# UIAlertView

## 1.Title

获取或设置UIAlertView上的标题。

## 2.Message

获取或设置UIAlertView上的消息

UIAlertView \*alertView = [[UIAlertView alloc] initWithTitle:@"Title" message:@"message" delegate:self cancelButtonTitle:@"取消" otherButtonTitles:@"确定", nil];

alertView.title = @"T";

alertView.message = @"M";

[alertView show];



## 3.numberOfButtons (只读)

返回UIAlertView上有多少按钮.

UIAlertView \*alertView = [[UIAlertView alloc] initWithTitle:@"Title" message:@"message" delegate:self cancelButtonTitle:@"取消" otherButtonTitles:@"确定", nil];

NSLog(@"%d",alertView.numberOfButtons);

[alertView show];



## 4.cancelButtonIndex

UIAlertView \*alert = [[UIAlertView alloc]initWithTitle:@"提示"

message:@"请选择一个按钮："

delegate:nil

cancelButtonTitle:@"取消"

otherButtonTitles:@"按钮一", @"按钮二", @"按钮三",nil];

[alert show];

NSLog(@"UIAlertView中取消按钮的角标是%d",alert.cancelButtonIndex);

效果：



注意不要认为取消按钮的角标是4，“取消”，“按钮一”，“按钮二”，“按钮三”的索引buttonIndex分别是0，1，2，3

## 5. alertViewStyle

### 5.1 UIAlertViewStyleLoginAndPasswordInput

UIAlertView \*alert = [[UIAlertView alloc] initWithTitle:@"产品信息展示" message:p.name delegate:nil cancelButtonTitle:@"取消" otherButtonTitles:@"确定", nil];

alert.alertViewStyle = UIAlertViewStyleLoginAndPasswordInput;

// 弹出UIAlertView

[alert show];



### 5.2 UIAlertViewStylePlainTextInput

UIAlertView \*alert = [[UIAlertView alloc] initWithTitle:@"产品信息展示" message:p.name delegate:nil cancelButtonTitle:@"取消" otherButtonTitles:@"确定", nil];

alert.alertViewStyle = UIAlertViewStylePlainTextInput;

// 弹出UIAlertView

[alert show];



### 5.3UIAlertViewStyleSecureTextInput

UIAlertView \*alert = [[UIAlertView alloc] initWithTitle:@"产品信息展示" message:p.name delegate:nil cancelButtonTitle:@"取消" otherButtonTitles:@"确定", nil];

alert.alertViewStyle = UIAlertViewStyleSecureTextInput;

// 弹出UIAlertView

[alert show];



## 6. - (UITextField \*)textFieldAtIndex:(NSInteger)textFieldIndex

返回textFieldIndex角标对应的文本框。

取出文本框文字 

## 7.手动的取消对话框

[alert dismissWithClickedButtonIndex:0 animated:YES];

## 8. delegate

作为UIAlertView的代理，必须遵守UIAlertViewDelegate。

### 1.当点击UIAlertView上的按钮时，就会调用，并且当方法调完后，UIAlertView会自动消失。

- (void)alertView:(UIAlertView \*)alertView clickedButtonAtIndex:(NSInteger)buttonIndex;

### 2.当UIAlertView即将出现的时候调用

- (void)willPresentAlertView:(UIAlertView \*)alertView;

### 3. 当UIAlertView完全出现的时候调用

- (void)didPresentAlertView:(UIAlertView \*)alertView;

### 4. 当UIAlertView即将消失的时候调用

- (void)alertView:(UIAlertView \*)alertView willDismissWithButtonIndex:(NSInteger)buttonIndex;

### 5. 当UIAlertView完全消失的时候调用

- (void)alertView:(UIAlertView \*)alertView didDismissWithButtonIndex:(NSInteger)buttonIndex;

## 9.注意UIAlertView调用show显示出来的时候，系统会自动强引用它，不会被释放。

## 10. 为UIAlertView添加子视图

在为UIAlertView对象太添加子视图的过程中，有点是需要注意的地方，如果删除按钮，也就是取消UIAlerView视图中所有的按钮的时候，可能会导致整个显示结构失衡。按钮占用的空间不会消失，我们也可以理解为这些按钮没有真正的删除，仅仅是他不可见了而已。如果在UIAlertview对象中仅仅用来显示文本，那么，可以在消息的开头添加换行符（@"\n）有助于平衡按钮底部和顶部的空间。

下面的代码用来演示如何为UIAlertview对象添加子视图的方法。

UIAlertView\*alert = [[UIAlertView alloc]initWithTitle:@"请等待"

                                                 message:nil

                                                 delegate:nil

                                                 cancelButtonTitle:nil

                                                 otherButtonTitles:nil];

[alert show];

UIActivityIndicatorView\*activeView = [[UIActivityIndicatorView alloc]initWithActivityIndicatorStyle:UIActivityIndicatorViewStyleWhiteLarge];

activeView.center = CGPointMake(alert.bounds.size.width / 2.0f, alert.bounds.size.height - 40.0f);

[activeView startAnimating];

[alert addSubview:activeView];



## 11. UIAlertView小例子

 UIAlertView默认情况下所有的text是居中对齐的。 那如果需要将文本向左对齐或者添加其他控件比如输入框时该怎么办呢？ 不用担心， iPhone SDK还是很灵活的， 有很多delegate消息供调用程序使用。 所要做的就是在

- (void)willPresentAlertView:(UIAlertView \*)alertView

中按照自己的需要修改或添加即可， 比如需要将消息文本左对齐，下面的代码即可实现:

-(void) willPresentAlertView:(UIAlertView \*)alertView

{

      for( UIView \* view in alertView.subviews )

      {

            if( [view isKindOfClass:[UILabel class]] )

            {

                  UILabel\* label = (UILabel\*) view;

                  label.textAlignment=UITextAlignmentLeft;

            }

      }

}

