

Использование WinAPI в C#

DLL Import

```
using System.Runtime.InteropServices;
```

```
public class Win32
```

```
{
```

```
    [DllImport("User32.dll")]
```

```
    public static extern void SetWindowText(IntPtr hwnd, String lpString);
```

```
}
```

Вызов функций Windows API, имеющих выходной строковый параметр char*

```
using System.Text; // для StringBuilder
[DllImport("user32.dll")]
public static extern int GetWindowText(IntPtr hwnd,
                                     StringBuilder buf, int nMaxCount);

StringBuilder sTitleBar = new StringBuilder(255);
GetWindowText(this.Handle, sTitleBar, sTitleBar.Capacity);
MessageBox.Show(sTitleBar.ToString());
```

Вызов функций, требующих struct

```
[StructLayout(LayoutKind.Sequential)]
```

```
public struct RECT
```

```
{
```

```
    public int left ;
```

```
    public int top;
```

```
    public int right;
```

```
    public int bottom;
```

```
}
```

```
[DllImport("user32.dll")]
```

```
public static extern int GetWindowRect(IntPtr hwnd, ref RECT rc);
```

Изменение типа, применяемого для маршалинга по умолчанию

Например, мы хотим вызвать функцию `GetClassName`, который принимает параметр `LPSTR (char)` даже в `Unicode`-версиях. Если вы передадите строку, общезыковая исполняющая среда (`CLR`) преобразует ее в серию `TCHAR`. Но с помощью атрибута `MarshalAs` можно переопределить то, что предлагается по умолчанию:*

```
[DllImport("user32.dll")]  
public static extern int GetClassName(IntPtr hwnd,  
    [MarshalAs(UnmanagedType.LPStr)] StringBuilder buf,  
    int nMaxCount);
```

Использование функций, требующих struct

```
int w, h;  
RECT rc = new RECT();  
GetWindowRect(this.Handle, ref rc);  
w = rc.right - rc.left;  
h = rc.bottom - rc.top;  
MessageBox.Show("Ширина формы: " + w + "\n\rВысота формы: " + h);
```

Использование функций, требующих struct

```
int w, h;  
RECT rc = new RECT();  
GetWindowRect(this.Handle, ref rc);  
w = rc.right - rc.left;  
h = rc.bottom - rc.top;  
MessageBox.Show("Ширина формы: " + w + "\n\rВысота формы: " + h);
```