**Coding Exercise Number 1. Payroll**

1. Develop the design for application by reading the code and drawing it on a piece of paper.
2. Complete a table that includes the objects, names, text and comments.
3. Copy the code as stated below; make sure the program runs without any errors.
   1. If there are errors, use your debugging skills to sort them out and fix them.
4. The code:
5. 'Pluchino
6. 'Payroll
7. 'October 3, 2016
8. Public Class frmPayroll
9. Private Sub btnPay\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPay.Click
10. Dim hoursWorked As Integer
11. Dim hourlyRate As Decimal
12. Dim grossWage As Decimal
13. Dim overtime As Integer
14. hoursWorked = Val(Me.txtHours.Text)
15. hourlyRate = Val(Me.txtRate.Text)
16. 'Calculate gross wages
17. If hoursWorked > 40 Then 'calculate overtime pay
18. overtime = hoursWorked - 40
19. grossWage = (40 \* hourlyRate) + (overtime \* (hourlyRate \* 1.5))
20. Else 'no overtime pay
21. grossWage = hoursWorked \* hourlyRate
22. End If
23. Me.lblAnswerLabel.Text = "Gross wages = $"
24. Me.lblAnswer.Text = grossWage
25. End Sub
26. Private Sub Form1\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
27. End Sub
28. Private Sub lblAnswerLabel\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles lblAnswerLabel.Click
29. End Sub
30. Private Sub ExitToolStripMenuItem\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ExitToolStripMenuItem.Click
31. Application.Exit()
32. End Sub
33. End Class

**Coding Exercise Number 2. Printing Prices**

1. Develop the design for application by reading the code and drawing it on a piece of paper.
2. Complete a table that includes the objects, names, text and comments.
3. Copy the code as stated below; make sure the program runs without any errors.
   1. If there are errors, use your debugging skills to sort them out and fix them.
4. The code:
5. 'Pluchino
6. 'PrintingPrices
7. 'October 4, 2016
8. Public Class frmPrices
9. Private Sub btnPrice\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPrice.Click
10. Dim copies As Integer
11. Dim price As Decimal
12. copies = Val(Me.txtNumCopies.Text)
13. 'Calculate price based on the number of copies
14. Me.lblPerCopyLabel.Text = "Price per copy is $"
15. If copies >= 1000 Then '1000 copies or more, price is .25 per copy
16. price = 0.25
17. Me.lblPerCopyAnswer.Text = "0.25"
18. ElseIf copies >= 750 Then '750-999 copies, price is .27 per copy
19. price = 0.27
20. Me.lblPerCopyAnswer.Text = "0.27"
21. ElseIf copies >= 500 Then '500-749 copies, price is .28 per copy
22. price = 0.28
23. Me.lblPerCopyAnswer.Text = "0.28"
24. Else '0-499 copies, price is .30 per copy
25. price = 0.3
26. Me.lblPerCopyAnswer.Text = "0.30"
27. End If
28. Me.lblTotalLabel.Text = "The total price is $"
29. Me.lblTotalPriceAnswer.Text = price \* copies
30. End Sub
31. Private Sub txtNumCopies\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtNumCopies.TextChanged
32. Me.lblPerCopyLabel.Text = Nothing
33. Me.lblPerCopyAnswer.Text = Nothing
34. Me.lblTotalLabel.Text = Nothing
35. Me.lblTotalPriceAnswer.Text = Nothing
36. End Sub
37. Private Sub Form1\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
38. End Sub
39. End Class

**Coding Exercise Number 3. Package Check**

1. Develop the design for application by reading the code and drawing it on a piece of paper.
2. Complete a table that includes the objects, names, text and comments.
3. Copy the code as stated below; make sure the program runs without any errors.
   1. If there are errors, use your debugging skills to sort them out and fix them.
4. The code:
5. 'PackageCheck
6. Public Class Form1
7. Private Sub btnCheckPackage\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnCheckPackage.Click
8. Dim weight As Single
9. Dim length As Single
10. Dim width As Single
11. Dim height As Single
12. Dim cubicMeters As Single
13. weight = Val(Me.txtWeight.Text)
14. length = Val(Me.txtLength.Text)
15. width = Val(Me.txtWidth.Text)
16. height = Val(Me.txtHeight.Text)
17. cubicMeters = length \* width \* height
18. 'Check package requirements and display appropriate message
19. If weight > 27 And cubicMeters > 100000 Then 'package too large and too heavy
20. Me.lblAnswer.Text = "Rejected: Too heavy and too large"
21. ElseIf weight > 27 Then 'package too heavy
22. Me.lblAnswer.Text = "Rejected: Too heavy"
23. ElseIf cubicMeters > 100000 Then 'package too large
24. Me.lblAnswer.Text = "Rejected: Too large"
25. Else 'package is acceptable size and weight
26. Me.lblAnswer.Text = "Package accepted"
27. End If
28. End Sub
29. Private Sub txtWeight\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtWeight.TextChanged
30. Me.lblAnswer.Text = Nothing
31. End Sub
32. Private Sub txtHeight\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtHeight.TextChanged
33. Me.lblAnswer.Text = Nothing
34. End Sub
35. Private Sub txtWidth\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtWidth.TextChanged
36. Me.lblAnswer.Text = Nothing
37. End Sub
38. Private Sub txtLength\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtLength.TextChanged
39. Me.lblAnswer.Text = Nothing
40. End Sub
41. Private Sub lblAnswer\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles lblAnswer.Click
42. End Sub
43. Private Sub Form1\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
44. End Sub
45. End Class

**Coding Exercise Number 4. Computer Troubleshooting**

1. Develop the design for application by reading the code and drawing it on a piece of paper.
2. Complete a table that includes the objects, names, text and comments.
3. Copy the code as stated below; make sure the program runs without any errors.
   1. If there are errors, use your debugging skills to sort them out and fix them.
4. The code:
5. 'ComputerTroubleshooting
6. Public Class Form1
7. Private Sub btnWhatToDo\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnWhatToDo.Click
8. Dim beep As Char
9. Dim spin As Char
10. beep = Me.txtBeep.Text
11. spin = Me.txtSpin.Text
12. 'Determine repair message
13. If (spin = "Y" Or spin = "y") And (beep = "Y" Or beep = "y") Then 'hard drive spins and computer beeps
14. Me.lblAnswer.Text = "Contact tech support."
15. ElseIf (spin = "N" Or spin = "n") And (beep = "Y" Or beep = "y") Then 'hard drive doesn't spin and computer beeps
16. Me.lblAnswer.Text = "Check drive contacts."
17. ElseIf (spin = "Y" Or spin = "y") And (beep = "N" Or beep = "n") Then 'hard drive spins and computer doesn't beep
18. Me.lblAnswer.Text = "Check the speaker connections."
19. Else 'hard drive doesn't spin and computer doesn't beep
20. Me.lblAnswer.Text = "Bring computer to repair center."
21. End If
22. End Sub
23. Private Sub txtSpin\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtSpin.TextChanged
24. Me.lblAnswer.Text = Nothing
25. End Sub
26. Private Sub txtBeep\_TextChanged(ByVal sender As Object, ByVal e As System.EventArgs) Handles txtBeep.TextChanged
27. Me.lblAnswer.Text = Nothing
28. End Sub
29. Private Sub Form1\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
30. End Sub
31. End Class

**Coding Exercise Number 5. Game of 21 (Blackjack)**

1. Develop the design for application by reading the code and drawing it on a piece of paper.
2. Complete a table that includes the objects, names, text and comments.
3. Copy the code as stated below; make sure the program runs without any errors.
   1. If there are errors, use your debugging skills to sort them out and fix them.
4. The code:

'GameOf21

Public Class Form1

Dim playerTotal As Integer

Dim computerTotal As Integer

Dim computerCard1 As Integer

Dim computerCard2 As Integer

Dim computerCard3 As Integer

Dim playerCard1 As Integer

Dim playerCard2 As Integer

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

playerTotal = 0

computerTotal = 0

Me.lblPlayerDrewCard.Text = Nothing

Me.lblComputersScore.Text = Nothing

'Deal cards to player

Randomize()

playerCard1 = Int(10 \* Rnd() + 1)

playerCard2 = Int(10 \* Rnd() + 1)

Me.lblPlayerCard1.Text = playerCard1

Me.lblPlayerCard2.Text = playerCard2

playerTotal = playerCard1 + playerCard2

'Determine computer cards

computerCard1 = Int(10 \* Rnd() + 1)

computerCard2 = Int(10 \* Rnd() + 1)

computerCard3 = Int(10 \* Rnd() + 1)

Me.lblComputerCard1.Text = "\*" 'hide cards for now

Me.lblComputerCard2.Text = "\*"

Me.lblComputerCard3.Text = "\*"

computerTotal = computerCard1 + computerCard2 + computerCard3

Me.lblPlayersScore.Text = playerTotal

End Sub

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

Me.lblComputerCard1.Text = computerCard1

Me.lblComputerCard2.Text = computerCard2

Me.lblComputerCard3.Text = computerCard3

Me.lblComputersScore.Text = computerTotal

If computerTotal > 21 And playerTotal > 21 Then 'draw, both players over 21

MessageBox.Show("Both players went over 21. It is a draw")

ElseIf computerTotal > playerTotal And computerTotal <= 21 Then 'computer has higher score and wins

MessageBox.Show("Computer wins.")

ElseIf playerTotal > computerTotal And playerTotal <= 21 Then 'player has higher score and wins

MessageBox.Show("Player wins.")

ElseIf playerTotal = computerTotal And playerTotal <= 21 And \_

computerTotal <= 21 Then 'draw, scores are equal

MessageBox.Show("Both scores are equal. It is a draw")

ElseIf playerTotal > 21 Then 'player went over 21

MessageBox.Show("Player went over 21. Computer wins.")

Else 'computer went over 21

MessageBox.Show("Computer went over 21. Player wins.")

End If

End Sub

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

Dim drawCard As Integer = Int(10 \* Rnd()) + 1

Me.lblPlayerDrewCard.Text = drawCard

playerTotal = playerCard1 + playerCard2 + drawCard

Me.lblPlayersScore.Text = playerTotal

End Sub

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

Application.Exit()

End Sub

Private Sub Form1\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

End Sub

Private Sub lblPlayerCard1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles lblPlayerCard1.Click

End Sub

End Class

Congratulations! You have completed the assignments for today.

Now start to study for your Unit 4 Test.