```vba
Attribute VB_Name = "mod01"
Option Explicit

Global Const rmv_modeInitial As Integer = 1
Global Const rmv_modeCreation As Integer = 2
Global Const rmv_modeAffichage As Integer = 3
Global Const rmv_modeModification As Integer = 4


Global rmv_state As Integer

Global savedClient As String
Global savedActivite As String
Global savedHeures As String
Global savedFacturable As String
Global savedCommNote As String

Sub FilterProfDate()

    'Minimum - Professionnel + Date
    If Trim(frmSaisieHeures.cmbProfessionnel.value) = "" Or _
       Trim(frmSaisieHeures.txtDate.value) = "" Then
        Exit Sub
    End If

    'Date in the appropriate format (AAAA-MM-JJ)
    Dim dateUSFormated As String
    dateUSFormated = Format(CDate(frmSaisieHeures.txtDate.value), "m/d/yyyy")

    'Base worksheet (Heures) contains all entries
    Dim sh As Worksheet
    Set sh = ThisWorkbook.Sheets("Heures")
    sh.AutoFilterMode = False

    'Once filtered, worksheet should only show the filtered records
    Dim shFiltered As Worksheet
    Set shFiltered = ThisWorkbook.Sheets("HeuresFiltered")
    shFiltered.UsedRange.Clear
    'shFiltered.Activate

    sh.Activate
    With sh.UsedRange
        .AutoFilter Field:=2, Criteria1:=frmSaisieHeures.cmbProfessionnel.value
        .AutoFilter Field:=3, Operator:=xlFilterValues, _
                    Criteria2:=Array(2, dateUSFormated)
        .AutoFilter Field:=12, Criteria1:="FAUX"
    End With

    sh.UsedRange.Select
    sh.UsedRange.Copy shFiltered.Range("A1")

    shFiltered.Activate

    sh.Activate
    sh.AutoFilterMode = False
```

```vba
End Sub

Sub ImportClientsList()    '---------------------------    'v1.0 - 2023-03-23  @ 07:40

    'Delete  all cells, but the headers  in the current  worksheet
    shImportedClients.Range("A1").CurrentRegion.Offset(1,       0).Clear

    'Source  workbook  (closed  Excel  file) - MUST BE IN THE SAME  DIRECTORY
    Dim sourceWorkbook   As String
    Dim sourceWorksheet   As String
    sourceWorkbook   = ThisWorkbook.Path   & Application.PathSeparator    & _
        "GCF_Clients.xlsx"
    sourceWorksheet   = "Clients"

    'ADODB  connection
    Dim connStr  As ADODB.Connection
    Set connStr  = New ADODB.Connection

    'Connection   String  specific  to EXCEL
    connStr.ConnectionString     = _
        "Provider  = Microsoft.ACE.OLEDB.12.0;"     & _
        "Data  Source = " & sourceWorkbook   & ";" & _
        "Extended  Properties  = 'Excel  12.0  Xml; HDR = YES';"
    connStr.Open

    'Recordset
    Dim recSet  As ADODB.Recordset
    Set recSet  = New ADODB.Recordset

    recSet.ActiveConnection     = connStr
    recSet.Source  = "SELECT  * FROM [" & sourceWorksheet   & "$]"
    recSet.Open

    'Copy  to destination  workbook  (actual)  into the 'Top2000'  worksheet
    shImportedClients.Range("A2").CopyFromRecordset       recSet
    shImportedClients.Range("A1").CurrentRegion.EntireColumn.AutoFit

    'Close  resource
    recSet.Close
    connStr.Close

    MsgBox  _
        Prompt:="J'ai   un total  de " & _
        Format(shImportedClients.Range("A1").CurrentRegion.Rows.count        - 1, _
        "## ##0")  & " clients",  _
        Title:="Vérification    du nombre  de clients",  _
        Buttons:=vbInformation

End Sub

'****************************************************************          EffaceFormulaire
Sub EffaceFormulaire()

    'Empty  the dynamic  fields  after  reseting  the form
```

```vba
            .txtActivite.value    = ""
            .txtHeures.value    = ""
            .txtCommNote.value    = ""
            .cmbProfessionnel.Enabled    = True
            .txtDate.Enabled    = True
        End With


        Call  FilterProfDate
        Call  RefreshListBox

        With  frmSaisieHeures
            .cmdClear.Enabled   = False
            .cmdAdd.Enabled   = False
            .cmdDelete.Enabled   = False
            .cmdUpdate.Enabled   = False
        End With


        frmSaisieHeures.txtClient.SetFocus

End Sub

'************************************************************                AjouteLigneDetail
Sub  AjouteLigneDetail()

        'Validations   first  (one  field  at  a time)
        If frmSaisieHeures.cmbProfessionnel.value       = "" Then
            MsgBox  _
                Prompt:="Le   professionnel   est OBLIGATOIRE   !", _
                Title:="Vérification",       _
                Buttons:=vbCritical
            frmSaisieHeures.cmbProfessionnel.SetFocus
            Exit  Sub
        End If

        If frmSaisieHeures.txtDate.value    = "" Or _
            IsDate(frmSaisieHeures.txtDate.value)       = False  Then
            MsgBox  _
                Prompt:="La   date est OBLIGATOIRE   !", _
                Title:="Vérification",       _
                Buttons:=vbCritical
            frmSaisieHeures.txtDate.SetFocus
            Exit  Sub
        End If

        If frmSaisieHeures.txtClient.value     = "" Then
            MsgBox  _
                Prompt:="Le   client  est OBLIGATOIRE   !", _
                Title:="Vérification",       _
                Buttons:=vbCritical
            frmSaisieHeures.txtClient.SetFocus
        Exit  Sub
        End If

        If frmSaisieHeures.txtHeures.value       = "" Or _
```

```vba
            Prompt:="Le    nombre  d'heures   est OBLIGATOIRE   !", _
            Title:="Vérification",           _
            Buttons:=vbCritical
        frmSaisieHeures.txtHeures.SetFocus
        Exit Sub
    End If

    Dim sh As Worksheet
    Set sh = ThisWorkbook.Sheets("Heures")
    'Always delete the TotalRow, before adding a new row to the table
    'sh.ListObjects("tCharges").TotalsRowRange.Delete

    Dim lastRow As Long
    lastRow = Application.WorksheetFunction.CountA(sh.Range("A:A"))

    'Load the cmb & txt into the 'Heures' worksheet
    With sh
        .Range("A" & lastRow + 1).value = "=row()-1"
        .Range("B" & lastRow + 1).value = frmSaisieHeures.cmbProfessionnel.value
        .Range("C" & lastRow + 1).value = CDate(frmSaisieHeures.txtDate.value)
        .Range("D" & lastRow + 1).value = frmSaisieHeures.txtClient.value
        .Range("E" & lastRow + 1).value = frmSaisieHeures.txtActivite.value
        .Range("F" & lastRow + 1).value = Format(frmSaisieHeures.txtHeures.value, _
  "#0.00")
        .Range("G" & lastRow + 1).value = frmSaisieHeures.txtCommNote.value
        .Range("H" & lastRow + 1).value = frmSaisieHeures.chbFacturable.value
        .Range("I" & lastRow + 1).value = Now
        .Range("J" & lastRow + 1).value = False
        .Range("K" & lastRow + 1).value = ""
        .Range("L" & lastRow + 1).value = False
    End With

    'Empty the fields after saving
    frmSaisieHeures.txtClient.value      = ""
    frmSaisieHeures.txtActivite.value    = ""
    frmSaisieHeures.txtHeures.value      = ""
    frmSaisieHeures.txtCommNote.value    = ""

    Call FilterProfDate
    Call RefreshListBox

    With frmSaisieHeures
        .cmdClear.Enabled  = False
        .cmdAdd.Enabled    = False
        .cmdUpdate.Enabled = False
    End With

    frmSaisieHeures.txtClient.SetFocus

End Sub


'*************************************************************          ModifieLigneDetail
Sub ModifieLigneDetail()
```

```vba
            Prompt:="Vous   devez  choisir  un enregistrement   à modifier  !", _
            Title:="",    _
            Buttons:=vbCritical
        Exit Sub
    End If


    'Validations  first  (one field  at a time)
    If frmSaisieHeures.cmbProfessionnel.value    = "" Then
        MsgBox _
            Prompt:="Le  professionnel  est OBLIGATOIRE  !", _
            Title:="Vérification",    _
            Buttons:=vbCritical
        frmSaisieHeures.cmbProfessionnel.SetFocus
        Exit Sub
    End If


    If frmSaisieHeures.txtDate.value   = "" Or _
        IsDate(frmSaisieHeures.txtDate.value)    = False Then
        MsgBox _
            Prompt:="La  date est OBLIGATOIRE  !", _
            Title:="Vérification",    _
            Buttons:=vbCritical
        frmSaisieHeures.txtDate.SetFocus
        Exit Sub
    End If


    If frmSaisieHeures.txtClient.value    = "" Then
        MsgBox _
            Prompt:="Le   client est OBLIGATOIRE  !", _
            Title:="Vérification",    _
            Buttons:=vbCritical
        frmSaisieHeures.txtClient.SetFocus
    Exit Sub
    End If


    If frmSaisieHeures.txtHeures.value   = "" Or _
        IsNumeric(frmSaisieHeures.txtHeures.value)    = False Then
        MsgBox _
            Prompt:="Le  nombre d'heures  est OBLIGATOIRE  !", _
            Title:="Vérification",    _
            Buttons:=vbCritical
        frmSaisieHeures.txtHeures.SetFocus
        Exit Sub
    End If

    Dim sh As Worksheet
    Set sh = ThisWorkbook.Sheets("Heures")

    Dim selectedRow  As Long
    selectedRow  = Application.WorksheetFunction.Match(CLng(frmSaisieHeures.txtID.value), _
                    sh.Range("A:A"),   0)

    With frmSaisieHeures
```

```vba
        sh.Range("D"   & selectedRow).value    = .txtClient.value
        sh.Range("E"   & selectedRow).value    = .txtActivite.value
        sh.Range("F"   & selectedRow).value    = Format(.txtHeures.value,     "#0.00")
        sh.Range("G"   & selectedRow).value    = .txtCommNote.value
        sh.Range("H"   & selectedRow).value    = .chbFacturable.value
        sh.Range("I"   & selectedRow).value    = Now
        sh.Range("J"   & selectedRow).value    = False
        sh.Range("K"   & selectedRow).value    = ""
        sh.Range("L"   & selectedRow).value    = False

        frmSaisieHeures.txtClient.value     = ""
        frmSaisieHeures.txtActivite.value       = ""
        frmSaisieHeures.txtHeures.value     = ""
        frmSaisieHeures.txtCommNote.value       = ""
    End With

    frmSaisieHeures.cmbProfessionnel.Enabled      = True
    frmSaisieHeures.txtDate.Enabled     = True
    rmv_state   = rmv_modeCreation

    Call  FilterProfDate
    Call  RefreshListBox

    frmSaisieHeures.txtClient.SetFocus

End Sub

'**************************************************************              EffaceLigneDetail
Sub EffaceLigneDetail()

    If frmSaisieHeures.txtID.value    = "" Then
        MsgBox _
            Prompt:="Vous   devez  choisir  un enregistrement   à DÉTRUIRE  !", _
            Title:="", _
            Buttons:=vbCritical
        Exit  Sub
    End If

    Dim answerYesNo  As Integer
    answerYesNo  = MsgBox("Êtes-vous   certain  de vouloir  DÉTRUIRE  cet
enregistrement   ? ", _
                            vbYesNo  + vbQuestion,  "Confirmation  de DESTRUCTION")
    If answerYesNo  = vbNo  Then
        MsgBox _
            Prompt:="Cet   enregistrement   ne sera PAS détruit  ! ", _
            Title:="Confirmation",    _
            Buttons:=vbCritical
        Exit  Sub
    End If

    Dim  sh As Worksheet
    Set  sh = ThisWorkbook.Sheets("Heures")

    Dim  selectedRow   As Long
```

```vba
                               sh.Range("A:A"),    0)


        'Assign  'VRAI'  to colomn  12,  since  it is deleted
        sh.Range("I"   & selectedRow).value   = Now
        sh.Range("L"   & selectedRow).value   = True


        'Empty  the dynamic  fields  after  deleting
        With  frmSaisieHeures
            .txtClient.value   = ""
            .txtActivite.value   = ""
            .txtHeures.value   = ""
            .txtCommNote.value   = ""
        End  With


        MsgBox  _
            Prompt:="L'enregistrement    a été DÉTRUIT  !", _
            Title:="Confirmation",     _
            Buttons:=vbCritical


        frmSaisieHeures.cmbProfessionnel.Enabled      = True
        frmSaisieHeures.txtDate.Enabled      = True
        rmv_state  = rmv_modeCreation


        Call  FilterProfDate
        Call  RefreshListBox


        frmSaisieHeures.txtClient.SetFocus


End  Sub


'********************    Reload  listBox  from HeuresFiltered   and reset  the buttons
Sub  RefreshListBox()


    If  Trim(frmSaisieHeures.cmbProfessionnel)      = ""  _
            Or  Trim(frmSaisieHeures.txtDate)      = ""  Then
        GoTo  EndOfProcedure
    End  If


    frmSaisieHeures.txtTotalHeures.value      = ""


    Dim  shFiltered  As  Worksheet
    Set  shFiltered  = ThisWorkbook.Sheets("HeuresFiltered")
    shFiltered.Activate


    'Last  Row used  in column  A
    Dim  lastRow  As  Long
    lastRow  = Application.WorksheetFunction.CountA(shFiltered.Range("A:A"))


    With  frmSaisieHeures.lstData
        .ColumnHeads   = True
        .ColumnCount   = 9
        .ColumnWidths   = "25;  30;  55;  120;  190;  30;  90;  35;  70"


        If  lastRow  = 1 Then
```

```vba
            .RowSource  = "HeuresFiltered!A2:K"    & lastRow
        End If
    End With


    'Add  hours  to totalHeures
    Dim nbrRows,  i As Integer
    nbrRows  = frmSaisieHeures.lstData.ListCount
    Dim totalHeures  As Double

    If nbrRows  > 0 Then
        For  i = 0 To nbrRows  - 1
            totalHeures  = totalHeures  + CCur(frmSaisieHeures.lstData.List(i,      5))
        Next
        frmSaisieHeures.txtTotalHeures.value     = Format(totalHeures,    "#0.00")
    End If

EndOfProcedure:
    frmSaisieHeures.cmdClear.Enabled     = False
    frmSaisieHeures.cmdAdd.Enabled     = False
    frmSaisieHeures.cmdUpdate.Enabled      = False
    frmSaisieHeures.cmdDelete.Enabled      = False

    'frmSaisieHeures.txtClient.SetFocus

End Sub
```