```
' ************************************************************************************
'  Surname, Initials:  Kuni, D
'  Student Number:  201320596
'  Practical: P05
'  Class name: frmUPD
' ************************************************************************************
Option Explicit On
Option Strict On
Option Infer Off
Public Class frmUPD
#Region "Variable Declarations "
    Private Frauds() As Fraud
    Private Fences() As Fencing
    Private launderies() As Laundering
    Private nfe, nfr, nl As Integer
#End Region
    Private Sub btndisplayfr_Click(sender As Object, e As EventArgs) Handles
btndisplayfr.Click
        'displays all necessary data values for fraud cases
        Dim answer As Double
        Dim rate As Integer
        grdDisplay.Cols = nfr + 1
        lblgrid.Text = "A table showing all the Fraud Case details"
        For i As Integer = 1 To nfr
            pig(i, 0, "Case " + CStr(i))
            pig(i, 1, Frauds(i).CaseID)
            pig(i, 2, Frauds(i).OfficerRank + Space(1) + Frauds(i).OfficerSurname)
            pig(i, 3, CStr(Frauds(i).loss))
            Rate = Frauds(i).CalcOfficerSuccess 'stores value , calls the function once
            pig(i, 4, CStr(Math.Round(Rate, 2)))
            answer = 0
            Frauds(i).CalcRecovery(Frauds(i).CalcSuccessFactor(rate), answer)
            pig(i, 5, CStr(Math.Round(answer)))
        Next
    End Sub

    Private Sub btndisplayfe_Click(sender As Object, e As EventArgs) Handles
btndisplayfe.Click
        'displays all necessary data values for fencing cases
        Dim answer As Double
        Dim rate As Integer
        grdDisplay.Cols = nfe + 1
        lblgrid.Text = "A table showing all the Fencing Case details"
        For i As Integer = 1 To nfe
            pig(i, 0, "Case " + CStr(i))
            pig(i, 1, Fences(i).CaseID)
            pig(i, 2, Fences(i).OfficerRank + Space(1) + Fences(i).OfficerSurname)
            pig(i, 3, CStr(Fences(i).loss))
            rate = Fences(i).CalcOfficerSuccess
            pig(i, 4, CStr(Math.Round(rate, 2)))
            answer = 0
            Fences(i).CalcRecovery(Fences(i).CalcSuccessFactor(rate), answer)
            pig(i, 5, CStr(Math.Round(answer)))
        Next
    End Sub

    Private Sub btndisplayl_Click(sender As Object, e As EventArgs) Handles
btndisplayl.Click
```

```
        'displays all necessary data values for laundry cases
        Dim answer As Double
        Dim rate As Integer
        grdDisplay.Cols = nl + 1
        lblgrid.Text = "A table showing all the Laundry Case details"
        For i As Integer = 1 To nl
            pig(i, 0, "Case " + CStr(i))
            pig(i, 1, launderies(i).CaseID)
            pig(i, 2, launderies(i).OfficerRank + Space(1) +
launderies(i).OfficerSurname)
            pig(i, 3, CStr(launderies(i).loss))
            rate = launderies(i).CalcOfficerSuccess 'stores value , calls the function
once
            pig(i, 4, CStr(Math.Round(rate, 2)))
            answer = 0
            launderies(i).CalcRecovery(launderies(i).CalcSuccessFactor(rate), answer)
            pig(i, 5, CStr(Math.Round(answer)))
        Next
    End Sub
End Class



' ******************************************************************************
'  Surname, Initials:  Kuni, D
'  Student Number:  201320596
'  Practical: P05
'  Class name: frmUPD
' ******************************************************************************
Option Explicit On
Option Strict On
Option Infer Off
Partial Class frmUPD
    Private Sub pig(c As Integer, r As Integer, t As String) 'stands for place in grid
        grdDisplay.Col = c
        grdDisplay.Row = r
        grdDisplay.Text = t
    End Sub
    Private Sub frmUPD_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Randomize()
        grdDisplay.set_ColWidth(0, 120)
        pig(0, 1, "Case ID")
        pig(0, 2, "Investigating Officer")
        pig(0, 3, "Value of Loss")
        pig(0, 4, "Officer's Success Rate")
        pig(0, 5, "Value Recovered")
    End Sub
    Private Sub btnSetup_Click(sender As Object, e As EventArgs) Handles btnSetup.Click
        nfr = CInt(txtfraud.Text)
        nfe = CInt(txtfencing.Text)
        nl = CInt(txtlaundering.Text)
        ReDim Frauds(nfr)
        ReDim Fences(nfe)
        ReDim launderies(nl)
        MsgBox("Program settings have been set", MsgBoxStyle.OkOnly, "Settings altered")
    End Sub
#Region "Inputted Data buttons"
```

```
    Private Sub btnfraud_Click(sender As Object, e As EventArgs) Handles btnfraud.Click
        'gets all necessary values for fraud cases
        For i As Integer = 1 To nfr
            Frauds(i) = New Fraud
            Frauds(i).CaseID = InputBox("Please enter the Case ID for Fraud case " +
CStr(i), "Fraud case " + CStr(i))
            Frauds(i).OfficerSurname = InputBox("Please enter the Investigating Officer
Surname for Fraud case " + CStr(i), "Fraud case " + CStr(i))
            Frauds(i).OfficerRank = InputBox("Please enter the Investigating Officer Rank
for Fraud case " + CStr(i), "Fraud case " + CStr(i))
            Frauds(i).loss = CDbl(InputBox("Please enter the Value of Loss for Fraud case
" + CStr(i), "Fraud case " + CStr(i)))
            Frauds(i).Duration = CInt(InputBox("Please enter the Days since crime took
place for Fraud case " + CStr(i), "Fraud case " + CStr(i)))
        Next i
        MsgBox("All necessary data has been entered for all fraud cases",
MsgBoxStyle.OkOnly, "fraud cases")
    End Sub
    Private Sub btnfence_Click(sender As Object, e As EventArgs) Handles btnfence.Click
        'gets all necessary values for fence cases
        For i As Integer = 1 To nfe
            Fences(i) = New Fencing
            Fences(i).CaseID = InputBox("Please enter the Case ID for Fencing case " +
CStr(i), "Fencing case " + CStr(i))
            Fences(i).OfficerSurname = InputBox("Please enter the Investigating Officer
Surname for Fencing case " + CStr(i), "Fencing case " + CStr(i))
            Fences(i).OfficerRank = InputBox("Please enter the Investigating Officer Rank
for Fencing case " + CStr(i), "Fencing case " + CStr(i))
            Fences(i).loss = CDbl(InputBox("Please enter the Value of Loss for Fencing
case " + CStr(i), "Fencing case " + CStr(i)))
            Fences(i).MOG = CInt(InputBox("Please enter the Mobility of Goods(value
between 1-10) for Fencing case " + CStr(i), "Fencing case " + CStr(i)))
        Next i
        MsgBox("All necessary data has been entered for all fencing cases",
MsgBoxStyle.OkOnly, "fencing cases")
    End Sub
    Private Sub btnlaunder_Click(sender As Object, e As EventArgs) Handles
btnlaunder.Click
        'gets all necessary values for laurndry cases
        For i As Integer = 1 To nl
            launderies(i) = New Laundering
            launderies(i).CaseID = InputBox("Please enter the Case ID for Laundering case
" + CStr(i), "Laundering case " + CStr(i))
            launderies(i).OfficerSurname = InputBox("Please enter the Investigating
Officer Surname for Laundering case " + CStr(i), "Laundering case " + CStr(i))
            launderies(i).OfficerRank = InputBox("Please enter the Investigating Officer
Rank for Laundering case " + CStr(i), "Laundering case " + CStr(i))
            launderies(i).loss = CDbl(InputBox("Please enter the Value of Loss for
Laundering case " + CStr(i), "Laundering case " + CStr(i)))
            launderies(i).numbCurrencies = CInt(InputBox("Please enter the Number of
different currencies involved for Laundering case " + CStr(i), "Laundering case " +
CStr(i)))
            launderies(i).numbAccounts = CInt(InputBox("Please enter the Number of
accountants involved for Laundering case " + CStr(i), "Laundering case " + CStr(i)))
        Next i
        MsgBox("All necessary data has been entered for all laundry cases",
MsgBoxStyle.OkOnly, "laundry cases")
    End Sub
```

```
#End Region
End Class



' ****************************************************************************
'  Surname, Initials: Kuni, D
'  Student Number:  201320596
'  Practical: P05
'  Class name: CommercialCrime
' ****************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public MustInherit Class CommercialCrime
#Region "Attributes"
    Protected iCaseID As String
    Protected iOfficerSurname As String
    Protected iOfficerRank As String
    Protected iLoss As Double
#End Region
#Region "Propert methods"
    Public Property CaseID As String
        Get
            Return iCaseID
        End Get
        Set(value As String)
            iCaseID = value
        End Set
    End Property
    Public Property OfficerSurname As String
        Get
            Return iOfficerSurname
        End Get
        Set(value As String)
            iOfficerSurname = value
        End Set
    End Property
    Public Property OfficerRank As String
        Get
            Return iOfficerRank
        End Get
        Set(value As String)
            iOfficerRank = value
        End Set
    End Property
    Public Property loss As Double
        Get
            Return iLoss
        End Get
        Set(value As Double)
            iLoss = value
        End Set
    End Property
#End Region
#Region "Methods"
    Public Function CalcOfficerSuccess() As Integer
```

```
        Return CInt(Math.Floor((100 - 10 + 1) * Rnd())) + 10
    End Function
    Public MustOverride Function CalcSuccessFactor(SuccessRate As Integer) As Double
'forces derived classes to edit code
    Public Sub CalcRecovery(ByVal successfactor As Double, ByRef ans As Double)
        ans = (successfactor / 100) * loss
    End Sub
#End Region
End Class
```

```
' ************************************************************************************
'  Surname, Initials: Kuni, D
'  Student Number:  201320596
'  Practical: P05
'  Class name: Fencing
' ************************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public Class Fencing
    Inherits CommercialCrime
    Private fMOG As Double
    Public Property MOG As Double
        Get
            Return fMOG
        End Get
        Set(value As Double)
            fMOG = value
        End Set
    End Property
    Public Overrides Function CalcSuccessFactor(SuccessRate As Integer) As Double
        Return SuccessRate - MOG
    End Function
    Public Sub New()
        CaseID = ""
        OfficerRank = ""
        OfficerSurname = ""
        loss = 0
        fMOG = 0
    End Sub
End Class
```

```
' ************************************************************************************
'  Surname, Initials: Kuni, D
'  Student Number:  201320596
'  Practical: P05
'  Class name: Fraud
' ************************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public Class Fraud
```

```
    Inherits CommercialCrime
    Private fDuration As Integer 'Days since crime took place
    Public Property Duration As Integer
        Get
            Return fDuration
        End Get
        Set(value As Integer)
            If value > 0 Then
                fDuration = value
            End If
        End Set
    End Property
    Public Overrides Function CalcSuccessFactor(SuccessRate As Integer) As Double
        Return SuccessRate / Duration
    End Function
    Public Sub New()
        CaseID = ""
        OfficerRank = ""
        OfficerSurname = ""
        loss = 0
        fDuration = 0
    End Sub
End Class
```

```
' ***************************************************************************
'  Surname, Initials: Kuni, D
'  Student Number:  201320596
'  Practical: P05
'  Class name: Laundering
' ***************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public Class Laundering
    Inherits CommercialCrime
#Region "Attributes"
    Private Currencies As Integer
    Private Accounts As Integer
#End Region
#Region "Property Methods"
    Public Property numbCurrencies As Integer
        Get
            Return Currencies
        End Get
        Set(value As Integer)
            If value > 0 Then
                Currencies = value
            End If
        End Set
    End Property
    Public Property numbAccounts As Integer
        Get
            Return Accounts
        End Get
        Set(value As Integer)
```

```
            If value > 0 Then
                Accounts = value
            End If
        End Set
    End Property
#End Region
    Public Overrides Function CalcSuccessFactor(SuccessRate As Integer) As Double
        Return (Currencies / Accounts) + SuccessRate
    End Function
    Public Sub New()
        CaseID = ""
        OfficerRank = ""
        OfficerSurname = ""
        loss = 0
        Currencies = 0
        Accounts = 0
    End Sub
End Class
```