#### CAButton

###### 类说明

按钮类， CAButton有五种状态，每种状态又分为独立的前景与背景，背景支持纯色层、图片、与九宫格拉升的图片，前景支持图片与文字。

CAButton分别可获取点击按下、多次重复点击按下、点击按下后在按钮范围内滑动、点击按下后再按钮外部滑动、点击按下并在按钮内部抬起与点击按下再任意地方抬起五种触摸事件的回调，且在回调函数中还可以获取到相对于按钮的触摸坐标。

###### 基类

[CAControl](../CAControl.docx)

###### 属性

|  |  |  |
| --- | --- | --- |
| 访问修饰符 | 属性名 | 说明 |
| public | [Frame](#Frame) | button的frame |
| public | [BackGroundView](#BackGroundView) | 背景视图 |
| public | [Image](#Image) | 不同状态的图片 |
| public | [Type](#Type) | button的类型 |
| public | [Title](#Title) | button的标题 |
| public | [State](#State) | button的状态 |
| protected | [TextTag](#TextTag) | 文本标签 |
| public | [Opacity](#Opacity) | 透明度 |
| public | [Color](#Color) | 颜色 |

###### 方法

|  |  |  |
| --- | --- | --- |
| 访问修饰符 | 方法名 | 说明 |
| public | [create](#create) | 创建一个button，并指定type |
| public | [createWithFrame](#createWithFrame) | 创建一个button，并指定frame和type |
| public | [createWithCenter](#createWithCenter) | 创建一个button，并指定frame和type |
| public | [setBackGroundViewForState](#setBackGroundViewForState) | 设置button各个状态的背景视图 |
| public | [setImageForState](#setImageForState) | 设置button各个状态的显示图片 |
| public | [setTitleForState](#setTitleForState) | 设置button各个状态的显示标题 |
| public | [setImageColorForState](#setImageColorForState) | 设置button各个状态的图片显示颜色 |
| public | [setTitleColorForState](#setTitleColorForState) | 设置button各个状态的标题显示颜色 |
| public | [setTitleFontName](#setTitleFontName) | 设置标题的字体 |
| public | [setControlState](#setControlState) | 设置button的状态 |
| public | [isTextTagEqual](#isTextTagEqual) | 判断文本标签是否相等 |
| public | [interruptTouchState](#interruptTouchState) | 中断按钮回调事件 |
| public | [setOpacity](#setOpacity) | 设置button透明度 |

###### 属性介绍

**Frame**

类型：CCRect

解释：[CAView](../../CAView.docx)子类共有属性，get/set{}。

**BackGroundView**

类型：CAView\*

解释：button不同状态下的背景view

**Image**

类型：CAImage\*

解释：button不同状态下的图片

**Type**

类型：CAButtonType

解释：button的类型，这是一个枚举类型。

typedef enum

{

CAButtonTypeCustom = 0,

CAButtonTypeSquareRect,

CAButtonTypeRoundedRect,

CAButtonTypeRounded3DRect

} CAButtonType;

包括四种button类型，采用第一种button创建后看不见任何明显效果，需要自己设置；第二种button，默认为创建的button添加了矩形的外边框；第三种button，默认为创建的button添加圆角的外边框；第四种button在第三种的基础上进行了填充。

CAButtonTypeCustom示例：

CAButton\* btn = CAButton::create(CAButtonTypeCustom);

btn->setTitleForState(CAControlStateAll,"Hello");

btn->setFrame(CCRect(winRect.size.width\*0.5, winRect.size.height\*0.5, 100, 50));

this->getView()->addSubview(btn);

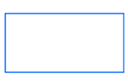


CAButtonTypeSquareRect示例：

CAButton\* btn = CAButton::create(CAButtonTypeSquareRect);

btn->setFrame(CCRect(winRect.size.width\*0.5, winRect.size.height\*0.5, 100, 50));

this->getView()->addSubview(btn);



CAButtonTypeRoundedRect示例：

CAButton\* btn = CAButton::create(CAButtonTypeRoundedRect);

btn->setFrame(CCRect(winRect.size.width\*0.5, winRect.size.height\*0.5, 100, 50));

this->getView()->addSubview(btn);



CAButtonTypeRounded3DRect示例：

CAButton\* btn = CAButton::create(CAButtonTypeRounded3DRect);

btn->setFrame(CCRect(winRect.size.width\*0.5, winRect.size.height\*0.5, 100, 50));

this->getView()->addSubview(btn);



**Title**

类型：string

解释：可设置button各个状态的显示标题

**State**

类型：CAControlState

解释：按钮的状态

**TextTag**

类型：string

解释：按钮的文本标签，get/set{}。

**Opacity**

类型：GLubyte

解释：按钮的透明度

**Color**

类型：ccColor4B

解释：按钮背景view的颜色

###### 方法介绍

**static CAButton\*** **create(CAButtonType buttonType)**

返回值：CAButton\*

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| [CAButtonType](#Type) | buttonType | 按钮类型 |

解释：创建一个button，并指定按钮的类型。

**static CAButton\*** **createWithFrame(const CCRect& rect, CAButtonType buttonType)**

返回值：CAButton\*

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CCRect | rect | 按钮的frame |
| [CAButtonType](#Type) | buttonType | 按钮的类型 |

解释：创建一个button，并指定按钮的frame和类型。

**static CAButton\*** **createWithCenter(const CCRect& rect, CAButtonType buttonType)**

返回值：CAButton\*

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CCRect | rect | 按钮的frame |
| CAButtonType | buttonType | 按钮的类型 |

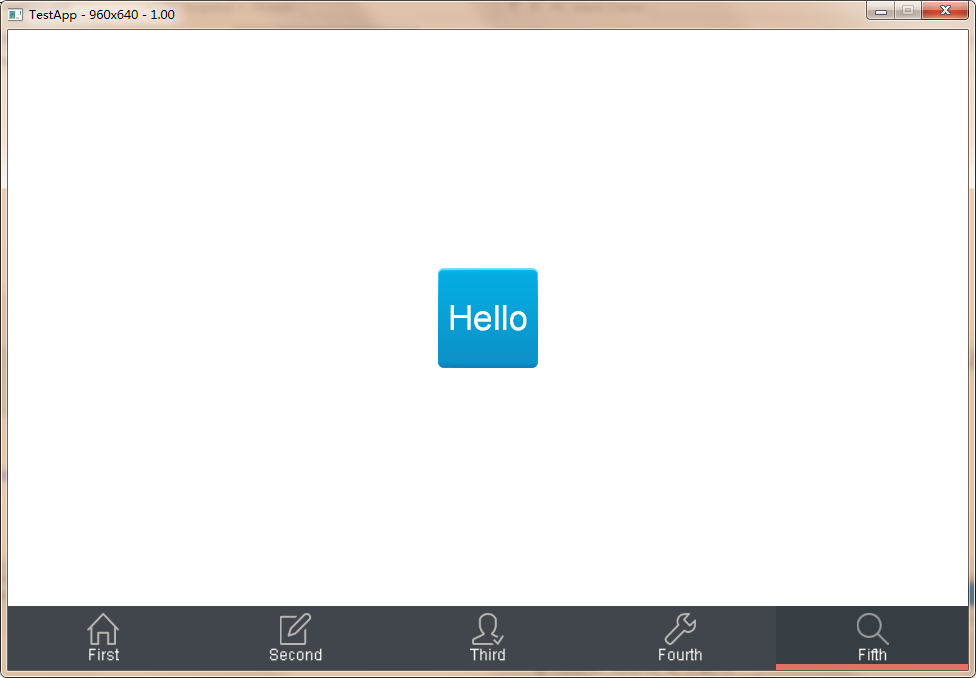
解释：创建一个button，并指定按钮的frame和类型，其中frame的size成员指的是按钮中心位置在屏幕上的坐标。

示例：

CAButton\* btn = CAButton::createWithCenter(CCRect(winRect.size.width\*0.5, winRect.size.height\*0.5, 100, 100), CAButtonTypeRounded3DRect);

btn->setTitleForState(CAControlStateAll,"Hello");

this->getView()->addSubview(btn);

****

**void** **setBackGroundViewForState(CAControlState controlState, CAView \*var)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CAControlState | controlState | 按钮的状态 |
| CAView\* | var | 背景视图 |

解释：设置按钮各个状态的背景视图

示例：

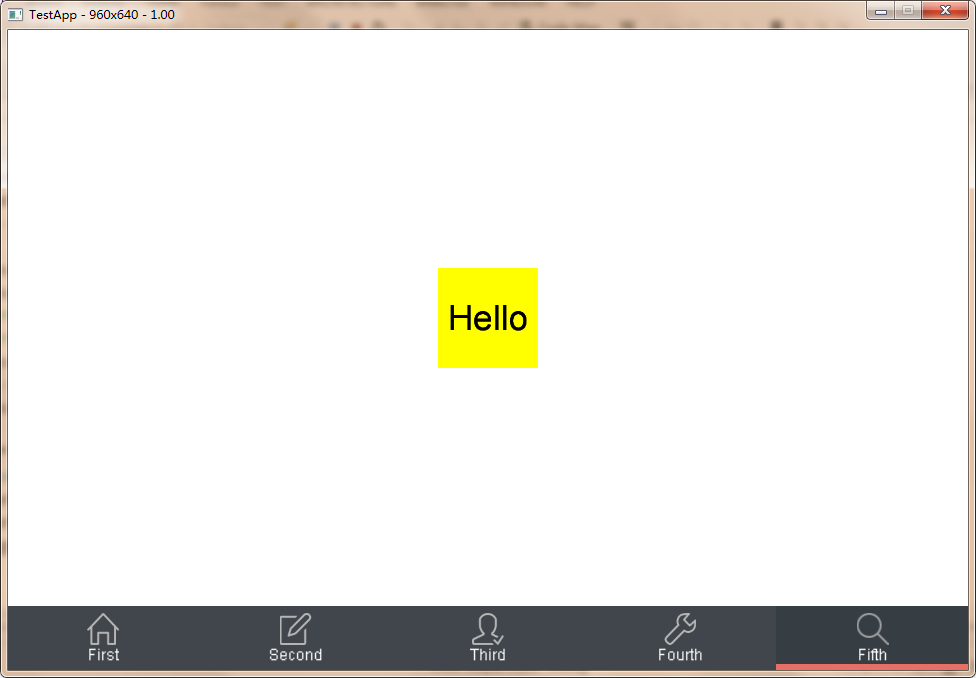
CAView\* p\_View=CAView::createWithColor(ccc4(255,255,0,255));

CAButton\* btn = CAButton::createWithCenter(CCRect(winRect.size.width\*0.5, winRect.size.height\*0.5, 100, 100), CAButtonTypeCustom);

btn->setTitleForState(CAControlStateAll,"Hello");

btn->setBackGroundViewForState(CAControlStateAll, p\_View);

this->getView()->addSubview(btn);



**void** **setImageForState(CAControlState controlState, CAImage\* var)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CAControlState | controlState | 按钮的状态 |
| CAImage\* | var | 背景图片 |

解释：设置不同状态下按钮的背景图片

**void** **setTitleForState(CAControlState controlState, std::string var)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CAControlState | controlState | 按钮的状态 |
| string | var | 按钮上显示的标题 |

解释：设置不同状态下按钮的显示标题

**void** **setImageColorForState(CAControlState controlState, ccColor3B var)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CAControlState | controlState | 按钮的状态 |
| ccColor3B | var | 背景图片的颜色 |

解释：设置不同状态下按钮背景图片的颜色

**void** **setTitleColorForState(CAControlState controlState, ccColor3B var)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CAControlState | controlState | 按钮的状态 |
| ccColor3B | var | 标题的颜色 |

解释：设置不同状态下按钮标题的颜色

**void** **setTitleFontName(std::string var)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| string | var | 标题的字体 |

解释：设置按钮标题的字体

**void** **setControlState(CAControlState var)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CAControlState | var | 按钮的状态 |

解释：设置按钮当前的状态

**bool** **isTextTagEqual(const char\* text)**

返回值：bool

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| char\* | text | 指定文本 |

解释：判断当前按钮标签与指定文本是否相同

**void** **interruptTouchState()**

返回值：void

解释：中断触摸事件，可以根据不同的条件中断触摸的响应。

**virtual void** **setOpacity(GLubyte opacity)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| GLubyte | opacity | 按钮的透明度 |

解释：设置按钮的透明度