Self Learning Exercises

Week 1. Objective-C.

ANSWERS

1. In the following sentence:

```
[String alloc];
```

we are invoking a class method or an instance method?

We are invoking a class method. Do note that the class methods have the follwing syntax: `[ ClassName message ];`

where ClassName is the name of the class, NSString in this case.

---

2. In the following sentence `NSString * name = [[NSString alloc] init ];` ¿we are invoking class methods or instance methods?

We are invoking a class method and then an instance method. First we are invoking:

```
   [NSString alloc] ;
```

which returns an object of id type. To this object we send the instance method init, the first line is equivalent to:

```
    id i = [NSString alloc];
NSString *nombre = [i init];
```

Do note thtat class methods have the follwoing syntax:

```
[ ClassName message ];
```

where ClassName is the name of the class, here NSString.

In the second line we call an instance method. Instance methods have the follwing syntax: `[object message ];` where *object* is a variable that points to an object who receives a *message*.

---

3. Could you write the following sentence `[[NSString alloc] init] ;` in two lines?

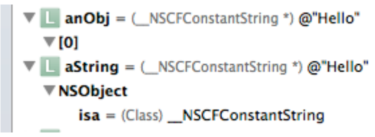
```
 id i = [NSString alloc];
NSString *nombre = [i init];
```

---

4. What does mean the *id* type?

id is a special keyword used in Objective-C to mean "some kind of object." It does not contain isa pointer (isa, gives the object access to its class and, through the class, to all the classes it inherits from), So you lose compile-time information about the object. Example:

```
NSString* aString = @"Hello";
id anObj = aString;
```



```
▼ L anObj = (__NSCFConstantString *) @"Hello"
  ▼ [0]
▼ L aString = (__NSCFConstantString *) @"Hello"
  ▼ NSObject
      isa = (Class) __NSCFConstantString
```

---

5. Can I use self in a class method?

Yes. Inside class methods, *self* refers to the object representing the corresponding Class: For example:

```
+ (id)create {
  return [[self alloc] init];
}
```

is the same as:

```
+ (id)create {
  return [[SomeClass alloc] init];
}
```

6. What is a protocol?

In a protocol we declare methods and properties that are not related with a specific class.

https://developer.apple.com/library/ios/documentation/Cocoa/Conceptual/ProgrammingWithObjectiveC/WorkingwithProtocols/WorkingwithProtocols.htn

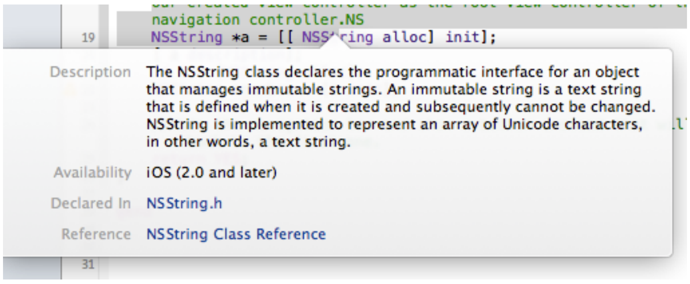
7. What is the syntax to indicate that a class implements a protocol?

In order to indicate that MyClass conforms the protocol myProtocol we will write:

```
@interface MyClass : NSObject <MyProtocol>
…
@end
```

8. How can I access the documentations of a class in Xcode?

(Just place the cursos over the class and press ⌥ + click)



```
navigation controller.NS
NSString *a = [[ NSS·ring alloc] init];
```

Description The NSString class declares the programmatic interface for an object that manages immutable strings. An immutable string is a text string that is defined when it is created and subsequently cannot be changed. NSString is implemented to represent an array of Unicode characters, in other words, a text string.

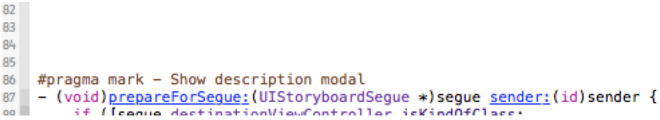
Availability iOS (2.0 and later)

Declared In NSString.h

Reference NSString Class Reference

9. How can I to highlight the signature of a method?

Just place the cursor over the method and press ⌘



```
#pragma mark - Show description modal
- (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender {
    if ([segue destinationViewController isKindOfClass:
```

10. NSUInteger is an Objective-C class or a primitive of C? ¿How I know it?

Is a C structure, access the documentations and note that NSUInteger is a typedef

11. NSString is an Objective-C or a primitve type of C?

NSString is an object of Objective-C, access the documentation and note that is an object.

12.What is a delegate? How can indicate that a class is a delegate?

A delegate is an object that acts on behalf of, or in coordination with, another object when that object encounters an event in a program. The delegating object is often a responder object—that is, an object inheriting from NSResponder in AppKit or UIResponder in UIKit—that is responding to a user event. The delegate is an object that is delegated control of the user interface for that event, or is at least asked to interpret the event in an application-specific manner.

https://developer.apple.com/library/ios/documentation/general/conceptual/CocoaEncyclopedia/DelegatesandDataSources/DelegatesandDataSources.ht

13. For the followoing variables:

```
 int i = 1;
  NSUInteger count = 2;
NSNumber number = @1;
```

Write code to find out the address memory where these variables are stored.

```
NSLog(@"value:%p", &i);
NSLog(@"count:%p", &count);
NSLog(@"myString:%p", myString);
NSLog(@"myArr:%p", &number);
```

We get:

```
value:   0xbfffc968                                                                                    count:  0xbfffc964
myString:  0x7e14                                                                                     myArr:  0xbfffc95c
```

Results will be different in different computers.

---

14. ¿What does mean the keyword static on a local variable?

When you use the static modifier on a local variable, the function “remembers” its value across invocations. Ejemplo:

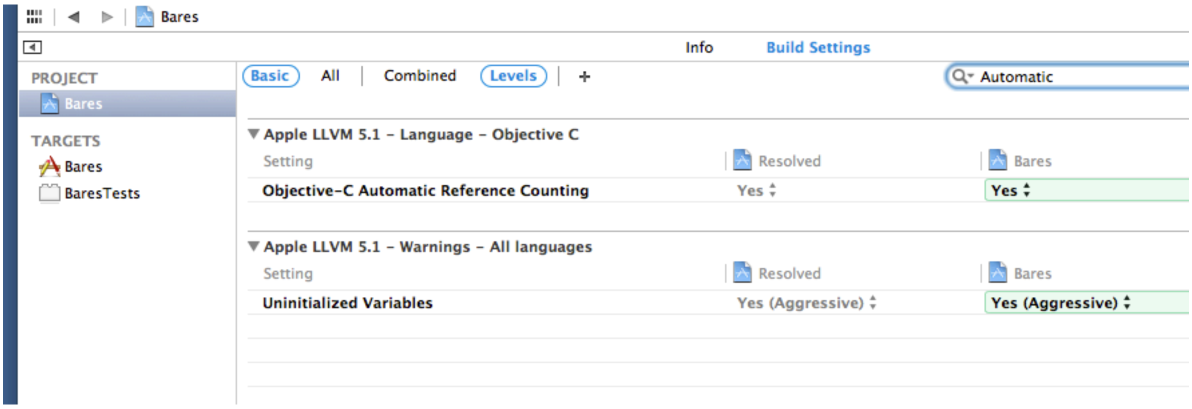
```
// main.m
#import <Foundation/Foundation.h>
int countByTwo() {
 static int currentCount = 0;
currentCount += 2; return currentCount;
}
 int main(int argc, const char * argv[]) {
 @autoreleasepool {
   NSLog(@"%d", countByTwo()); // 2
   NSLog(@"%d", countByTwo()); // 4
   NSLog(@"%d", countByTwo()); // 6
    return 0;
    ]
 }
```

---

15. iOS8 uses by default Manual Reference Counting or ARC (Automatic Reference Counting)? How can I force Xcode not to use ARC?

By default XCode I uses ARC. In order to use Manual Reference Counting, do the following:

In Xcode 5.0.2, select your project in Navigatior, select Build Settings, search for Apple LLVM 5.0 - Language - Objective C and change Objective-C Automatic Refence Counting to NO



▼ Apple LLVM 5.1 – Language – Objective C

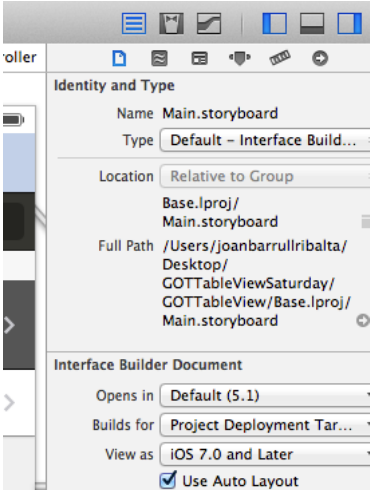
Setting	Resolved	Bares
Objective-C Automatic Reference Counting	Yes	Yes

▼ Apple LLVM 5.1 – Warnings – All languages

Setting	Resolved	Bares
Uninitialized Variables	Yes (Aggressive)	Yes (Aggressive)

16. XCode uses Autolayout? How can If orce Xcode not to use AutoLayout?

Also from Interface Builder selecting Main.storyboardwe can enable o disable Autolayout with a checkbox:



oller

Identity and Type

Name Main.storyboard

Type Default – Interface Build...

Location Relative to Group

Base.lproj/
Main.storyboard

Full Path /Users/joanbarrullribalta/
Desktop/
GOTTableViewSaturday/
GOTTableView/Base.lproj/
Main.storyboard

Interface Builder Document

Opens in Default (5.1)

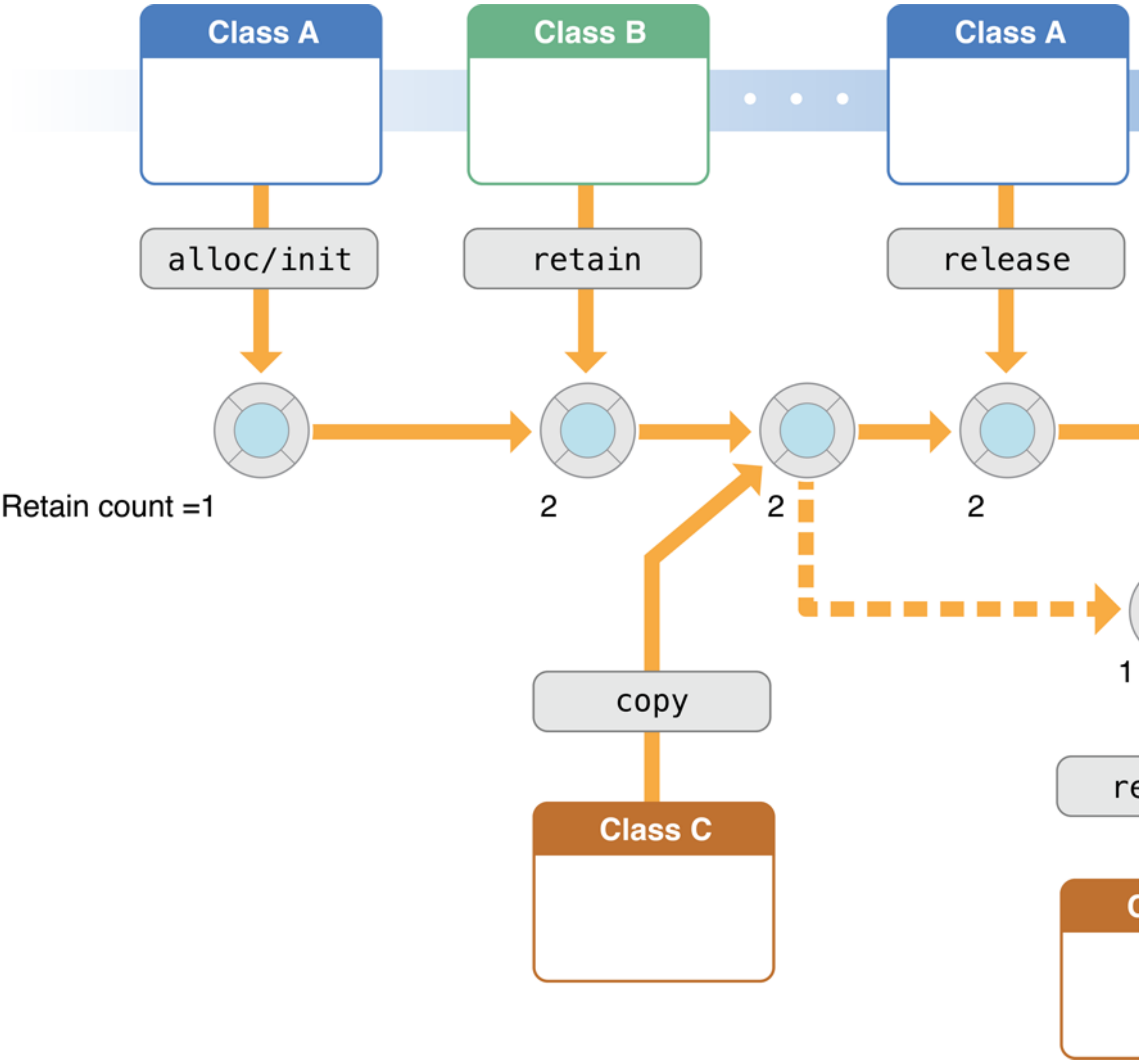
Builds for Project Deployment Tar...

View as iOS 7.0 and Later

☑ Use Auto Layout

---

17. After learning Manual Memory Management, how do you interpret the following drawing?



Answer: From left to right, First, an instance of Class A creates a new object - it has a retain count of one. In code: `@interface A { -instanceMethod { X x = [[ X alloc] init ]; // after creation object X has a retain count of 1 } }` An instance of Class B retains the new object - it now has a retain count of 2. An instance of class C makes a copy of the new object - this is another new object, with a retain count of one. The instance of Class A releases the new object - it now has a retain count of 1, so is not deallocated. Class C releases its copy - the copied instance is deallocated Class B releases its reference to the new object. The reference count returns to zero and it is deallocated.

18. What is the result of the following sentence?

```
[nil isEqual: nil];      // returns NO!
```

Objective-C allows to send any message to nil, and it will ever return 0 (or false).Objective-C allows sending any mesage to nil, and will always return 0, even in cases likes this where it makes no sense.

19. What is the result of the following code?

```
BOOL result = [[NSMutableArray array] isMemberOfClass:[NSArray class]];
NSLog (@"%@", result? @"Yes!" : @"No!");   // returns No!
```

Mehtod isMeemberOfClass: only will return TRUE if the instance is exactly of class [NSArray class]. But in this case instance pertains to class NSMutableArray, different of NSArray so it returns NO.

20. What is the result of the following code?

```
BOOL result = [[NSMutableArray array] isKindOfClass:[NSArray class]];
NSLog (@"%@", result? @"Yes!" : @"No!");   // returns YES!
```

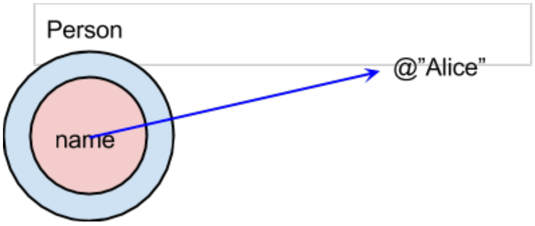
Method isKindOfClass: only returns TRUE if the instance is exactly of class [NSArray class] or a subclass. In this case, instance pertains to class NSMutableArray, a subclass de NSArray, so it returns YES.

Self Learning Exercises. Week 2 (View)

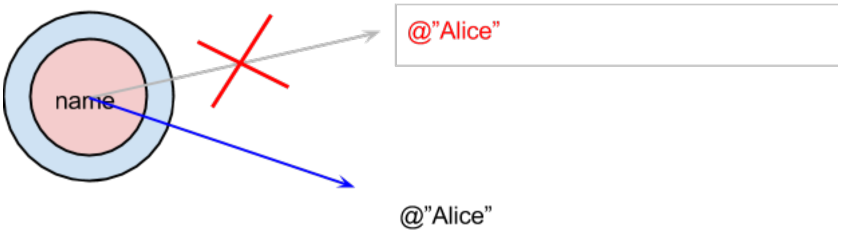
1. To modelize a text string we have NSString classes and their mutable version NSMutableString. In the following example we modify the value of a NSString varibale. How is it possible? Is is it not NSString immutable?

```
#import <Foundation/Foundation.h>
 @interface Person : NSObject
@property (nonatomic,strong)
NSString *name; // strong should be copy
@end
@implementation Person
@end
 int main(int argc, char *argv[]) {
   @autoreleasepool {
      Person *p = [Person new];
      NSMutableString *name = [[NSMutableString alloc] initWithString:@"Alice"];
      p.name = name;
      NSLog(@"%@",p.name); // prints Alice
      [name stringByAppendString:@"xxx"];
      NSLog(@"%@",p.name); // prints Alicexxx }
}
```

Answer. NSString is immutable. That is, the content pointed by the pointer (@"Alice" in our case) can not change.



But it is possible force the pointer in order to point another address, so the content pointed can be different. @"Alice" continues to be immutable, but now is no more referenced by anyone.



2. Following the prior example, how can we define the property in order to not modify the value?

```
#import <Foundation/Foundation.h>
 @interface Person : NSObject
@property (nonatomic,copy) NSString *name; // strong should be copy
@end
@implementation Person
@end
 int main(int argc, char *argv[]) {
   @autoreleasepool {
      Person *p = [Person new];
      NSMutableString *name = [[NSMutableString alloc] initWithString:@"Alice"];
      p.name = name;
      NSLog(@"%@",p.name); // prints Alice
      [name stringByAppendString:@"xxx"];
      NSLog(@"%@",p.name); // prints Alice }
}
```

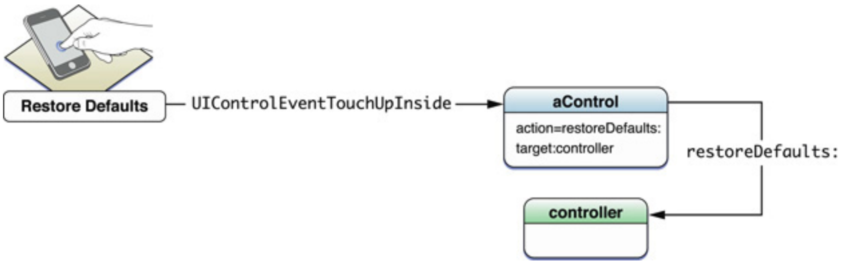
Do note that when we execute the sentence: p.name = name;

we execute the method set with copy:

```
-(NSString *) setName: (NSString * string) {
        return string.copy
}
```

3. Could you explain the target-action pattern?

Cocoa uses the target-action mechanism for communication between a control and another object. This mechanism allows the control and, in OS X its cell or cells, to encapsulate the information necessary to send an application-specific instruction to the appropriate object. The receiving object—typically an instance of a custom class—is called the target. The action is the message that the control sends to the target. The object that is interested in the user event—the target—is the one that imparts significance to it, and this significance is usually reflected in the name it gives to the action.



https://developer.apple.com/library/ios/documentation/general/conceptual/Devpedia-CocoaApp/TargetAction.html

4. Could you present an code example of the pattern target'action?

Here we create an UIButton, and then we set a method that will be invoked when we touch it.

```
{
    UIButton *button = [UIButton buttonWithType:UIButtonTypeSystem];
    button.frame = CGRectMake(10, 10, 100, 50);
    [self.view addSubview:button];

//   1.Set button's title for normal and highlighted state
    [button setTitle:@"Normal" forState:UIControlStateNormal];
    [button setTitle:@"Highlighted" forState:UIControlStateHighlighted];

//   2.When you press it, it should log "You pressed me!" or something similar
    [button addTarget:self
               action:@selector(buttonPressed)
     forControlEvents:UIControlEventTouchDown];
}
- (void)buttonPressed
{
    NSLog(@"button pressed");
}
```

5. How would you define the MVC Pattern?

The model-view-controller pattern proposes three main components or objects to be used:

- A Model , which represents the underlying, logical structure of data in a software application and the high-level class associated with it. This object model does not contain any information about the user interface.
- A View , which is a collection of classes representing the elements in the user interface (all of the things the user can see and respond to on the screen, such as buttons, display boxes, and so forth)
- A Controller , which represents the classes connecting the model and the view, and is used to communicate between classes in the model and view.

6. In InterfaceBuilder how can I find out the distance in points from an element to their superview?

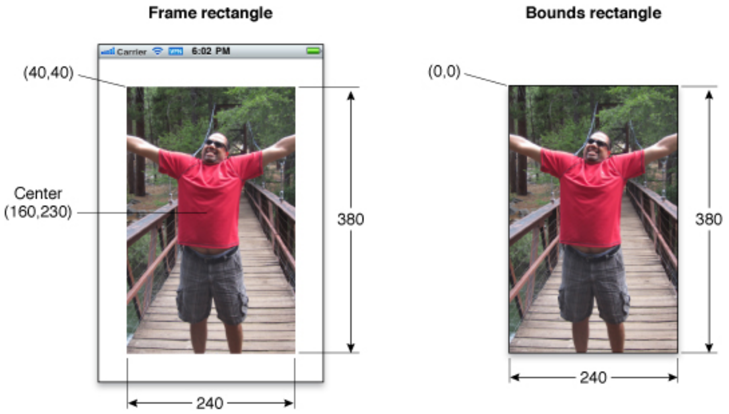
We select th eelement and press key ⌥ (option key), Interface Builder will show us 4 distances to top, bottom, left an right.

7. How can we add a view to the main view?

We add a view invoking addSubView method Añadiremos la vista invocando el método addSubView to the view of the ViewController (self.view). For instance: `self.switchLabel = [[UILabel alloc] initWithFrame:CGRectMake(200, 200, 100, 50)]; self.switchLabel.text = @"Hola caracola"; [self.view addSubview:self.switchLabel];`

8. Could you explain the difference betweem frame and bounds of a view?

frame represents where is the view from the superview. bounds represents where in the view we can paint (from the view itself) For instance, in the follwoing sentence: `[[UIimageView alloc] initWithFrame:CGRectMake(40, 40, 240, 380)];` we make an UIView on the coordinates 40,40 (referenced from the superview:



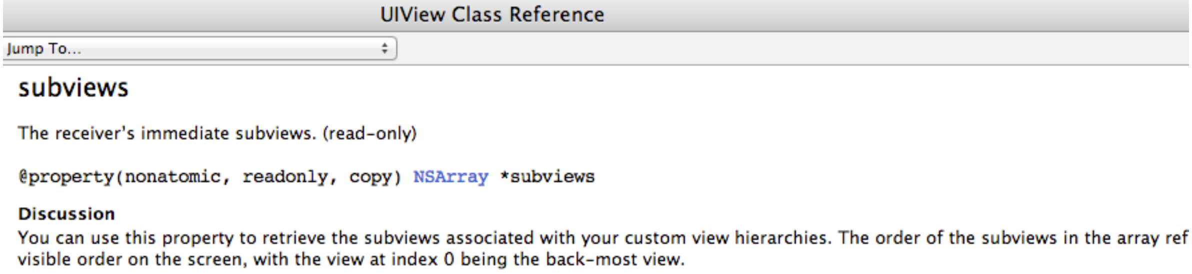
Instead, the bounds property of the view is: CGRect(0,0,240,380).

9. For UIView what is the difference between initilization methods initWithFrame: and initWithCoder: ?

We will use initWithFrame: when creating a view from code. We will use initWithCoder: when we want to load a NIB file created through Interface Builder.

10. How to access the subviews property of a UIView?

Through the property subviews that contains an NSArray with all subviews.



UIView Class Reference

subviews

The receiver's immediate subviews. (read-only)

```
@property(nonatomic, readonly, copy) NSArray *subviews
```

Discussion

You can use this property to retrieve the subviews associated with your custom view hierarchies. The order of the subviews in the array ref visible order on the screen, with the view at index 0 being the back-most view.

11. In the following example we have an Employee class with a property age that contains the employee age. We can get the age this way:

```
NSString age = employee.age;
```

or to modify it: `employee.age = @”20”;`

---

12. How can we access tot the age property through KVC?

Through KVC we can access to the age with: `[employee valueForKey:@"age"];`

---

13. How can we, through KVO, make sure the inspector get notifications of change for the property openingBalance and specify that the old and new value being get by the observer method

```
[account addObserver: inpector
    forKeyPath: @”openingBlance”
    options: (NSKeyValueObservingOptionNew |
            NSKEyValueObservingOptionOld)
    context: NULL ];
```

---

14. Can I put an element Popover or an SplitViewController on an iPhone

Only in iOS8. In prior iOs versions Popover and SplitViewController only can be used on an iPad, not an iPhone.

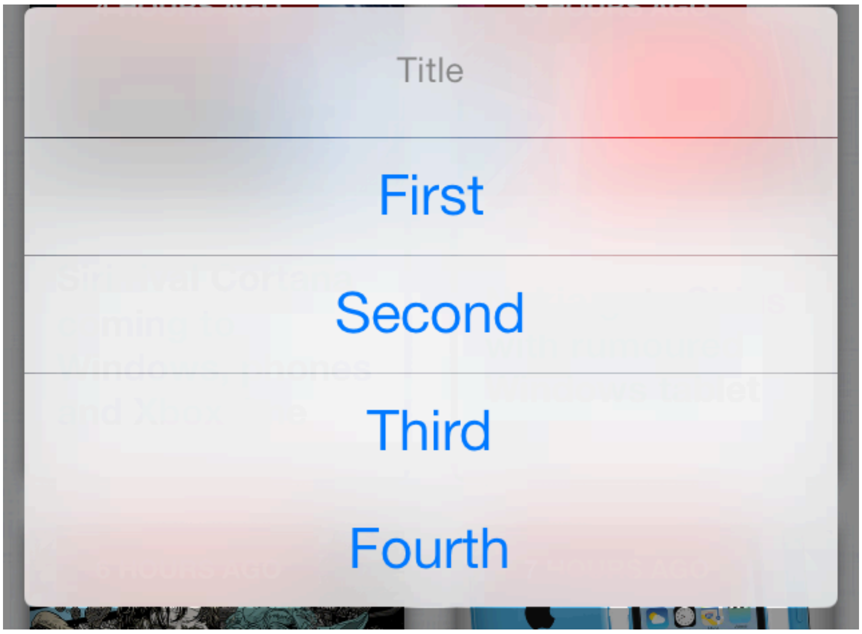
---

15. How can I instantiate and present a popover on an iPad?.

```
MyViewController* content = [[MyViewController alloc] init];
UIPopoverController* aPopover = [[UIPopoverController alloc]
initWithContentViewController:content];
aPopover.delegate = self; !
self.popoverController = aPopover;!
[self.popoverController presentPopoverFromBarButtonItem:sender permittedArrowDirections:UIPopoverArrowDirectionAny
animated:YES];
```

---

16. How can you create a n ActionSheet by code?



An ActionSheet is a graphic element that presents to the user a list of choices to execute a job. In order to create we will write in the viewDidLoad method: ``` - (void)viewDidLoad { [super viewDidLoad];

UIActionSheet *actionSheet = [[UIActionSheet alloc] initWithTitle:@"Delegate Example" delegate:self // telling this class to implement UIActionSheetDelegate cancelButtonTitle:@"Cancel" destructiveButtonTitle:@"Destructive Button" otherButtonTitles:@"Other Button",nil

[actionSheet showInView:self.view];

[actionSheet release]; } ```

We simply ask the controller to implement the protocol UIActionSheet in order to deal with user clicks. We will implement this method:

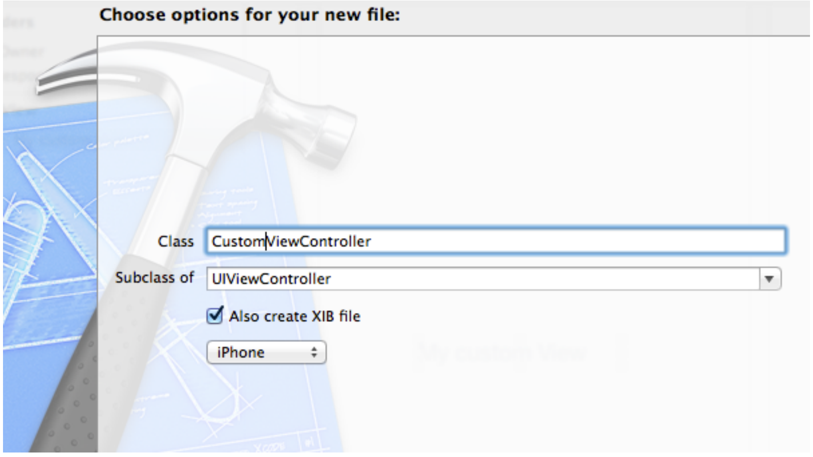
```
- (void)actionSheet:(UIActionSheet *)actionSheet clickedButtonAtIndex:(NSInteger)buttonIndex {
  NSLog(@"hello world!");
}
```

Finally, our class need to impelement the protocol UIActionSheetDelegate ,so we go to the .h and add: ``` @interface DelegateExampleViewController : UIViewController {

} ```

---

17. How to create a xib file in our project?



Choose options for your new file:

Class: CustomViewController

Subclass of: UIViewController

☑ Also create XIB file

iPhone

If we are creating a new ViewController, we select the checkbox *Also create XIB file*. This way we will create the file *CustomView.xib*.

---

18. Following the prior example ¿how to load the XIB file in our application?

This cn be do with the following code:

```
CustomView* customView = [NSBundle mainBundle] loadNibNamed:@"CustomView" owner:nil options:nil] lastObject];
[self.view addSubview:customView];
```

19. If we have the follwing Array:

```
        NSArray *cars = @[@"Mercedes-Benz", @"BMW", @"Porsche",
                        @"Opel", @"Volkswagen", @"Audi"]
```

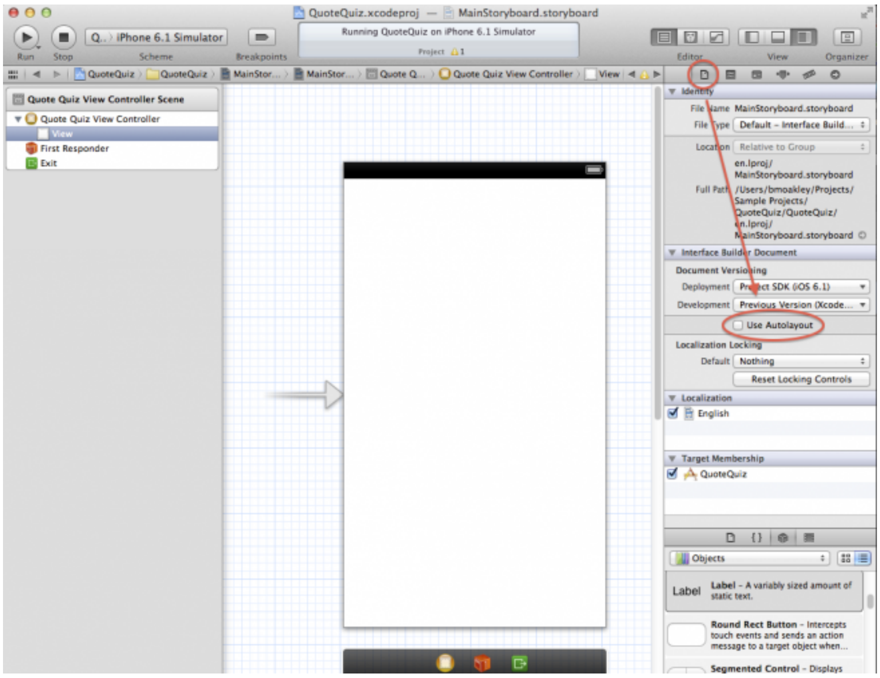
how can access the last elementof the array?

```
cars.lastObject;  // returns @”Audi”
```

20. In the prior example, which is the best way to access to the first element of the Array?

```
cars.firstObject;  // returns @”Mercedes-Benz”
```

21. New projects have AutoLayout selecected by default. Where can check this setting in InterfaceBuilder?


QuoteQuiz.xcodeproj — MainStoryboard.storyboard
Run
Stop
iPhone 6.1 Simulator
Scheme
Breakpoints
Running QuoteQuiz on iPhone 6.1 Simulator
Project
Editor
View
Organizer
QuoteQuiz
MainStor...
Quote Quiz View Controller
View
Quote Quiz View Controller Scene
Quote Quiz View Controller
View
First Responder
Exit
Identity
Interface Builder Document
Localization
Target Membership
Objects
Label
Label – A variably sized amount of static text.
Round Rect Button – Intercepts touch events and sends an action message to a target object when...
Segmented Control – Displays

File Name MainStoryboard.storyboard
File Type Default – Interface Build...
Location Relative to Group
en.lproj/
MainStoryboard.storyboard
Full Path /Users/bmoakley/Projects/
Sample Projects/
QuoteQuiz/QuoteQuiz/
en.lproj/
MainStoryboard.storyboard

Document Versioning
Deployment Project SDK (iOS 6.1)
Development Previous Version (Xcode...
□ Use Autolayout

Localization Locking
Default Nothing
Reset Locking Controls

☑ English

☑ QuoteQuiz

First Version: 14-10-2014 @Joan Barrull 2014
Reviewed: Reviewed: