



Василий Федоров,
Senior iOS Developer,
<http://napoleonit.ru>

Среда разработки и язык программирования для iOS

2014, Челябинск



Необходимое оборудование



28.10.14
© В.Федоров,
<http://napoleonit.ru>



Необходимое оборудование

Необходим один из следующих вариантов:

1. Компьютер под управлением Mac OS X:
 - iMac
 - MacBook
 - Mac Mini

2. Эмулятор
 - Virtual Machine (VMWare, Virtual Box)
 - Hackintosh

3. Веб-платформа
 - <http://MacInCloud.com>
 - <http://VirtualMacOSX.com>



Основы языка Objective-C



Основы языка Objective-C

- На базе C
- Объектно-ориентированный
- Всё - объекты
- Посылка сообщений объектам, а не вызовы функций:
 - [receiver message]
 - [receiver message:argument]
 - [receiver message:argument1 :argument2]
 - [receiver messagePart1:argument1 messagePart2:argument2]
 - [receiver message:argument1, argument2, argument3, nil]

```
- (int)changeColorToRed:(float)red green:(float)green blue:(float)blue;  
[myColor changeColorToRed:5.0 green:2.0 blue:6.0];
```



Основы языка Objective-C

□ Создание класса

```
@interface NameClass : NSObject {  
    NSString *name;  
}  
  
@property (strong) NSString * secondName;  
  
+(id) newName;  
  
-(NSString*) fullName;  
  
@end
```



Основы языка Objective-C

□ Реализация методов класса

```
#import "NameClass.h"

@implementation NameClass

+(id)newName {
    NameClass *instance = [[NameClass alloc] init];
    return instance;
}

-(NSString *)fullName {
    return [name stringByAppendingString:self.secondName];
}

@end
```



Основы языка Objective-C

□ Протоколы

```
@protocol NameProtocol <NSObject>  
  
@end
```

```
@interface NameClass : NSObject <NameProtocol> {  
    NSString *name;  
}
```



ОСНОВЫ ЯЗЫКА Objective-C

□ Категории

```
#import "NameClass.h"  
  
@interface NameClass (shorter)  
-(NSString*) shortName;  
  
@end
```



ОСНОВЫ ЯЗЫКА Objective-C

□ Блоки

```
void(^someBlock)(void) = ^(void) {  
    //implementation;  
};  
  
someBlock();  
  
/////////  
  
NSString* (^otherBlock)(BOOL) = ^(BOOL flag) {  
    if (flag) {  
        //implementation;  
    } else {  
        //implementation;  
    }  
    return @"";  
};  
|  
otherBlock(YES);
```

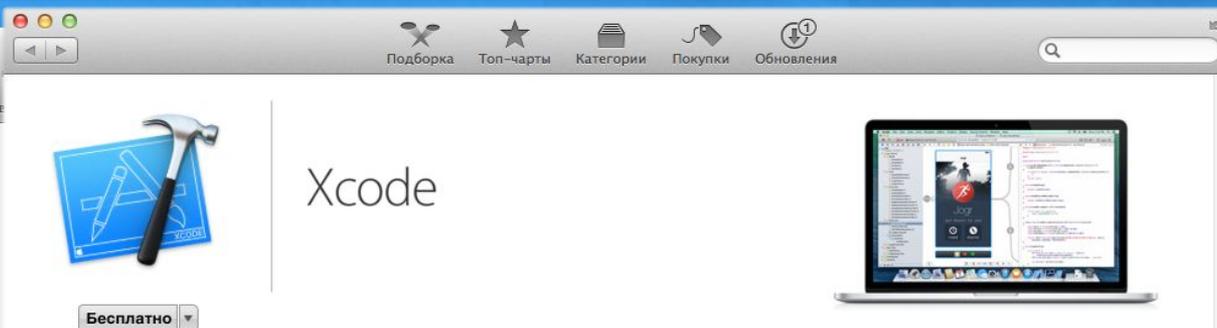


Приступаем к разработке

28.10.14

© В.Федоров,
<http://napoleonit.ru>

Качаем и устанавливаем Xcode



Xcode 4+

Самое необходимое

Xcode provides everything developers need to create great applications for Mac, iPhone, and iPad. Xcode brings user interface design, coding, testing, and debugging all into a unified workflow. The Xcode IDE combined with the Cocoa and Cocoa Touch frameworks, and the Swift programming language make developing apps easier and more fun than ever before....

...Еще

Что нового в версии 6.1

Includes SDKs for OS X 10.10 Yosemite, OS X 10.9 Mavericks, and iOS 8.1.

...

...Еще

Веб-сайт Apple >

«Xcode» – поддержка >

Лицензионное соглашение >

Политика конфиденциальности >

Информация

Категория: Инструменты разработчика

Обновлено: 20.10.2014

Версия: 6.1

Цена: Бесплатно

Размер: 2.49 Гб

Семейный доступ: Да

Язык: английский

Разработчик: iTunes S.a.r.l.

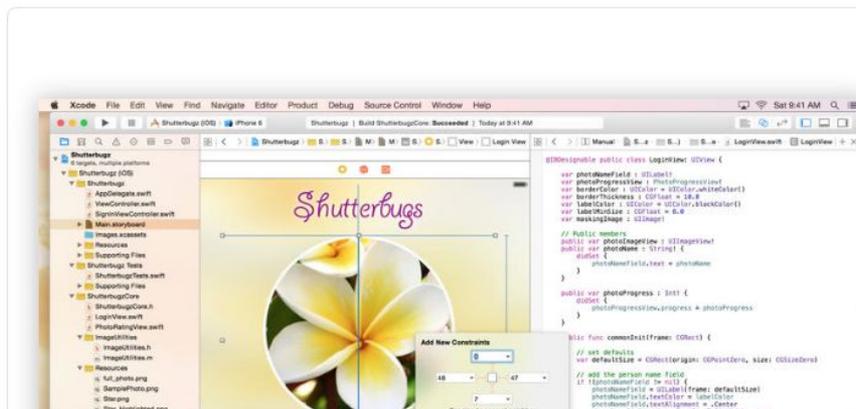
© 1999–2013 Apple Inc.

Возраст 4+

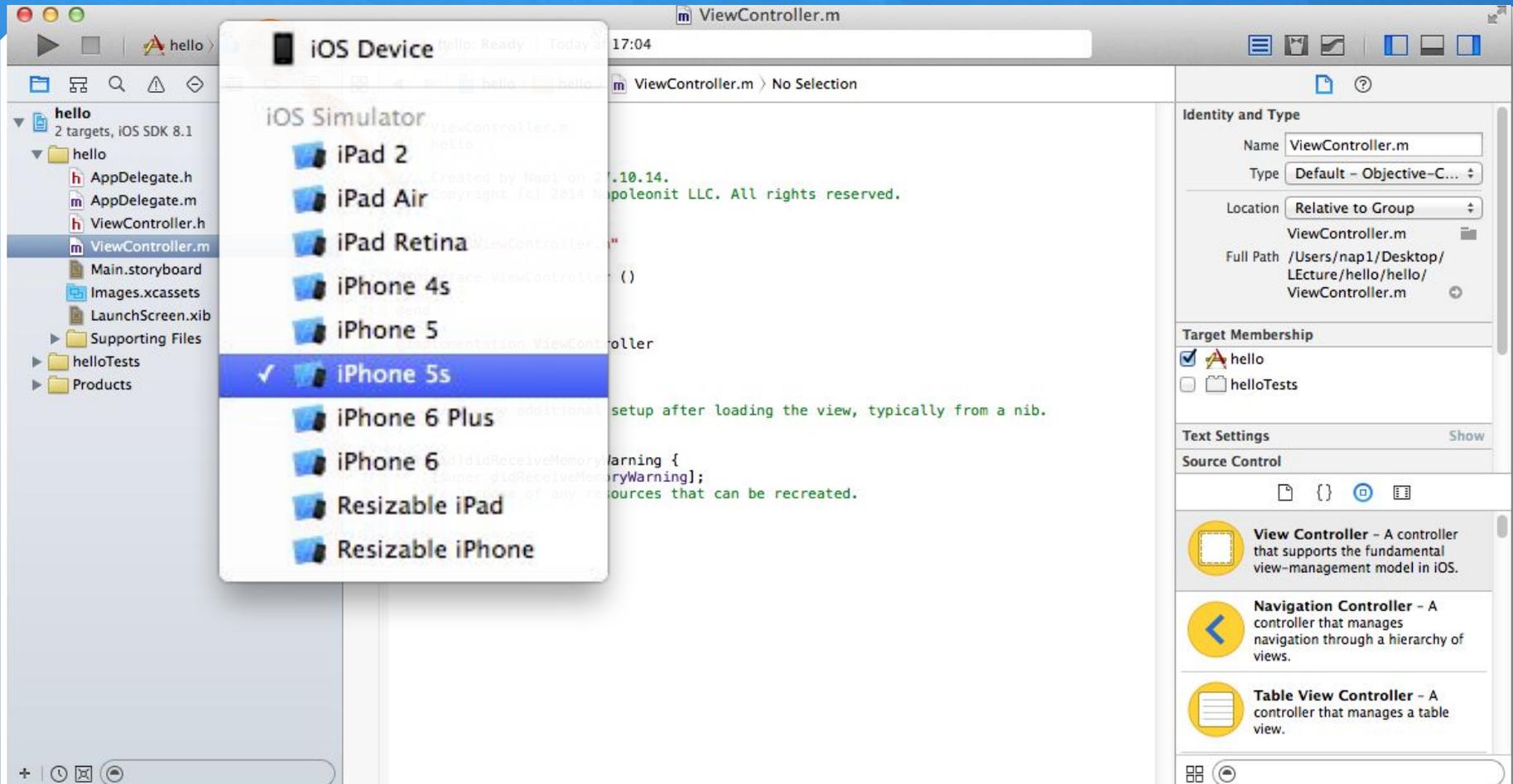
Совместимость:
OS X 10.9.4 или более поздняя версия

Apple: другие материалы

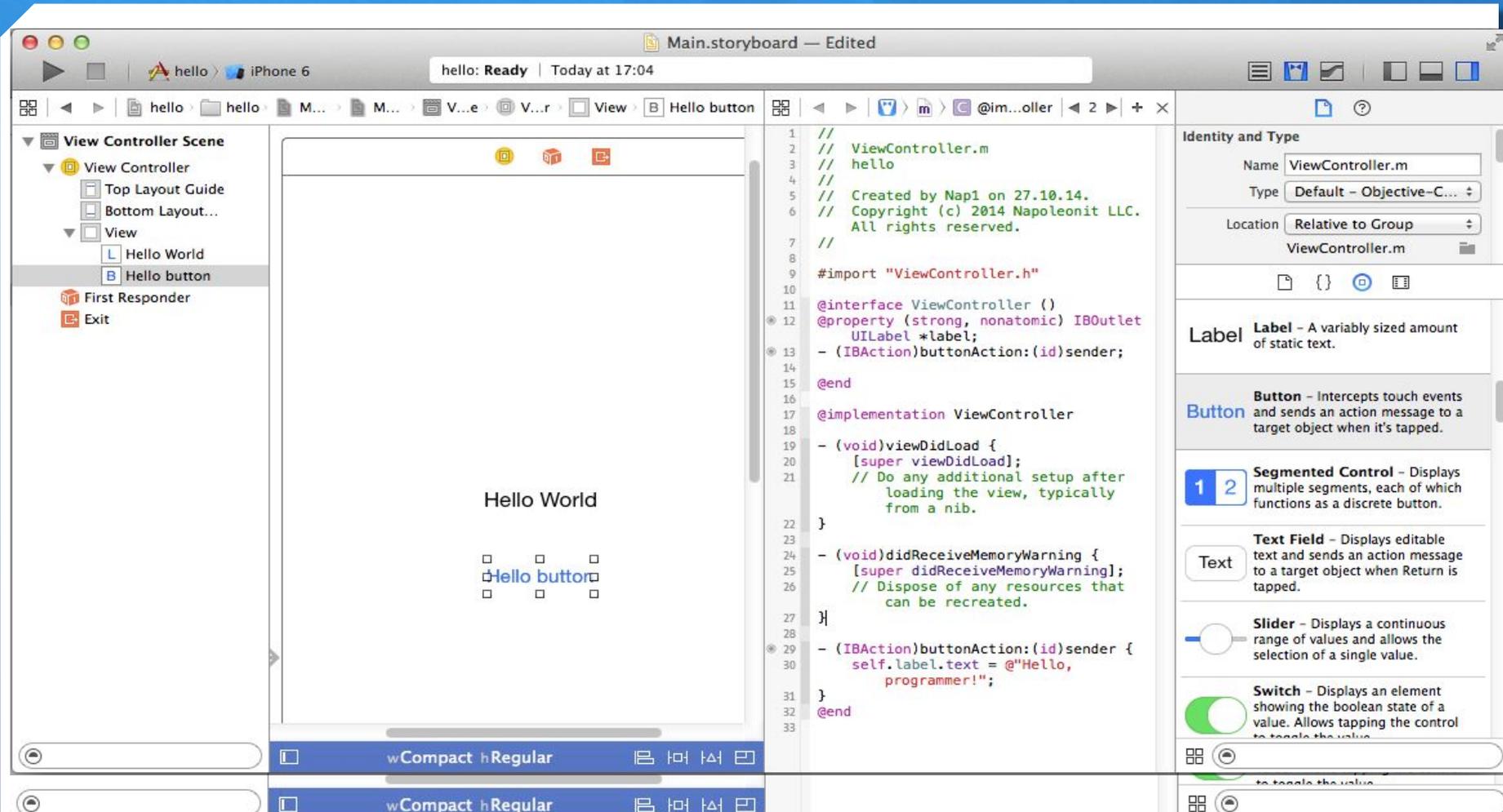
OS X Yosemite



Главное окно Xcode



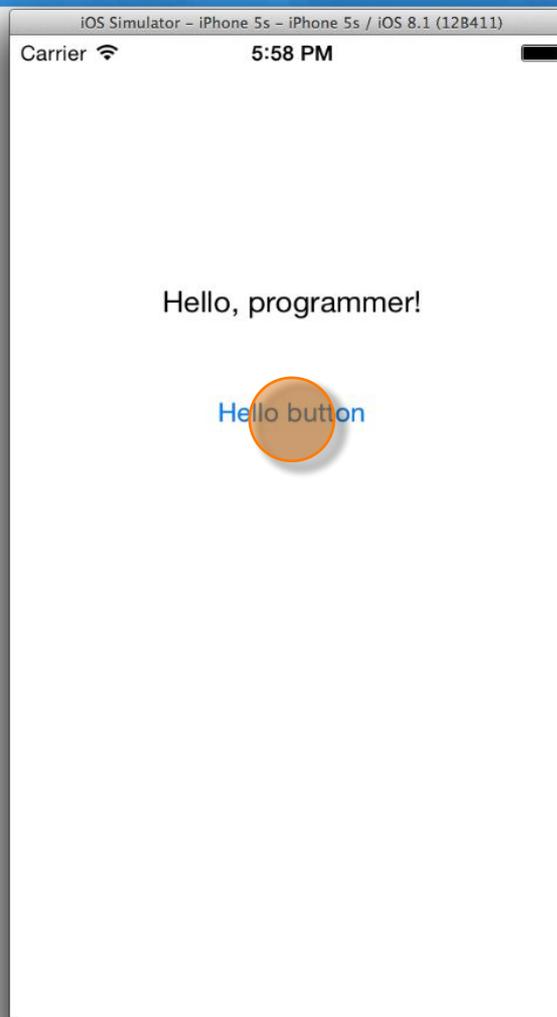
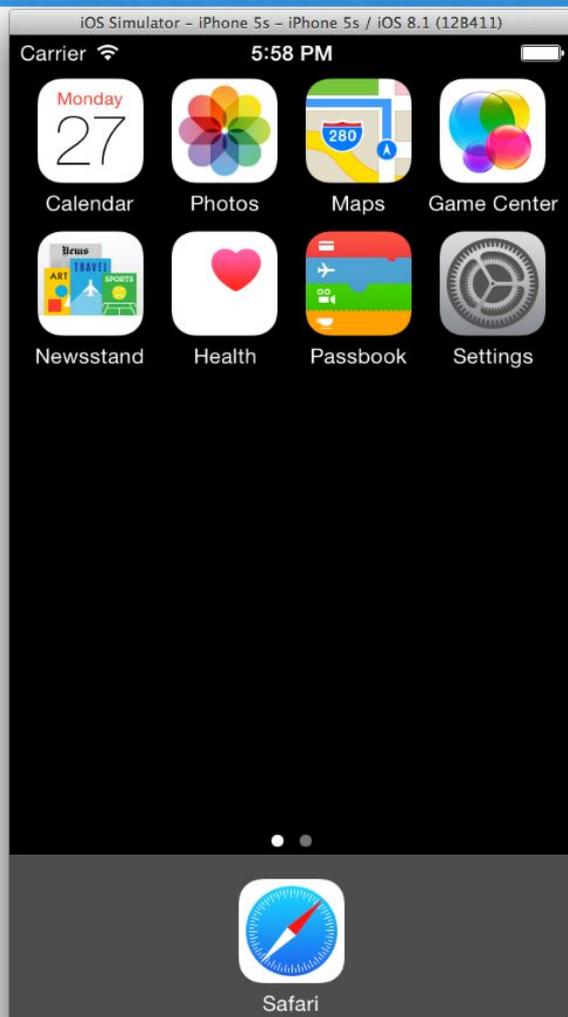
Storyboard



The screenshot displays the Xcode development environment. On the left, the 'View Controller Scene' tree shows a hierarchy: View Controller > View > Hello World > Hello button. The main canvas shows a storyboard with a 'Hello World' label and a 'Hello button' below it. The center pane shows the Objective-C code for the ViewController.m file, including imports, interface definitions, and implementation methods like viewDidLoad and buttonAction. The right pane shows the 'Identity and Type' inspector for the selected 'Hello button', with fields for Name, Type, and Location. Below this, there are descriptions for 'Label', 'Button', 'Segmented Control', 'Text Field', 'Slider', and 'Switch' UI elements.

```
1 //
2 // ViewController.m
3 // hello
4 //
5 // Created by Nap1 on 27.10.14.
6 // Copyright (c) 2014 Napoleonit LLC.
7 // All rights reserved.
8 //
9
10 #import "ViewController.h"
11
12 @interface ViewController ()
13 @property (strong, nonatomic) IBOutlet
14     UILabel *label;
15 - (IBAction)buttonAction:(id)sender;
16 @end
17
18 @implementation ViewController
19 - (void)viewDidLoad {
20     [super viewDidLoad];
21     // Do any additional setup after
22     // loading the view, typically
23     // from a nib.
24 }
25 - (void)didReceiveMemoryWarning {
26     [super didReceiveMemoryWarning];
27     // Dispose of any resources that
28     // can be recreated.
29 }
30 - (IBAction)buttonAction:(id)sender {
31     self.label.text = @"Hello,
32     programmer!";
33 }
34 @end
```

iOS Simulator





Источники информации

- Дейв Марк, “Разработка приложений для iPhone, ...”
- Matt Galloway, “Effective Objective-C 2.0”
- <http://www.raywenderlich.com/>
- <http://stackoverflow.com/>



Вопросы

28.10.14

© В.Федоров,
<http://napoleonit.ru>

Василий Федоров,
Senior iOS Developer,
<http://napoleonit.ru>

+7-905-833-74-12

9058337412@list.ru