

Разработване на приложения за iOS



Петко Пенчев, TapBits.com

“Клавиатура”

- Keyboard Notifications
- UITextField, UITextFieldDelegate
- UITextView, UITextViewDelegate



“Клавиатура”

- Каква “клавиатура” ще се визуализира?

UITextField прилага `UITextInputTraits` protocol, чийто свойства са

```
@property UITextAutocapitalizationType autocapitalizationType; // words, sentences, etc.
```

```
@property UITextAutocorrectionType autocorrectionType; // UITextAutocorrectionTypeYES/NO
```

```
@property UIReturnKeyType returnKeyType; // Go, Search, Google, Done, etc.
```

```
@property BOOL secureTextEntry; // for passwords, for example
```

```
@property UIKeyboardType keyboardType; // ASCII, URL, PhonePad, etc.
```

Пример:

```
myUITextField.autocapitalizationType = UITextAutocapitalizationTypeNone;
```

“Клавиатура”

- Keyboard Notifications

UIKeyboardWillShowNotification
UIKeyboardDidShowNotification
UIKeyboardWillHideNotification
UIKeyboardDidHideNotification

```
[[NSNotificationCenter defaultCenter] addObserver:self  
                                     selector:@selector(keyboardDidShow:)  
                                     name:UIKeyboardDidShowNotification  
                                     object:nil];
```

```
[[NSNotificationCenter defaultCenter] addObserver:self  
                                     selector:@selector(keyboardDidHide:)  
                                     name:UIKeyboardDidHideNotification  
                                     object:nil];
```

“Клавиатура”

- Keyboard Notifications Handlers

```
-(void) keyboardDidShow: (NSNotification *)notif
{
    // If keyboard is visible, return
    if (keyboardVisible)
    {
        NSLog(@"Keyboard is already visible.");
        return;
    }

    // Get the size of the keyboard.
    NSDictionary* info = [notif userInfo];
    NSValue* aValue = [info objectForKey:UIKeyboardBoundsUserInfoKey];
    CGSize keyboardSize = [aValue CGRectValue].size;

    // Keyboard is now visible
    keyboardVisible = YES;
}
```

```
-(void) keyboardDidHide: (NSNotification *)notif
{
    // Is the keyboard already shown
    if (!keyboardVisible)
    {
        NSLog(@"Keyboard is already hidden.");
        return;
    }

    // Keyboard is no longer visible
    keyboardVisible = NO;
}
```

Keyboard Notification User Info Keys

```
(NSNotification *)notif
```

```
...
```

```
NSDictionary* info = [notif userInfo];
```

```
NSValue* aValue = [info objectForKey:UIKeyboardBoundsUserInfoKey];
```

```
...
```

```
NSString *const UIKeyboardFrameBeginUserInfoKey;
```

```
NSString *const UIKeyboardFrameEndUserInfoKey;
```

```
NSString *const UIKeyboardAnimationDurationUserInfoKey;
```

```
NSString *const UIKeyboardAnimationCurveUserInfoKey;
```

Декларирани в UIWindow.h

UITextField

- Подобен на **UILabel**, но с възможност да бъде редактиран.

Можем да установим `textcolor`, `alignment`, `font` и др. по подобие на `UILabel`

- Клавиатурата се появява когато **UITextField** става “first responder”

програмно можем изпратим `becomeFirstResponder` за да покажем клавиатурата или да я скрием с `resignFirstResponder`

UITextField

- Основните свойства

```
@property(n nonatomic, copy) NSString *text
@property(n nonatomic) NSTextAlignment textAlignment
@property(n nonatomic, retain) UIColor *textColor
@property(n nonatomic, retain) UIFont *font
@property(n nonatomic, readonly, getter=isEditing) BOOL editing
@property(n nonatomic) BOOL clearsOnBeginEditing
@property(n nonatomic) UITextBorderStyle borderStyle
@property(n nonatomic, retain) UIImage *background
@property(n nonatomic) BOOL adjustsFontSizeToFitWidth
@property(n nonatomic) CGFloat minimumFontSize
```

- Overlays

```
@property(n nonatomic, retain) UIView *leftView
@property(n nonatomic) UITextFieldViewMode leftViewMode
@property(n nonatomic, retain) UIView *rightView
@property(n nonatomic) UITextFieldViewMode rightViewMode
```

UITextField

- UITextFieldDelegate

Следните няколко метода е добре да бъдат “пренаписани” когато използваме UITextField за да си осигурим по-добро прилагане като цяло, но те разбира се не са задължителни.

`-(BOOL)textFieldShouldReturn:(UITextField*)sender;`

Изпраща се при натискане на “return” key, който и да е той

`-(void)textFieldDidEndEditing:(UITextField *)sender;`

Типично това е метода при който извличаме съдържанието на текстовото поле

UITextField

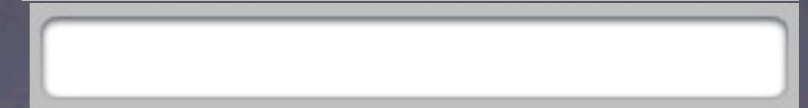
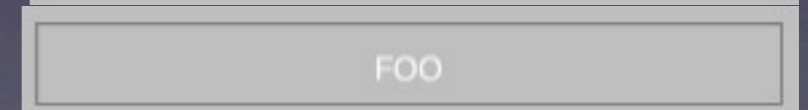
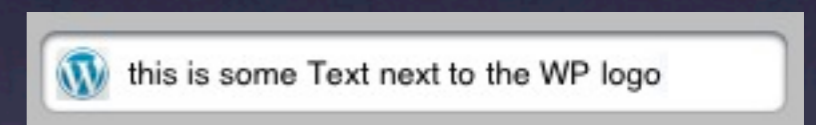
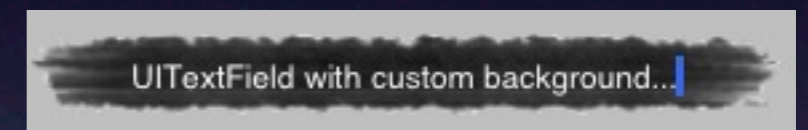
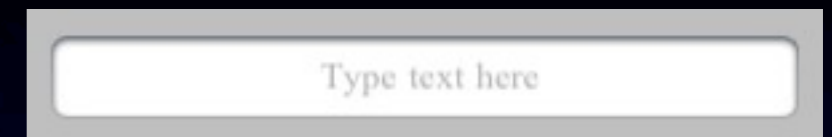
- **Примери:**

```
[myTextField setText:@"This is some text!"];  
[myTextField setPlaceholder:@"Type text here"];  
[myTextField setFont:[UIFont fontWithName:@"Times New Roman" size:14]];  
[myTextField setTextColor:[UIColor blueColor]];  
[myTextField setTextAlignment:UITextAlignmentCenter];
```

```
myTextField.textAlignment = NSTextAlignmentCenter;  
myTextField.textColor = [UIColor whiteColor];  
myTextField.borderStyle = UITextBorderStyleNone;  
myTextField.background = [UIImage imageNamed:@"bg.png"];
```

```
UIImageView * myView = [[ UIImageView alloc ] initWithImage :  
    [UIImage imageNamed : @"wordpress.png" ]];  
[myTextField setLeftView :myView];  
[ myTextField setLeftViewMode: UITextFieldViewModeAlways];  
[myView release ];
```

```
[myTextField setBorderStyle:UITextBorderStyleNone];  
[myTextField setBorderStyle:UITextBorderStyleLine];  
[myTextField setBorderStyle:UITextBorderStyleBezel];  
[myTextField setBorderStyle:UITextBorderStyleRoundedRect];
```



UITextField

- Примери:

```
@interface test_appViewController : UIViewController <UITextFieldDelegate>
...

UITextField *textfield;
...

@implementation

textfield = [[UITextField alloc] initWithFrame:CGRectMake(20, 240, 280, 30)];
textfield.placeholder = @"Textfield 1";
textfield.delegate = self;
textfield.returnKeyType = UIReturnKeyDone;
textfield.borderStyle = UITextBorderStyleBezel;
textfield.enablesReturnKeyAutomatically = TRUE;
[self.view addSubview:textfield];
...

```

UITextField

- Примери:

```
- (BOOL)textFieldShouldReturn:(UITextField *)textField {
    if (textField == MyTextField)
    {
        [MyTextField resignFirstResponder];
    }
    MyLabel.text = MyTextField.text;

    return YES;
}
```

```
- (void)textFieldDidEndEditing:(UITextField *)textField {

    double value = 0.0;
    NSString* text = textField.text;

    if ([text length] > 0) {
        value = [text doubleValue];
    }
    ...
}
```

UITextView

- UITextView
 - много редово текстово поле
 - поддържа скролиране
 - може да се установи шрифт и цвят за цялото поле
 - не поддържа форматиране на ниво СИМВОЛ

UITextView

- UITextView няколко специфични метода

Метод за скролиране в текста

- (void)scrollRangeToVisible:(NSRange)range

- (BOOL)hasText;

Наш изглед за виртуалната клавиатура

@property (readwrite, retain) UIView *inputView

UITextView

- UITextViewDelegate

- (void)textViewDidBeginEditing:(UITextView *)textView
- (BOOL)textViewShouldBeginEditing:(UITextView *)textView

- (void)textViewDidEndEditing:(UITextView *)textView
- (BOOL)textViewShouldEndEditing:(UITextView *)textView

- (BOOL)textView:(UITextView *)textView shouldChangeTextInRange:(NSRange)range replacementText:(NSString *)text

- (void)textViewDidChange:(UITextView *)textView

- (void)textViewDidChangeSelection:(UITextView *)textView

UITextView

- Пример

```
CGRect tfFrame = CGRectMake(10, 10, 300, 100);
messageTextView = [[UITextView alloc] initWithFrame:tfFrame];

// make sure that it is editable
messageTextView.editable = YES;

// add the controller as the delegate
messageTextView.delegate = self;
...

- (BOOL)textView:(UITextView *)textView shouldChangeTextInRange:(NSRange)range
replacementText:(NSString *)text
{
    // Any new character added is passed in as the "text" parameter
    if ([text isEqualToString:@"\n"]) {
        // Be sure to test for equality using the "isEqualToString" message
        [textView resignFirstResponder];

        // Return FALSE so that the final '\n' character doesn't get added
        return FALSE;
    }
    // For any other character return TRUE so that the text gets added to the view
    return TRUE;
}
```