temp = CurrentPlayer.HitPoints - ((RandomNumber() Mod 3) + 1)
If temp <= 0 Then
    MsgBox ("You die from the poison burning in your blood")
    GameForm.QuitButton_Click
Else
    CurrentPlayer.HitPoints = temp
    AttackMessages.Caption = "The Poison makes you ill!"
    Pause (2)
    PlayerHitPoints.Value = CurrentPlayer.HitPoints
End If
' checks to determine who has the present turn (the monster, the player)
' if the turn number is equal, then calls the initiative function.
If Monsterturn = Playerturn Then
    Initiative
ElseIf Monsterturn < Playerturn Then
    CheckMonFlee
Else
    PlayerOption
End If
End Sub

Private Sub CancelCommand_Click()
' set for returning to the player option menu if the person chooses not to use a skill.
    KickCommand.Visible = False
    PunchCommand.Visible = False
    ParryCommand.Visible = False
    CancelCommand.Visible = False
End Sub

Private Sub DefendButton_Click()
' drops a player into a defensive stance decreasing the likelihood of being hit.
' also allows for the regeneration of some fatigue points in battle.
    AttackButton.Enabled = False
    DefendButton.Enabled = False
    SkillButton.Enabled = False
    FleeButton.Enabled = False
    ItemButton.Enabled = False
    AttackMessages.Caption = "You have dropped into a defensive stance."
    Evade = Evade + 10
    CurrentPlayer.Fatigue = CurrentPlayer.Fatigue + 2
    If CurrentPlayer.Fatigue > CurrentPlayer.Max_Fp Then
    End If
    PlayerFatiguePoints.Value = CurrentPlayer.Fatigue
    Playerturn = Playerturn + 1
    CheckTurn
End Sub

Private Sub Died()
' called if the player is killed by the monster,
' displays a screen telling the player they are dead.
    BlackFrame.Visible = True
    BlackFrame.Width = 800
    BlackFrame.Height = 600
    NameLabel.Caption = CurrentPlayer.Name & Chr(13) & "died on day " & GameForm.timeclock1.days
    KilledFrame.Left = 5500 - (KilledFrame.Width / 2)
    KilledFrame.Top = 3000 - (KilledFrame.Height / 2)
    KilledFrame.Visible = True
    Pause (5)
Sub EndAttack()
    ' when the fight is over, unload the form. control returned to main game form.
    NewPlayerForm.Calculate_HitBonus
    NewPlayerForm.Calculate_DamageBonus
    CurrentMonsterNumber = -1
    Unload Me
End Sub

Private Sub FleeButton_Click()
    ' sometimes you just can't win and it is better to run away. This enables the player
    ' to try and escape combat with some hitpoints and fatigue left
    AttackButton.Enabled = False
    SkillButton.Enabled = False
    FleeButton.Enabled = False
    ItemButton.Enabled = False
    DefendButton.Enabled = False
    Dim choice As VbMsgBoxResult
    Dim GetAway As Single
    choice = MsgBox("Are you sure you want to flee?", vbYesNo + vbQuestion, "Flee")
    ' checks to be sure the player wants to run, if he does,
    If choice = vbYes Then
        GetAway = RandomNumber()
        If GetAway >= 50 Then ' check to see if you can get away
            MonPursue ' if you do, the monster may try to pursue you to stop you
        Else ' sometimes you just can't run.
            AttackMessages.Caption = " You were unable to escape. "
            Playerturn = Playerturn + 1
            CheckTurn
        End If
    Else ' if you made a mistake with flee, then you go back to the options menu
        PlayerOption
    End If
End Sub

' Written by Karen Woznick
' This subroutine sets up the locations and sizes of the objects
' on the form and initializes the variables.
Private Sub Form_Load()
    ' Paints the form's picture to the background of the screen
    For i = 0 To 6
        For j = 0 To 8
            AttackForm.PaintPicture AttackForm.Picture, j * 100, i * 100, 100, 100
        Next j, i
    Next j, i
    PlayerImage.Picture = LoadPicture(CurrentImageDrive + CurrentPlayer.Picture)
    PlayerImage.Height = 320
    PlayerImage.Width = 300
    PlayerImage.Top = 25
    PlayerImage.Left = 25
    Opponent.Top = PlayerImage.Top
    Opponent.Left = 800 - 25 - PlayerImage.Width
    Opponent.Height = 320
    Opponent.Width = 300
    AttackButton.Left = PlayerImage.Left + PlayerImage.Width + 25
    AttackButton.Top = PlayerImage.Top
    DefendButton.Left = AttackButton.Left
    DefendButton.Top = AttackButton.Top + AttackButton.Height
    SkillButton.Left = AttackButton.Left
    SkillButton.Top = DefendButton.Top + DefendButton.Height
ItemButton.Left = AttackButton.Left
ItemButton.Top = SkillButton.Top + SkillButton.Height
FleeButton.Left = AttackButton.Left
FleeButton.Top = ItemButton.Top + ItemButton.Height
MediumAttack.Left = LightAttack.Left
HeavyAttack.Left = LightAttack.Left
KickCommand.Left = LightAttack.Left
KickCommand.Top = SkillButton.Top
PunchCommand.Left = LightAttack.Left
PunchCommand.Top = KickCommand.Top + KickCommand.Height
ParryCommand.Left = LightAttack.Left
ParryCommand.Top = PunchCommand.Top + PunchCommand.Height
CancelCommand.Left = LightAttack.Left
CancelCommand.Top = ParryCommand.Top + ParryCommand.Height
' Set up progress bars
PlayerHitPoints.Top = PlayerImage.Top + PlayerImage.Height + 25
PlayerHitPoints.Left = PlayerImage.Left
PlayerHitPoints.Max = CurrentPlayer.Max_Hp
PlayerHitPoints.Value = CurrentPlayer.HitPoints
PlayerFatiguePoints.Top = PlayerHitPoints.Top + PlayerHitPoints.Height + 10
PlayerFatiguePoints.Left = PlayerImage.Left
PlayerFatiguePoints.Max = CurrentPlayer.Max_Fp
PlayerFatiguePoints.Value = CurrentPlayer.Fatigue
PlayerTurn = 0
Monsterturn = 0
Evade = CurrentPlayer.Evade
MonsterName.Top = Opponent.Top + Opponent.Height + 25
MonsterName.Left = Opponent.Left
MonsterHitPoints.Left = Opponent.Left
MonsterHitPoints.Top = PlayerFatiguePoints.Top
AttackMessages.Left = 25
AttackMessages.Width = 750
AttackMessages.Height = 100
AttackMessages.Top = MonsterHitPoints.Top + MonsterHitPoints.Height + 25
' calls these functions to perform skill checks
Armorlore
Archery
Blindfighting
Bluntweaponfighting
Brawling
Knifefighting
Swordfighting
Weaponlore

Private Sub HeavyAttack_Click()
' choses the strongest level of attack for the player to use
' closes out the attack sub-menu, and calls fatigue to ensure the attack can be used
LightAttack.Visible = False
MediumAttack.Visible = False
HeavyAttack.Visible = False
AttackLevel = 3
CheckFatigue

Private Sub Initiative()
' called when turns are equal, determines who goes first
MonsterInit = (RandomNumber() \ 5) + 1
PlayerInit = (RandomNumber() \ 5) + 1
If (MonsterInit > PlayerInit) Then 'greater initiative wins...
Private Sub ItemButton_Click()
'for the moment does nothing.
    AttackButton.Enabled = False
    DefendButton.Enabled = False
    SkillButton.Enabled = False
    FleeButton.Enabled = False
    ItemButton.Enabled = False
    PlayerOption
End Sub

Private Sub Kick()
' checks the fatigue of the player, if player has enough fatigue,
' executes the kick command, otherwise a return to player option.
    If CurrentPlayer.Fatigue < 5 Then
        AttackMessages.Caption = " You can't do that! too tired!! "
        PlayerOption
    Else
        Dim hitval As Integer
        hitval = RandomNumber() Mod 50 + 1
        ' checks to see if the kick hit
        If hitval + CurrentPlayer.HitBonus < CurrentMonster.Evade Then
            ' if not, the player has an opportunity of falling down
            If hitval > 10 Then
                Playerturn = Playerturn + 2
                CheckTurn
            Else
                Playerturn = Playerturn + 1
                CheckTurn
            End If
        Else
            ' otherwise the kick hits and does damage
            Dim temp As Integer
            If temp <= 0 Then
                MsgBox("You have killed the " + CurrentMonster.Name + ".")
                AttackMessages.Caption = " You kicked the monster. "
                PlayerFatiguePoints.Value = CurrentPlayer.Fatigue
            EndAttack
        Else
            CurrentMonster.Current_HitPoints = temp
            MonsterHitPoints.Value = CurrentMonster.Current_HitPoints
            AttackMessages.Caption = " You kicked the monster. "
            PlayerFatiguePoints.Value = CurrentPlayer.Fatigue
            Playerturn = Playerturn + 1
            CheckTurn
        End If
    End If
End Sub

Private Sub KickCommand_Click()
'calls the kick command from the skills submenu
    KickCommand.Visible = False
    ParryCommand.Visible = False
    PunchCommand.Visible = False
    CancelCommand.Visible = False
    Kick
End Sub

Private Sub Killed()
'if the monster is killed, determines the various goodies that may
'be added to the players inventory
    If CurrentMonster.Name = "golem" Then
        Call Show_MessageBox(CurrentImageDrive + ItemArray(20).Picture, "You have taken a gem from the forehead of the golem.")
        Call GameForm.Add(ItemArray(20))
        Call GameForm.CheckItems
    ElseIf CurrentMonster.Name = "barbarian leader" Then
        Call Show_MessageBox(CurrentImageDrive + ItemArray(24).Picture, "You have taken a pommel from the corpse of the barbarian leader.")
        Call GameForm.Add(ItemArray(24))
        Call GameForm.CheckItems
    End If
    If CurrentMonster.MonsterType = 0 Then
        Call GameForm.Add(ItemArray(49))
        MsgBox ("You have obtained a carcass.")
    ElseIf CurrentMonster.MonsterType = 1 Then
        CurrentPlayer.Money = CurrentPlayer.Money + (RandomNumber() \ 10) + 1
        MsgBox ("You have pilfered some money from the corpse.")
    End If
End Attack
End Sub

Private Sub Knifefighting()
' adds bonuses for a player wielding a knife type weapon (dagger
' or knife) depending on if the skill check is made and the
' level of proficiency
    If CurrentPlayer.Wielded.ItemType = 2 And RandomNumber() <= CurrentPlayer.Knifefighting Then
        If CurrentPlayer.Knifefighting < 21 Then
            CurrentPlayer.Hitbonus = CurrentPlayer.Hitbonus + 1
            CurrentPlayer.Damagebonus = CurrentPlayer.Damagebonus + 1
        ElseIf CurrentPlayer.Knifefighting > 20 And CurrentPlayer.Knifefighting < 41 Then
        ElseIf CurrentPlayer.Knifefighting > 40 And CurrentPlayer.Knifefighting < 61 Then
        ElseIf CurrentPlayer.Knifefighting > 60 And CurrentPlayer.Knifefighting < 81 Then
            CurrentPlayer.Damagebonus = CurrentPlayer.Damagebonus + 4
        ElseIf CurrentPlayer.Knifefighting > 80 Then
        End If
    Else
    End If
End Sub

Private Sub LightAttack_Click()
' calls the light attack
    LightAttack.Visible = False
MediumAttack.Visible = False
HeavyAttack.Visible = False
AttackLevel = 1
CheckFatigue

Private Sub MediumAttack_Click()
' calls the medium attack
LightAttack.Visible = False
MediumAttack.Visible = False
HeavyAttack.Visible = False
AttackLevel = 2
CheckFatigue
End Sub

Private Sub MonFlee()
' the monster attempts to get away.
' checks to see if the monster gets away
Dim GetAway As Single
Dim choice As VbMsgBoxResult
GetAway = RandomNumber()
If GetAway >= 50 Then
' if he does, offer the option of pursuit
choice = MsgBox("The monster is fleeing. Would you like to pursue?", vbYesNo + vbQuestion, "Pursue")
If choice = vbYes Then
    Pursue
Else endAttack
End If
Else
    Monsterturn = Monsterturn + 1
    CheckTurn
End If
End Sub

Private Function MonPursue()
' the monster's attempt to block the player is determined by no flee
If CurrentMonster.NoFlee <= RandomNumber() Mod 10 Then
    If RandomNumber() >= 50 Then
        AttackMessages.Caption = "You were not permitted to escape."
        Playerturn = Playerturn + 1
        CheckTurn
    Else
        MsgBox ("You have successfully escaped.")
        EndAttack
    End If
Else
    'otherwise, combat continues.
    MsgBox ("You have successfully escaped.")
    EndAttack
End If
End Function

Private Function MonsterAttack()
' works to determine if the monster hits
If (RandomNumber() Mod 50 + 1 + CurrentMonster.ToHit) < (Evade - (100 * CurrentPlayer.Fatigue / 200) * Evade) Then
    AttackMessages.Caption = "The monster misses!"
    If CurrentPlayer.Parry = True Then

'if the player is parrying, check for a parry attack
If RandomNumber() Mod 10 = 1 Then
    Dim temphp As Integer
    If temphp <= 0 Then
        MsgBox("You have killed the " + CurrentMonster.Name + ")."
    EndAttack
Else
    CurrentMonster.Current_HitPoints = temphp
    MonsterHitPoints.Value = CurrentMonster.Current_HitPoints
End If
End If
CurrentPlayer.Parry = False
End If
'if the player has fencing, heck for a counter offensive
If RandomNumber() <= CurrentPlayer.Fencing And CurrentPlayer.Wielded.ItemType = 1 Then
    AttackMessages.Caption = "you mount a counter offensive while the monster is off balance!"
    If temphp <= 0 Then
        MsgBox("You have killed the " + CurrentMonster.Name + ")."
    EndAttack
Else
    CurrentMonster.Current_HitPoints = temphp
    MonsterHitPoints.Value = CurrentMonster.Current_HitPoints
End If
Else
    CurrentPlayer.Fencing = CurrentPlayer.Fencing + GameForm.Improve_Skill
End If
Monsterturn = Monsterturn + 1
CheckTurn
Else
'otherwise, the monster mounts his attack. at one of 3 attack levels
'determined by a random number.
    AttackLevel = RandomNumber()
    Dim temp As Integer
    If AttackLevel <= CurrentMonster.LightAttackPercent Then
        temp = CurrentPlayer.HitPoints -
        ((CurrentMonster.DamageDice * ((RandomNumber() Mod CurrentMonster.Damage) + 1) + CurrentMonster.ToDam) / 2) * (100 - CurrentPlayer.Armor) / 100
        If temp <= 0 Then
            Died
        Else
            CurrentPlayer.HitPoints = temp
            AttackMessages.Caption = "The monster hits for damage. You have " & temp & " hit points left."
            PlayerHitPoints.Caption = CurrentPlayer.HitPoints
        End If
        Monsterturn = Monsterturn + 1
        CurrentPlayer.Parry = False
        CheckTurn
        If temp <= 0 Then
            Died
        Else
            CurrentPlayer.HitPoints = temp
            AttackMessages.Caption = "The monster hits for damage. You have " & temp & " hit points left."
            PlayerHitPoints.Caption = CurrentPlayer.HitPoints
        End If
        Monsterturn = Monsterturn + 1
        CurrentPlayer.Parry = False
        CheckTurn
    EndIf
    Monsterturn = Monsterturn + 1
    CheckTurn
Else
    CurrentPlayer.HitPoints = temp
    AttackMessages.Caption = "The monster hits for damage. You have " & temp & " hit points left."
    PlayerHitPoints.Value = CurrentPlayer.HitPoints
End If
Monsterturn = Monsterturn + 1
CurrentPlayer.Parry = False
CheckTurn
Else
If temp <= 0 Then
    Died
Else
    CurrentPlayer.HitPoints = temp
    AttackMessages.Caption = "The monster hits for damage. You have " & temp & " hit points left."
    PlayerHitPoints.Value = CurrentPlayer.HitPoints
End If
Monsterturn = Monsterturn + 1
CurrentPlayer.Parry = False
CheckTurn
End If
End If
End Function
Private Sub MonsterSpecial()
'checks the monsters special percent, if it is met, the special is called, other wise monattack is called.
If RandomNumber() <= CurrentMonster.SpecialAttackPercent Then
    Select Case CurrentMonster.Special
        Case 0: Call FireBreath
        Case 1: Call Paralyze
        Case 2: Call Haymaker
        Case 3: Call Hamstring
        Case 4: Call Poison
        Case 5: Call ColdBreath
    End Select
Else
    MonsterAttack
End If
End Sub
' Show Message Box
Private Sub Show_MessageBox(ByVal image As String, ByVal text As String)
    MessageBox.MessagePicture = image
    MessageBox.LabelText = text
    MessageBox.Top = 150
    MessageBox.Left = 75
    MessageBox.Height = 225
    MessageBox.Width = 350
    MessageBox.Visible = True
End Sub
Private Sub Swordfighting()
'checks to determine if the player is wielding a sword. if he is, then apply the appropriate modifiers if a skill check is met.
If CurrentPlayer.Wielded.ItemType = 1 And RandomNumber() <= CurrentPlayer.Swordfighting Then
    If CurrentPlayer.Swordfighting < 21 Then
        CurrentPlayer.Hitbonus = CurrentPlayer.Hitbonus + 1
CurrentPlayer.Damagebonus = CurrentPlayer.Damagebonus + 1
ElseIf CurrentPlayer.Swordfighting > 20 And CurrentPlayer.Swordfighting < 41 Then
ElseIf CurrentPlayer.Swordfighting > 40 And CurrentPlayer.Swordfighting < 61 Then
ElseIf CurrentPlayer.Swordfighting > 60 And CurrentPlayer.Swordfighting < 81 Then
    CurrentPlayer.Damagebonus = CurrentPlayer.Damagebonus + 4
ElseIf CurrentPlayer.Swordfighting > 80 Then
End If
Else
    CurrentPlayer.Swordfighting = CurrentPlayer.Swordfighting + GameForm.Improve_Skill
End If

Private Sub Parry()
' engages the parry ability of the player, offers a better evade
' and the opportunity to mount a counter offensive.
    If CurrentPlayer.Fatigue < 2 Then
        AttackMessages.Caption = "You can't do that! You're too tired!!"
        PlayerOption
    Else
        If RandomNumber() > CurrentPlayer.Parrying Then
            If RandomNumber Mod 10 = 0 Then
                Playerturn = Playerturn + 1
                CheckTurn
            Else
                Playerturn = Playerturn + 1
                CheckTurn
            End If
        Else
            Evade = Evade + 2
            Player.Fatigue = CurrentPlayer.Fatigue
            Playerturn = Playerturn + 1
            CheckTurn
        End If
    End If
End Sub

Private Function PlayerOption()
' enables the player's options during the appropriate combat turn.
    AttackButton.Enabled = True
    SkillButton.Enabled = True
    DefendButton.Enabled = True
    FleeButton.Enabled = True
    ItemButton.Enabled = True
End Function

Private Sub Punch()
' called by choosing the punch skill, this function throws a punch
' at your opponent provided you have the fatigue for it.
    If CurrentPlayer.Fatigue < 3 Then
        AttackMessages.Caption = "You can't do that! too tired!!"
    Else
        AttackMessages.Caption = "You can't do that! too tired!!"
    End If
End Sub
Else
    Dim hitval As Integer
    hitval = RandomNumber() Mod 50 + 1
    If hitval + CurrentPlayer.Hitbonus < CurrentMonster.Evade Then
        If hitval > 5 Then
            PlayerTurn = PlayerTurn + 2
            CheckTurn
        Else
            PlayerTurn = PlayerTurn + 1
            CheckTurn
        End If
    Else
        Dim temp As Integer
        If temp <= 0 Then
            MsgBox ("You have killed the " + CurrentMonster.Name + ").")
            EndAttack
        Else
            CurrentMonster.Current_HitPoints = temp
            MonsterHitPoints.Value = CurrentMonster.Current_HitPoints
            AttackMessages.Caption = "You punched the monster."
            PlayerTurn = PlayerTurn + 1
            CheckTurn
        End If
    End If
End If
End If

Private Sub Pursue()
'provides the player with a chance to stop a monster from fleeing
'if successful, the monster's attempt to flee is stopped
    Dim pursuit As Single
    pursuit = RandomNumber()
    If pursuit >= 50 Then
        AttackMessages.Caption = "You Block the Monster's attempt to flee."
        MonsterTurn = MonsterTurn + 1
        CheckTurn
    Else
        MsgBox ("The Monster gets away!")
        EndAttack
    End If
End Sub

Private Sub ParryCommand_Click()
' this calls the parry function when the parry button is clicked.
    KickCommand.Visible = False
    ParryCommand.Visible = False
    PunchCommand.Visible = False
    CancelCommand.Visible = False
    Parry
End Sub

Private Sub PunchCommand_Click()
'this calls the punch function if the punch button is selected
'from the skills submenu
    KickCommand.Visible = False
    ParryCommand.Visible = False

Private Function RandomNumber() As Single
'generates a random number from 1-100, set by system clock
    Randomize
    RandomNumber = Rnd() * 100 Mod 100 + 1
End Function

Private Sub SkillButton_Click()
    'calls the skills submenu when the skills button is clicked
    KickCommand.Visible = True
    ParryCommand.Visible = True
    PunchCommand.Visible = True
    CancelCommand.Visible = True
    If CurrentPlayer.Kicking < 1 Then
        KickCommand.Enabled = False
    End If
    If CurrentPlayer.Parrying < 1 Then
        ParryCommand.Enabled = False
    End If
    If CurrentPlayer.Punching < 1 Then
        PunchCommand.Enabled = False
    End If
End Sub

Private Sub Weaponlore()
    'checks weapons lore skill, if successful, apply appropriate bonuses
    If RandomNumber() <= CurrentPlayer.Weaponlore Then
    Else
        CurrentPlayer.Weaponlore = CurrentPlayer.Weaponlore + GameForm.Improve_Skill
    End If
End Sub

Private Sub ColdBreath()
    'first monster special. activates a breath attack for certain
    'monsters, hard to avoid (lower evade) and not soaked by armor.
    Dim temp As Integer
    If (RandomNumber() + CurrentMonster.ToHit) >= (Evade - ((100 * CurrentPlayer.Fatigue / 200) * Evade)) - 10 Then
        temp = CurrentPlayer.HitPoints - ((3 * (RandomNumber() Mod 5 + 1)) + 5)
        If temp <= 0 Then
            Died
        Else
            CurrentPlayer.HitPoints = temp
        End If
        AttackMessages.Caption = "The monster hits for damage. You have " & temp & " hit points left."
    End If
    Monsterturn = Monsterturn + 1
    CheckTurn
Else
    Monsterturn = Monsterturn + 1
    CheckTurn
End If
End Sub

Private Sub FireBreath()
' activates a breath attack for certain
'monsters, hard to avoid (lower evade) and not soaked by armor.

Dim temp As Integer
If (RandomNumber() + CurrentMonster.ToHit) >= (Evade - ((100 * CurrentPlayer.Fatigue / 200) * Evade)) - 10 Then
    temp = CurrentPlayer.HitPoints - ((3 * (RandomNumber() Mod 5 + 1)) + 5)
    If temp <= 0 Then
        Died
    Else
        CurrentPlayer.HitPoints = temp
        AttackMessages.Caption = "The monster hits for damage. You have " & temp & " hit points left."
        PlayerHitPoints.Value = CurrentPlayer.HitPoints
    End If
    Monsterturn = Monsterturn + 1
    CheckTurn
Else
    Monsterturn = Monsterturn + 1
    CheckTurn
End If

Private Sub Paralyze()
    ' another monster special, this one does no damage, but does
    ' stop the player from doing anything for a few turns.
    If (RandomNumber() + CurrentMonster.ToHit) >= (Evade - ((100 * CurrentPlayer.Fatigue / 200) * Evade)) Then
        Playerturn = Playerturn + (RandomNumber() Mod 4) + 1
        Monsterturn = Monsterturn + 1
        CheckTurn
    Else
        Monsterturn = Monsterturn + 1
        CheckTurn
    End If
End Sub

Private Sub Hamstring()
    ' theives special, lessens player's evade. also a physical attack deals
    ' damage.
    Dim temp As Integer
    If (RandomNumber() + CurrentMonster.ToHit) >= (Evade - ((100 * CurrentPlayer.Fatigue / 200) * Evade)) Then
        temp = CurrentPlayer.HitPoints - ((RandomNumber() Mod 5) + 2)
        If temp <= 0 Then
            Died
        Else
            CurrentPlayer.HitPoints = temp
            AttackMessages.Caption = "The monster hits for damage. You have " & temp & " hit points left."
            PlayerHitPoints.Value = CurrentPlayer.HitPoints
        End If
        Dim tempevad As Integer
        tempevade = Evade - 10
        If tempevade <= 0 Then
            Evade = 0
        Else
            Evade = tempevade
            Monsterturn = Monsterturn + 1
            CheckTurn
        End If
    Else
        Else
Private Sub Haymaker()
' Special, monster throws a powerful punch at the player, may stun
' for a turn or two, providing free attacks.
Dim temp As Integer
If (RandomNumber() + CurrentMonster.ToHit) >= (Evade - ((100 * CurrentPlayer.Fatigue / 200) * Evade)) Then
    temp = CurrentPlayer.HitPoints - ((RandomNumber() Mod 10 + 1) + 3)
    If temp <= 0 Then
        Died
    Else
        CurrentPlayer.HitPoints = temp
        AttackMessages.Caption = " The monster hits for damage. You have " & temp & " hit points left."
        PlayerHitPoints.Value = CurrentPlayer.HitPoints
    End If
End If
Monsterturn = Monsterturn + 1
CheckTurn
Else
    Monsterturn = Monsterturn + 1
    CheckTurn
End If
End Sub

Private Sub Poison()
' Sets the flag poisoned to true. When this flag is true,
' poison hurts the player every even numbered combat round.
If (RandomNumber() + CurrentMonster.ToHit) >= (Evade - ((100 * CurrentPlayer.Fatigue / 200) * Evade)) Then
    CurrentPlayer.Poisoned = True
    Monsterturn = Monsterturn + 1
    CheckTurn
Else
    AttackMessages.Caption = " The monster misses with its attack."
    Monsterturn = Monsterturn + 1
    CheckTurn
End If
End Sub

' Written by Karen Woznick
' Turns off timer and reads in the data for the current monster
Private Sub StartTimer_Timer()
    StartTimer.Enabled = False
    With MonsterData.Recordset
        .MoveFirst
        For i = 1 To CurrentMonsterNumber - 1
            .MoveNext
        Next i
        CurrentMonster.Name = .Fields("Name").Value
        CurrentMonster.Current_HitPoints = .Fields("Current_HitPoints").Value
        CurrentMonster.Total_HitPoints = .Fields("Total_Hp").Value
        CurrentMonster.FleePercent = .Fields("FleePercent").Value
        CurrentMonster.ToHit = .Fields("ToHit").Value
        CurrentMonster.ToDam = .Fields("ToDam").Value
        CurrentMonster.MediumAttackPercent = .Fields("MediumAttackPercent").Value
    End With
CurrentMonster.HasSpecial = .Fields("HasSpecial").Value
CurrentMonster.Special = .Fields("Special").Value
CurrentMonster.SpecialAttackPercent = .Fields("SpecialPercent").Value
CurrentMonster.NoFlee = .Fields("NoFlee").Value
CurrentMonster.Armor = .Fields("Armor").Value
CurrentMonster.Evade = .Fields("Evade").Value
CurrentMonster.Damage = .Fields("Damage").Value
CurrentMonster.DamageDice = .Fields("Dice").Value
CurrentMonster.Picture = .Fields("Picture").Value
CurrentMonster.MonsterType = .Fields("Type").Value
End With

' Displays monster's picture
Opponent.Picture = LoadPicture(CurrentImageDrive + CurrentMonster.Picture)

' Displays monster's name
MonsterName.Caption = CurrentMonster.Name

' Sets up progress bars
MonsterHitPoints.Max = CurrentMonster.Total_HitPoints
MonsterHitPoints.Value = CurrentMonster.Current_HitPoints
Call Initiative
End Sub
'All functions on this form were written by
Karen Woznick unless otherwise noted.

Private Sub ExitCommand_Click()
    Dim frm As Form
    For Each frm In Forms
        Unload frm
    Next frm
End Sub

Private Sub Form_Load()
    Paint the background to the screen
    WelcomeLabel.Top = 100
    WelcomeLabel.Left = 400 - (WelcomeLabel.Width / 2)
    NewPlayerCommand.Top = WelcomeLabel.Top + WelcomeLabel.Height + 50
    NewPlayerCommand.Left = WelcomeLabel.Left + ((WelcomeLabel.Width / 2) - (NewPlayerCommand.Width / 2))
    LoadPlayerCommand.Left = NewPlayerCommand.Left
    ExitCommand.Top = LoadPlayerCommand.Top + LoadPlayerCommand.Height + 20
    ExitCommand.Left = LoadPlayerCommand.Left + ((LoadPlayerCommand.Width / 2) - (ExitCommand.Width / 2))
End Sub

Private Sub Initialize_Items()
    Dim i As Integer
    With ItemData.Recordset
        .MoveFirst
        i = 0
        While (.EOF <> True)
            ItemArray(i).Name = .Fields("Name").Value
            If .Fields("Picture").Value <> "" Then
                ItemArray(i).Picture = .Fields("Picture").Value
            End If
            ItemArray(i).Description1 = .Fields("Description1").Value
            ItemArray(i).Description2 = .Fields("Description2").Value
            ItemArray(i).Description3 = .Fields("Description3").Value
            ItemArray(i).Height = .Fields("Height").Value
            ItemArray(i).Weight = .Fields("Weight").Value
            ItemArray(i).Wear_Position = .Fields("Wear_Position").Value
            ItemArray(i).ItemType = .Fields("Type").Value
            ItemArray(i).Special1 = .Fields("Special1").Value
            ItemArray(i).Special2 = .Fields("Special2").Value
            ItemArray(i).Value = .Fields("Value").Value
            ItemArray(i).IDNumber = .Fields("IDNumber").Value
            i = i + 1
        .MoveNext
        Wnd
    End With
    ' Elements in ItemArray
    ' 0 Buckler
FirstScreen - 2

1 Small Shield
2 Large Shield
3 Plate Boots
4 Helmet
5 Leather Armor
6 Chain Mail Armor
7 Full Plate Mail Armor
8 Full Plate Armor
9 Long Sword
10 Bastard Sword
11 Dagger
12 Knife
13 Mace
14 Warhammer
15 Axe
16 Giant Axe
17 Bow
18 Final Sword
19 Special Dagger
20 Gem1
21 Gem2
22 Crosspiece
23 Hilt
24 Pommel
25 Scabbard
26 Loaf
27 Cheese
28 Ale
29 Wine
30 Meat
31 Fish
32 Fruit
33 Grapes
34 Berries
35 Herbs
36 Roots
37 Picks
38 Fishing Pole
39 Wood
40 First Aid Kit
41 Fur Pelt
42 Cloak
43 Pants and Shirt
44 Boots
45 Dress
46 Tunic and Leggings
47 Sandals
48 Monk's Robe
49 Carcass
50 Dinner
51 Chair
52 Bench
53 Arrow
54 Nothing

Private Sub LoadPlayerCommand_Click()
    Dim tmp_player As playertype
    Dim i As Integer

Written by Jeff Hanks
' Read in all item information from database
' into an array
Initialize_Items

' Load player information
' Obtain correct file to open
CommonDialog1.DefaultExt = ".dst"
CommonDialog1.DialogTitle = "Retrieve Character..."
CommonDialog1.Filter = "Character Files (*.dst)|*.dst|All Files (*.*)|*.*"
CommonDialog1.ShowOpen
If CommonDialog1.filename <> "" Then
  Open CommonDialog1.filename For Binary As #256
  Get #256, , tmp_player
  SetStats tmp_player
  Close #256
  Unload Me
End If
End Sub

' Written by Jeff Hanks
Private Sub SetStats(tmp As playertype)
With CurrentPlayer
Set .Aboutbody = ItemArray(tmp.Aboutbody)
  .Age = tmp.Age
  .Agility = tmp.Agility
  .Appraising = tmp.Appraising
  .Archery = tmp.Archery
  .Armor = tmp.Armor
  .Armorlore = tmp.Armorlore
  .Bandaged = tmp.Bandaged
  .Bandaging = tmp.Bandaging
  .Begged = tmp.Begged
  .Begging = tmp.Begging
  .Blindfighting = tmp.Blindfighting
  .Bluntweaponfighting = tmp.Bluntweaponfighting
Set .Body = ItemArray(tmp.Body)
  .Brawling = tmp.Brawling
  .Butchery = tmp.Butchery
  .Camping = tmp.Camping
  .Carpentry = tmp.Carpentry
  .Carving = tmp.Carving
  .Chopping = tmp.Chopping
  .Commerce = tmp.Commerce
  .Cooking = tmp.Cooking
  GameForm.timeclock1.minutes = tmp.Cur_minutes
  GameForm.Cur_x = tmp.Cur_x
  GameForm.Cur_y = tmp.Cur_y
  .Damagebonus = tmp.Damagebonus
  .Evade = tmp.Evade
  .Fatigue = tmp.Fatigue
Set .Feet = ItemArray(tmp.Feet)
  .Fencing = tmp.Fencing
  .Firebuilding = tmp.Firebuilding
  .Fishing = tmp.Fishing
  .Fitness = tmp.Fitness
  .Foraging = tmp.Foraging
  .Handtohandcombat = tmp.Handtohandcombat
Set .Head = ItemArray(tmp.Head)
  .Health = tmp.Health
  .Height = tmp.Height
  .Herblore = tmp.Herblore
.Hiking = tmp.Hiking
.Hitbonus = tmp.Hitbonus
.HitPoints = tmp.HitPoints
.Hunting = tmp.Hunting
.Intelligence = tmp.Intelligence
' not sure about the indexing.
For i = 0 To tmp.Nextindex - 1
    GameForm.AddItemArray(tmp.Inventory(i))
Next i
.Itemlore = tmp.Itemlore
.Jogging = tmp.Jogging
.Kicking = tmp.Kicking
.Knifefighting = tmp.Knifefighting
.Max_Fp = tmp.Max_Fp
.Max_Hp = tmp.Max_Hp
.Money = tmp.Money
.Mountaineering = tmp.Mountaineering
.Name = tmp.Name
.Nextindex = tmp.Nextindex
.Parry = tmp.Parry
.Parrying = tmp.Parrying
.Performance = tmp.Performance
.Performed = tmp.Performed
.Personality = tmp.Personality
.PickPocketing = tmp.PickPocketing
.Picture = tmp.Picture
.Poisoned = tmp.Poisoned
.Punching = tmp.Punching
.Sex = tmp.Sex
Set .Shield = ItemArray(tmp.Shield)
.Skinning = tmp.Skinning
.Social = tmp.Social
.Stealth = tmp.Stealth
.Strength = tmp.Strength
.Swimming = tmp.Swimming
.Swordfighting = tmp.Swordfighting
.Weaponlore = tmp.Weaponlore
.Weight = tmp.Weight
Set .Wielded = ItemArray(tmp.Wielded)
.Wrestling = tmp.Wrestling
End With
End Sub

' NewPlayerCommand_Click
' allows the player to start a new character
Private Sub NewPlayerCommand_Click()
' Reads items into an array from a database
Initialize_Items
StoryForm.Show vbModal, Me
NewPlayerForm.Show vbModal, Me
SkillsForm.Show vbModal, Me
Unload Me
End Sub

' PaintBack
' tiles the form's picture to all of the screen
Private Sub PaintBack()
    For i = 0 To 3
        For j = 0 To 4
            FirstScreen.PaintPicture FirstScreen.Picture, j * 200, i * 200, 200, 200
        Next j, i
    Next i
End Sub
FirstScreen - 5
End Sub
Option Explicit

' Create a global variable to keep track of which button
' was pushed last. Used in Map_Click subroutine.
Dim State As String
Dim Map(100, 100) As Integer
Public Cur_x As Integer
Public Cur_y As Integer
Dim Fire As Boolean

' Adds a new item to the current player's inventory
' and the combo box on the form.
Public Sub Add(NewItem As Item)
' First checks to see if the inventory is full.
If CurrentPlayer.NextIndex >= 99 Then
    MsgBox("You may not add any more items to your inventory.")
' If not full, then add the item.
Else
    Set CurrentPlayer.Inventory(CurrentPlayer.NextIndex) = NewItem
    Equipment.AddItem (CurrentPlayer.Inventory(CurrentPlayer.NextIndex) .Name)
    CurrentPlayer.NextIndex = CurrentPlayer.NextIndex + 1
End If
End Sub

' Appraising
' Allows the player to use his appraising skill to determine
' an approximate value of an item in his inventory.
Private Sub Appraising()
' First checks to see if an item is selected.
    If Equipment.text = "" Then
        MsgBox("You must be holding something in order to appraise it.")
' If so, then calculate and display a value.
    Else
        Dim Value As Integer
        Value = CurrentPlayer.Inventory(Equipment.ListIndex) .Value * (CurrentPlayer.Appraising / 100)
        PassTime (1)
        MsgBox("You have appraised the item for " + CStr(Value) + " silver pieces.")
    End If
' Give the player a chance to improve his skill.
    If CurrentPlayer.Appraising < RandomNumber() Then
        CurrentPlayer.Appraising = CurrentPlayer.Appraising + Improve_Skill
    End If
End Sub

' ArrowButton_Click
' Craft an arrow from wood.
Private Sub ArrowButton_Click()
    CarvingFrame.Visible = False
' Make sure that the player has enough fatigue.
    If CurrentPlayer.Fatigue < 2 Then
        MsgBox("You are too tired.")
' If so, perform a skill check.
    ElseIf CurrentPlayer.Carpentry > RandomNumber() Then
        PassTime (10)
        ' Subtract fatigue points
    End If
End Sub
' Show messagebox with picture of an arrow
Call Show_MessageBox(CurrentImageDrive + ItemArray(53).Picture, "You have successfully carved an arrow.")
' Remove the wood from the inventory
Remove (Equipment.ListIndex)
' Add an arrow to the inventory
Call Add(ItemArray(53))
' If player fails the skill check
Else
' Pass five minutes of gameplay
PassTime (5)
' Subtract fatigue points
CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 1
' Show a messagebox telling the player that they could not make an arrow.
MsgBox ("You were unable to make a arrow.")
' Remove the wood from his inventory
Remove (Equipment.ListIndex)
' Give player a chance to improve his skill
End If
' Reset the fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Changes state to attack if button was not previously pushed; else cancels last click and returns state to normal. Changes mousepointer icon appropriately.
Private Sub Attack_Click()
If State = "Attack" Then
    State = "Normal"
    MousePointer = 0
Else
    State = "Attack"
    MouseIcon = LoadPicture(CurrentImageDrive + "sword.ico")
    MousePointer = 99
End If
End Sub

' Bandaging
' Allows the player to bandage his wounds to restore some of his hit points
Private Sub Bandaging()
    Dim number As Byte
    ' Make sure player is holding a first aid kit
    If Equipment.text <> "firstaid kit" Then
        MsgBox ("You must be holding a FirstAid Kit to bandage.")
    ' Make sure player has enough fatigue points
    ElseIf CurrentPlayer.Fatigue < 4 Then
        MsgBox ("You are too tired.")
    ' If player can still bandage this day
    ElseIf CurrentPlayer.Bandaged = False Then
        number = CurrentPlayer.HitPoints + (CurrentPlayer.Bandaging / 10) + (RandomNumber() Mod 6) + 1
        ' Make sure to not exceed the player's max hit points
        If CurrentPlayer.HitPoints + number > CurrentPlayer.Max_Hp Then
            CurrentPlayer.Bandaged = True
        Else
            CurrentPlayer.HitPoints = CurrentPlayer.HitPoints + number
            CurrentPlayer.Bandaged = True
        End If
' Update the hit point progress bar
HitPoints.Value = CurrentPlayer.HitPoints
' Pass 5 minutes of game time
PassTime (5)
' Subtract fatigue points
CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 4
' Give player a chance to improve his skill
If CurrentPlayer.Bandaging < RandomNumber() Then
End If
' If player has already bandaged today
Else
    MsgBox ("You can only bandage once a day.")
End If
' Update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Craft a bench
Private Sub BenchCommand_Click()
    CarpentryFrame.Visible = False
    ' Make sure the player has enough fatigue points
    If CurrentPlayer.Fatigue < 10 Then
        MsgBox ("You are too tired.")
    End If
    ' If player passes the skill check
    ElseIf CurrentPlayer.Carpentry > RandomNumber() Then
        ' Pass game time
        PassTime (45)
        ' Subtract fatigue points
        CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 10
        ' Show a messagebox with a picture of a bench
        Call Show_MessageBox(CurrentImageDrive + ItemArray(52).Picture, "You have successfully crafted a bench.")
        ' Remove the wood from the equipment list
        Remove (Equipment.ListIndex)
        ' Add a bench to the player's inventory
        Call Add(ItemArray(52))
    ' If player fails skill check
    Else
        ' Pass some game time
        PassTime (20)
        ' Subtract fatigue points
        ' Display appropriate message
        MsgBox ("You were unable to make a bench.")
        ' Remove the wood from the equipment list
        Remove (Equipment.ListIndex)
        ' Give player a chance to improve his skill
        CurrentPlayer.Carpentry = CurrentPlayer.Carpentry + Improve_Skill
    End If
    ' Update fatigue progress bar
    Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Forage for berris
Private Sub BerriesButton_Click()
    Dim i, j As Byte
    ForageFrame.Visible = False
    ' Make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 2 Then
        MsgBox ("You are too tired.")
' If player fails skill check
ElseIf RandomNumber() > (CurrentPlayer.Foraging + 25) Then
   ' Pass some game time
   PassTime (5)
   ' Subtract fatigue points
   ' Display appropriate message
   MsgBox ("You were unsuccessful at foraging for berries.")
   ' Give player a chance to improve
   CurrentPlayer.Foraging = CurrentPlayer.Foraging + Improve_Skill
Else
   ' Pass some game time
   PassTime (10)
   ' Subtract fatigue points
   ' Add some berries to the player's inventory
   For i = 1 To (2 * ((RandomNumber() Mod 4) + 1))
      Call Add(ItemArray(34))
   Next i
   ' Show a messagebox with a picture of berries
   Call Show_MessageBox(CurrentImageDrive + ItemArray(34).Picture, "You have successfully foraged some berries.")
End If
   ' Update fatigue progress bar
   Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Craft a bow
Private Sub BowCommand_Click()
   CarpentryFrame.Visible = False
   ' Make sure player has enough fatigue
   If CurrentPlayer.Fatigue < 6 Then
      MsgBox ("You are too tired.")
   ' If player passes skill check
   ElseIf CurrentPlayer.Carpentry > RandomNumber() Then
      ' Pass some game time
      PassTime (30)
      ' Subtract fatigue points
      CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 6
      ' Show a messagebox with a picture of a bow
      Call Show_MessageBox(CurrentImageDrive + ItemArray(17).Picture, "You have successfully crafted a bow.")
      ' Remove the wood from the equipment list
      Remove (Equipment.ListIndex)
      ' Add a bow to the player's inventory
      Call Add(ItemArray(17))
   Else
      ' Pass some game time
      PassTime (15)
      ' Subtract fatigue points
      ' Display appropriate message
      MsgBox ("You were unable to make a bow.")
      ' Give player a chance to improve his skill
      CurrentPlayer.Carpentry = CurrentPlayer.Carpentry + Improve_Skill
      ' Remove the wood from the equipment list
      Remove (Equipment.ListIndex)
   End If
   ' Update fatigue progress bar
   Fatigue.Value = CurrentPlayer.Fatigue
End Sub
'Butcher a carcass
Private Sub Butchery()
  ' Make sure player is holding a carcass
  If Equipment.text <> "carcass" Then
    MsgBox ("You must be holding the carcass in order to skin it.")
  ' Make sure player has enough fatigue
  ElseIf CurrentPlayer.Fatigue < 5 Then
    MsgBox ("You are too tired.")
  Else
    ' If player fails skill check
    If CurrentPlayer.Butchery < RandomNumber() Then
      ' Pass some time
      PassTime (10)
      ' Subtract fatigue points
      ' Display appropriate message
      MsgBox ("You were unable to butcher the carcass.")
      ' Give player a chance to improve skill
      ' Remove the carcass from the inventory
      Remove (Equipment.ListIndex)
    Else
      ' If player passes skill check
      Else
        ' Pass some game time
        PassTime (20)
        ' Subtract fatigue points
        ' Show a messagebox with a picture of a meat
        Call Show_MesageBox(CurrentImageDrive + ItemArray(30).Picture, "You successfully butchered the carcass.")
        ' Pause to let them read the messagebox.
        Pause (3)
        ' Add meat to player's inventory
        Call Add(ItemArray(30))
        ' See if player also wants to skin carcass
        If CurrentPlayer.Skinning > 0 Then
          Dim choice As VbMsgBoxResult
          choice = MsgBox("Would you like to skin the carcass also?", vbYesNo)
          ' If player wants to skin carcass
          If choice = vbYes Then
            ' Make sure player has enough fatigue
            If CurrentPlayer.Fatigue < 4 Then
              MsgBox ("You are too tired.")
            ' If player fails skill check
            ElseIf CurrentPlayer.Skinning < RandomNumber() Then
              ' Pass some game time
              PassTime (5)
              ' Subtract fatigue points
              CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 1
              ' Display appropriate message
              MsgBox ("You were unable to skin the carcass.")
              ' Give player a chance to improve his skill
              CurrentPlayer.Skinning = CurrentPlayer.Skinning + Improve_Skill
            Else
              ' If player passes skill check
              Else
                ' Pass some game time
                PassTime (10)
                ' Subtract fatigue points
                CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 4
                ' Show a messagebox with a picture of a fur pelt

Call Show_MessageBox(CurrentImageDrive + ItemArray(41).Picture, "You have successfully skinned the carcass.")
' Add a fur pelt to the player's inventory
Call Add(ItemArray(41))
End If
' Remove the carcass from the equipment list
Remove (Equipment.ListIndex)
End If
End If
' Update the fatigue progress bar
Fatigue.Value = currentPlayer.Fatigue
End Sub

' Camping
Private Sub Camping()
' Check to see if player can camp at current location
' If Player CAN camp here
If (Map(Cur_x + 3, Cur_y + 3)) = 32 Or (Map(Cur_x + 3, Cur_y + 3)) = 33 _
Or (Map(Cur_x + 3, Cur_y + 3)) = 43 Or (Map(Cur_x + 3, Cur_y + 3)) = 44 _
Or (Map(Cur_x + 3, Cur_y + 3)) = 45 Or (Map(Cur_x + 3, Cur_y + 3)) = 46 _
Or (Map(Cur_x + 3, Cur_y + 3)) = 40 Then
' If player passes skill check
If currentPlayer.Camping > RandomNumber() Then
' Pass some game time
PassTime (15)
' Display appropriate message
Call Show_MessageBox(CurrentImageDrive + "campsite.jpg", "You have successfully set up camp.")
'Replace player image with tent image
mapgrid(49).Picture = LoadPicture(CurrentImageDrive + "tent.gif")
'Display Camping Menu
CampingMenu.Left = MapPicture.Left
CampingMenu.Top = MapPicture.Top
CampingMenu.Visible = True
' Check to see whether player knows how to cook
If currentPlayer.Cooking = 0 Then
  CookCommand.Enabled = False
End If
Else
' Pass some game time
PassTime (7)
' Display appropriate message
MsgBox ("You were unable to make a camp.")
' Give player a chance to improve his skill
currentPlayer.Camping = currentPlayer.Camping + Improve_Skill
End If
' If player can't camp on this terrain
Else
  MsgBox ("You are unable to camp on this terrain.")
End If
End Sub

' Turns player to game form
Private Sub CancelButton_Click()
  DiskOptions.Visible = False
End Sub

' Exit from Carpentry frame
Private Sub CancelCarpentryButton_Click()
CarpentryFrame.Visible = False
End Sub

' Exit from Carving frame
Private Sub CancelCarvingButton_Click()
    CarvingFrame.Visible = False
End Sub

' Exit from Foraging frame
Private Sub CancelForageButton_Click()
    ForageFrame.Visible = False
End Sub

' Carpentry
Private Sub Carpentry()
' Check to make sure player is holding some wood
    If Equipment.text <> "wood" Then
        MsgBox ("You must be holding some wood in order to craft it.")
    ' If he is holding wood, display the menu
    Else
        CarpentryFrame.Visible = True
        CarpentryFrame.Left = 200
        CarpentryFrame.Top = 200
        CarpentryFrame.Visible = True
    End If
End Sub

' Carving
Private Sub Carving()
' Check to make sure player is holding some wood
    If Equipment.text <> "wood" Then
        MsgBox ("You must be holding some wood in order to carve it.")
    ' If he is holding wood, display the menu
    Else
        CarvingFrame.Visible = True
        CarvingFrame.Left = 200
        CarvingFrame.Top = 200
        CarvingFrame.Visible = True
    End If
End Sub

' Craft a chair
Private Sub ChairCommand_Click()
    CarpentryFrame.Visible = False
    ' Make sure the player has enough fatigue
    If CurrentPlayer.Fatigue < 10 Then
        MsgBox ("You are too tired.")
    ' If player passes skill check
    ElseIf CurrentPlayer.Carpentry > RandomNumber() Then
        ' Pass some game time
        PassTime (60)
        ' Subtract fatigue points
        CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 10
        ' Show a messagebox with a picture of a chair
        Call Show_MessageBox(CurrentImageDrive + ItemArray(Sl).Picture, "You have successfully crafted a chair.")
        ' Remove the wood from the equipment list
        Remove (Equipment.ListIndex)
        ' Add a chair to the player's inventory
        Call Add(ItemArray(51))
    ' If player fails skill check
    Else
        MsgBox ("You cannot craft this chair with your current skill level.")
    End If
End Sub
Else
' Pass some game time
PassTime (30)
' Subtract fatigue points
' Display appropriate message
MsgBox ("You were unable to make a chair.")
' Remove the wood from the equipment list
Remove (Equipment.ListIndex)
' Give player a chance to improve his skill
CurrentPlayer.Carpentry = CurrentPlayer.Carpentry + Improve_Skill
End If
' Update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Updates information on statistics label and makes it visible, makes character picture invisible
Private Sub Character_Click()
Character.Visible = False
Statistics.PlayerName = CurrentPlayer.Name
Statistics.PlayerAge = CurrentPlayer.Age
Statistics.PlayerArmor = CurrentPlayer.Armor
Statistics.PlayerEvade = CurrentPlayer.Evade
Statistics.PlayerFitness = CurrentPlayer.Fitness
Statistics.PlayerHeight = CurrentPlayer.Height
Statistics.PlayerIntelligence = CurrentPlayer.Intelligence
Statistics.PlayerPersonality = CurrentPlayer.Personality
Statistics.PlayerStrength = CurrentPlayer.Strength
Statistics.PlayerWeight = CurrentPlayer.Weight
Statistics.Top = TopLine.Y1 + 0.5
Statistics.Left = LeftLine.X1 + 0.5
Statistics.Height = LeftLine.Y2 - LeftLine.Y1 - 0.5
Statistics.Width = TopLine.X2 - TopLine.X1 - 0.5
Statistics.Visible = True
End Sub

' Written by Tim Zoch
Public Sub Check_Armor()
CurrentPlayer.Armor = 1
' first, check the armor worn and add the ac bonus
' next, calculate the evade modifier
Dim tempevade As Integer
If tempevade < 0 Then
    CurrentPlayer.Evade = 0
Else
    CurrentPlayer.Evade = tempevade
End If
End Sub
CheckItems
Checks to see if the player has all seven items, and if so,
replaces them with the final sword

Sub CheckItems()
    Dim HasDagger As Boolean
    HasDagger = False
    Dim DaggerIndex As Byte
    DaggerIndex = 0
    Dim HasGem1 As Boolean
    HasGem1 = False
    Dim Gem1Index As Byte
    Gem1Index = 0
    Dim HasGem2 As Boolean
    HasGem2 = False
    Dim Gem2Index As Byte
    Gem2Index = 0
    Dim HasCrosspiece As Boolean
    HasCrosspiece = False
    Dim CrosspieceIndex As Byte
    CrosspieceIndex = 0
    Dim HasPommel As Boolean
    HasPommel = False
    Dim PommelIndex As Byte
    PommelIndex = 0
    Dim HasHilt As Boolean
    HasHilt = False
    Dim HiltIndex As Byte
    HiltIndex = 0
    Dim HasScabbard As Boolean
    HasScabbard = False
    Dim ScabbardIndex As Byte
    ScabbardIndex = 0
    Dim i As Byte
    For i = 0 To CurrentPlayer.NextIndex - 1
        If CurrentPlayer.Inventory(i).IDNumber = 19 Then
            HasDagger = True
        ElseIf CurrentPlayer.Inventory(i).IDNumber = 20 Then
            HasGem1 = True
        ElseIf CurrentPlayer.Inventory(i).IDNumber = 21 Then
            HasGem2 = True
        ElseIf CurrentPlayer.Inventory(i).IDNumber = 22 Then
            HasCrosspiece = True
        ElseIf CurrentPlayer.Inventory(i).IDNumber = 24 Then
            HasPommel = True
        ElseIf CurrentPlayer.Inventory(i).IDNumber = 23 Then
            HasHilt = True
        ElseIf CurrentPlayer.Inventory(i).IDNumber = 25 Then
            HasScabbard = True
        End If
    Next i
    If (HasDagger = True And HasGem1 = True And HasGem2 = True And HasCrosspiece = True And HasPommel = True And HasHilt = True And HasScabbard = True) Then
        Remove (DaggerIndex)
        Remove (Gem1Index)
        Remove (Gem2Index)
        Remove (CrosspieceIndex)
        Remove (PommelIndex)
        Remove (HiltIndex)
        Remove (ScabbardIndex)
' Written by Jeff Hanks
' This is the function for determining wandering monster types.
' This function is only called if there is definitely a wandering
' monster to be found. This function will then determine what
' exactly the wandering monster is based on the terrain type that
' the character is currently walking on.
Function DetermineWanderingType(i As Integer) As Integer
Dim x, cum As Integer

cum = 0
With MonsterData.Recordset
  .MoveFirst
  While .EOF = False
    If .Fields("Found1").Value = i Then
      cum = cum + (.Fields("Encounter1").Value * .Fields("Rarity").Value)
    ElseIf .Fields("Found2").Value = i Then
      cum = cum + (.Fields("Encounter2").Value * .Fields("Rarity").Value)
    ElseIf .Fields("Found3").Value = i Then
      cum = cum + (.Fields("Encounter3").Value * .Fields("Rarity").Value)
    End If
  .MoveNext
  Wend
  x = Int(Rnd() * cum) + 1
  .MoveFirst
  cum = 0
  While ((.EOF = False) And (x > cum))
    If .Fields("Found1").Value = i Then
      cum = cum + (.Fields("Encounter1").Value * .Fields("Rarity").Value)
    ElseIf .Fields("Found2").Value = i Then
      cum = cum + (.Fields("Encounter2").Value * .Fields("Rarity").Value)
    ElseIf .Fields("Found3").Value = i Then
      cum = cum + (.Fields("Encounter3").Value * .Fields("Rarity").Value)
    End If
  .MoveNext
  Wend
  .MovePrevious
  MsgBox ("You have been attacked by a " & .Fields("Name").Value & ",")
  DetermineWanderingType = .Fields("IDnum").Value
End With
End Function

' Written by Jeff Hanks
' This function determines if movement from one location to the
' produces a wandering monster encounter. The percentage chance
' of a wandering monster encounter is dependent on the type of the
' terrain the character is presently on. If a wandering monster is
' found then the above DetermineWanderingType() function is called
' to determine the exact type of the wandering monster encountered.
Function CheckWandering(i As Integer)
Dim rand As Integer
Dim cum As Integer

cum = 0
rand = RandomNumber()
With MonsterData.Recordset

Select Case i
  Case 43, 45: ' Grass Terrain
    If rand <= 10 Then
      CurrentMonsterNumber = DetermineWanderingType(0)
      AttackForm.Show vbModal, Me
    End If
  Case 44, 46, 57, 58, 59, 60: ' Snow Terrain
    If rand <= 12 Then
      CurrentMonsterNumber = DetermineWanderingType(1)
      AttackForm.Show vbModal, Me
    End If
  Case 40, 41: ' hills
    If rand <= 15 Then
      CurrentMonsterNumber = DetermineWanderingType(2)
      AttackForm.Show vbModal, Me
    End If
  Case 41: ' snowy hills
    If rand <= 17 Then
      CurrentMonsterNumber = DetermineWanderingType(3)
      AttackForm.Show vbModal, Me
    End If
  Case 34 To 37: ' stone/rocks
    If rand <= 12 Then
      CurrentMonsterNumber = DetermineWanderingType(4)
      AttackForm.Show vbModal, Me
    End If
  Case 30, 53, 54, 55, 56: ' bridge
    If rand <= 10 Then
      CurrentMonsterNumber = DetermineWanderingType(6)
      AttackForm.Show vbModal, Me
    End If
  Case 47, 48, 66, 67, 68, 69: ' desert
    If rand <= 15 Then
      CurrentMonsterNumber = DetermineWanderingType(7)
      AttackForm.Show vbModal, Me
    End If
  Case 42, 61: ' swamp
    If rand <= 20 Then
      CurrentMonsterNumber = DetermineWanderingType(8)
      AttackForm.Show vbModal, Me
    End If
  Case 32: ' forest
    If rand <= 15 Then
      CurrentMonsterNumber = DetermineWanderingType(9)
      AttackForm.Show vbModal, Me
    End If
  Case 33: ' winter forest
    If rand <= 18 Then
      CurrentMonsterNumber = DetermineWanderingType(10)
      AttackForm.Show vbModal, Me
    End If
End Select
End With
' Update hitpoints and fatigue progress bars
  HitPoints.Value = CurrentPlayer.HitPoints
  Fatigue.Value = CurrentPlayer.Fatigue
End Function

'Written by Jeff Hanks
'This function performs the movement from one tile on the map
' to the next tile. This function takes into consideration a
characters skill in hiking, jogging, and mountaineering in order to give the character positive modifiers on their fatigue and the amount of time that it takes them to move.

```vba
Function PerformMovement(i As Integer, j As Integer)
    Dim y As Integer
    Dim x As Single
    Dim a, b As Single
    If RandomNumber() < CurrentPlayer.Hiking Then
        y = (Rnd() * 5) + 1
    End If
    If RandomNumber() < CurrentPlayer.Jogging Then
        y = y + (Rnd() * 10) + 1
    End If
    If RandomNumber() < CurrentPlayer.Mountaineering Then
        y = y + (Rnd() * 15) + 1
    End If
    x = (100 - y) / 100
    a = i * x
    b = j * x
    timeclock1.minutes = timeclock1.minutes + b
    y = CurrentPlayer.Fatigue - a
    If y < 1 Then
        CurrentPlayer.Fatigue = 0
        MsgBox("You are now exhausted!")
    Else
        CurrentPlayer.Fatigue = y
    End If
End Function
```

This function performs the movement throughout the game. Every movement that the character makes goes through this function which will determine the terrain type that the character is moving onto and will adjust the clock and the player's fatigue accordingly.

```vba
Function CheckTerrain(tx As Integer, ty As Integer) As Integer
    Dim i As Integer
    Select Case (Map(tx + 3, ty + 3))
        Case 0, 1, 2, 3, 11 To 29, 38 To 39, 62:
        Case Else
            Cur_x = tx
            Cur_y = ty
            For i = 0 To 48
                PastePicture (i)
            Next i
            ' The following part of this function increments the clock on the display in the lower right corner depending on the type of terrain they are moving onto (i.e. moving onto hills takes longer than moving onto grass.
            CheckWandering (Map(tx + 3, ty + 3))
            Select Case (Map(tx + 3, ty + 3))
                Case 43, 45: 'grass terrain
                    PerformMovement 1, 20
                Case 44, 46: 'snow terrain
                    PerformMovement 2, 25
                Case 36 To 37: 'stones
                    PerformMovement 3, 15
            End Select
        End Case
    End Select
End Function
```
PerformMovement 3, 45
Case 34 To 35: 'rock
   PerformMovement 4, 55
Case 30: 'bridge
   PerformMovement 1, 30
Case 47, 48, 66, 67, 68, 69, 65: 'desert
   PerformMovement 3, 40
Case 42, 61: 'swamp
   PerformMovement 3, 50
Case 40 To 41: 'hills
   PerformMovement 3, 75
Case 32 To 33: 'forest
   PerformMovement 2, 60
Case 4 To 11: '1/4 water squares
   PerformMovement 2, 35
Case 31: 'waterfall
   PerformMovement 2, 40
Case 49 To 56, 70 To 75: 'various buildings
   PerformMovement 1, 15
Case 57 To 60: 'snow/grass transitions
   PerformMovement 2, 23
Case Else 'This message box should no longer occur.
   MsgBox ("Error! Please report the current " & _
   Chr(13) & "terrain type to the developers")
End Select
End Select
' Following written by Karen Woznick
' Checks to see if the tile the player has just
' entered is a town or one of the special monsters
' Village
If (((tx + 3) = 60) And ((ty + 3) = 37)) Then
   CurrentTown = 0
   TownForm.Show vbModal, Me
   ' Update hitpoints and fatigue progress bars
   HitPoints.Value = CurrentPlayer.HitPoints
   Fatigue.Value = CurrentPlayer.Fatigue
' Southern City
ElseIf (((tx + 3) = 62) And ((ty + 3) = 48)) Then
   CurrentTown = 1
   TownForm.Show vbModal, Me
   ' Update hitpoints and fatigue progress bars
   HitPoints.Value = CurrentPlayer.HitPoints
   Fatigue.Value = CurrentPlayer.Fatigue
' Desert City
ElseIf (((tx + 3) = 5) And ((ty + 3) = 41)) Then
   CurrentTown = 2
   TownForm.Show vbModal, Me
   ' Update hitpoints and fatigue progress bars
   HitPoints.Value = CurrentPlayer.HitPoints
   Fatigue.Value = CurrentPlayer.Fatigue
' Ocean City
ElseIf (((tx + 3) = 17) And ((ty + 3) = 57)) Then
   CurrentTown = 3
   TownForm.Show vbModal, Me
   ' Update hitpoints and fatigue progress bars
   HitPoints.Value = CurrentPlayer.HitPoints
   Fatigue.Value = CurrentPlayer.Fatigue
' StoneWall
ElseIf (((tx + 3) = 54) And ((ty + 3) = 34)) Then
   CurrentTown = 4
   TownForm.Show vbModal, Me
' Update hitpoints and fatigue progress bars
HitPoints.Value = CurrentPlayer.HitPoints
Fatigue.Value = CurrentPlayer.Fatigue

' Waterfall
ElseIf (((tx + 3) = 26) And ((ty + 3) = 34)) Then
    Call Show_MessageBox(CurrentImageDrive + ItemArray(22).Picture, "You have recovered a gem from behind the waterfall.")
    Call Add(ItemArray(21))
    Call CheckItems
' Abandoned castle
ElseIf (((tx + 3) = 33) And ((ty + 3) = 12)) Then
    Call Show_MessageBox(CurrentImageDrive + ItemArray(25).Picture, "You have discovered a scabbard in the abandoned castle.")
    Call Add(ItemArray(25))
    Call CheckItems
' Golem
ElseIf (((tx + 3) = 76) And ((ty + 3) = 60)) Then
    CurrentMonsterNumber = 28
    AttackForm.Show vbModal, Me
    ' Update hitpoints and fatigue progress bars
    HitPoints.Value = CurrentPlayer.HitPoints
    Fatigue.Value = CurrentPlayer.Fatigue
' Vorga
ElseIf (((tx + 3) = 12) And ((ty + 3) = 24)) Then
    CurrentMonsterNumber = 30
    AttackForm.Show vbModal, Me
    ' Update hitpoints and fatigue progress bars
    HitPoints.Value = CurrentPlayer.HitPoints
    Fatigue.Value = CurrentPlayer.Fatigue
    End If
End Function

' Chopping
Private Sub Chopping()
    ' Make sure there is a tree to chop
    If (Map(Cur_x + 3, Cur_y + 3)) <> 32 And (Map(Cur_x + 3, Cur_y + 3)) <> 33 Then
        MsgBox ("You must be standing by a tree in order to chop it down.")
    ' Make sure player is wielding something that can chop down a tree
    ElseIf CurrentPlayer.Wielded.ItemType <> 1 And CurrentPlayer.Wielded.ItemType <> 4 _
        And CurrentPlayer.Wielded.ItemType <> 2 Then 'Must be wielding a bladed weapon
        MsgBox ("You must be wielding a bladed weapon to chop down a tree.")
    ' Make sure player has enough fatigue
    ElseIf CurrentPlayer.Fatigue < (100 / CurrentPlayer.Chopping) Then 'must have enough fatigue points to chop
        MsgBox ("You are too tired to chop down a tree, better rest some.")
    ' Chop down the tree
    Else
        PassTime ((100 / CurrentPlayer.Chopping) * 10)
        ' Subtract fatigue points
        ' Show appropriate message
        Call Show_MessageBox(CurrentImageDrive + ItemArray(39).Picture, "You have finished chopping down the tree.")
    ' Add some wood to the player's inventory
    Call Add(ItemArray(39))
    ' See if player gets a chance to improve his skill
    If CurrentPlayer.Chopping < RandomNumber() Then
        CurrentPlayer.Chopping = CurrentPlayer.Chopping + Improve_Skill
    End If
End If
End If
' Update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' okCommand
Private Sub CookCommand_Click()
' Calls cooking function
Cooking
End Sub

Private Sub Cooking()
' Make sure the player has built a fire
If Fire = False Then
    MsgBox ("You must first build a fire.")
' Make sure player is holding some meat or a fish
ElseIf Equipment.text <> "meat" And Equipment.text <> "fish" Then
    MsgBox ("You need to be holding the meat or fish in order to cook it.")
Else
' Make sure player has enough fatigue
If CurrentPlayer.Fatigue < 1 Then
    MsgBox ("You are too tired.")
' If player passes the skill check
ElseIf CurrentPlayer.Cooking > RandomNumber() Then
    ' Pass some game time
    PassTime (25)
    ' Subtract fatigue points
    CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 1
    ' Display appropriate message
    Call Show_MessageBox(CurrentImageDrive + "potboil.gif", "You have successfully cooked dinner.")
    ' Remove the meat from the player's inventory
    Remove (Equipment.ListIndex)
    ' Add some dinner to the player's inventory
    Call Add(ItemArray(50))
' If player fails the skill check
Else
    ' Pass some game time
    PassTime (10)
    ' Subtract fatigue points
    CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 1
    ' Display appropriate message
    MsgBox ("You were unable to cook yourself dinner.")
    ' Remove the incorrectly cooked item
    Remove (Equipment.ListIndex)
    ' Give player a chance to improve his skill
End If
End If
'Update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Makes disk options menu visible
Private Sub Disk_Click()
    MousePointer = 0
    DiskOptions.Left = 200
    DiskOptions.Top = 200
    DiskOptions.Visible = True
End Sub

' Written by Jeff Hanks
This function will load the initial game map at the beginning of the game. Additionally, it will create the 7 * 7 grid which is the background for the movement system of the game.

Private Sub DisplayMap()
    Dim i As Integer

    For i = 1 To 48
        Load mapgrid(i)
    Next i

    For i = 0 To 49
        mapgrid(i).Visible = True
        mapgrid(i).Stretch = True
        mapgrid(i).Width = 65
        mapgrid(i).Height = 67
        mapgrid(i).BorderStyle = 0
        mapgrid(i).Left = (i Mod 7) * 65
        mapgrid(i).Top = (i \ 7) * 65
    Next i

    mapgrid(49).Left = mapgrid(24).Left
    mapgrid(49).Top = mapgrid(24).Top
    mapgrid(49).Picture = LoadPicture(DRIVE & "femaleright.gif")
End Sub

' Equipment_Click
Private Sub Equipment_Click()
    ' First make sure there is something selected
    If Equipment.text <> "" Then
        ' If player wants to identify the item
        If State = "Eye" Then
            State = "Normal"
            MousePointer = 0
            Call Identify_Item(CurrentPlayer.Inventory(Equipment.ListIndex))
        ' If player wants to remove the item from his inventory
        ElseIf State = "Hand" Then
            State = "Normal"
            MousePointer = 0
            ' Don't let him remove the special items!!
            If ((CurrentPlayer.Inventory(Equipment.ListIndex).IDNumber > 17) And (CurrentPlayer.Inventory(Equipment.ListIndex).IDNumber < 26)) Then
                MsgBox ("You cannot remove this item from your inventory.")
            Else
                ' Make sure player really wants to discard item
                Dim choice As VbMsgBoxResult
                choice = MsgBox("Are you sure you want to remove " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & " from your inventory?", vbYesNo)
                ' If he does want to discard it
                If choice = vbYes Then
                    ' Remove the item from the player's inventory
                    Remove (Equipment.ListIndex)
                End If
            End If
        End If
    End If

    Select Case CurrentPlayer.Inventory(Equipment.ListIndex).ItemType
    Case 8:
        ' Display appropriate message
        MsgBox ("You have eaten the " + CurrentPlayer.Inventory(Equipment.ListIndex).Name + ")
    Case 8 + "."
        ' Remove item from inventory
Remove (Equipment.ListIndex)
' Add fatigue points
CurrentPlayer.Fatigue = CurrentPlayer.Fatigue + 2
' Make sure we have not exceeded the player's max fatigue points
If CurrentPlayer.Fatigue > CurrentPlayer.Max_Fp Then
End If
' Add hit points
CurrentPlayer.HitPoints = CurrentPlayer.HitPoints + 2
' Make sure we have not exceeded the player's max hit points
If CurrentPlayer.HitPoints > CurrentPlayer.Max_Hp Then
End If
' Elseif item is Clothing
Case 7:
    Select Case CurrentPlayer.Inventory(Equipment.ListIndex).Wear_Position
        Case 1:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", on your head.")
            Set CurrentPlayer.Head = CurrentPlayer.Inventory(Equipment.ListIndex)
        Case 2:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", on your body.")
            Set CurrentPlayer.Body = CurrentPlayer.Inventory(Equipment.ListIndex)
        Case 3:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", on your feet.")
            Set CurrentPlayer.Feet = CurrentPlayer.Inventory(Equipment.ListIndex)
        Case 4:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", about your body.")
            Set CurrentPlayer.Aboutbody = CurrentPlayer.Inventory(Equipment.ListIndex)
        Case 6:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", as a shield.")
    End Select
' Recalculate player's armor rating
Check_Armor
' Elseif item is Armor
Case 6:
    Select Case CurrentPlayer.Inventory(Equipment.ListIndex).Wear_Position
        Case 1:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", on your head.")
            Set CurrentPlayer.Head = CurrentPlayer.Inventory(Equipment.ListIndex)
        Case 2:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", on your body.")
            Set CurrentPlayer.Body = CurrentPlayer.Inventory(Equipment.ListIndex)
        Case 3:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", on your feet.")
            Set CurrentPlayer.Feet = CurrentPlayer.Inventory(Equipment.ListIndex)
        Case 4:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", about your body.")
            Set CurrentPlayer.Aboutbody = CurrentPlayer.Inventory(Equipment.ListIndex)
        Case 6:
            MsgBox("You are now wearing " & CurrentPlayer.Inventory(Equipment.ListIndex).Name & ", as a shield.")
    End Select
' Recalculate player's armor rating
Check_Armor
' Elseif item is a Weapon
Case 1 To 5:
    Set CurrentPlayer.Wielded = CurrentPlayer.Inventory(Equipment.ListIndex)
MsgBox ("You are now wielding " & CurrentPlayer.Inventory(Equipment.ListIndex).N
ame & ")."
End Select
End If
Elseif item is a Weapon
    Set CurrentPlayer.Wielded = CurrentPlayer.Inventory(Equipment.ListIndex)
    MsgBox ("You are now wielding " & CurrentPlayer.Inventory(Equipment.ListIndex).N
ame & ")."
End Select
Elseif item is a Weapon
    Set CurrentPlayer.Wielded = CurrentPlayer.Inventory(Equipment.ListIndex)
    MsgBox ("You are now wielding " & CurrentPlayer.Inventory(Equipment.ListIndex).N
ame & ")."
End Select
Else
    Update hitpoints and fatigue progress bars
    Fatigue.Value = CurrentPlayer.Fatigue
    HitPoints.Value = CurrentPlayer.HitPoints
End Sub

' Changes State to "eye" if button was not previously
' pushed; else cancels last click and returns state
' to normal. Changes mousepointer appropriately.
Private Sub Eyeball_Click()
    If State = "Eye" Then
        State = "Normal"
        MousePointer = 0
    Else
        MouseIcon = LoadPicture(CurrentImageDrive + "bullseye.ico")
        MousePointer = 99
        State = "Eye"
    End If
End Sub

' -re Building
Private Sub Fire_Building()
    ' First check to see if player is holding wood
    If Equipment.text = "wood" Then
        Make sure player has enough fatigue
        If CurrentPlayer.Fatigue < 2 Then
            MsgBox ("You are too tired.")
        ' If terrain is snowy :
        ' 33 : tree in snow
        ' 35 : rocks in snow
        ' 37 : stones in snow
        ' 41 : snow-covered hill
        ' 44 : snowy grass
        ' 46 : normal snow
        ' > 56 and < 61 : snow transitions
        ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 33 Or (Map(Cur_x + 3, Cur_y + 3)) = 35 _
            Or (Map(Cur_x + 3, Cur_y + 3)) = 37 Or (Map(Cur_x + 3, Cur_y + 3)) = 41 _
            Or (Map(Cur_x + 3, Cur_y + 3)) = 44 Or (Map(Cur_x + 3, Cur_y + 3)) = 46 _
            Or ((Map(Cur_x + 3, Cur_y + 3)) > 56 And (Map(Cur_x + 3, Cur_y + 3)) < 61) Then
            ' Pass some game time
            PassTime (10)
            ' Subtract fatique points
        ' If player passes skill check
        If CurrentPlayer.Firebuilding - 15 > RandomNumber() Then
            Call Show_MessageBox(CurrentImageDrive + "fire.jpg", "You have successfully buil
t fire.")
            Fire = True
            ' Remove the wood from his inventory
            Remove (Equipment.ListIndex)
        ' If player fails skill check
        Else
            Remove (Equipment.ListIndex)
        End If
    End If
    Else
        Update hitpoints and fatigue progress bars
        Fatigue.Value = CurrentPlayer.Fatigue
        HitPoints.Value = CurrentPlayer.HitPoints
    End Sub
MsgBox ("You were unable to start a fire.")
' Give player a chance to improve his skill
CurrentPlayer.Firebuilding = CurrentPlayer.Firebuilding + Improve_Skill
End If
' If terrain is swampy
' 42 : swamp
' 61 : rock in swamp
ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 42 Or (Map(Cur_x + 3, Cur_y + 3)) = 61 Then
    ' Pass some game time
    PassTime (10)
    ' Subtract fatigue points
    ' If player passes the skill check
    If CurrentPlayer.Firebuilding - 10 > RandomNumber() Then
        Call Show_MessageBox(CurrentImageDrive + "fire.jpg", "You have successfully built a fire.")
        Fire = True
        ' Remove the wood from the player's inventory
        Remove (Equipment.ListIndex)
        ' If player fails the skill check
    Else
        MsgBox ("You were unable to start a fire.")
        ' Give player a chance to improve his skill
        CurrentPlayer.Firebuilding = CurrentPlayer.Firebuilding + Improve_Skill
    End If
' All other terrain types
Else
' Pass some game time
    PassTime (10)
    ' Subtract fatigue points
    ' If player passes skill check
    If CurrentPlayer.Firebuilding > RandomNumber() Then
        Call Show_MessageBox(CurrentImageDrive + "fire.jpg", "You have successfully built a fire.")
        Fire = True
        ' Remove the wood from the player's inventory
        Remove (Equipment.ListIndex)
        ' If player fails the skill check
    Else
        MsgBox ("You were unable to start a fire.")
        ' Give player a chance to improve his skill
        CurrentPlayer.Firebuilding = CurrentPlayer.Firebuilding + Improve_Skill
    End If
End If
' If player is not holding some wood
Else
    MsgBox ("You need to be holding some wood in order to start a fire.")
End If
' Update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub

Private Sub Fishing()
    ' If player is standing next to water
    If (Map(Cur_x + 3, Cur_y + 3)) < 32 Or (Map(Cur_x + 2, Cur_y + 2)) < 32 _
       Or (Map(Cur_x + 3, Cur_y + 2)) < 32 Or (Map(Cur_x + 4, Cur_y + 2)) _
       Or (Map(Cur_x + 2, Cur_y + 3)) < 32 Or (Map(Cur_x + 4, Cur_y + 3)) _
       Or (Map(Cur_x + 2, Cur_y + 4)) < 32 Or (Map(Cur_x + 3, Cur_y + 4)) _
       Or (Map(Cur_x + 4, Cur_y + 4)) Then
GameForm - 20

' Make sure player is holding a fishing pole
If Equipment.text <> "fishing pole" Then
    MsgBox ("You must be holding a fishing pole in order to fish.")
Else
    ' Make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 5 Then
        MsgBox ("You are too tired.")
    ' If player fails skill check
    ElseIf CurrentPlayer.Fishing < RandomNumber Then
        ' Pass some game time
        PassTime (15)
        ' Subtract fatigue points
        ' Display appropriate message
        MsgBox ("You were unable to catch a fish.")
        ' Give player a chance to improve his skill
        CurrentPlayer.Fishing = CurrentPlayer.Fishing + Improve_Skill
    ' If player passes the skill check
    Else
        ' Pass some game time
        PassTime (20)
        ' Subtract fatigue points
        ' Add a fish to the player's inventory
        Call Add(ItemArray(31))
        ' Display an appropriate message
        Call Show_MessageBox(CurrentImageDrive + ItemArray(31).Picture, "You have successfully caught a fish.")
    End If
ElseIf player is not standing next to water. Then
    MsgBox ("You must be standing next to water in order to fish.")
End If
' Update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Foraging
Private Sub Foraging()
    'Determining if player is in an area that allows foraging
    'If player CAN forage
    If (Map(Cur_x + 3, Cur_y + 3)) = 32 Or (Map(Cur_x + 3, Cur_y + 3)) = 33
    Or (Map(Cur_x + 3, Cur_y + 3)) = 43 Or (Map(Cur_x + 3, Cur_y + 3)) = 45
    Or (Map(Cur_x + 3, Cur_y + 3)) = 53 Then
        'Display Forage Menu
        ForageFrame.Left = 200
        ForageFrame.Top = 200
        ForageFrame.Visible = True
        'Else the player cannot forage here
        Else
            MsgBox ("You are unable to forage in this area.")
    End If
End Sub

' aft a fishing pole
Private Sub FishingPoleCommand_Click()
    CarvingFrame.Visible = False
    ' Make sure player has enough fatigue points
    If CurrentPlayer.Fatigue < 2 Then
        MsgBox ("You are too tired.")
' If player passes skill check
ElseIf CurrentPlayer.Carving > RandomNumber() Then
  ' Pass some game time
  PassTime (10)
  ' Subtract fatigue points
  ' Show a messagebox with a picture of a fishing pole
  Call Show_MessageBox(CurrentImageDrive + ItemArray(38).Picture, "You have successfully carved a fishing pole.")
  ' Remove the wood from the player’s equipment list
  Remove (Equipment.ListIndex)
  ' Add a fishing pole to the player’s inventory
  Call Add(ItemArray(38))
Else
  ' If player fails skill check
  ' Pass some game time
  PassTime (5)
  ' Subtract fatigue points
  CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 1
  ' Display appropriate message
  MsgBox ("You were unable to make a fishing pole.")
  ' Remove the wood from the player’s inventory
  Remove (Equipment.ListIndex)
  ' Give player a chance to improve his skill
End If
' Update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue

Private Sub Form_Load()
  ' Initializes state
  State = "Normal"
  ' Sets up game form
  Call Initial_Setup

  PlaceInBuffer
  DisplayMap
  LoadFromFile

  ' Shows startup screen
  FirstScreen.Show vbModal, Me
  LoadInitialPosition Cur_x, Cur_y
  ' Loads the correct character picture
  Character.Picture = LoadPicture(CurrentImageDrive + CurrentPlayer.Picture)
  ' Sets the max value of the progress bar to the player’s max_hp
  HitPoints.Max = CurrentPlayer.Max_Hp
  ' Set the value of the health progress bar to the player’s health.
  HitPoints.Value = CurrentPlayer.HitPoints
  ' Sets the max value of the progress bar to the player’s max_fp
  Fatigue.Max = CurrentPlayer.Max_Fp
  ' Set the value of the fatigue progress bar to the player’s current fatigue.
  Fatigue.Value = CurrentPlayer.Fatigue

  ' Add all of the player’s currently available skills to the combo box
  Update_Skills
  ' Enable the timer
  Timer1.Enabled = True
  ' Initialize variables from module
  CurrentMonsterNumber = -1
  CurrentTown = -1
Fire = False
End Sub

FruitButton_Click
Private Sub FruitButton_Click()
    ForageFrame.Visible = False
    ' Make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 5 Then
        MsgBox("You are too tired.")
    ' If player fails skill check
    ElseIf RandomNumber() > (CurrentPlayer.Foraging + 30) Then
        ' Pass some game time
        PassTime (5)
        ' Subtract fatigue points
        ' Display appropriate message
        Call MsgBox("You were not successful at foraging for fruit.")
        ' Give player a chance to improve his skill
        CurrentPlayer.Foraging = CurrentPlayer.Foraging + Improve_Skill
    ' If player passes the skill check
    Else
        ' Pass some game time
        PassTime (15)
        ' Subtract fatigue points
        ' Add some fruit to the player's inventory
        Dim i As Byte
        For i = 1 To (2 * ((RandomNumber() Mod 4) + 1))
            Call Add(ItemArray(32))
        Next i
        ' Show a messagebox with a picture of fruit
        Call Show_MessageBox(CurrentImageDrive + ItemArray(32).Picture, "You have successfully foraged some fruit.")
    End If
    ' Update the fatigue progress bar
    Fatigue.Value = CurrentPlayer.Fatigue
End Sub

Hand_Click
Private Sub Hand_Click()
    If State = "Hand" Then
        State = "Normal"
        MousePointer = 0
    Else
        State = "Hand"
        MouseIcon = LoadPicture(CurrentImageDrive + "arw11sw.ico")
        MousePointer = 99
    End If
End Sub

HerbsButton_Click
Private Sub HerbsButton_Click()
    ForageFrame.Visible = False
    ' Make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 8 Then
        MsgBox("You are too tired.")
    ' If player fails the skill check
    ElseIf RandomNumber() > (CurrentPlayer.Foraging + 10) Then
        ' Pass some time
        PassTime (15)
        ' Subtract fatigue points
        ' Add some fruit to the player's inventory
        Dim i As Byte
        For i = 1 To (2 * ((RandomNumber() Mod 4) + 1))
            Call Add(ItemArray(32))
        Next i
        ' Show a messagebox with a picture of fruit
        Call Show_MessageBox(CurrentImageDrive + ItemArray(32).Picture, "You have successfully foraged some fruit.")
    End If
    ' Update the fatigue progress bar
    Fatigue.Value = CurrentPlayer.Fatigue
End Sub
' Display appropriate message
MsgBox ("You were not successful at foraging for herbs.")
' Give player a chance to improve his skill
CurrentPlayer.Foraging = CurrentPlayer.Foraging + Improve_Skill
' If player passes the skill check
Else
   ' Pass some game time
   PassTime (20)
   ' Subtract fatigue points
   CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 8
   ' Add some herbs to the player's inventory
   Dim i As Byte
   For i = 1 To ((RandomNumber() Mod 4) + 1)
      Call Add(ItemArray(35))
   Next i
   ' Show a messagebox with a picture of some herbs
   Call Show_MessageBox(CurrentImageDrive + ItemArray(35).Picture, "You have successfully foraged some herbs.")
End If
' Update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Hunting
Private Sub Hunting()
   ' If player is on a bridge
   If (Map(Cur_x + 3, Cur_y + 3)) = 30 Then
      MsgBox ("You are unable to hunt while on a bridge")
   ' Player is on grasslands
   ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 43 Or (Map(Cur_x + 3, Cur_y + 3)) = 45 Then
      ' Make sure player has enough fatigue
      If CurrentPlayer.Fatigue < 10 Then
         MsgBox ("You are too tired.")
      Else
         ' Pass some game time
         PassTime (60)
         ' Subtract fatigue points
         CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 10
         ' If player fails skill check
         If (CurrentPlayer.Hunting + (CurrentPlayer.AnimalLore \ 10) + (CurrentPlayer.Stealth \ 25) - 5) < RandomNumber() Then
            MsgBox ("You were unable to kill anything.")
            ' Give player a chance to improve his skill
            CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
            If CurrentPlayer.AnimalLore < RandomNumber() Then
            End If
            If CurrentPlayer.Stealth < RandomNumber() Then
               CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
            End If
         ' If player passes skill check
         Else
            ' Add a carcass to the player's inventory
            Call Add(ItemArray(49))
            MsgBox ("You were successful.")
         End If
      End If
   Else
      ' Player is on hills
      ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 40 Then
         ' Make sure player has enough fatigue

If CurrentPlayer.Fatigue < 20 Then
    MsgBox ("You are too tired.")
Else
    ' Pass game time
    PassTime (60)
    ' Subtract fatigue
    CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 20
    ' If player fails skill check
        MsgBox ("You were unable to kill anything.")
        ' Give player a chance to improve his skill
        CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
        If CurrentPlayer.AnimalLore < RandomNumber() Then
        End If
        If CurrentPlayer.Stealth < RandomNumber() Then
            CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
        End If
    End If
    Else
    ' If player passes skill check
    Else
    ' Add a carcass to the player's inventory
    Call Add(ItemArray(49))
    MsgBox ("You were successful.")
End If
End If
' Player is on swamps
ElseIf (Map(Cur_x + 3, Cur_y + 3) = 42 Or (Map(Cur_x + 3, Cur_y + 3) = 61 Then
    ' Make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 20 Then
        MsgBox ("You are too tired.")
    Else
        ' Pass game time
        PassTime (60)
        ' Subtract fatigue points
        CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 20
        ' If player fails skill check
            MsgBox ("You were unable to kill anything.")
            ' Give player a chance to improve his skill
            CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
            If CurrentPlayer.AnimalLore < RandomNumber() Then
            End If
            If CurrentPlayer.Stealth < RandomNumber() Then
                CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
            End If
        End If
        Else
        ' If player passes the skill check
        Else
        ' Add a carcass to the player's inventory
        Call Add(ItemArray(49))
        MsgBox ("You were successful.")
    End If
End If
' Player is on trees
ElseIf (Map(Cur_x + 3, Cur_y + 3) = 32 Then
    ' Make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 12 Then
        MsgBox ("You are too tired.")
    Else
' Pass game time
PassTime (60)
' Subtract fatigue points
CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 12
' If player fails skill check
    MsgBox ("You were unable to kill anything.")
' Give player a chance to improve his skill
CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
If CurrentPlayer.AnimalLore < RandomNumber() Then
End If
If CurrentPlayer.Stealth < RandomNumber() Then
    CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
End If
' If player passes skill check
Else
' Add a carcass to the player's inventory
Call Add(ItemArray(49))
MsgBox ("You were successful.")
End If
End If
' Player is on desert
ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 47 Then
    ' Make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 20 Then
        MsgBox ("You are too tired.")
    Else
        ' Pass game time
        PassTime (60)
        ' Subtract fatigue points
        CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 20
        ' If player fails skill check
        If (CurrentPlayer.Hunting + (CurrentPlayer.AnimalLore \ 10) + (CurrentPlayer.Stealth \ 25) - 40) < RandomNumber() Then
            MsgBox ("You were unable to kill anything.")
' Give player a chance to improve his skill
CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
If CurrentPlayer.AnimalLore < RandomNumber() Then
End If
If CurrentPlayer.Stealth < RandomNumber() Then
    CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
End If
' If player passes skill check
Else
' Add a carcass to the player's inventory
Call Add(ItemArray(49))
MsgBox ("You were successful.")
End If
End If
' Player is snowy grasslands
ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 44 Then
    ' Make sure player has enough fatigue points
    If CurrentPlayer.Fatigue < 15 Then
        MsgBox ("You are too tired.")
    Else
        ' Pass game time
        PassTime (60)
        ' Subtract fatigue points
' if player fails skill check
If (CurrentPlayer.Hunting + (CurrentPlayer.AnimaLore \ 10) + (CurrentPlayer.Stealth - 25) - 10) < RandomNumber() Then
    MsgBox ('You were unable to kill anything.")
    ' give player a chance to improve his skill
    CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
    If CurrentPlayer.AnimaLore < RandomNumber() Then
    End If
    If CurrentPlayer.Stealth < RandomNumber() Then
        CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
    End If
Else
    ' player passes skill check
    If CurrentPlayer.AnimaLore < RandomNumber() Then
    End If
    If CurrentPlayer.Stealth < RandomNumber() Then
        CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
    End If
End If
' if player passes skill check
Else
    ' Add a carcass to the player's inventory
    Call Add(ItemArray(49))
    MsgBox ("You were successful.")
End If
End If
' Player is on snowy hills
ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 41 Then
    ' make sure player has enough fatigue points
    If CurrentPlayer.Fatigue < 20 Then
        MsgBox ("You are too tired.")
    Else
        ' pass game time
        PassTime (60)
        ' subtract fatigue
        CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 20
    ' if player fails skill check
        MsgBox ("You were unable to kill anything.")
        ' give player a chance to improve his skill
        CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
        If CurrentPlayer.AnimaLore < RandomNumber() Then
        End If
        If CurrentPlayer.Stealth < RandomNumber() Then
            CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
        End If
    End If
Else
    ' Add a carcass to the player's inventory
    Call Add(ItemArray(49))
    MsgBox ("You were successful.")
End If
End If
' Player is on snowy tree
ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 33 Then
    ' make sure player has enough fatigue points
    If CurrentPlayer.Fatigue < 15 Then
        MsgBox ("You are too tired.")
    Else
        ' pass game time
        PassTime (60)
        ' subtract fatigue points
        ' if player fails skill check
            MsgBox ("You were unable to kill anything.")
            ' give player a chance to improve his skill
            CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
            If CurrentPlayer.AnimaLore < RandomNumber() Then
            End If
            If CurrentPlayer.Stealth < RandomNumber() Then
                CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
            End If
        End If
    ' if player passes skill check
    Else
        If CurrentPlayer.AnimaLore < RandomNumber() Then
        End If
        If CurrentPlayer.Stealth < RandomNumber() Then
            CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
        End If
    End If
Else
    ' Add a carcass to the player's inventory
    Call Add(ItemArray(49))
    MsgBox ("You were successful.")
End If
End If
' Player is on snowy tree
ElseIf (Map(Cur_x + 3, Cur_y + 3)) = 33 Then
    ' make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 15 Then
        MsgBox ("You are too tired.")
    Else
        ' pass game time
        PassTime (60)
        ' subtract fatigue points
        ' if player fails skill check
            MsgBox ("You were unable to kill anything.")
            ' give player a chance to improve his skill
            CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
            If CurrentPlayer.AnimaLore < RandomNumber() Then
            End If
            If CurrentPlayer.Stealth < RandomNumber() Then
                CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
            End If
        End If
    ' if player passes skill check
    Else
        ' Add a carcass to the player's inventory
        Call Add(ItemArray(49))
        MsgBox ("You were successful.")
End If
End If
' Player is on snowy tree
GameForm - 27

\ 25) < RandomNumber() Then
  MsgBox ("You were unable to kill anything.")
  ' give player a chance to improve his skill
  CurrentPlayer.Hunting = CurrentPlayer.Hunting + Improve_Skill
  If CurrentPlayer.AnimalLore < RandomNumber() Then
  End If
  If CurrentPlayer.Stealth < RandomNumber() Then
    CurrentPlayer.Stealth = CurrentPlayer.Stealth + Improve_Skill
  End If
  ' if player passes the skill check
  Else
    ' Add a carcass to the player's inventory
    Call Add(ItemArray(49))
    MsgBox ("You were successful.")
  End If
  ' if player can not hunt on this terrain
  Else
    MsgBox ("You are unable to hunt on this land")
  End If
  ' update fatigue progress bar
  Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Identify_Item
' Displays a message describing what the item is that
' the player wants to identify. A more descriptive message
' is give if the player's item lore skill rating is high
Public Sub Identify_Item(thing As Item)
  ' if player has not learned item lore, then give a very vague message
  If CurrentPlayer.Itemlore = 0 Then
    MsgBox ("This item is a " + thing.Name + ".")
  ' else if item is Herbs or Roots
  ' then use the player's herb lore skill rating as well
  ' as the player's skill rating
  ElseIf thing.Name = "herbs" Or thing.Name = "roots" Then
    If (CurrentPlayer.Itemlore / 2) + (CurrentPlayer.Herblore / 2) < 34 Then
      MsgBox ("You found " + thing.Description1)
    ElseIf (CurrentPlayer.Itemlore / 2) + (CurrentPlayer.Herblore / 2) < 67 Then
      MsgBox ("You found " + thing.Description2)
    Else
      MsgBox ("You found " + thing.Description3)
    End If
  ' else if item is armor
  ' then use the player's armor lore rating as well
  ' as the player's skill rating
  ElseIf thing.ItemType = 6 Then
    If (CurrentPlayer.Itemlore / 2) + (CurrentPlayer.Armorlore / 2) < 34 Then
      MsgBox ("You found a " + thing.Description1)
    ElseIf (CurrentPlayer.Itemlore / 2) + (CurrentPlayer.Armorlore / 2) < 67 Then
      MsgBox ("You found a " + thing.Description2)
    Else
      MsgBox ("You found a " + thing.Description3)
    End If
  ' else if item is weapon
  ' then use the player's weapon lore rating as well
  ' as the player's skill rating
  ElseIf thing.ItemType < 6 And thing.ItemType > 0 Then
    If (CurrentPlayer.Itemlore / 2) + (CurrentPlayer.Weaponlore / 2) < 34 Then
      MsgBox ("You found a " + thing.Description1)
    ElseIf (CurrentPlayer.Itemlore / 2) + (CurrentPlayer.Weaponlore / 2) < 67 Then
      MsgBox ("You found a " + thing.Description2)
    Else
      MsgBox ("You found a " + thing.Description3)
    End If
'
ElseIf (CurrentPlayer.Itemlore / 2) + (CurrentPlayer.Weaponlore / 2) < 67 Then
    MsgBox ("You found a " + thing.Description2)
Else
    MsgBox ("You found a " + thing.Description3)
End If
' Any other item
Else
    If CurrentPlayer.Itemlore < 34 Then
        MsgBox ("You found a " + thing.Description1)
    ElseIf CurrentPlayer.Itemlore < 67 Then
        MsgBox ("You found a " + thing.Description2)
    Else
        MsgBox ("You found a " + thing.Description3)
    End If
End If
' give the player a chance to improve his skill
If CurrentPlayer.Itemlore < RandomNumber() Then
    CurrentPlayer.Itemlore = CurrentPlayer.Itemlore + Improve_Skill
End If
End Sub
'
' Improve_Skill
' gives the player a 20% chance to improve his skill
' returns a 1 if the player improves,
' returns a 0 otherwise
Public Function Improve_Skill()
    If RandomNumber() < 21 Then
        Improve_Skill = 1
    Else
        Improve_Skill = 0
    End If
End Function
'
' Initial_Setup
' Sets up the initial position of all the objects
' on the form
Private Sub Initial_Setup()
    Dim i, j As Integer
    ' Paint the background of the form
    For i = 0 To 4
        For j = 0 To 3
            GameForm.PaintPicture GameForm.Picture, i * 200, j * 200, 200, 200
        Next j,
    Next i
    GameForm.Height = 600
    GameForm.Width = 800
    MapPicture.Left = 35
    MapPicture.Top = 72
    MapPicture.Height = 455
    MapPicture.Width = 455
    LeftLine.X1 = MapPicture.Left + MapPicture.Width + 40
    LeftLine.X2 = LeftLine.X1
    LeftLine.Y1 = MapPicture.Top
    LeftLine.Y2 = MapPicture.Top + 180
    TopLine.X1 = LeftLine.X1
    TopLine.X2 = TopLine.X1 + 210
    TopLine.Y1 = MapPicture.Top
    TopLine.Y2 = TopLine.Y1
    RightLine.X1 = LeftLine.X1 + 210
    RightLine.X2 = RightLine.X1
    RightLine.Y1 = LeftLine.Y1
    RightLine.Y2 = LeftLine.Y2
BottomLine.X1 = TopLine.X1
BottomLine.X2 = TopLine.X2
BottomLine.Y1 = TopLine.Y1 + 180
BottomLine.Y2 = BottomLine.Y1
Character.Width = 150
Character.Height = 160
Character.Left = LeftLine.X1 + 30 ' LeftLine.X1 + ((105/2) - (Character.Width / 2))
Character.Top = MapPicture.Top + 10 ' TopLine.Y1 + ((180 / 2) - (Character.Height / 2))
HpLabel.Left = MapPicture.Left + MapPicture.Width + 20
HpLabel.Top = BottomLine.Y1 + 10
HitPoints.Left = HpLabel.Left + HpLabel.Width + 5
HitPoints.Top = BottomLine.Y1 + 10
FatigueLabel.Left = HpLabel.Left
FatigueLabel.Top = HitPoints.Top + 35
Fatigue.Left = HitPoints.Left
Fatigue.Top = HitPoints.Top + 35 ' Health.Top + Health.Height + 10
ItemsLabel.Left = HpLabel.Left
ItemsLabel.Top = Fatigue.Top + 35
Equipment.Top = Fatigue.Top + 35 ' Fatigue.Top + Fatigue.Height + 10
Equipment.Left = HitPoints.Left
SkillsLabel.Top = Equipment.Top + Equipment.Height + 10
SkillsLabel.Left = HpLabel.Left
Skills.Top = SkillsLabel.Top
Skills.Left = Equipment.Left
Eyeball.Top = 539
Eyeball.Left = MapPicture.Left + 55
Hand.Top = Eyeball.Top
Hand.Left = Eyeball.Left + 70 ' Eyeball.Left + Eyeball.Width + 11
Mouth.Top = Eyeball.Top
Mouth.Left = Hand.Left + 70 ' Hand.Left + Hand.Width + 11
Attack.Top = Eyeball.Top
Attack.Left = Mouth.Left + 70 ' Mouth.Left + Mouth.Width + 11
Disk.Left = Attack.Left + 70 ' Attack.Left + Attack.Width + 11
Disk.Top = Eyeball.Top
timeclock1.Left = HpLabel.Left
'timeclock1.Top = Skills.Top + Skills.Height + 10
' North
MoveButton(0).Top = MapPicture.Top - (MoveButton(0).Height / 2)
MoveButton(0).Left = MapPicture.Left + (MapPicture.Width / 2) - (MoveButton(0).Width / 2)
' East
MoveButton(1).Top = MapPicture.Top + (MapPicture.Height / 2) - (MoveButton(1).Height / 2)
MoveButton(1).Left = MapPicture.Left + MapPicture.Width - (MoveButton(3).Width / 2)
' South
MoveButton(2).Top = MapPicture.Top + MapPicture.Height - (MoveButton(2).Height / 2)
MoveButton(2).Left = MapPicture.Left + (MapPicture.Width / 2) - (MoveButton(2).Width / 2)
' West
MoveButton(3).Top = MapPicture.Top + (MapPicture.Height / 2) - (MoveButton(1).Height / 2)
MoveButton(3).Left = MapPicture.Left - (MoveButton(1).Width / 2)
End Sub

' Initialize_Characters
' reads in all the character information from
' the database
Private Sub Initialize_Characters()
Dim i As Integer
With CharacterData.Recordset
  .MoveFirst
  i = 0
  While (.EOF <> True)
    CharacterArray(i).Name = .Fields("Name").Value
    If .Fields("Picture").Value <> "" Then
CharacterArray(i).Picture = .Fields("Picture").Value
End If
CharacterArray(i).Description1 = .Fields("Description1").Value
CharacterArray(i).Description2 = .Fields("Description2").Value
CharacterArray(i).Description3 = .Fields("Description3").Value
CharacterArray(i).IDNumber = .Fields("IDNumber").Value
CharacterArray(i).Speech = .Fields("Speech").Value
i = i + 1
MoveNext
Wend
End With
End Sub

' Learn_Skill
' always allows the player to improve
' his skill, and also allows the player
' to improve his skill to a greater degree
' also, gives the player a bonus based on his
' intelligence rating
Public Function Learn_Skill() As Byte
    Dim bonus As Byte
    If CurrentPlayer.Intelligence < 11 Then
        bonus = 2
    ElseIf CurrentPlayer.Intelligence > 10 And CurrentPlayer.Intelligence < 18 Then
        bonus = 4
    ElseIf CurrentPlayer.Intelligence > 17 And CurrentPlayer.Intelligence < 25 Then
        bonus = 6
    ElseIf CurrentPlayer.Intelligence > 24 And CurrentPlayer.Intelligence < 31 Then
        bonus = 8
    End If
    Learn_Skill = ((RandomNumber() Mod 10) + 1) + bonus
End Function

' Written by Jeff Hanks
Sub LoadFromFile()
    'This function is called initially and will open the map file
    'and read all of the data from the file into the map array
    'where it can be easily accessed later. The large case
    'statement is used to convert all of the ASCII values into
    'a representative integer value for the map int array.
    Dim land As Byte
    Dim temp As String * 1
    Dim b(100) As String
    Dim i, k, m As Integer
    i = 0
    Open DRIVE & "map.txt" For Input As #1
    Do While Not EOF(1)
        Line Input #1, b(i) 'Read line of data.
        Debug.Print b(i) 'Print to Debug window.
        For m = 1 To Len(b(i))
            Select Case Mid(b(i), m, 1)
                Case "0" 'water in all but bottom-right corner
                    land = 0
                Case "1" 'water in all but bottom-left corner
                    land = 1
                Case "2" 'water in all but upper-left corner
                    land = 2
                Case "3" 'water in all but upper-right corner
                    land = 3
                Case "4" 'water on left half only
                    land = 4
            End Select
            Debug.Print land
        End For
    Loop
End Sub
land = 4
Case "5" 'water on right half only
  land = 5
Case "6" 'water on top half only
  land = 6
Case "7" 'water on bottom half only
  land = 7
Case "8" 'water in bottom-right corner
  land = 8
Case "9" 'water in bottom-left corner
  land = 9
Case "a" 'water in upper-left corner
  land = 10
Case "b" 'water in upper-right corner
  land = 11
Case "0" 'full water (ocean 1)
  land = 12
Case "o" 'full water (ocean 2)
  land = 13
Case "L" 'lake
  land = 14
Case "c" 'north-south river
  land = 15
Case "d" 'east-west river
  land = 16
Case "e" 'west-north river
  land = 17
Case "f" 'north-east river
  land = 18
Case "y" 'west-south river
  land = 19
Case "i" 'south-east river
  land = 20
Case "j" 'west-north-east river
  land = 21
Case "k" 'south-west-north river
  land = 22
Case "l" 'north-south-east-west river
  land = 23
Case "p" 'north-east-south river
  land = 24
Case "q" 'south river
  land = 25
Case "u" 'east river
  land = 26
Case "v" 'north river
  land = 27
Case "x" 'west river
  land = 28
Case "E" 'ocean entrance from north
  land = 29
Case "B" 'bridge across north-south river
  land = 30
Case "w" 'waterfall from north to the west
  land = 31
Case "T" 'tree
  land = 32
Case "t" 'tree in the snow
  land = 33
Case "R" 'rocks
  land = 34
Case "r" 'rocks in the snow
  land = 35
Case "S" 'stones
  land = 36
Case "s" 'stones in the snow
  land = 37
Case "M" 'mountain
  land = 38
Case "m" 'snow-capped mountain
  land = 39
Case "E" 'hill
  land = 40
Case "h" 'snow-covered hill
  land = 41
Case "W" 'swamp
  land = 42
Case "G" 'grass
  land = 43
Case "g" 'snowy grass
  land = 44
Case "N" 'normal grass terrain
  land = 45
Case "n" 'normal snow terrain
  land = 46
Case "D" 'desert
  land = 47
Case "C" 'causts
  land = 48
Case "^" 'small village
  land = 49
Case "%" 'small village in the snow
  land = 50
Case "+" 'monastary
  land = 51
Case "*" 'city
  land = 52
Case "+" 'cave entrance
  land = 53
Case "#" 'hermit's hut
  land = 54
Case ">" 'abandoned castle
  land = 55
Case "/" 'final tower
  land = 56
 ' snow-land transitions:
Case "[" 'snow in upper left, grass in bottom right
  land = 57
Case "]" 'snow in upper right, grass in bottom left
  land = 58
Case "{" 'snow in bottom left, grass in upper right
  land = 59
Case "}" 'snow in bottom right, grass in upper left
  land = 60
Case "?" 'tree in swamp
  land = 61
Case "V" 'oasis
  land = 65
Case ">" 'D/G
  land = 66
Case ">" 'G\D
  land = 67
Case "\G/D" land = 68
Case "/\D/G" land = 69
Case "," 'farmer land = 70
Case ":" 'guard land = 71
Case "_" 'adventurer land = 72
Case ":-" 'bridgekeeper land = 73
Case "\~" 'fisherman land = 74
Case "-_" 'barbarian land = 75
Case Else land = 62
End Select
Map((m - 1), i) = land
Next m
i = i + 1
Loop
Close #1
End Sub

'Written by Jeff Hanks
' This function has been revised and will now create the initial screen
' based on the images stored in the BufPicture control.
Sub LoadInitialPosition(xpos As Integer, ypos As Integer)
    Dim x, y As Integer
    For y = 0 To 6
        For x = 0 To 6
            mapgrid(x + (y * 7)).Picture = BufImage(Map(xpos + x, ypos + y))
        Next x
    Next y
End Sub

Private Sub MapGrid_Click(i As Integer)
    Select Case State
        Case "Normal"
            ' Nothing needs to be done if the state is normal
        Case "Eye"
            ' Currently unimplemented as there are no
            ' items on the map
            ' Determine what is being pointed to
            Call Identify_Item (item)
            State = "Normal"
            MousePointer = 0
        Case "Mouth"
            ' Determine if the player is on a square
            ' containing a character
            ' If player is pointing to hermit
            If (Cur_x + (i Mod 7)) = 72 And (Cur_y + (i \ 7)) = 9 Then
                Call Show_SpeechBox(CurrentImageDrive + CharacterArray(23).Picture, CharacterArray(23).Speech)
                Pause (2)
                Call Show_MessageBox(CurrentImageDrive + ItemArray(23).Picture, "The hermit has
given you the hilt to the sword.")
                Add (ItemArray(23))
Call CheckItems
Pause (1)
If CurrentPlayer.Camping > 10 And CurrentPlayer.Foraging > 10 Then
  ' Learn Herb Lore
  If CurrentPlayer.Herblore < 60 Then
    MsgBox ("You have acquired additional knowledge of herb lore from the hermit.")
  End If
  PassTime (60)
Else
  MsgBox ("You are too knowledgeable in this skill to learn any more here.")
End If
Else
  MsgBox ("You may not learn herb lore until you are sufficiently knowledgeable in camping and foraging.")
End If
If CurrentPlayer.Herblore > 10 Then
  If CurrentPlayer.Bandaging < 60 Then
    If CurrentPlayer.Bandaging = 0 Then
      Skills.AddItem ("Bandaging")
    End If
    ' Learn Bandaging
    MsgBox ("You have acquired additional knowledge of bandaging from the hermit.")
  End If
  PassTime (60)
Else
  MsgBox ("You are too knowledgeable in this skill to learn any more here.")
End If
Else
  MsgBox ("You may not learn bandaging until you are sufficiently knowledgeable in herb lore.")
End If
' If player is pointing to the bridgekeeper
ElseIf (Cur_x + (i Mod 7) = 27 And (Cur_y + (i \ 7)) = 46 Then
  Call Show_SpeechBox(CurrentImageDrive + CharacterArray(9).Picture, CharacterArray(9).Speech)
Pause (2)
If CurrentPlayer.Carving > 10 Then
  If CurrentPlayer.Carpentry < 60 Then
    If CurrentPlayer.Carpentry = 0 Then
      Skills.AddItem ("Carpentry")
    End If
    ' Learn Carpentry
    CurrentPlayer.Carpentry = CurrentPlayer.Carpentry + Learn_Skill()
    MsgBox ("You have acquired additional knowledge of carpentry from the bridgekeeper.")
  End If
  PassTime (60)
Else
  MsgBox ("You are too knowledgeable in this skill to learn any more here.")
End If
Else
  MsgBox ("You may not learn carpentry until you are sufficiently knowledgeable in carving.")
End If
If CurrentPlayer.Hunting > 10 Then
  If CurrentPlayer.Butchery < 60 Then
    If CurrentPlayer.Butchery = 0 Then
      Skills.AddItem ("Butchery")
    End If
    ' Learn Butchery
    MsgBox ("You have acquired additional knowledge of butchery from the hunter.")
  End If
  PassTime (60)
Else
  MsgBox ("You are too knowledgeable in this skill to learn any more here.")
End If
Else
  MsgBox ("You may not learn butchery until you are sufficiently knowledgeable in hunting.")
End If
GameForm - 35

Skills.AddItem ("Butchery")
End If
'
Learn Butchery
MsgBox ("You have acquired additional knowledge of butchery from the bridgekeeper.")
PassTime (60)
Else
MsgBox ("You are too knowledgeable in this skill to learn any more here.")
End If
Else
MsgBox ("You may not learn butchery until you are sufficiently & Chr(13) & "knowledgeable in hunting.")
End If
'
If player is pointing to fisherman
ElseIf (Cur_x + (i Mod 7)) = 27 And (Cur_y + (i \ 7)) = 42 Then
Call Show_SpeechBox(CurrentImageDrive + CharacterArray(15).Picture, CharacterArray(15).Speech)
Pause (2)
If CurrentPlayer.Camping > 10 And CurrentPlayer.Firebuilding > 10 Then
    If CurrentPlayer.Cooking < 60 Then
        ' Learn Cooking
        MsgBox ("You have acquired additional knowledge of cooking from the fisherman.")
    Else
        PassTime (60)
        Else
            MsgBox ("You are too knowledgeable in this skill to learn any more here.")
    End If
Else
    MsgBox ("You may not learn cooking until you are sufficiently & Chr(13) & "knowledgeable in camping and fire building.")
End If
If CurrentPlayer.Fishing < 60 Then
    ' Learn Fishing
    If CurrentPlayer.Fishing = 0 Then
        Skills.AddItem ("Fishing")
    End If
    MsgBox ("You have acquired additional knowledge of fishing from the fisherman.")
    CurrentPlayer.Fishing = CurrentPlayer.Fishing + Learn_Skill()
    PassTime (60)
Else
    MsgBox ("You are too knowledgeable in fishing to learn any more here.")
End If
If CurrentPlayer.Hiking > 10 Then
    If CurrentPlayer.Swimming < 60 Then
        If CurrentPlayer.Swimming = 0 Then
            Skills.AddItem ("Swimming")
        End If
        ' Learn Swimming
        CurrentPlayer.Swimming = CurrentPlayer.Swimming + Learn_Skill()
        MsgBox ("You have acquired additional knowledge of swimming from the fisherman.")
    Else
        PassTime (60)
        Else
            MsgBox ("You are too knowledgeable in swimming to learn any more here.")
    End If
Else
    MsgBox ("You are too knowledgeable in swimming to learn any more here.")
End If
Else
MsgBox ("You may not learn swimming until you are sufficiently knowledgeable in hiking.")

End If

' If the player is pointing to the adventurer1
ElseIf (Cur_x + (i Mod 7)) = 54 And (Cur_y + (i \ 7)) = 48 Then
    Call Show_SpeechBox(CurrentImageDrive + CharacterArray(8).Picture, CharacterArray(8).Speech)
    Pause (2)
    If CurrentPlayer.Camping < 60 Then
        If CurrentPlayer.Camping = 0 Then
            Skills.AddItem ("Camping")
        End If
        ' Learn Camping
        CurrentPlayer.Camping = CurrentPlayer.Camping + Learn_Skill()
        MsgBox ("You have acquired additional knowledge of camping from the adventurer.")
    End If
    PassTime (60)
Else
    MsgBox ("You are too knowledgeable in camping to learn any more here.")
End If

' Learn FireBuilding
If CurrentPlayer.Firebuilding < 60 Then
    CurrentPlayer.Firebuilding = CurrentPlayer.Firebuilding + Learn_Skill()
    MsgBox ("You have acquired additional knowledge of fire building from the adventurer.")
    PassTime (60)
Else
    MsgBox ("You are too knowledgeable in firebuilding to learn any more here.")
End If

' Learn Foraging
If CurrentPlayer.Foraging < 60 Then
    If CurrentPlayer.Foraging = 0 Then
        Skills.AddItem ("Foraging")
    End If
    ' Learn Foraging
    MsgBox ("You have acquired additional knowledge of foraging from the adventurer.")
    PassTime (60)
Else
    MsgBox ("You are too knowledgeable in foraging to learn any more here.")
End If

' Learn Hunting
If CurrentPlayer.Hunting < 60 Then
    If CurrentPlayer.Hunting = 0 Then
        Skills.AddItem ("Hunting")
    End If
    ' Learn Hunting
    MsgBox ("You have acquired additional knowledge of hunting from the adventurer.")
    PassTime (60)
Else
    MsgBox ("You are too knowledgeable in hunting to learn any more here.")
End If

' Learn Wrestling
If CurrentPlayer.Wrestling < 60 Then
    ' Learn Wrestling
    MsgBox ("You have acquired additional knowledge of wrestling from the adventurer.")
    PassTime (60)
Else
    MsgBox ("You are too knowledgeable in wrestling to learn any more here.")
End If
' If the player is pointing to the townguard1
ElseIf (Cur_x + (i Mod 7)) = 63 And (Cur_y + (i \ 7)) = 48 Then
  Call Show_SpeechBox(CurrentImageDrive + CharacterArray(5).Picture, CharacterArray(5).Speech)
  Pause (2)
  If CurrentPlayer.Itemlore > 10 Then
    If CurrentPlayer.Weaponlore < 60 Then
      ' Learn Weapon Lore
      CurrentPlayer.Weaponlore = CurrentPlayer.Weaponlore + Learn_Skill()
      MsgBox ("You have acquired additional knowledge of weapon lore from the townguard.")
    End If
    PassTime (60)
  Else
    MsgBox ("You are too knowledgeable in weapon lore to learn any more here.")
  End If
Else
  MsgBox ("You may not learn weapon lore until you are sufficiently" & Chr(13) & "knowledgeable in item lore.")
End If
If CurrentPlayer.Weaponlore > 10 Then
  If CurrentPlayer.Knifefighting < 60 Then
    ' Learn KnifeFighting
    MsgBox ("You have acquired additional knowledge of knife fighting from the townguard.")
  End If
  PassTime (60)
Else
  MsgBox ("You are too knowledgeable in knifefighting to learn any more here.")
End If
If CurrentPlayer.Swordfighting < 60 Then
  ' Learn SwordFighting
  CurrentPlayer.Swordfighting = CurrentPlayer.Swordfighting + Learn_Skill()
  MsgBox ("You have acquired additional knowledge of sword fighting from the townguard.")
  PassTime (60)
Else
  MsgBox ("You are too knowledgeable in swordfighting to learn any more here.")
End If
Else
  MsgBox ("You may not learn knife fighting or sword fighting until you are sufficiently" & Chr(13) & "knowledgeable in weapon lore.")
End If
' If the player is pointing to the townguard2
ElseIf (Cur_x + (i Mod 7)) = 6 And (Cur_y + (i \ 7)) = 41 Then
  Call Show_SpeechBox(CurrentImageDrive + CharacterArray(10).Picture, CharacterArray(10).Speech)
  Pause (2)
  If CurrentPlayer.Itemlore > 10 Then
    If CurrentPlayer.Armorlore < 60 Then
      ' Learn ArmorLore
      MsgBox ("You have acquired additional knowledge of armor lore from the townguard.")
    End If
    PassTime (60)
  Else
    MsgBox ("You are too knowledgeable in armor lore to learn any more here.")
  End If
Else
  MsgBox ("You may not learn armor lore until you are sufficiently" & Chr(13) & "knowledgeable in weapon lore.")
End If
msgbox ("you are too knowledgeable in armor lore to learn any more here.")

end if

else
    msgbox ("you may not learn armor lore until you are sufficiently" & chr(13) & "knowledgeable in item lore.")
end if

if currentplayer.brawling > 10 then
    if currentplayer.punching < 60 then
        ' learn punching
currentplayer.punching = currentplayer.punching + learn_skill()
    msgbox ("you have acquired additional knowledge of punching from the town guard.")
    pass time (60)
else
    msgbox ("you are too knowledgeable in punching to learn any more here.")
end if

if currentplayer.kicking < 60 then
    ' learn kicking
currentplayer.kicking = currentplayer.kicking + learn_skill()
msgbox ("you have acquired additional knowledge of kicking from the town guard.")
    pass time (60)
else
    msgbox ("you are too knowledgeable in kicking to learn any more here.")
end if
else
    msgbox ("you may not learn punching or kicking until you are sufficiently" & chr(13) & "knowledgeable in brawling.")
end if

' if the player is pointing to the farmer
elseif (cur_x + (i mod 7)) = 61 and (cur_y + (i \\ 7)) = 36 then
    call show_speechbox(currentimage drive + characterarray(2).picture, characterarray(2).speech)

pause (2)
if currentplayer.foraging < 60 then
    if currentplayer.foraging = 0 then
        skills.additem ("foraging")
    end if
    ' learn foraging
currentplayer.foraging = currentplayer.foraging + learn_skill()
msgbox ("you have acquired additional knowledge of foraging from the farmer.")
    pass time (60)
else
    msgbox ("you are too knowledgeable in foraging to learn any more here.")
end if
else can't talk
else
    ' error message
    msgbox ("there is no one there to talk to.")
end if
state = "normal"
mousepointer = 0

case "hand"
' currently unimplemented as there are no
' items on the map
' determine what is being pointed to
' determine if player can get item
' if can get item
' get item
' Place in inventory
State = "Normal"
MousePointer = 0
' Else
' Error message
' End if
Case "Attack"
' Determine what is being pointed to
' Determine if player can attack object
' Currently, there is only one monster
' on the map that can be attacked
If (Cur_x + (i Mod 7)) = 59 And (Cur_y + (i \ 7)) = 21 Then
' If player can attack
CurrentMonsterNumber = 29
AttackForm.Show vbModal, Me
State = "Normal"
MousePointer = 0
HitPoints.Value = CurrentPlayer.Hitpoints
Fatigue.Value = CurrentPlayer.Fatigue
PassTime (10)
Else
' If player can not attack
' Error message
MsgBox ("There is no one there for you to attack.")
End If
Case "Begging"
' if player is talking to the hermit
If (Cur_x + (i Mod 7)) = 72 And (Cur_y + (i \ 7)) = 9 Then
If CurrentPlayer.Begging < 30 And CurrentPlayer.Begging > 0 Then
Call Show_MessageBox(CurrentImageDrive + ItemArray(26).Picture, "The hermit has given you some bread.")
Add (ItemArray(26))
CurrentPlayer.Begged = CurrentPlayer.Begged + 1
PassTime (5)
ElseIf CurrentPlayer.Begging < 60 And CurrentPlayer.Begging > 29 Then
Dim temp As Byte
temp = (RandomNumber() Mod 10) + 1
MsgBox ("The hermit has given you " & temp & " silver pieces.")
CurrentPlayer.Begged = CurrentPlayer.Begged + 1
PassTime (5)
ElseIf CurrentPlayer.Begged > 59 Then
temp = (RandomNumber() Mod 15) + 1
MsgBox ("The hermit has given you " & temp & " silver pieces.")
CurrentPlayer.Begged = CurrentPlayer.Begged + 1
PassTime (5)
End If
ElseIf (Cur_x + (i Mod 7)) = 27 And (Cur_y + (i \ 7)) = 46 Then
Call TownForm.Begging("bridgekeeper")
ElseIf (Cur_x + (i Mod 7)) = 27 And (Cur_y + (i \ 7)) = 42 Then
Call TownForm.Begging("fisherman")
ElseIf (Cur_x + (i Mod 7)) = 54 And (Cur_y + (i \ 7)) = 48 Then
Call TownForm.Begging("adventurer")
ElseIf (Cur_x + (i Mod 7)) = 63 And (Cur_y + (i \ 7)) = 48 Then
Call TownForm.Begging("townguard")
ElseIf (Cur_x + (i Mod 7)) = 61 And (Cur_y + (i \ 7)) = 36 Then
Call TownForm.Begging("farmer")
ElseIf (Cur_x + (i Mod 7)) = 6 And (Cur_y + (i \ 7)) = 41 Then
Call TownForm.Begging("townguard")
Else
MsgBox ("There is no one there for you to beg from.")
End If
Case "PickPocketing"
' if player is talking to the hermit
If (Cur_x + (i Mod 7)) = 72 And (Cur_y + (i \ 7)) = 9 Then
Call TownForm.PickPocketing("hermit")
ElseIf (Cur_x + (i Mod 7)) = 27 And (Cur_y + (i \ 7)) = 46 Then
Call TownForm.PickPocketing("bridgekeeper")
ElseIf (Cur_x + (i Mod 7)) = 27 And (Cur_y + (i \ 7)) = 42 Then
Call TownForm.PickPocketing("fisherman")
ElseIf (Cur_x + (i Mod 7)) = 54 And (Cur_y + (i \ 7)) = 48 Then
Call TownForm.PickPocketing("adventurer")
ElseIf (Cur_x + (i Mod 7)) = 63 And (Cur_y + (i \ 7)) = 48 Then
Call TownForm.PickPocketing("townguard")
ElseIf (Cur_x + (i Mod 7)) = 61 And (Cur_y + (i \ 7)) = 36 Then
Call TownForm.PickPocketing("farmer")
ElseIf (Cur_x + (i Mod 7)) = 6 And (Cur_y + (i \ 7)) = 41 Then
Call TownForm.PickPocketing("townguard")
Else
MsgBox ("There is no one there for you to pickpocket.")
End If
End Select
End Sub

Private Sub MessageBox_GotFocus()
    MessageBox.Visible = False
End Sub

' Written by Jeff Hanks
' Function to handle screen shifts in various directions
Private Sub MoveButton_Click(Index As Integer)
    Dim x, w As Integer
    If CurrentPlayer.Fatigue <> 0 Then
        Select Case (Index)
        Case 0 ' north shift
            ' top 6 shift down one, one is pulled from database for top
            If Cur_y <> 0 Then ' check to see if already at top
                ' x = BitBlt(Picture1.hDC, 1, 66, 455, 390, Picture1.hDC, 1, 1, SRCCOPY)
                w = CheckTerrain(Cur_x, Cur_y - 1)
                cur_y = cur_y - 1
                ' fill in posx, posy through posx+6, posy
                ' For w = 0 To 48
                ' PastePicture (w)
                ' Next w
End If
Case 1 'east shift
  ' right 6 shift left one, one is pulled from database for right
  If Cur_x < 84 Then
    'x = BitBlt(Picture1.hDC, 1, 1, 389, 455, Picture1.hDC, 66, 1, SRCCOPY)
    cur_x = cur_x + 1
    w = CheckTerrain(Cur_x + 1, Cur_y)
    'fill in posx+6, posy through posx+6, posy+6
    'For w = 1 To 7
    'For w = 0 To 48
    '  PastePicture (w)
    '  PastePicture ((w * 7) - 1)
    'Next w
    mapgrid(49).Picture = BufImage(64).Picture
  End If
Case 2 'south shift
  ' bottom 6 shift up one, one is pulled from database for bottom
  If Cur_y < 64 Then
    'x = BitBlt(Picture1.hDC, 1, 1, 455, 390, Picture1.hDC, 1, 66, SRCCOPY)
    cur_y = cur_y + 1
    w = CheckTerrain(Cur_x, Cur_y + 1)
    'fill in posx, posy+6 through posx+6, posy+6
    'For w = 0 To 48
    '  PastePicture (w)
    '  PastePicture (w + 6)
    'Next w
  End If
Case 3 'west shift
  ' left 6 shift right one, one is pulled from database for left
  If Cur_x <> 0 Then
    'x = BitBlt(Picture1.hDC, 65, 1, 390, 455, Picture1.hDC, 1, 1, SRCCOPY)
    cur_x = cur_x - 1
    w = CheckTerrain(Cur_x - 1, Cur_y)
    'fill in posx, posy through posx, posy+6
    'For w = 0 To 6
    '  PastePicture (w)
    '  PastePicture (w + 6)
    'Next w
    mapgrid(49).Picture = BufImage(63).Picture
  End If
End Select
Else
  MsgBox ("You are totally exhausted. " & Chr(13) & "Rest before you travel any further.")
End If
End Sub

' PassTime
'  passes game time by incrementing the timeclock
'  by the number of minutes passed to the subroutine
Public Sub PassTime(min As Integer)
  ' min minutes pass
  timeclock1.minutes = timeclock1.minutes + min
End Sub

'Written by Jeff Hanks
Sub PastePicture(pict As Integer)
  'This function receives an integer as a parameter which is the box on
  'the map which needs to be filled in (e.g. 0 to 48 where 7 is the first
' element of the second row, 14 the 1st element of the third row, etc...
' Then, using the global variables cur_x and cur_y to determine where the character is, the map array is searched to determine the int value of the terrain graphic that needs to go in the specified square.

'The following functions sets the specified image box picture equal to the picture specified in the map array.
   Dim temp As Integer
   temp = Map(Cur_x + (pict Mod 7), Cur_y + (pict \ 7))
   mapgrid(pict).Picture = BufImage(temp).Picture
   End Sub

'Written by Jeff Hanks
Sub PlaceInBuffer()
   'This subroutine writes all of the images to a buffer in an invisible second picturebox where the images for the map can be easily accessed. All 60+ images should be written to the buffer eventually. This routine also creates the necessary image boxes in runtime.
   Dim i As Integer
   For i = 1 To 75 'dynamically creates 61 additional image boxes for buffer
      Load BufImage(i)
   Next i

   'load all 62 images into buffer:
   BufImage(0).Picture = LoadPicture(DRIVE & "seland.jpg")
   BufImage(1).Picture = LoadPicture(DRIVE & "swland.jpg")
   BufImage(2).Picture = LoadPicture(DRIVE & "nwland.jpg")
   BufImage(3).Picture = LoadPicture(DRIVE & "neland.jpg")
   BufImage(4).Picture = LoadPicture(DRIVE & "oceanwest.jpg")
   BufImage(5).Picture = LoadPicture(DRIVE & "oceaneast.jpg")
   BufImage(6).Picture = LoadPicture(DRIVE & "oceansnorth.jpg")
   BufImage(7).Picture = LoadPicture(DRIVE & "oceansouth.jpg")
   BufImage(8).Picture = LoadPicture(DRIVE & "seoccean.jpg")
   BufImage(9).Picture = LoadPicture(DRIVE & "swocean.jpg")
   BufImage(10).Picture = LoadPicture(DRIVE & "nwocean.jpg")
   BufImage(11).Picture = LoadPicture(DRIVE & "neoccean.jpg")
   BufImage(12).Picture = LoadPicture(DRIVE & "ocean.jpg")
   BufImage(13).Picture = LoadPicture(DRIVE & "ocean.jpg")
   BufImage(14).Picture = LoadPicture(DRIVE & "ocean.jpg")
   BufImage(15).Picture = LoadPicture(DRIVE & "nsriver.jpg")
   BufImage(16).Picture = LoadPicture(DRIVE & "triver.jpg")
   BufImage(17).Picture = LoadPicture(DRIVE & "triver.jpg")
   BufImage(18).Picture = LoadPicture(DRIVE & "nriver.jpg")
   BufImage(19).Picture = LoadPicture(DRIVE & "nriver.jpg")
   BufImage(20).Picture = LoadPicture(DRIVE & "eriver.jpg")
   BufImage(21).Picture = LoadPicture(DRIVE & "swriver.jpg")
   BufImage(22).Picture = LoadPicture(DRIVE & "nswriver.jpg")
   BufImage(23).Picture = LoadPicture(DRIVE & "nesriver.jpg")
   BufImage(24).Picture = LoadPicture(DRIVE & "nesriver.jpg")
   BufImage(25).Picture = LoadPicture(DRIVE & "sriver.jpg")
   BufImage(26).Picture = LoadPicture(DRIVE & "triver.jpg")
   BufImage(27).Picture = LoadPicture(DRIVE & "nriver.jpg")
   BufImage(28).Picture = LoadPicture(DRIVE & "nriver.jpg")
   BufImage(29).Picture = LoadPicture(DRIVE & "oceanentrance.jpg")
   BufImage(30).Picture = LoadPicture(DRIVE & "bridge.jpg")
   BufImage(31).Picture = LoadPicture(DRIVE & "waterfall.jpg")
   BufImage(32).Picture = LoadPicture(DRIVE & "tree.jpg")
   BufImage(33).Picture = LoadPicture(DRIVE & "snowtree.jpg")
   BufImage(34).Picture = LoadPicture(DRIVE & "bigrock.jpg")
BufImage(35).Picture = LoadPicture(DRIVE & "snowrock.jpg")
BufImage(36).Picture = LoadPicture(DRIVE & "stones.jpg")
BufImage(37).Picture = LoadPicture(DRIVE & "snowstones.jpg")
BufImage(38).Picture = LoadPicture(DRIVE & "mountain.jpg")
BufImage(39).Picture = LoadPicture(DRIVE & "mountainsnow.jpg")
BufImage(40).Picture = LoadPicture(DRIVE & "hill.jpg")
BufImage(41).Picture = LoadPicture(DRIVE & "snowhill.jpg")
BufImage(42).Picture = LoadPicture(DRIVE & "swamp.jpg")
BufImage(43).Picture = LoadPicture(DRIVE & "grass.jpg")
BufImage(44).Picture = LoadPicture(DRIVE & "snow.jpg")
BufImage(45).Picture = LoadPicture(DRIVE & "grass.jpg")
BufImage(46).Picture = LoadPicture(DRIVE & "snow.jpg")
BufImage(47).Picture = LoadPicture(DRIVE & "desert.jpg")
BufImage(48).Picture = LoadPicture(DRIVE & "cactus.jpg")
BufImage(49).Picture = LoadPicture(DRIVE & "small_village.jpg")
BufImage(50).Picture = LoadPicture(DRIVE & "snow_village.jpg")
BufImage(51).Picture = LoadPicture(DRIVE & "monastery.jpg")
BufImage(52).Picture = LoadPicture(DRIVE & "city.jpg")
BufImage(53).Picture = LoadPicture(DRIVE & "cave_entrance.jpg")
BufImage(54).Picture = LoadPicture(DRIVE & "hermittile.jpg") 'hut
BufImage(55).Picture = LoadPicture(DRIVE & "castletile.jpg") 'castle
BufImage(56).Picture = LoadPicture(DRIVE & "tower.jpg") 'tower
BufImage(57).Picture = LoadPicture(DRIVE & "snownwgrassse.jpg")
BufImage(58).Picture = LoadPicture(DRIVE & "snownegrasssw.jpg")
BufImage(59).Picture = LoadPicture(DRIVE & "snowswgrassne.jpg")
BufImage(60).Picture = LoadPicture(DRIVE & "black.jpg")
BufImage(61).Picture = LoadPicture(DRIVE & "swamptree.jpg")
BufImage(62).Picture = LoadPicture(DRIVE & "black.jpg") 'Shouldn't happen
BufImage(63).Picture = LoadPicture(DRIVE & "femaleleft.gif")
BufImage(64).Picture = LoadPicture(DRIVE & "femaleright.gif")
BufImage(65).Picture = LoadPicture(DRIVE & "oasis.jpg")
BufImage(66).Picture = LoadPicture(DRIVE & "desertnwgrassse.jpg")
BufImage(67).Picture = LoadPicture(DRIVE & "desertnegrasssw.jpg")
BufImage(68).Picture = LoadPicture(DRIVE & "desertsegrassnw.jpg")
BufImage(69).Picture = LoadPicture(DRIVE & "desertswgrassne.jpg")
BufImage(70).Picture = LoadPicture(DRIVE & "farmertile.jpg")
BufImage(71).Picture = LoadPicture(DRIVE & "guardtile.jpg")
BufImage(72).Picture = LoadPicture(DRIVE & "adventile.jpg")
BufImage(73).Picture = LoadPicture(DRIVE & "bridgekeepertile.jpg")
BufImage(74).Picture = LoadPicture(DRIVE & "fishermantile.jpg")
BufImage(75).Picture = LoadPicture(DRIVE & "barbariantile.jpg")

End Sub

' Quit_Button_Click
' ends the game
' unloads any currently loaded forms
Public Sub QuitButton_Click()
    Dim frm As Form
    For Each frm In Forms
        Unload frm
    Next frm
End Sub

' Function RandomNumber
' returns a random number between 1 and 100
Public Function RandomNumber() As Single
    Randomize
    RandomNumber = Rnd() * 100 Mod 100 + 1
End Function

' Remove Item from Inventory
Public Sub Remove(theIndex As Byte)
    ' Remove item from inventory
    Set CurrentPlayer.Inventory(theIndex) = Nothing
    ' Remove item from equipment list
    Equipment.RemoveItem (theIndex)
    ' Reset array to account for empty slot
    Dim i As Byte
    For i = theIndex To 99
        Set CurrentPlayer.Inventory(i) = CurrentPlayer.Inventory(i + 1)
    Next i
    ' Subtract one from the next index
    CurrentPlayer.NextIndex = CurrentPlayer.NextIndex - 1
End Sub

' Rest
' allows player to rest for a specified number of
' hours; restores fatigue and hit points
Private Sub Rest(i As Byte)
    ' rest for i hours
    PassTime (i * 60)
    Dim tempfatigue As Integer
    tempfatigue = CurrentPlayer.Fatigue + (i * (RandomNumber() Mod 10) + 1 + 10)
    ' Make sure we have not exceeded the player's max_fp
    If tempfatigue > CurrentPlayer.Max_Fp Then
    Else
        CurrentPlayer.Fatigue = tempfatigue
    End If
    CurrentPlayer.HitPoints = CurrentPlayer.HitPoints + ((RandomNumber() Mod 10) + 1) * i
    ' Make sure we have not exceeded the player's max_hp
    If CurrentPlayer.HitPoints > CurrentPlayer.Max_Hp Then
    End If
    ' Update hitpoint and fatigue progress bars
    HitPoints.Value = CurrentPlayer.HitPoints
    Fatigue.Value = CurrentPlayer.Fatigue
    ' Display appropriate message
    MsgBox ("You feel much better after " & i & " hours of rest.")
End Sub

' RestCommand_Click
' shows the rest frame
Private Sub RestCommand_Click()
    CampingMenu.Visible = False
    RestEditBox.text = 0
    RestMenu.Left = 200
    RestMenu.Top = 200
    RestMenu.Visible = True
End Sub

' RestOkayCommand_Click
' sets the rest frame to not visible
' calls the rest subroutine with the number of hours
GameForm - 45

' the player supplied
' shows the camping menu
Private Sub RestOkayCommand_Click()
    RestMenu.Visible = False
    Rest (CByte(RestEditBox.text))
    CampingMenu.Visible = True
End Sub

' RootsButton_Click
' player forages for roots
Private Sub RootsButton_Click()
    ForageFrame.Visible = False
    ' make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 10 Then
        MsgBox ("You are too tired.")
    ' if player fails the skill check
    ElseIf RandomNumber() > (CurrentPlayer.Foraging) Then
        ' pass some game time
        PassTime (10)
        ' subtract fatigue points
        CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 8
        ' display appropriate message
        MsgBox ("You were not successful at foraging for roots.")
        ' give player a chance to improve his skill
        CurrentPlayer.Foraging = CurrentPlayer.Foraging + Improve_Skill
    ' if player passes the skill check
    Else
        ' pass some time
        PassTime (15)
        ' subtract fatigue points
        CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 10
        ' Add some roots to the player's inventory
        Dim i As Byte
        For i = 1 To ((RandomNumber() Mod 2) + 1)
            Call Add(ItemArray(36))
        Next i
        ' Show a messagebox with a picture of some roots
        Call Show_MessageBox(CurrentImageDrive + ItemArray(36).Picture, "You have successfully foraged some roots.")
    End If
    ' update the fatigue progress bar
    Fatigue.Value = CurrentPlayer.Fatigue
End Sub

' Show Message_Box
' Shows a message box with the given image and message
Private Sub Show_MessageBox(ByVal image As String, ByVal text As String)
    MessageBoxIcon = image
    LabelText = text
    Top = 150
    Left = 75
    Height = 225
    Width = 350
    Visible = True
End Sub

' Show SpeechBox
' Shows a speech box with the given image and message
Private Sub Show_SpeechBox(ByVal image As String, ByVal text As String)
    MessageBox.image = image
    LabelText = text
private sub savebutton_click()
    commondialog1.defaultext = ".dst"
    commondialog1.dialogtitle = "save character...
    commondialog1.filter = "character files (*.dst)|*.dst"
    commondialog1.filename = currentplayer.name
    commondialog1.showsave
    filletype commondialog1.filename
end sub

public sub filletype(fil as string)
    dim i as integer
    dim tmp as playertype
    with currentplayer
        tmp.aboutbody = .aboutbody.idnumber
        tmp.age = .age
        tmp.agility = .agility
        tmp.appraising = .appraising
        tmp.archery = .archery
        tmp.armor = .armor
        tmp.armorrow = .armorlore
        tmp.bandaged = .bandaged
        tmp.bandaging = .bandaging
        tmp.barter = .barter
        tmp.begged = .begged
        tmp.begging = .begging
        tmp.blindfighting = .blindfighting
        tmp.bluntweaponfighting = .bluntweaponfighting
        tmp.body = .body.idnumber
        tmp.brawling = .brawling
        tmp.butchery = .butchery
        tmp.camping = .camping
        tmp.carpentry = .carpentry
        tmp.carving = .carving
        tmp.chopping = .chopping
        tmp.commerce = .commerce
        tmp.cooking = .cooking
        tmp.cur_minutes = timeclock1.minutes
        tmp.cur_x = cur_x
        tmp.cur_y = cur_y
        tmp.damagebonus = .damagebonus
        tmp.evade = .evade
        tmp.fatigue = .fatigue
        tmp.feet = .feet.idnumber
        tmp.fencing = .fencing
    end with
end sub
tmp.Firebuilding = .Firebuilding
tmp.Fishing = .Fishing
tmp.Fitness = .Fitness
tmp.Foraging = .Foraging
tmp.Handtohandcombat = .Handtohandcombat
tmp.Head = .Head.IDNumber
tmp.Health = .Health
tmp.Height = .Height
tmp.Herbole = .Herbole
tmp.Hiking = .Hiking
tmp.Hitbonus = .Hitbonus
tmp.HitPoints = .HitPoints
tmp.Hunting = .Hunting
tmp.Intelligence = .Intelligence
' not sure about the indexing
For i = 0 To 99
   If i < .Nextindex Then
      tmp.Inventory(i) = .Inventory(i).IDNumber
   Else
      tmp.Inventory(i) = 54
   End If
Next i
tmp.Itemlore = .Itemlore
tmp.Jogging = .Jogging
tmp.Kicking = .Kicking
tmp.Knifefighting = .Knifefighting
tmp.Max_Fp = .Max_Fp
tmp.Max_Hp = .Max_Hp
tmp.Money = .Money
tmp.Mountaineering = .Mountaineering
tmp.Name = .Name
tmp.Nextindex = .Nextindex
tmp.Parry = .Parry
tmp.Parrying = .Parry
     tmp.Performance = .Performance
tmpPerformed = .Performed
tmp.Personality = .Personality
tmp.PickPocketing = .PickPocketing
tmp.Picture = .Picture
tmp.Poisoned = .Poisoned
tmp.Punching = .Punching
tmp.Sex = .Sex
tmp.Shield = .Shield.IDNumber
tmp.Skinning = .Skinning
tmp.Social = .Social
tmp.Stealth = .Stealth
tmp.Strength = .Strength
tmp.Swimming = .Swimming
tmp.Swordfighting = .Swordfighting
tmp.Weaponlore = .Weaponlore
tmp.Weight = .Weight
tmp.Wielded = .Wielded.IDNumber
tmp.Wrestling = .Wrestling
End With
Open fil For Binary As #256
Put #256, , tmp
Close #256

End Sub
'
Skills_Click
' Checks to see what skill was selected and calls the appropriate function
Private Sub Skills_Click()
    Select Case Skills.text
        Case "Appraising": Appraising
        Case "Bandaging": Bandaging
        Case "Butchery": Butchery
        Case "Camping": Camping
        Case "Carpentry": Carpentry
        Case "Carving": Carving
        Case "Chopping": Chopping
        Case "Fishing": Fishing
        Case "Foraging": Foraging
        Case "Hunting": Hunting
        Case "Pick Pocketing": State = "PickPocketing"
        Case "Skinning": Skinning
    End Select
End Sub
'
'Skinning
' player wants to skin a carcass
Private Sub Skinning()
    ' make sure player is holding a carcass
    If Equipment.text <> "Carcass" Then
        MsgBox ("You must be holding the carcass in order to skin it.")
    ElseIf CurrentPlayer.Fatigue < 4 Then
        MsgBox ("You are too tired.")
    Else
        ' if player fails skill check
        If CurrentPlayer.Skinning < RandomNumber() Then
            ' pass game time
            PassTime (5)
            ' subtract fatigue points
            CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 1
            ' display appropriate message
            MsgBox ("You were unable to skin the carcass.")
            ' give player a chance to improve his skill
            CurrentPlayer.Skinning = CurrentPlayer.Skinning + Improve_Skill
            ' if player passes skill check
        Else
            ' pass game time
            PassTime (10)
            ' subtract fatigue points
            CurrentPlayer.Fatigue = CurrentPlayer.Fatigue - 4
            ' show a messagebox with a picture of a fur pelt
            Call Show_MessageBox(CurrentImageDrive + ItemArray(41).Picture, "You have successfully skinned the carcass.")
            ' add the fur pelt to the player's inventory
            Call Add(ItemArray(41))
            ' give the player a chance to butcher the carcass
            Dim choice As VbMsgBoxResult
            choice = MsgBox("Would you like to butcher the carcass?", vbYesNo)
            ' if player wants to butcher the carcass
            If choice = vbYes Then
                ' make sure player has enough fatigue
                If CurrentPlayer.Fatigue < 5 Then
                    MsgBox ("You are too tired.")
                ' if player fails the skill check
                ElseIf CurrentPlayer.Butchery < RandomNumber() Then
                    ' pass time
Pass'Time (10)
' subtract fatigue points
' display appropriate message
MsgBox ("You were unable to butcher the carcass.")
' give player a chance to improve his skill
' if player passes the skill check
Else
' pass game time
Pass'Time (20)
' subtract fatigue points
' Show a messagebox with a picture of meat
Call Show_MessageBox(CurrentImageDrive + ItemArray(30).Picture, "You success fully butchered the carcass.")
' add the meat to the player's inventory
Call Add(ItemArray(30))
End If
End If
' remove the carcass from the player's inventory
Remove (Equipment.ListIndex)
End If
' update the fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue

Private Sub SpeechBox_GotFocus()
    SpeechBox.Visible = False
End Sub

    ' calls the fire_building subroutine
Private Sub StartFireCommand_Click()
    Fire_Building
End Sub

    ' replaces the statistics box with the player's picture
Private Sub Statistics_GotFocus()
    Statistics.Visible = False
    Character.Visible = True
End Sub

    ' StopCampingCommand_Click
Private Sub StopCampingCommand_Click()
    CampingMenu.Visible = False
    Remove tent icon
    ' Place character on map
    mapgrid(49) = LoadPicture(DRIVE & "femaleright.gif")
    ' set Fire to false since there is no longer a fire burning
    Fire = False
End Sub

    ' date_Skills
    ' adds those skills that the player knows to the skills
    ' combo box on the form
Private Sub Update_Skills()
    If CurrentPlayer.Appraising > 0 Then Skills.AddItem ("Appraising")
    If CurrentPlayer.Bandaging > 0 Then Skills.AddItem ("Bandaging")
If CurrentPlayer.Butchery > 0 Then Skills.AddItem ("Butchery")
If CurrentPlayer.Camping > 0 Then Skills.AddItem ("Camping")
If CurrentPlayer.Carpentry > 0 Then Skills.AddItem ("Carpentry")
If CurrentPlayer.Carving > 0 Then Skills.AddItem ("Carving")
If CurrentPlayer.Chopping > 0 Then Skills.AddItem ("Chopping")
If CurrentPlayer.Fishing > 0 Then Skills.AddItem ("Fishing")
If CurrentPlayer.Foraging > 0 Then Skills.AddItem ("Foraging")
If CurrentPlayer.Hunting > 0 Then Skills.AddItem ("Hunting")
If CurrentPlayer.PickPocketing > 0 Then Skills.AddItem ("Pick Pocketing")
If CurrentPlayer.Skinning > 0 Then Skills.AddItem ("Skinning")

End Sub

' when a new day starts, set the player's flags
' back to 0 so he can perform those skills again
' during the new day
Private Sub timeclock1_newday()
    CurrentPlayer.Bandaged = 0
    CurrentPlayer.Poisoned = 0
    CurrentPlayer.Performed = 0
    CurrentPlayer.Begged = 0
End Sub

' turns the time off
' calls initialize_characters to read in the
' characted information into an array
Private Sub Timer1_Timer()
    Timer1.Enabled = False
    Initialize_Characters
End Sub

' craft a walking stick
Private Sub WalkingStickCommand_Click()
    CarvingFrame.Visible = False
    ' make sure player has enough fatigue
    If CurrentPlayer.Fatigue < 5 Then
        MsgBox ("You are too tired.")
    ' if player passes skill check
    ElseIf CurrentPlayer.Carving > RandomNumber() Then
        ' pass game time
        PassTime (10)
        ' subtract fatigue points
        ' show a message box with a picture of a walking stick
        Call Show_MessageBox(ItemArray54.Picture, "You have successfully carved a walking stick.")
        ' remove the wood from the player's equipment list
        Remove (Equipment.ListIndex)
        ' add a walking stick to the player's inventory
        Call Add(ItemArray54)
    ' if player fails skill check
    Else
        ' pass game time
        PassTime (5)
        ' subtract fatigue points
        ' display appropriate message
        MsgBox ("You were unable to make a walking stick.")
        ' remove wood from the player's inventory
        Remove (Equipment.ListIndex)
        ' give player a chance to improve his skill
    End If
End Sub
End If
' update fatigue progress bar
Fatigue.Value = CurrentPlayer.Fatigue
End Sub
' All functions on this form written by
'   Karen Woznick unless otherwise noted.

Stats_Number As Byte
Dim LastStrengthValue As Byte
Dim LastAgilityValue As Byte
Dim LastFitnessValue As Byte
Dim LastHealthValue As Byte
Dim LastIntelligenceValue As Byte
Dim LastPersonalityValue As Byte

' Written by Tim Zoch
Public Sub Calculate_DamageBonus()
' takes player's strenght and applies a bonus too the power of the
' player's attack
    If CurrentPlayer.Strength = 8 Then
        CurrentPlayer.Damagebonus = -2
    ElseIf CurrentPlayer.Strength > 8 And CurrentPlayer.Strength < 12 Then
        CurrentPlayer.Damagebonus = -1
    ElseIf CurrentPlayer.Strength > 11 And CurrentPlayer.Strength < 15 Then
        CurrentPlayer.Damagebonus = 0
    ElseIf CurrentPlayer.Strength > 14 And CurrentPlayer.Strength < 18 Then
        CurrentPlayer.Damagebonus = 1
    ElseIf CurrentPlayer.Strength > 17 And CurrentPlayer.Strength < 21 Then
        CurrentPlayer.Damagebonus = 2
    ElseIf CurrentPlayer.Strength > 20 And CurrentPlayer.Strength < 24 Then
        CurrentPlayer.Damagebonus = 3
    ElseIf CurrentPlayer.Strength > 23 And CurrentPlayer.Strength < 27 Then
        CurrentPlayer.Damagebonus = 4
    ElseIf CurrentPlayer.Strength > 26 And CurrentPlayer.Strength < 30 Then
        CurrentPlayer.Damagebonus = 5
    Else
        CurrentPlayer.Damagebonus = 6
    End If
End Sub

' Written by Tim Zoch
Private Sub Calculate_FatigueBonus()
' resets player's max fatigue based on how fit they are
    CurrentPlayer.Max_Fp = 100
    If CurrentPlayer.Fitness = 8 Then
    ElseIf CurrentPlayer.Fitness > 8 And CurrentPlayer.Fitness < 12 Then
    ElseIf CurrentPlayer.Fitness > 11 And CurrentPlayer.Fitness < 15 Then
        CurrentPlayer.Max_Fp = CurrentPlayer.Max_Fp + 1
    ElseIf CurrentPlayer.Fitness > 14 And CurrentPlayer.Fitness < 18 Then
    ElseIf CurrentPlayer.Fitness > 17 And CurrentPlayer.Fitness < 21 Then
    ElseIf CurrentPlayer.Fitness > 20 And CurrentPlayer.Fitness < 24 Then
    ElseIf CurrentPlayer.Fitness > 23 And CurrentPlayer.Fitness < 27 Then
    Else
    End If
End Sub
'Written by Tim Zoch

Public Sub Calculate_HitBonus()
'Adds a bonus to hit a monster based on agility
    If CurrentPlayer.Agility = 8 Then
        CurrentPlayer.Hitbonus = -2
    ElseIf CurrentPlayer.Agility > 8 And CurrentPlayer.Agility < 12 Then
        CurrentPlayer.Hitbonus = -1
    ElseIf CurrentPlayer.Agility > 11 And CurrentPlayer.Agility < 15 Then
        CurrentPlayer.Hitbonus = 0
    ElseIf CurrentPlayer.Agility > 14 And CurrentPlayer.Agility < 18 Then
        CurrentPlayer.Hitbonus = 1
    ElseIf CurrentPlayer.Agility > 17 And CurrentPlayer.Agility < 21 Then
        CurrentPlayer.Hitbonus = 2
    ElseIf CurrentPlayer.Agility > 20 And CurrentPlayer.Agility < 24 Then
        CurrentPlayer.Hitbonus = 3
    ElseIf CurrentPlayer.Agility > 23 And CurrentPlayer.Agility < 27 Then
        CurrentPlayer.Hitbonus = 4
    ElseIf CurrentPlayer.Agility = 26 And CurrentPlayer.Agility < 30 Then
        CurrentPlayer.Hitbonus = 5
    ElseIf CurrentPlayer.Agility > 29 Then
        CurrentPlayer.Hitbonus = 6
    End If
End Sub

'Written by Tim Zoch
Private Sub Calculate_HitPointBonus()
'determines a players max hitpoints based on health
    CurrentPlayer.Max_Hp = 100
    If CurrentPlayer.Health = 8 Then
    ElseIf CurrentPlayer.Health > 8 And CurrentPlayer.Health < 12 Then
        CurrentPlayer.Max_Hp = CurrentPlayer.Max_Hp - 1
    ElseIf CurrentPlayer.Health > 11 And CurrentPlayer.Health < 15 Then
    ElseIf CurrentPlayer.Health > 14 And CurrentPlayer.Health < 18 Then
        CurrentPlayer.Max_Hp = CurrentPlayer.Max_Hp + 1
    ElseIf CurrentPlayer.Health > 17 And CurrentPlayer.Health < 21 Then
    ElseIf CurrentPlayer.Health > 20 And CurrentPlayer.Health < 24 Then
    ElseIf CurrentPlayer.Health > 23 And CurrentPlayer.Health < 27 Then
    ElseIf CurrentPlayer.Health > 26 And CurrentPlayer.Health < 30 Then
    ElseIf CurrentPlayer.Health = 30 Then
    End If
End Sub

' Centers the frame around the selected image
' Sets the player's image to "female1.gif"
' Sets the player's sex to female
Private Sub FemaleImage1_Click()
    Frame.Left = 413
    Frame.Visible = True
    CurrentPlayer.Picture = "female1.gif"
    CurrentPlayer.Sex = 1
End Sub

'Centers the frame around the selected image