IOS - ALERTS

Use of Alerts

Alerts are used to give important informations to user. Only after selecting the option in the alert view, we can proceed further using the app.

Important Properties

- alertViewStyle
- cancelButtonIndex
- delegate
- message
- numberOfButtons
- title

Important Methods

```
- (NSInteger)addButtonWithTitle:(NSString *)title
- (NSString *)buttonTitleAtIndex:(NSInteger)buttonIndex
- (void)dismissWithClickedButtonIndex:
(NSInteger)buttonIndex animated:(BOOL)animated
- (id)initWithTitle:(NSString *)title message:
(NSString *)message delegate:(id)delegate
cancelButtonTitle:(NSString *)cancelButtonTitle
otherButtonTitles:(NSString*)otherButtonTitles, ...
- (void)show
```

Update ViewController.h as follows —

Make your class conform to alert view delegate protocol by adding **< UIAlertViewDelegate>** as shown below in **ViewController.h.**

```
#import <UIKit/UIKit.h>

@interface ViewController : UIViewController<UIAlertViewDelegate>{

}
@end
```

Add Custom Method addAlertView

```
-(void)addAlertView{
    UIAlertView *alertView = [[UIAlertView alloc]initWithTitle:
    @"Title" message:@"This is a test alert" delegate:self
    cancelButtonTitle:@"Cancel" otherButtonTitles:@"Ok", nil];
    [alertView show];
}
```

Implement Alert View Delegate Method

```
#pragma mark - Alert view delegate
- (void)alertView:(UIAlertView *)alertView clickedButtonAtIndex:
  (NSInteger)buttonIndex{
    switch (buttonIndex) {
        case 0:
            NSLog(@"Cancel button clicked");
            break;
```

```
        case 1:
            NSLog(@"OK button clicked");
            break;

        default:
            break;
    }
}
```

Update viewDidLoad in ViewController.m as follows −

```
(void)viewDidLoad
{
    [super viewDidLoad];
    [self addAlertView];
}
```

Output

When we run the application, we'll get the following output −