

نکته : توابع عملیاتی API در صورتیکه کار خود را بدرستی انجام ندهند مقدار صفر، در غیر این صورت مقدار غیر صفر را برمی‌گردانند.

### ShellAbout()

با استفاده از این تابع API می‌توان پنجره About ویندوز را باز کرد و همچنین می‌توان در متن آن تغییراتی را اعمال کرد.

```
[DllImport("shell32.DLL")]
[DllImport("User32.DLL")]
public string Lpwindow { get; set; }
public IntPtr Hicon { get; set; }
public static extern int ShellAbout(IntPtr hwnd,string szApp,string szotherstuff,IntPtr hicon );
public static extern int FindWindow(string str,string lpwindow);
```

طریقه استفاده از تابع فوق

```
private void btnbutton1_Click(object sender, EventArgs e)
{
    var Window = (IntPtr)FindWindow(null, Lpwindow);
    ShellAbout(Window, "===WINDOWS===", "HELLO", Hicon);
}
```

### SetSuspendState ()

با استفاده از این تابع API می‌توان سیستم را به حالت Hibernate برد.

```
[DllImport("powrprof.dll")]
public static extern int SetSuspendState(int hibernate,int forcecritical,int DisablewakeEvent);
```

طریقه استفاده از تابع فوق

```
private void btnbutton1_Click(object sender, EventArgs e)
{
    SetSuspendState(1, 0, 0);
}
```

### LockWorkStation()

با اجرای این تابع سیستم به حالت Lock می‌رود و کاربر برای استفاده از سیستم باید کلمه عبور را وارد کند.

```
[DllImport("user32.DLL")]
public static extern int LockWorkStation();
```

طریقه استفاده از تابع فوق

```
private void btnbutton1_Click(object sender, EventArgs e)
{
    LockWorkStation();
}
```

## FatalAppExit()

در صورت اجرای این تابع برنامه یک پیغام خطا نمایش میدهد و بعد از بستن این پیام برنامه بسته می‌شود.

```
DllImport("kernel32.DLL")
public static extern int FatalAppExit(int uAction, string ipmesseg);
```

طریقه استفاده از تابع فوق

```
private void btnbutton1_Click(object sender, EventArgs e)
{
    FatalAppExit(0, "ERROR PROGRAM");
}
```

## timeGetTime()

این تابع مدت زمان روشن بودن سیستم را به میلی ثانیه برمیگرداند.

```
[DllImport("Winmm.DLL")]
public static extern long timeGetTime();

private void timer1_Tick(object sender, EventArgs e)
{
    var sd = Convert.ToString(timeGetTime());
    int i = Convert.ToInt32(sd.Substring(0, sd.Length - 3));
    int m;
    if (i < 60)
        textBox1.Text = @"00:00:" + i;
    else if (i >= 60 && i < 3600)
    {
        m = i / 60;
        S = i % 60;
        textBox1.Text = @"00:" + m + @":";
    }
    else
    {
        var h = i / 3600;
        var mm = i % 3600;

        if (mm > 59)
        {
            m = mm / 60;
            S = mm % 60;
            textBox1.Text = h + @":" + m.ToString(CultureInfo.InvariantCulture) + @":" + S;
        }
        else
        {
            textBox1.Text = h + @":00:" + mm;
        }
    }
}
```

طریقه استفاده از تابع فوق

```
private void Form1_Load(object sender, EventArgs e)
{
```

```
timer1.Start();  
}
```