#### CAWindow

###### 类说明

CAWindow主要的作用是作为所有view的载体、容器，分发触摸消息，协同viewController完成对应用程序的管理。应用程序通常只有一个window，即使存在多个window，也只能有一个window能够接收屏幕事件。应用程序启动时创建这个窗口，并往窗口中加入一或多个视图并显示出来，之后我们很少需要再次引用它。CAWindow是所有[CAView](../CAView.docx)的根，管理和协调应用程序的显示。

###### 基类

[CAView](../CAView.docx)

###### 属性

|  |  |  |
| --- | --- | --- |
| 访问修饰符 | 属性名 | 说明 |
| public | [RootViewController](#RootViewController) | 根视图管理器 |
| public | [ModalViewController](#ModalViewController) | 新弹出的viewController |

###### 方法

|  |  |  |
| --- | --- | --- |
| 访问修饰符 | 方法名 | 说明 |
| public | [init](#init) | 初始化window |
| public | [create](#create) | 创建一个window |
| public | [presentModalViewController](#presentModalViewController) | 弹出一个新viewController |
| public | [dismissModalViewController](#dismissModalViewController) | 隐藏新弹出的viewController |

###### 属性介绍

**RootViewController**

类型：CAViewController\*

解释：通过get/set可以获取和设置当前window的根视图管理器。

示例：

void CAWindow::setRootViewController(CAViewController \*var)

{

if (m\_pRootViewController)

{

m\_pRootViewController->removeViewFromSuperview();

m\_pRootViewController->release();

m\_pRootViewController = NULL;

}

if (var)

{

var->retain();

m\_pRootViewController = var;

m\_pRootViewController->addViewFromSuperview(this);

}

}

CAViewController\* CAWindow::getRootViewController()

{

return m\_pRootViewController;

}

**ModalViewController**

类型：CAViewController\*

解释：只读属性，获取新生成的viewController。

###### 方法介绍

**init()**

返回值：bool

解释：初始化window数据

**create()**

返回值：CAWindow\*

解释：创建一个window，并返回window对象,内部调用init方法进行初始化。

示例：

CAWindow\* \_window = CAWindow::create();

**void** **presentModalViewController(CAViewController\* controller, bool animated)**

返回值：void

参数：

|  |  |  |
| --- | --- | --- |
| 类型 | 参数名 | 说明 |
| CAViewController\* | controller | 新的viewController |
| bool | animated | 是否播放弹出动画 |

解释：生成一个新的viewController，并从底部将新生成的viewController从屏幕的底部弹出，覆盖原先的画面，可以自己实现新弹出的内容。

**void** **dismissModalViewController(bool animated)**

返回值：void

解释：将新生成的viewController隐藏。