctr).wage, 2)
        Me.Text = "Payroll System - Employee " & ctr + 1 & "/" & n + 1
    ElseIf ctr < 0 Then
        MessageBox.Show("You cannot move pass the first record!")
        ctr = ctr + 1
    ElseIf ctr > n Then
        MessageBox.Show("You cannot move pass the last record!")
        ctr = ctr - 1
    Else
        AddEmployee()
    End If

End Sub

Sub AddEmployee()
    '-----
    'Subprogram Name: ReadFile          -
    '-----
    'Written By: Binh Dang              -
    'Written On: January 15, 2022      -
    '-----
    'Subprogram Purpose:                -
    '
    'This subroutine will show the button save and cancel when the
    'employee is being read. When adding employee, reset all the txtboxes and make
    'them editable
    'Also Hide the button add new employee, next, and back. Then change it to the
    'appropriate title
    '-----
    'Parameter Dictionary (in parameter order):    -
    'none
    '-----
    'Local Variable Dictionary (alphabetically):    -
    'none
    '-----

    'empty all the textbox
    txtName.Text = ""
    txtID.Text = ""
    txtHours.Text = ""
    txtWages.Text = ""

    ' Make all textboxes editable
    txtName.ReadOnly = False
    txtHours.ReadOnly = False
    txtID.ReadOnly = False
    txtWages.ReadOnly = False

    'Show button save and cancel
    btnSave.Visible = True
    btnCancel.Visible = True

    'Hide button Next, Back, and Add New Employee
    btnBack.Visible = False
    btnNext.Visible = False
    btnAdd.Visible = False

    'Change the form title
    Me.Text = "Payroll System - Add New Employee"
End Sub

End Class

```