

# Tuxis Input System

Документация



## КЛАСС INPUT

*Класс Input отвечает за обработку ввода в рамках окна или всей рабочей области.*

Заголовочный файл: « **input.h** »

Статичная библиотека: « **InputModule.lib** »

Динамическая библиотека: « **InputModule.dll** »

### Основные методы:

#### **Input( HWND pWindowHandler );**

##### **Параметры**

Конструктор. Используется для инициализации устройства DirectInput.

##### **Описание**

pWindowHandler – идентификатор окна в системе Windows. Если передать нулевое значение то класс будет действовать в пределах всей рабочей области экрана.

#### **void Update();**

##### **Параметры**

Нет.

##### **Описание**

Обновление состояния устройств.

## Клавиатура

**bool KeyDown( BYTE key );**

### Параметры

key – код клавиши (См. пространство имен Key)

### Описание

Метод используется, чтобы проверить, была ли нажата клавиша в момент последнего вызова метода Update.

Возвращает **false**, если клавиша не была нажата, и **true**, если клавиша была нажата.

**bool KeyHit( BYTE key );**

### Параметры

key – код клавиши (См. пространство имен Key)

### Описание

Метод используется, чтобы обнаружить, произошло ли нажатие клавиши с момента последнего вызова этого метода.

Возвращает **false**, если клавиша не была нажата, и **true**, если произошло нажатие клавиши.

**bool KeyUp( BYTE key );**

### Параметры

key – код клавиши (См. пространство имен Key)

### Описание

Метод используется, чтобы обнаружить, произошло ли отпускание клавиши с момента последнего вызова этого метода.

Возвращает **false**, если клавиша не была отпущена, и **true**, если произошло отпускание клавиши.

## Мышь

**bool MouseDown( BYTE key );**

### Параметры

key – код клавиши (См. пространство имен Key (MOUSE\_ префикс))

### Описание

Метод используется, чтобы проверить, была ли нажата клавиша мыши в момент последнего вызова метода Update.

Возвращает **false**, если клавиша не была нажата, и **true**, если клавиша была нажата.

**bool MouseHit( BYTE key );**

### Параметры

key – код клавиши (См. пространство имен Key (MOUSE\_ префикс))

### Описание

Метод используется, чтобы обнаружить, произошло ли нажатие клавиши мыши с момента последнего вызова этого метода.

Возвращает **false**, если клавиша не была нажата, и **true**, если произошло нажатие клавиши.

**bool MouseUp( BYTE key );**

### Параметры

key – код клавиши (См. пространство имен Key (MOUSE\_ префикс))

### Описание

Метод используется, чтобы обнаружить, произошло ли отпускание клавиши мыши с момента последнего вызова этого метода.

Возвращает **false**, если клавиша не была отпущена, и **true**, если произошло отпускание клавиши.

**int MouseSpeedX();**

**int MouseSpeedY();**

**int MouseSpeedZ();**

### Параметры

Нет.

### Описание

Методы **MouseSpeedX** и **MouseSpeedY** возвращают зарегистрированную скорость перемещения указателя (курсора) мыши по координатам **X** и **Y** в момент последнего вызова **Update**.

Метод **MouseSpeedZ** предназначен для определения скорости вращения колёсика мыши в момент последнего вызова **Update**.

Возвращает **false**, если клавиша не была отпущена, и **true**, если произошло отпускание клавиши.

## Курсор

```
void SetMousePosition(int x, int y);
```

### Параметры

x, y - координаты курсора.

### Описание

Метод позволяет установить курсор в координаты X,Y.

```
void GetMousePosition(int& x, int& y);
```

### Параметры

x, y – координаты для записи.

### Описание

Метод позволяет получить координаты позиции курсора.