

任务 4.5 创建简单文字动画

在本任务中，利用计时器控件的 Timer 事件来移动标签控件在窗体上的位置，使标签文字向上方移动，从而生成文字动画效果，并允许通过复选框来开启或停止动画，如图 4.12 所示。

任务目标

- 掌握计时器控件的常用属性。
- 掌握计时器控件的常用事件。

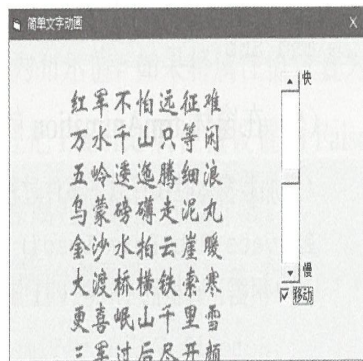



图 4.12 简单文字动画

任务分析

如果想要使标签文字在窗体上移动，则可以使用计时器控件的 Timer 事件在预定的时间间隔后重复执行一段代码，用来改变标签控件的 Top 属性。如果想要控制标签文字移动或停止，则可以使用复选框控件的 Click 事件来设置计时器控件的 Enabled 属性。如果想要调整标签文字移动速度的快慢，则可以使用垂直滚动条控件来设置计时器控件的 Interval 属性。

任务实施

- (1) 在 Visual Basic 6.0 集成开发环境中创建一个标准 EXE 工程。
- (2) 将窗体 Form1 命名为 frmAnimation，然后将其 Caption 属性值设置为“简单文字动画”，BorderStyle 属性值设置为 1。
- (3) 在窗体 frmAnimation 上添加以下控件。
 - 添加一个标签控件并命名为 lblPoetry，然后将其 Caption 属性值设置为“红军不怕远征难 万水千山只等闲 五岭逶迤腾细浪 乌蒙磅礴走泥丸 金沙水拍云崖暖 大渡桥横铁索寒 更喜岷山千里雪 三军过后尽开颜”，Font 属性值设置为“华文新魏楷”、二号，ForeColor 属性值设置为红色。

- 添加一个垂直滚动条控件并命名为 VScroll1，然后将其 Max 属性值设置为 30，Min 属性值设置为 10，SmallChange 属性值设置为 1，LargeChange 属性值设置为 2。
- 添加两个标签控件并分别命名为 lblFast 和 lblSlow，然后将它们的 Caption 属性值分别设置为“快”和“慢”。
- 添加一个复选框控件并命名为 chkMove，然后将其 Caption 属性值设置为“移动”，Value 属性值设置为 1。
- 在工具箱窗口中双击 Timer 控件图标，添加计时器控件并保留其默认名称 Timer1，然后将其 Interval 属性值设置为 30。

(4) 在窗体 frmAnimation 的代码窗口中编写复选框控件 chkMove 的 Click 事件过程，程序代码如下：

```
'当单击复选框时执行以下事件过程
Private Sub chkMove_Click()
    '根据复选框的值设置是否启用计时器
    Timer1.Enabled = chkMove.Value
End Sub
```

(5) 在窗体 frmAnimation 的代码窗口中编写该窗体的 Load 事件过程，程序代码如下：

```
'当加载窗体时执行以下事件过程
Private Sub Form_Load()
    '根据计时器的 Interval 属性设置垂直滚动条的 Value 属性
    VScroll1.Value = Timer1.Interval
End Sub
```

(6) 在窗体 frmAnimation 的代码窗口中编写计时器控件 Timer1 的 Timer 事件过程，程序代码如下：

```
'每当经过预定的时间间隔后执行以下事件过程
Private Sub Timer1_Timer()
    If lblPoetry.Top + lblPoetry.Height < 0 Then
        lblPoetry.Top = Me.ScaleHeight
    End If
    lblPoetry.Top = lblPoetry.Top - 5
End Sub
```

(7) 在窗体 frmAnimation 的代码窗口中编写垂直滚动条控件 VScroll1 的 Change 事件过程，程序代码如下：

```
'当垂直滚动条的值发生变化时执行以下事件过程

Private Sub VScroll1_Change()
    '根据垂直滚动条的 Value 属性设置计时器的 Interval 属性
    Timer1.Interval = VScroll1.Value
End Sub
```

(8) 将窗体文件命名为 frmAnimation.frm，工程文件命名为工程 04-05.vbp。

程序测试

- (1) 按下 F5 键，以运行程序，此时标签文字将按照预先设定的速度自下而上移动。
- (2) 将滚动条的滑块向下移动，则标签文字的移动速度变慢；将滚动条的滑块向上移动，则标签文字的移动速度变快。
- (3) 取消对“移动”复选框的勾选，则标签文字停止移动；再次勾选“移动”复选框，则标签文字恢复移动。

相关知识

计时器 (Timer) 控件用于背景进程中，它是不可见的。通过引发 Timer 事件，计时器控件可以有规律地隔一段时间执行一次代码。在一个窗体上可以添加多个计时器控件。

1. 计时器控件的常用属性

计时器控件的常用属性如下所述。

- **Enabled:** 返回或设置计时器控件的有效性，该属性的值为布尔值，如果将属性值设置为 True，则每经过指定的时间间隔将触发 Timer 事件。通过把 Enabled 属性值设置为 False 可以使计时器控件无效，将取消由计时器控件的 Interval 属性所建立的倒计时。
- **Interval:** 返回或设置对计时器控件的计时事件各调用之间的毫秒数。语法格式如下：

```
oTimer.Interval [= milliseconds]
```

其中，oTimer 表示计时器控件；参数 milliseconds 为数值表达式，用于指定毫秒数，如果将其设置为 0（默认值），则计时器控件无效。如果将其设置为 1~65,535 之间的数值，则对计时器控件设置一个时间间隔（以毫秒计），在计时器控件的 Enabled 属性值设置为 True 时开始有效。例如，10,000 毫秒等于 10 秒；最大值为 65,535 毫秒，等于 1 分钟多一些。

提示：可以在程序设计时或在程序运行时设置计时器控件的 Interval 属性。在使用 Interval 属性时，应当记住：计时器控件的 Enabled 属性决定该控件是否对时间的推移做出响应。将 Enabled 属性值设置为 False 会关闭计时器控件，将 Enabled 属性值设置为 True 则会打开计时器控件。当计时器控件设置为有效时，倒计时总是从其 Interval 属性的设置值开始的。

2. 计时器控件的事件

计时器控件有一个 Timer 事件，通过对该事件创建事件过程可以告诉 Visual Basic 在每次 Interval 到时该做什么。在任务 4.5 中，通过 Timer 事件过程包含的代码对标签控件的 Top 属性进行修改，从而实现文字动画效果。