

Namespace SKM

Classes

[Config](#)

参数配置类

[DpiAwareness](#)

[KMSimulatorService](#)

键鼠模拟器服务 封装skm.dll的函数，提供键鼠模拟器服务 x64环境需要复制x64\skm.dll，x86环境需要复制x86\skm.dll到当前目录，意思是：x64环境需要使用x64\skm.dll，x86环境需要使用x86\skm.dll 本服务类会自动根据当前环境复制DLL到当前目录，可以无感知使用

[SkmcCore](#)

注意: x86 和 x64 的 dll 是不同的，需要根据实际情况选择

x86 的 dll 是 x86\skm.dll，x64 的 dll 是 x64\skm.dll

使用 `KMSimulatorService` 服务类可以无感知使用，会自动根据当前环境复制 DLL 到当前目录
具体 `KMSimulatorService` 类使用请参考 [KMSimulatorService](#)

Class Config

Namespace: [SKM](#)

Assembly: SKSimulator.dll








参数配置类

```
public static class Config
```

Inheritance

[object](#)  ← Config

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,
[object.ToString\(\)](#) 

Properties

KMDeivicePID

键鼠模拟器设备PID

```
public static int KMDeivicePID { get; set; }
```

Property Value

[int](#) 

KMDeiviceVID

键鼠模拟器设备VID

```
public static int KMDeiviceVID { get; set; }
```

Property Value

[int](#)

KMMouseMoveMode

鼠标移动模式

```
public static int KMMouseMoveMode { get; set; }
```

Property Value

[int](#)

KMOffsetOfClick

点击偏移量,单位像素 为了避免每次点击都点击到同一个位置，可以设置一个偏移量，实际点击位置为点击位置减去偏移量的一个随机值

```
public static int KMOffsetOfClick { get; set; }
```

Property Value

[int](#)

KMOutputStringType

输出字符串编码类型,默认使用剪贴板粘贴输出字符串。优点是输出字符多时速度更快且不受输入法影响

```
public static int KMOutputStringType { get; set; }
```

Property Value

[int](#)

KMVerifyUserData

```
public static string KMVerifyUserData { get; set; }
```

Property Value

[string](#)

ProcessDpiAwareness

进程DPI感知值,如果使用库的应用已经设置DPI感知, 此参数无效。 0: 不设置,进程对DPI完全不知晓, 按逻辑像素绘制, 可能会出现点击不准确的情况。 1: PROCESS_SYSTEM_DPI_AWARE 默认值,进程只根据主显示器DPI绘制, DPI感知生效。 2: PROCESS_PER_MONITOR_DPI_AWARE, 进程根据每个显示器DPI绘制,DPI感知生效。

```
public static int ProcessDpiAwareness { get; set; }
```

Property Value

[int](#)

Class DpiAwareness

Namespace: [SKM](#)








Assembly: SKSimulator.dll

```
public static class DpiAwareness
```

Inheritance

[object](#)  ← DpiAwareness

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,
[object.ToString\(\)](#) 

Methods

SetProcessDPIAware()

设置进程DPI感知(旧方法) 注意：此方法必须在任何窗口创建之前调用，如果使用库的应用已经设置DPI感知，调用此方法无效

```
public static extern bool SetProcessDPIAware()
```

Returns

[bool](#) 

是否成功

SetProcessDpiAwareness(int)

设置进程DPI感知,如果使用库的应用已经设置DPI感知，此方法无效。 此方法必须在任何窗口创建之前调用

```
public static void SetProcessDpiAwareness(int processDpiAwareness)
```

Parameters

processDpiAwareness [int](#)

Exceptions

[Exception](#)

SetProcessDpiAwarenessContext(int)

设置进程DPI感知(新方法) 注意：此方法必须在任何窗口创建之前调用

```
public static extern int SetProcessDpiAwarenessContext(int value)
```

Parameters

value [int](#)

DPI感知值

Returns

[int](#)

是否成功

Class KMSimulatorService

Namespace: [SKM](#)

Assembly: SKSimulator.dll








键鼠模拟器服务 封装skm.dll的函数，提供键鼠模拟器服务 x64环境需要复制x64\skm.dll，x86环境需要复制x86\skm.dll到当前目录，意思是：x64环境需要使用x64\skm.dll，x86环境需要使用x86\skm.dll 本服务类会自动根据当前环境复制DLL到当前目录，可以无感知使用

```
public static class KMSimulatorService
```

Inheritance

[object](#)  ← KMSimulatorService

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,
[object.ToString\(\)](#) 

Properties

DeviceData

```
public static nint DeviceData { get; }
```

Property Value

[nint](#) 

Methods

CloseDevice()

关闭设备

```
public static void CloseDevice()
```

Init(int, int, string)

```
public static void Init(int deviceVID, int devicePID, string verifyUserData)
```

Parameters

deviceVID [int](#)

devicePID [int](#)

verifyUserData [string](#)

IsDeviceOpen()

判断设备是否打开

```
public static bool IsDeviceOpen()
```

Returns

[bool](#)

是否打开

OpenDevice(uint)

打开设备

```
public static void OpenDevice(uint deviceID)
```

Parameters

deviceID [uint](#)

设备ID

SearchDevice(int, int)

搜索设备

```
public static uint SearchDevice(int deviceVID, int devicePID)
```

Parameters

deviceVID [int](#)

设备VID

devicePID [int](#)

设备PID

Returns

[uint](#)

设备ID

Class SkmCore

Namespace: [SKM](#)

Assembly: SKSimulator.dll

注意: x86 和 x64 的 dll 是不同的, 需要根据实际情况选择

x86 的 dll 是 x86\skm.dll, x64 的 dll 是 x64\skm.dll








使用 [KMSimulatorService](#) 服务类可以无感知使用, 会自动根据当前环境复制 DLL 到当前目录
具体 [KMSimulatorService](#) 类使用请参考 [KMSimulatorService](#)

```
public class SkmCore
```

Inheritance

[object](#)  ← SkmCore

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,
[object.ToString\(\)](#) 

Methods

HKMCheckPressedKeysCS(uint)

```
public static string HKMCheckPressedKeysCS(uint Flags)
```

Parameters

Flags [uint](#) 

Returns

[string](#) 

HKMClose(nint)

```
public static extern bool HKMClose(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMDelayRnd(nint, uint, uint)

```
public static extern bool HKMDelayRnd(nint HKMData, uint MinTime, uint MaxTime)
```

Parameters

HKMData [nint](#)

MinTime [uint](#)

MaxTime [uint](#)

Returns

[bool](#)

HKMEnableOSMouseAccelerate(bool, bool)

```
public static extern bool HKMEnableOSMouseAccelerate(bool Enable, bool Save)
```

Parameters

Enable [bool](#)

Save [bool](#)

Returns

[bool](#)

HKMGetCursorPos(nint, ref int, ref int)

```
public static extern bool HKMGetCursorPos(nint HKMData, ref int X, ref int Y)
```

Parameters

HKMData [nint](#)

X [int](#)

Y [int](#)

Returns

[bool](#)

HKMGetCursorPos2(nint)

```
public static extern uint HKMGetCursorPos2(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[uint](#)

HKMGetDevInfo(nint, uint, bool)

```
public static extern uint HKMGetDevInfo(nint HKMData, uint Index, bool Mouse)
```

Parameters

HKMData [nint](#)

Index [uint](#)

Mouse [bool](#)

Returns

[uint](#)

HKMGetDevString(nint, uint, bool, ref uint)

```
public static extern nint HKMGetDevString(nint HKMData, uint Index, bool Mouse, ref
uint Length)
```

Parameters

HKMData [nint](#)

Index [uint](#)

Mouse [bool](#)

Length [uint](#)

Returns

[nint](#)

HKMGetDevStringCS(nint, uint, bool)

```
public static string HKMGetDevStringCS(nint HKMData, uint Index, bool Mouse)
```

Parameters

HKMData [nint](#)

Index [uint](#)

Mouse [bool](#)

Returns

[string](#)

HKMGetKeyboardLEDState(nint, uint)

```
public static extern bool HKMGetKeyboardLEDState(nint HKMData, uint Index)
```

Parameters

HKMData [nint](#)

Index [uint](#)

Returns

[bool](#)

HKMGetKeyboardMode(nint)

```
public static extern uint HKMGetKeyboardMode(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[uint](#)

HKMGetMouseMode(nint)

```
public static extern uint HKMGetMouseMode(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[uint](#)

HKMGetOSMouseSpeed()

```
public static extern int HKMGetOSMouseSpeed()
```

Returns

[int](#)

HKMGetSerialNumber(nint, bool)

```
public static extern uint HKMGetSerialNumber(nint HKMData, bool Mouse)
```

Parameters

HKMData [nint](#)

Mouse [bool](#)

Returns

[uint](#)

HKMIsKeyBusy(nint)

```
public static extern bool HKMIsKeyBusy(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMIsKeyDown(nint, string)

```
public static extern bool HKMIsKeyDown(nint HKMData, string KeyName)
```

Parameters

HKMData [nint](#)

KeyName [string](#)

Returns

[bool](#)

HKMIsMouseBusy(nint)

```
public static extern bool HKMIsMouseBusy(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMIsMouseButtonDown(nint, uint)

```
public static extern bool HKMIsMouseButtonDown(nint HKMData, uint Index)
```


Parameters

HKMData [nint](#)

Index [uint](#)

Returns

[bool](#)

HKMIsOSMouseAccelerateEnabled()

```
public static extern bool HKMIsOSMouseAccelerateEnabled()
```

Returns

[bool](#)

HKMIsOpen(nint, uint)

```
public static extern bool HKMIsOpen(nint HKMData, uint Flags)
```

Parameters

HKMData [nint](#)

Flags [uint](#)

Returns

[bool](#)

HKMKeyDown(nint, string)

```
public static extern bool HKMKeyDown(nint HKMData, string KeyName)
```

Parameters

HKMData [nint](#)

KeyName [string](#)

Returns

[bool](#)

HKMKeyPress(nint, string)

```
public static extern bool HKMKeyPress(nint HKMData, string KeyName)
```

Parameters

HKMData [nint](#)

KeyName [string](#)

Returns

[bool](#)

HKMKeyUp(nint, string)

```
public static extern bool HKMKeyUp(nint HKMData, string KeyName)
```

Parameters

HKMData [nint](#)

KeyName [string](#)

Returns

[bool](#)

HKMLeftClick(nint)

```
public static extern bool HKMLeftClick(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMLeftDoubleClick(nint)

```
public static extern bool HKMLeftDoubleClick(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMLeftDown(nint)

```
public static extern bool HKMLeftDown(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMLeftUp(nint)

```
public static extern bool HKMLeftUp(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMMiddleClick(nint)

```
public static extern bool HKMMiddleClick(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMMiddleDoubleClick(nint)

```
public static extern bool HKMMiddleDoubleClick(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMMiddleDown(nint)

```
public static extern bool HKMMiddleDown(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMMiddleUp(nint)

```
public static extern bool HKMMiddleUp(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMMouseWheel(nint, int)

```
public static extern bool HKMMouseWheel(nint HKMData, int Count)
```

Parameters

HKMData [nint](#)

Count [int](#)

Returns

[bool](#)

HKMMouseWheelP(nint, int)

```
public static extern bool HKMMouseWheelP(nint HKMData, int Count)
```

Parameters

HKMData [nint](#)

Count [int](#)

Returns

[bool](#)

HKMMoveR(nint, int, int)

```
public static extern bool HKMMoveR(nint HKMData, int X, int Y)
```

Parameters

HKMData [nint](#)

X [int](#)

Y [int](#)

Returns

[bool](#)

HKMMoveR2(nint, int, int)

```
public static extern bool HKMMoveR2(nint HKMData, int X, int Y)
```

Parameters

HKMData [nint](#)

X [int](#)

Y [int](#)

Returns

[bool](#)

HKMMoveRP(nint, int, int)

```
public static extern bool HKMMoveRP(nint HKMData, int X, int Y)
```

Parameters

HKMData [nint](#)

X [int](#)

Y [int](#)

Returns

[bool](#)

HKMMoveTo(nint, int, int)

```
public static extern bool HKMMoveTo(nint HKMData, int X, int Y)
```

Parameters

HKMData [nint](#)

X [int](#)

Y [int](#)

Returns

[bool](#)

HKMOpen(uint, uint)

```
public static extern nint HKMOpen(uint DeviceId, uint DpiMode)
```

Parameters

DeviceId [uint](#)

DpiMode [uint](#)

Returns

[nint](#)

HKMOpen2(uint, uint, uint)

```
public static extern nint HKMOpen2(uint DeviceId1, uint DeviceId2, uint DpiMode)
```

Parameters

DeviceId1 [uint](#)

DeviceId2 [uint](#)

DpiMode [uint](#)

Returns

[nint](#)

HKMOutputString(nint, string)

```
public static extern bool HKMOutputString(nint HKMData, string Str)
```


Parameters

HKMData [nint](#)

Str [string](#)

Returns

[bool](#)

HKMReleaseKeyboard(nint)

```
public static extern bool HKMReleaseKeyboard(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMReleaseMouse(nint)

```
public static extern bool HKMReleaseMouse(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMRightClick(nint)

```
public static extern bool HKMRightClick(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMRightDoubleClick(nint)

```
public static extern bool HKMRightDoubleClick(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMRightDown(nint)

```
public static extern bool HKMRightDown(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMRightUp(nint)

```
public static extern bool HKMRightUp(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMSearchDevice(uint, uint, uint)

```
public static extern uint HKMSearchDevice(uint Vid, uint Pid, uint DeviceType)
```

Parameters

Vid [uint](#)

Pid [uint](#)

DeviceType [uint](#)

Returns

[uint](#)

HKMSearchDevice2(uint, uint, uint, uint)

```
public static extern uint HKMSearchDevice2(uint Vid, uint Pid, uint SN, uint DeviceType)
```

Parameters

Vid [uint](#)

Pid [uint](#)

SN [uint](#)

DeviceType [uint](#)

Returns

[uint](#)

HKMSearchDeviceAll(uint, uint, uint)

```
public static extern nint HKMSearchDeviceAll(uint Vid, uint Pid, uint DeviceType)
```

Parameters

Vid [uint](#)

Pid [uint](#)

DeviceType [uint](#)

Returns

[nint](#)

HKMSetAbsMouseScrnRes(nint, int, int)

```
public static extern bool HKMSetAbsMouseScrnRes(nint HKMData, int Width, int Height)
```

Parameters

HKMData [nint](#)

Width [int](#)

Height [int](#)

Returns

[bool](#)

HKMSetKeyInterval(nint, uint, uint)

```
public static extern bool HKMSetKeyInterval(nint HKMData, uint MinTime, uint MaxTime)
```

Parameters

HKMData [nint](#)

MinTime [uint](#)

MaxTime [uint](#)

Returns

[bool](#)

HKMSetLightMode(nint, uint, bool)

```
public static extern bool HKMSetLightMode(nint HKMData, uint Mode, bool Mouse)
```

Parameters

HKMData [nint](#)

Mode [uint](#)

Mouse [bool](#)

Returns

[bool](#)

HKMSetMode(nint, uint, uint)

```
public static extern bool HKMSetMode(nint HKMData, uint Index, uint Mode)
```

Parameters

HKMData [nint](#)

Index [uint](#)

Mode [uint](#)

Returns

[bool](#)

HKMSetMouseInterval(nint, uint, uint)

```
public static extern bool HKMSetMouseInterval(nint HKMData, uint MinTime, uint MaxTime)
```

Parameters

HKMData [nint](#)

MinTime [uint](#)

MaxTime [uint](#)

Returns

[bool](#)

HKMSetMouseMoveTimeout(nint, uint)

```
public static extern bool HKMSetMouseMoveTimeout(nint HKMData, uint Timeout)
```

Parameters

HKMData [nint](#)

Timeout [uint](#)

Returns

[bool](#)

HKMSetMousePosMaxOffset(nint, uint)

```
public static extern bool HKMSetMousePosMaxOffset(nint HKMData, uint dwOffset)
```

Parameters

HKMData [nint](#)

dwOffset [uint](#)

Returns

[bool](#)

HKMSetMousePosPrecision(nint, uint)

```
public static extern bool HKMSetMousePosPrecision(nint HKMData, uint Precision)
```

Parameters

HKMData [nint](#)

Precision [uint](#)

Returns

[bool](#)

HKMSetMouseSpeed(nint, uint)

```
public static extern bool HKMSetMouseSpeed(nint HKMData, uint MouseSpeed)
```

Parameters

HKMData [nint](#)

MouseSpeed [uint](#)

Returns

[bool](#)

HKMSetOSMouseSpeed(int, bool)

```
public static extern bool HKMSetOSMouseSpeed(int Speed, bool Save)
```

Parameters

Speed [int](#)

Save [bool](#)

Returns

[bool](#)

HKMSetResetMode(nint, uint, bool)

```
public static extern bool HKMSetResetMode(nint HKMData, uint Mode, bool Mouse)
```

Parameters

HKMData [nint](#)

Mode [uint](#)

Mouse [bool](#)

Returns

[bool](#)

HKMSetResetTime(nint, uint, bool)


```
public static extern bool HKMSetResetTime(nint HKMData, uint Time, bool Mouse)
```

Parameters

HKMData [nint](#)

Time [uint](#)

Mouse [bool](#)

Returns

[bool](#)

HKMVerifyUserData(nint, string, bool)

```
public static extern bool HKMVerifyUserData(nint HKMData, string Str, bool Mouse)
```

Parameters

HKMData [nint](#)

Str [string](#)

Mouse [bool](#)

Returns

[bool](#)

HKMVerifyUserData2(nint, string, bool)

```
public static extern uint HKMVerifyUserData2(nint HKMData, string Str, bool Mouse)
```

Parameters

HKMData [nint](#)

Str [string](#)

Mouse [bool](#)

Returns

[uint](#)

HKMXBtn1Click(nint)

```
public static extern bool HKMXBtn1Click(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMXBtn1DoubleClick(nint)

```
public static extern bool HKMXBtn1DoubleClick(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMXBtn1Down(nint)

```
public static extern bool HKMXBtn1Down(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMXBtn1Up(nint)

```
public static extern bool HKMXBtn1Up(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMXBtn2Click(nint)

```
public static extern bool HKMXBtn2Click(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMXBtn2DoubleClick(nint)

```
public static extern bool HKMXBtn2DoubleClick(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMXBtn2Down(nint)

```
public static extern bool HKMXBtn2Down(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)

HKMXBtn2Up(nint)

```
public static extern bool HKMXBtn2Up(nint HKMData)
```

Parameters

HKMData [nint](#)

Returns

[bool](#)