

Option Explicit

```
, =====
', 数据变量
', =====
```

```
Private appList As Collection
Private swApp As SldWorks.SldWorks
Private configPath As String
```

```
, =====
', 窗体初始化
', =====
```

```
Private Sub UserForm_Initialize()
    On Error GoTo ErrorHandler

    ' 获取SolidWorks应用程序实例
    Set swApp = Application.SldWorks

    ' 获取配置文件路径
    configPath = mdlAppList.GetConfigPath(swApp)

    ' 加载程序列表
    LoadAppList

    ' 初始化状态栏
    UpdateStatus

    ' 显示初始说明
    ShowInstructions

Exit Sub
```

```
ErrorHandler:
    MsgBox "初始化错误 " & Err.Number & ": " & Err.Description & vbCrLf & _
        "配置文件路径: " & configPath, vbCritical, "窗体初始化错误"
    Unload Me
End Sub
```

```
, =====
', 数据管理
', =====
```

```
Private Sub LoadAppList()
    On Error GoTo ErrorHandler

    ' 清空列表框
    Me.lstApps.Clear

    ' 加载程序列表
    Set appList = mdlAppList.LoadAppList(swApp)

    ' 将程序添加到列表框
    Dim i As Integer
    For i = 1 To appList.Count
        Dim app As Object
        Set app = appList(i)
        Me.lstApps.AddItem app("DisplayName")
    Next i

    ' 选择第一项（如果有）
    If Me.lstApps.ListCount > 0 Then
        Me.lstApps.ListIndex = 0
    End If

Exit Sub
```

```
ErrorHandler:
    MsgBox "加载程序列表失败: " & Err.Description & vbCrLf & _
        "配置文件路径: " & configPath, vbExclamation, "错误"
    Set appList = New Collection
End Sub
```

```
, =====
', 显示使用说明
', =====
```

```
Private Sub ShowInstructions()
    Me.lblInstructions.Caption = "程序启动器使用说明:" & vbCrLf & vbCrLf & _
        "• 双击列表中的程序名称启动程序" & vbCrLf & vbCrLf & _
        "• 选择程序后按回车键启动程序" & vbCrLf & vbCrLf & _
        "• 如需修改程序列表请修改同目录下的SolidWorksAppList.txt" & vbCrLf & vbCrLf & _
        "• 程序列表会自动从配置文件加载" & vbCrLf & vbCrLf & _
```

```

        " • 配置文件路径: " & GetConfigPath(swApp) & vbCrLf & vbCrLf & _
        " • 当前共有 " & IIf(Not appList Is Nothing, appList.Count, 0) & " 个程序" & vbCrLf & vbCrLf & _
        "Made By Tika"
End Sub

', =====
', 列表交互 - 启动程序
', =====

Private Sub lstApps_Db1Click(ByVal Cancel As MSForms.ReturnBoolean)
    LaunchSelectedApp
End Sub

' 启动选中的程序
Private Sub LaunchSelectedApp()
    On Error GoTo ErrorHandler

    ' 检查是否有选中项
    If Me.lstApps.ListIndex = -1 Then
        MsgBox "请先选择要启动的程序", vbExclamation, "提示"
        Exit Sub
    End If

    ' 获取选中程序的信息
    Dim app As Object
    Set app = appList(Me.lstApps.ListIndex + 1)

    ' 检查文件是否存在
    If Dir(app("FilePath")) = "" Then
        MsgBox "错误: 文件不存在" & vbCrLf & app("DisplayName") & vbCrLf & app("FilePath"), vbCritical, "文件错误"
        Exit Sub
    End If

    ' 启动程序
    Dim result As Double
    result = Shell(app("FilePath"), vbNormalFocus)

    ' 显示启动成功信息
    MsgBox "已启动: " & app("DisplayName"), vbInformation, "程序启动"

    Exit Sub

ErrorHandler:
    MsgBox "启动失败: " & Err.Description, vbCritical, "错误"
End Sub

' 按回车键启动程序
Private Sub lstApps_KeyDown(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)
    If KeyCode = vbKeyReturn Then
        LaunchSelectedApp
    End If
End Sub

', =====
', 状态栏更新
', =====

Private Sub UpdateStatus(Optional ByVal message As String = "")
    On Error Resume Next

    If message = "" Then
        If Not appList Is Nothing Then
            lblStatus.Caption = "共 " & appList.Count & " 个程序"
        Else
            lblStatus.Caption = "程序列表未加载"
        End If
    Else
        lblStatus.Caption = message
    End If
End Sub

', =====
', 窗体生命周期
', =====

Private Sub UserForm_Terminate()
    ' 释放对象
    Set appList = Nothing
    Set swApp = Nothing
End Sub

' 获取配置文件路径（用于显示）
Private Function GetConfigPath(swApp As SldWorks.SldWorks) As String

```

```
On Error Resume Next
GetConfigPath = mdlAppList.GetConfigPath(swApp)
If GetConfigPath = "" Then
    GetConfigPath = "无法确定配置文件路径"
End If
End Function
```

main - 1

’ 模块名称: main  
’ 用途: 主入口点

Option Explicit

Sub main()  
’ 显示程序启动器窗体  
frmAppLauncher.Show  
End Sub

' 模块名称: mdlAPI  
' 用途: 定义Windows API函数

Option Explicit

#If VBA7 Then

```
Public Type OPENFILENAME
    lStructSize As Long
    hwndOwner As LongPtr
    hInstance As LongPtr
    lpstrFilter As String
    lpstrCustomFilter As String
    nMaxCustFilter As Long
    nFilterIndex As Long
    lpstrFile As String
    nMaxFile As Long
    lpstrFileName As String
    nMaxFileName As Long
    lpstrInitialDir As String
    lpstrTitle As String
    Flags As Long
    nFileOffset As Integer
    nFileExtension As Integer
    lpstrDefExt As String
    lCustData As LongPtr
    lpfnHook As LongPtr
    lpTemplateName As String
End Type
```

Public Declare PtrSafe Function GetOpenFileName Lib "comdlg32.dll" Alias "GetOpenFileNameA" (pOpenfilename As OPENFILENAME) As Long

#Else

```
Public Type OPENFILENAME
    lStructSize As Long
    hwndOwner As Long
    hInstance As Long
    lpstrFilter As String
    lpstrCustomFilter As String
    nMaxCustFilter As Long
    nFilterIndex As Long
    lpstrFile As String
    nMaxFile As Long
    lpstrFileName As String
    nMaxFileName As Long
    lpstrInitialDir As String
    lpstrTitle As String
    Flags As Long
    nFileOffset As Integer
    nFileExtension As Integer
    lpstrDefExt As String
    lCustData As Long
    lpfnHook As Long
    lpTemplateName As String
End Type
```

Public Declare Function GetOpenFileName Lib "comdlg32.dll" Alias "GetOpenFileNameA" (pOpenfilename As OPENFILENAME) As Long

#End If

’ 模块名称: mdlAppList  
 ’ 用途: 管理应用程序列表

Option Explicit

Public Const APP\_LIST\_FILE As String = "SolidWorksAppList.txt"

’ 获取配置文件完整路径（使用宏文件位置）

Public Function GetConfigPath(swApp As SldWorks.SldWorks) As String  
 On Error Resume Next

’ 获取当前宏的完整路径

Dim macroPath As String  
 macroPath = swApp.GetCurrentMacroPathName

’ 如果成功获取宏路径

If macroPath <> "" Then  
 ’ 提取目录部分（去掉文件名）  
 Dim lastSlashPos As Integer  
 lastSlashPos = InStrRev(macroPath, "\")  
 If lastSlashPos > 0 Then  
 GetConfigPath = Left(macroPath, lastSlashPos) & APP\_LIST\_FILE  
 Exit Function  
 End If  
End If

’ 如果无法获取宏路径（例如宏是从内存中运行的），回退到文档目录

GetConfigPath = Environ("USERPROFILE") & "\Documents\" & APP\_LIST\_FILE  
 End Function

’ 从配置文件加载程序列表

Public Function LoadAppList(swApp As SldWorks.SldWorks) As Collection  
 Dim fso As Object  
 Dim ts As Object  
 Dim filePath As String  
 Dim appList As New Collection

’ 获取配置文件路径

filePath = GetConfigPath(swApp)  
 Set fso = CreateObject("Scripting.FileSystemObject")

’ 文件不存在则创建默认项

If Not fso.FileExists(filePath) Then  
 Set ts = fso.CreateTextFile(filePath)  
 ts.WriteLine "计算器|C:\Windows\System32\calc.exe"  
 ts.WriteLine "记事本|C:\Windows\System32\notepad.exe"  
 ts.Close  
End If

’ 读取文件

On Error Resume Next  
 Set ts = fso.OpenTextFile(filePath, 1)  
 If Err.Number <> 0 Then  
 MsgBox "无法打开配置文件: " & filePath & vbCrLf & Err.Description, vbCritical, "错误"  
 Set LoadAppList = New Collection  
 Exit Function  
 End If  
 On Error GoTo 0

Do Until ts.AtEndOfStream

Dim line As String  
 Dim parts() As String

line = Trim(ts.ReadLine)  
 If line <> "" Then  
 parts = Split(line, "|", 2)

’ 检查格式是否正确

If UBound(parts) >= 1 Then  
 ’ 创建变体对象存储应用程序信息  
 Dim app As Object  
 Set app = CreateObject("Scripting.Dictionary")

’ 设置显示名称和文件路径

app.Add "DisplayName", parts(0)  
 app.Add "FilePath", parts(1)

’ 检查文件是否存在

If Dir(app("FilePath")) <> "" Then  
 appList.Add app  
Else

```
        ' 文件不存在时仍然添加，但给出警告
        appList.Add app
    End If
ElseIf UBound(parts) = 0 Then
    ' 兼容旧格式（只有路径）
    Dim appName As String
    Dim lastSlashPos As Integer
    lastSlashPos = InStrRev(parts(0), "\")
    If lastSlashPos > 0 Then
        appName = Mid(parts(0), lastSlashPos + 1)
    Else
        appName = parts(0)
    End If

    ' 创建变体对象存储应用程序信息
    Dim appOld As Object
    Set appOld = CreateObject("Scripting.Dictionary")
    appOld.Add "DisplayName", appName
    appOld.Add "FilePath", parts(0)

    If Dir(parts(0)) <> "" Then
        appList.Add appOld
    Else
        ' 文件不存在时仍然添加，但给出警告
        appList.Add appOld
    End If
End If
End If
Loop
ts.Close

Set LoadAppList = appList
End Function
```

mdlTypes - 1

' 模块名称: mdlTypes  
' 用途: 定义公共数据类型

Option Explicit

' 定义应用程序信息类型  
Public Type AppInfo  
    DisplayName As String  
    filePath As String  
End Type