Crack The iOS Interview

100% Proof

What is iOS?

iOS is an Operating system for running iOS Applications installed on iPhones / iPad and iPod Touches Devices manufactured by Apple

Who is founder of Apple?

Steve Jobs

Who is the current CEO of Apple?

Tim Cook

What is NextStep(NS)?

NextStep was a company started by Steve Jobs where the Macintosh (MAC) operating system was developed. Later it was merged with Apple Inc.

What are the different products of Apple?

iPhones, iPads, iPod Touches, iMac, Mac Minis, Mac Servers, iPods, Wireless Keyboards & Mouses etc

What are the base languages of the Objective-C?

C-Language and Smalltalk (OOPs)

Who are the developer of Objective-C?

Brad Cox and Tom Love in 1980

What is Smart Phone?

A mobile phone that performs many of the functions of a computer, typically having a touchscreen interface, Internet access, and an operating system capable of running downloaded apps.

What are the most popular Smart Phone Operating Systems?

- iOS developed my Apple in 2007
- Android developed by Google in 2007
- Windows developed by Microsoft in 2000
- Blackberry developed by Blackberry Ltd in 1999

When was the first iPhone launched?

What is the latest version of iOS?

iOS 12.x

Please Note: It varies

iOS 12 Features

SMS OTP Autofill Multiple faces on faceTime (Group Face Time Video Call)

Hey siri works on low power mode Performance Improvements

What is the latest version of MAC?

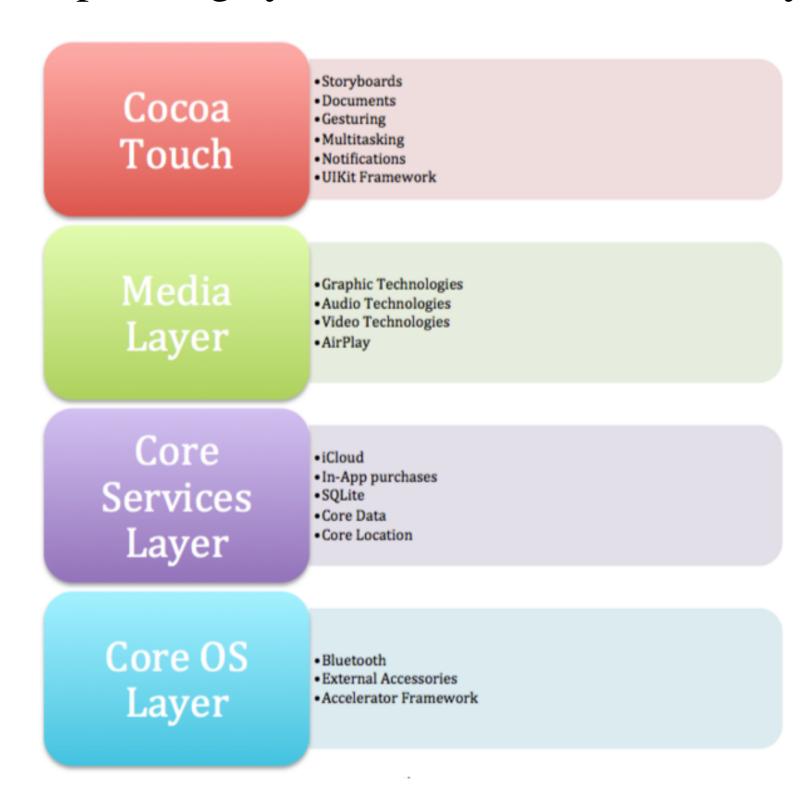
MAC 10.14

Name: Majova

Please Note: It varies

What is iOS Architecture?

iOS operating system is classified into 4 layers



iOS vs MAC OS

Cocoa
Media
Core Services
Core OS

Cocoa Touch
Media
Core Services
Core OS

MAC iOS

What is Cocoa Touch?

Cocoa Touch is a one of the layers in 4 layered architecture of iOS. It contains set of framework which deals with UI Appearance.

Available Frameworks: UIKit, MapKit, EventKit, AddressBookUI etc..,

What is Media Layer?

CoreMedia is a one of the layers in 4 layered architecture of iOS. It contains set of framework which deals with Multi Media and Graphics related functionality.

Available Frameworks: MediaPlayer, CoreAnimation, CoreGraphics, AVFoundation etc..,

What is Core Services?

Core Services is a one of the layers in 4 layered architecture of iOS. It contains set of framework which deals with Data Management and Network related functionality.

Available Frameworks: Foundation, CoreData, CFNetwork, CloudKit etc..,

What is Core OS?

Core OS is a one of the layers in 4 layered architecture of iOS. It contains set of framework which deals with Security and external hardware communication related functionality.

Available Frameworks: CoreBluetooth, Security, ExternalAccessary etc..,

What is iOS SDK?

iOS SDK is a Software Development Kit. It is a set of tools which are helpful to develop iOS / AppleTV applications.

i. Xcode: IDE for managing and develop iOS Apps ii. Simulator: Virtual Device to test the iOS Apps iii. Instruments: Profiling the iOS Applications to optimise the resources (Processing time, Memory Allocation etc...)

What is the framework used to do design UI in iOS? UIKit Framework

What is the framework used to do design UI in MAC OS?

AppKit Framework

What is the latest iPhone Modals?

iPhone 7, iPhone 7 +, SE



What are the resolutions of iOS Devices?

```
iPhone 3G (320x480)
iPhone 3GS (320x480)
iPhone 4 (320x480)*2
iPhone 4S (320x480)*2
```

What are the new features of latest iOS?

What are the new features of latest MAC OS?



What are the languages are used to develop iOS Apps?

Objective-C and Swift

Is Objective-C dynamic typed language?

YES



Is Swift dynamic typed language?

NO

What is the current version of Swift?

Swift 3.1 (It varies)



Objective-c vs Swift

Main method required	Main method not required
Every statement ends with semicolon	Semicolon free language
More code	Very less code
Methods reruns only one value	Methods can return multiple values using Tuples
Need to specify the datatype when declaring	No need to specify datatype when declaring the variable
Bit slower	More faster execution
Strings, Arrays are class types	Strings, Arrays are value types (Structures)

Note: Try to find some more differences from Internet

Swift 2 vs Swift 3

Function label for 1st parameter is not required names.indexOf("Taylor")	Function labels are must names.index(of: "Taylor")
NSDate	All Basic Classes dropped NS. Now Date, FileManager, Data, URL
	Omit useless words
	"Taylor".containsString("ayl")
	"Taylor".contains("ayl")
	"1,2,3,4,5".componentsSeparatedByString(", ")
	"1,2,3,4,5".components(separatedBy: ",")
	No ++ and —
	No General for-loop

Note: Try to find some more differences from Internet

What is Framework?

Framework is a collection of classes which serves some specific purpose.

Ex:

Foundation, UIKit, AddressBook, AddressBookUI, CoreData, MapKit etc..,

What is the Extension of iOS App?

.ipa is the extension of the iOS App.
.ipa stands for iPhone Application Achieve

What is the Extension of MAC App?

.app is the extension of the MAC App.

What are the different datatypes available in Objective-C?

int (smaller whole numbers)
long (bigger whole numbers)
float (smaller decimal values)
double (bigger decimal values)
char (characters, A-Z, a-z, 0-9 and special symbols)

BOOL (YES/NO, TRUE/FALSE, 1/0)
unichar (any language character)
NSInteger (equal to long)
NSUInteger (equal to unsigned long)
CGFloat (floats)
id (Any kind of object but not primitive datatypes)
SEL (which holds a method or pointer to a method)

What are the OOPs principles?

Class
Object
Encapsulation
Abstraction
Polymorphism
Inheritance
Message Passing(Method Calling)

What is class?

Class is a structure / template / blueprint to create objects. Once class is created you can create any number of objects out of it.

Class doesn't allocate any memory but object allocates the memory.

Ex: Building, Car, Person

Write a syntax of a Class?

In Objective-C class is constructed in two files.

- 1. interface file (ClassName.h file): contains declarations of instance variables and Methods.
- 2. implementation file (ClassName.m file): Contains Implementations.

```
.h file
@interface ClassName : NSObject
{
      // Instance variables declaration
}
// Properties / Methods declaration
@end
.m file
#import "ClassName.h"
@implementation ClassName
// Method implementations
@end
```

Object

Definition: Object is a physical live implementation of the class. Object can be touchable.

Ex: White House, Maruthi, Modi etc

How to create Object?

What is the syntax of the method?

```
Without input params:
```

+/-(ReturnType)methodName;

With input param:

+/-(ReturnType)methodName:(datatype)inputArg;

With multiple input params:

+/-(ReturnType)methodName:(datatype)inputArg:(datatype)inputArg;

Ex:

- -(void)display;
- -(void)display:(int)a;
- -(void)display:(int)a:(int)b;

Method Calling

```
[ClassName / objectName methodName];
With Parameters
[ClassName / objectName methodName: valueOne];
[ClassName / objectName methodName: valueOne :valueTwo];
 C-Language:
                               Obj-C:
 void sayHi();
                                -(void)sayHi;
                                [objectOne sayHi];
 sayHi();
 void wishHim(char *wish);
                               -(void)wishHim:(char*)wish;
                                [objectOne wishHim:"Hi"]
 wishHim("Hi")
                               -(void)sum:(int)a:(int)b;
 void sum(int a, int b);
 sum(10, 20);
                                [objectOne sum:10:20];
```

Note: Assume that we have a class, that has all the above methods and we created an object of that class called "objectOne"

Class method vs Instance method

Class method starts with + sign	Instance method starts with - sing
Class method must be called on top of class name	instance method must be called on top of object
Can't access instance variables in class method	Can access instance variables in instance method

Write an example for a class?

```
#import <Foundation/Foundation.h>
@interface BasicCalculator : NSObject
    int addRes, mulRes, subRes, divRes;
-(void)performAddition:(int)a : (int)b;
-(void)performSubtraction:(int)a : (int)b;
-(void)performMultiply:(int)a : (int)b;
-(void)performDivision:(int)a : (int)b;
@end
#import "BasicCalculator.h"
@implementation BasicCalculator
-(void)performAddition:(int)a : (int)b
   addRes = a = b;
-(void)performSubtraction:(int)a : (int)b
    subRes = a - b;
-(void)performMultiply:(int)a : (int)b
   mulRes = a * b;
-(void)performDivision:(int)a : (int)b
   divRes = a / b;
@end
```

```
#import "BasicCalculator.h"
int main(int argc, char * argv[]) {
    @autoreleasepool {
        BasicCalculator *aCalculator =
[[BasicCalculator alloc]init];
[aCalculator performAddition:10 :5];
[aCalculator performSubtraction:20 :5];
[aCalculator performMultiply:20 :5];
[aCalculator performDivision:20 :5];
return 0;
```

What is the Root or Super most class in Objective-C?

NSObject is the root class in Objective-C. NSObject class contains basic required functionality to create objects like alloc, init and type introspection methods



What is alloc?

alloc is a method which is declared in NSObject class which allocated memory for the Object.

What is init?

init is a method which is declared in NSObject class which initialises the proper default values for the newly allocated Object.

What is the default constructor in Objective-C?

Ans: init

Can we override init method?

Yes. We can override init method.

Override init method when you want to assign your own default values to the instance variables.

```
-(id)init
{
    if ([super init])
        {
             // assign required default values for the instance variables
        }
        return self;
}
```

What is new?

new is the method which serves for purpose of alloc and init.

new = alloc + init



Encapsulation

Encapsulation: Encapsulation is a process of binding instance variables and methods together into a single unit to keep safe from out side of that class.

Encapsulation is achieved through Class.

Data Encapsulation: Data encapsulation is the process of hiding instance variables' data to out side of the class/ Other classes. We can achieve data encapsulation using Accessor Specifiers

Q: How do you achieve the encapsulation in Objective-C?

Ans: Using Classes

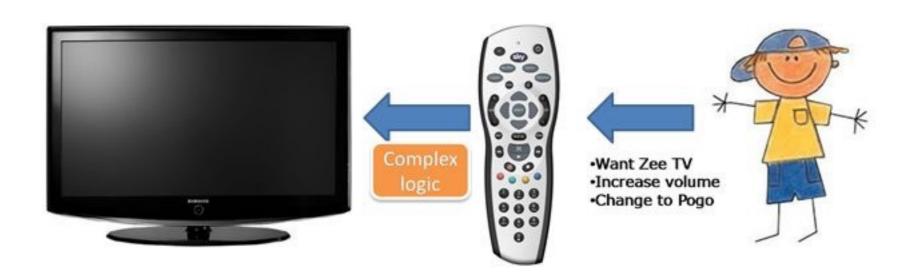
Q: How do you achieve the data encapsulation?

Ans: Using Accessor Specifiers

Abstraction

Abstraction: Abstraction is a process of exposing necessary functionality and hiding implementation Details. We can achieve abstraction using interface file.

As a user we just need remote to operate TV but we don't need the complex login behind the remote and TV sensors communication. So necessary functionality is REMOTE and implementation details are (complex sensors logic)



Inheritance

Inheritance is a process of deriving some properties and behaviours from its super class.

Syntax:

```
#import <Foundation/Foundation.h>
#import "BasicCalculator.h"
@interface ScientificCalculator : BasicCalculator
    // Additional Instance variables declaration
    int powerRes;
//Additional Properties / Methods declaration
-(void)power:(int)a;
@end
#import "ScientificCalculator.h"
@implementation ScientificCalculator
// Additional Method implementations
-(void)power:(int)a
   powerRes = a * a;
}
@end
NOTE: Here BasicCalculator is the super class and ScientificCalculator is the sub class.
```

Can we access subclass properties from super class object?

No

Ex:

If you purchase the Basic Calculator, you can perform Add, sub, mul and div operations but you can not perform power operation with Basic Calculator

If you purchase the Scientific calculator you can perform Add, Sub, Mul, Div + Power operations.

Tip: You can access super class + Sub Class properties from sub class.

You can not access sub class properties from super class.

What is Accessor Method?

Accessor methods are the methods who provides an access to instance variables form an out side of the class.

Here, name and rollNumber are the variables, without accessor methods you can not access them out side of the class (main or other class).

Accessor methods are combination of setter and getter.

setter: This method is used to set the value for an instance variable

getter: This method is used to get the value of an instance variable.

Once accessor methods are defined for a variable, you can access that instance variable in two ways.

- i. Method calling notation
- ii. Dot(.) notation

```
[firstStudent setName: "value"];
firstStudent.name = "value";
```

Note: Setter and getter methods must follow the syntax to access using Dot(.) notation.

How to declare setter method?

Assume we have two instance variables in a class as below.

```
char *name;
int rollNumber;
```

Setter syntax:

- -(void)setVariableNameInCCC:(VariableDatatype)argumentName
- i. Place -(void)set (This is common for all setters)
- -(void)set
- ii. Append the variable name in Capital Camel case.
- -(void)setName
- iii. Place the : and datatype in braces and give some inputArg name
- -(void)setName:(char*)aName;

int rollNumber;

-(void)setRollNumber:(int)aRollNumber;

Note: Setters sets value to the instance variable. So setter method doesn't return a value.

How to implement Setter Method

Syntax:

```
-(void)setVariableNameInCCC:(VariableDatatype)argumentName
   variableName = tempVariable;
CCC: CapitalCamelCase
Ex:
char *name;
int rollNumber;
-(void)setName:(char*)aName
    name = aName;
-(void)setRollNumber:(int)aRollNumber
    rollNumber = aRollNumber;
```

How to declare Getter Method?

Assume we have two instance variables in a class as below

```
char *name;
int rollNumber;
```

Getter syntax:

-(VariableDatatype)variableName

Ex:

- (char*)name;
- (int)rollNumber;
- i. Place sing and variable datatype in braces
- (char*)
- ii. Place variable name as it is.
- (char*)name;

Note: Getters returns the value to the instance variable. So getter method doesn't have input argument.

How to implement the getter method?

Syntax:

```
-(Datatype)variableName
  return variableName;
Ex:
char *name;
int rollNumber;
-(char*)name
    return name;
-(int)rollNumber
    return rollNumber;
```

Task

```
i. Write accessor methods for the following variables.
@interface ClassRoom : NSObject
    char *className;
    Student *aStudent;
    int classRoomNumber;
-(void)setClassName:(char*)tempName;
-(char*)className;
-(void)setAStudent:(Student*)tempStudent;
-(Student*)aStudent;
-(void)setClassRoomNumber:(int)tempNumber;
-(int)classRoomNumber;
@end
```

```
@implementation ClassRoom
-(void)setClassName:(char*)tempName
    className = tempName;
-(char*)className
    return className;
-(void)setAStudent:(Student*)tempStudent
    aStudent = tempStudent;
-(Student*)aStudent
    return aStudent;
-(void)setClassRoomNumber:(int)tempNumber
    classRoomNumber = tempNumber;
-(int)classRoomNumber
    return classRoomNumber;
@end
```

```
#import "Student.h"
#import "ClassRoom.h"
int main(int argc, const char * argv[]) {
    @autoreleasepool
        ClassRoom *aClass = [[ClassRoom alloc] init];
        aClass.aStudent = [[Student alloc] init];
        aClass.classRoomNumber = 100;
        aClass.className = "XII":
        NSLog(@"Student info: %@", aClass.aStudent);
        NSLog(@"Class room number: %i", aClass.classRoomNumber);
        NSLog(@"Class name: %s", aClass.className);
        ClassRoom *bClass = [[ClassRoom alloc] init];
        [bClass setAStudent:[[Student alloc] init]];
        [bClass setClassName:"XI"];
        [bClass setClassRoomNumber:101];
        NSLog(@"Student info: %@", [bClass aStudent]);
        NSLog(@"Class room number: %i", [bClass classRoomNumber]);
        NSLog(@"Class name: %s", [bClass className]);
    return 0:
Output:
Student info: <Student: 0x100200300>
Class room number: 100
Class name: XII
Student info: <Student: 0x1001071a0>
Class room number: 101
Class name: Xi
```

What is property? How to declare properties?

Properties are the ways to auto generate the accessor methods(setter and getter). We can use properties as an alternate to Accessor methods

```
Syntax:
h:
@property datatype variableName;
.m file:
@synthesize variableName;s
.h file:
int rollNumber;
-(void)setRollName:(int)tempNumber;
                                           @property int rollNumber;
-(int)rollNumber;
.m file:
-(void)setRollName:(int)tempNumber
  rollNumber = tempNumber;
                                            @synthesize rollNumber;
-(int)rollNumber
    return rollNumber;
```

Keep in Mind

- Generally you can access the ivars in same class methods (No Accessors / Properties required)
- Write setters to set values for iVars from outside of the class
- Write getter to access the iVar's value from outside of the class
- Use @property and @synthesize as alternate to writing setters and getters
- @property and @synthesize auto generates the accessor methods for the declared ivars.
- Declaring @property, @synthesize or Accessor methods allows us to access ivars outside of the class using two notations.
 Dot(.) notation and method calling notation
- @synthesize is not mandatory in implementation file
- Use _iVarName or self.iVarName to access iVars in implementation if @synthesize is not declared.
- @synthesize is used in implementation file specifies to complier that for which variables the accessor methods to be generated.

What is Polymorphism?

i) <u>Overloading</u>: In any class having same method names but difference in number of parameters / order of parameters / type of parameters is known as overloading.

Ex:

```
-(void) print; // prints some message || print
-(void) print:(int) a; // prints value of a || print:
-(void) print:(float b); // prints value of b || print:
-(void) print:(int) a :float b; // prints a and b || print::
-(void) print:(float) b :(int) a; // prints a and b || print::
```

Objective-C doesn't allow to have multiple methods with same name (print:, print: or print::, print::) in single class.

Q: Does Obj-c allow method Over loading?

Ans: No, Objective-c doesn't allow to write multiple methods with same name.

What is Overriding?

Overriding: In Inheritance relationship, having same methods in super class and sub class with difference in implementation.

To achieve Overriding, you must have inheritance relationship between two classes.

```
Super Class method:
    -(void)display
{
        NSLog(@"This is super class method");
}
Sub Class:
    -(void)display
{
        NSLog(@"This functionality is different");
}
```

Message Passing

Message passing is the process of calling a method.

[receiver message];

Ex:

[basicCalculator add];

Here, receiver could be instance or class message is a method name.

Memory Management

How the memory management works?

Memory management works based on the Object's retainCount value. Once the retain count of an object becomes 0, the object is removed from memory.



What is the retainCount of the newly created object?

The newly created object contains the retain count of 1.

How to clean the objects memory?

Object can be removed from memory by reducing the retainCount of the object to 0 by calling release message on top of object else setting the nil removes the object from memory.



What is release?

release is a method which instantly/immediately decrements the retainCount of the object by 1.

When the object's retainCount becomes 0, that object is removed from memory.

What is autorelease?

autorelease is a method which decrements the retainCount of the object by 1 in some point of time in future.

We can say autorelease is delayed release of the object.

The objects which are autoreleased are added to autoreleasepool.

What is autoreleasepool?

autoreleased objects are stored.



When the autoreleased objects are released?

Runtime system checks autoreleasepool in specific time intervals (we say as Run Loops). If runtime system finds any unused objects, it calls release message on top of those objects. So that the unused object is released immediately.

When should we use the autorelease?

There are two situations where we use autorelease.

- 1. Whenever you are not sure "when is your work is going to be finished with your object". Using autorelease makes sure the object retained till you need it.
- 2. When a method returning a newly created object.

```
Ex:
-(id)createNewObject
{
    id newObj = [[Class alloc] init];
    return [newObj autorelease];
}
```

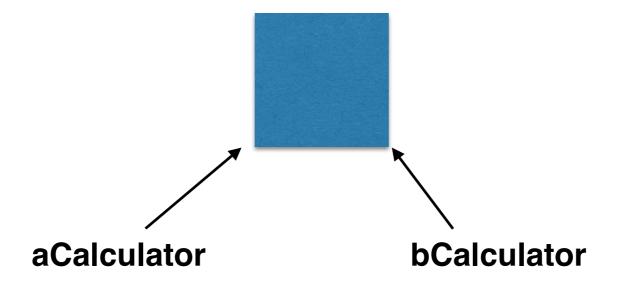
What is retain? when to use it?

retain is a method which increments the retain count of the object by 1.

Use retain when you are sharing the object's ownership with another object.

Ex:

```
BasicCalculator *aCalculator = [[BasicCalculator alloc]
init];
BasicCalculator *bCalculator = [aCalculator retain];
```

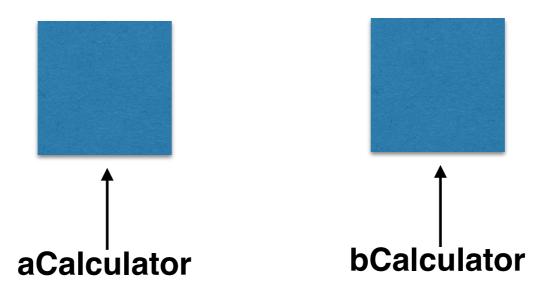


What is copy? when to use it?

copy is a method which creates new copy of the object and set the retainCount 1 to the newly copied object.

Use copy when you are sharing the object's ownership with another object.

```
Ex:
BasicCalculator *aCalculator = [[BasicCalculator alloc]
init];
BasicCalculator *bCalculator = [aCalculator copy];
```



What is the difference between retain vs copy?

```
retain:
 Fruit *aApple = [[Fruit alloc] init];
 Fruit *bApple = [aApple retain];
retain increments the retainCount of the R.H.S.
object(aApple) by 1 and assigns the retained object to
L.H.S (bApple).
Note: Modifying the aApple / bApple modifies the contents
on aApple and bApple.
copy:
 Fruit *aApple = [[Fruit alloc] init];
 Fruit *bApple = [aApple copy];
 copy creates new copy of the R.H.S (aApple) and assigns
 the object to L.H.S(bApple). Now the newly created
 object(bApple) contains retainCount value as 1.
```

Note: Modifying the contents of aApple doesn't affect the contents of bApple.

What is Memory Leak?

A leak is a situation where operating system fails to release unused Object's memory.



What is ARC?

ARC(Automatic Referencing Counting) is the process of Automatically Managing the Objects' Memory.

Automatic Reference Counting (ARC) is a compiler feature that provides automatic memory management of Objective-C objects. Rather than having to think about retain and release operations, ARC allows you to concentrate on the interesting code, the object graphs, and the relationships between objects in your application.

What is property and how to declare a property?

Properties are the ways to auto generate accessor methods.

Syntax:

@property (property Attributes) Datatype variableName;



ARC vc Garbage Collector

ARC doesn't handle retain cycles	GC handles the retain cycles
Compile time	Run time

What is property attribute?

Property attribute specifies to complier that what kind of accessor methods to be generated.

- readonly
- readwrite
- assign
- weak
- copy
- strong
- atomic
- nonatomic
- getter==
- setter==

What are the default property attributes?

assign, readwrite and atomic

atomic vs nonatomic?

atomic is the default behaviour of any property	it is not the default behaviour of the property
Only one process / thread can access the variable at given point of time	Multiple processes can access the variable at same point of time
is not fast, as it ensures the process is completed entirely	It is faster since multiple processes can access the property at same point of time
results are accurate	results are unexpected

strong vs weak?

strong increments the retainCount of an object	weak doesn't increment the retainCount of an object
modifying the contents of a/b reflects the two objects	modifying the contents of a/b reflects the two objects
releasing one object (a/b) just decrements the retain count by 1 and still keeps the object	releasing the object (a/b) decrements the retain count by 1 and removes the object from memory
Object a and b contains retain count 2	Object a/b contains retainCount 1.

assign vs weak?

assign is only used for primitive datatypes	weak is only used for reference types / objects
assign simply assign the value to instance variable	weak simply assigns the object to instance variable
releasing instance variable removed the actual object	releasing the object keeps nil at the object place. So it prevents the crashes

retain vs strong? strong is new word for retain in ARC?

What is zombi object?

The object whose retain count is zero called Zombi object.

Calling a method on zombi object leads to crash.

What happens calling a method on nil?

Nothing happens when you call a method on top of nil.

What is the output of the below statement?

```
Fruit *apple = [[[[Fruit alloc] init]
autorelease];
```

Ans: Calling autorelease for multiple times doesn't affect the app execution.

List some Foundation framework classes?

NSString NSArray NSDictionary NSSet NSNumber NSData NSDate NSFileManager **NSNotificationCenter NSTimer** etc..,



NSString vs NSMutableString?

NSString is a constant string	NSMutableString is a variables string
The content of the NSString class object can not be modified	Content of the string can be modified at the same memory location
Subsequent changes of the string leads to memory wastage	No wastage of memory

String Operations

```
1. How to find number of characters in aString?
NSString *aString = @"I am learning Strings";
NSLog(@"Number of characters: %i", aString.length); // 21
2. How to find number of words in a string?
NSArray *words = [aString componentsSeparatedByString:@" "];
NSLog(@"Number of words : %i", words count); // 4
3. How to find number of digits in a string?
NSString *bString = @"Hello, My roll number is 11 and I
am 12 years old";
int numberOfDigits = 0;
for (int i = 0; i < bString.length; i++)</pre>
   if ([bString characterAtIndex:i] >= 48 && [bString
characterAtIndex:i] <= 57)</pre>
        numberOfDigits++;
NSLog(@"number of digits : %i", numberOfDigits); // 4
NOTE: ASCII values for 0-9 are 48-57
                             Reach me @iPhoneDev1990@gmail.com
```

String Operations

4. Write a program for reversing a string?

```
NSString *bString = @"Hello, My roll numebr is 11 and
I am 12 years old";
NSMutableString *reversedString = [[NSMutableString]
alloc] init];
for (int i = bString.length-1; i >= 0; i--)
    unichar aChar = [bString characterAtIndex:i];
    [reversedString appendString:[NSString
stringWithFormat:@"%c", aChar]];
NSLog(@"Reversed String is: %@", reversedString);
0/P:
dlo sraey 21 ma I dna 11 si rbemun llor yM ,olleH
```

String Operations

4. Write a program for reversing all words of a string?

```
NSString *bString = @"Hello, My roll number is 11 and I am 12 years
old";
words = [bString componentsSeparatedByString:@" "];
NSMutableArray *reversedWordsArray = [[NSMutableArray alloc] init];
for (NSString *aWord in words)
    NSMutableString *reversedWord = [[NSMutableString alloc] init];
    for (NSInteger i = aWord.length-1; i >= 0; i--)
        unichar aChar = [aWord characterAtIndex:i];
        [reversedWord appendString: [NSString
stringWithFormat:@"%c", aChar]];
    [reversedWordsArray addObject:reversedWord];
NSString *reversedWordsString = [reversedWordsArray
componentsJoinedByString:@" "];
NSLog(@"Reversed words string: %@", reversedWordsString);
0/P:
olleH yM llor rebmun si 11 dna I ma 21 sraey dlo
```

How to get number from digits string?

```
NSString *a = @"10";
NSString *b = @"20";
int sum = a.intValue + b.intValue;
```

How to represent a number as a string?

```
NSString *numberString = [NSString
stringWithFormat:@"%i", sum];
```

What is the difference between isEqualToString: vs == ? isEqualToString: compares the contents of two strings.

== compares the addresses of two strings



How to trim starting and ending white space characters in a string?

```
NSString *aString = @" Hello World ";
aString = [aString
stringByTrimmingCharactersInSet:[NSCharacterSet
whitespaceCharacterSet]];
NSLog(@"Trimmed String is:%@", aString);

0/P: Trimmed String is:Hello World
```

What is id?

id is a generic datatype, which can hold any kind of object but not primitive datatype variable or value.

```
id aGeneric = [[NSArray alloc] init];
aGeneric = [[NSString alloc] init];
aGeneric = [[Car alloc] init];
aGeneric = [[Calculator alloc] init];
```

Here, aGeneric is capable of holding NSArray, NSString, Car and Calculator objects.

```
aGeneric = 10; // ERROR
aGeneric = 10.50; // ERROR
aGeneric = 'c'; // ERROR
NOTE: 10, 10.50, 'c' are the primitives
```

What is dynamic typing? How to achieve it?

The Process of identifying the datatype of an object at runtime is known as dynamic typing.

By using id we can achieve the dynamic typing. Runtime system doesn't know the type of the id variable until it executes the statement.

What is Type Introspection?

Type Introspection is process of identifying an object is kind of the given class or is it in the hierarchy of the given class.

We have two methods to identify the type introspection

```
- (BOOL)isKindOfClass:(Class)aClass;
```

- (BOOL)isMemberOfClass:(Class)aClass;

What is isKindOfClass:?

isKindOfClass: checks for the object is of given class type or its subclass type.

```
NSString *cString = @"Hello, My roll number is 11 and I am 12
years old";
if ([cString isKindOfClass:[NSMutableString class]])
    NSLog(@"bString is object of NSString or it's subclass's
object");
O/P: bString is object of NSString or it's subclass's object
if ([cString isKindOfClass:[NSString class]])
{
    NSLog(@"bString is of NSString or it's subclass's object");
else
    NSLog(@"bString is not an object of NSString or it's
subclass's object");
O/P: bString is object of NSString or it's subclass's object
```

What is isMemberClass:?

isMemberOfClass: checks for the receiver is exactly of given class type or not.

```
NSString *cString = @"Hello, My roll number is 11 and I
am 12 years old";
if ([cString isMemberOfClass:[NSMutableString class]])
    NSLog(@"bString is an object of NSMutableString
class");
else
    NSLog(@"bString is not an object of NSMutableString
class");
0/P:
bString is not an object of NSMutableString class
```

self vs super

We use self and super in methods.

self refers to the same class instance variables and methods whereas super refers to the super class instance variables and methods

use self when referring instance variables to differ from local variables which are having same name.

use super when calling a super class methods from sub class methods.

What is NSArray?

NSArray is collection of objects but not primitive datatype values. NSArray ends with nil. nil indicates the end of the Array. NSArray elements are accessed by index. NSArray index starts from 0 to size-1.



How to find number of objects in an Array?

```
NSArray *array = [[NSArray alloc]
initWithObjects:@"One", @"Two", @"Three", @"Four",
nil];
NSLog(@"%i", array.count); // 4
```

How to sort the elements in an Array?

```
NSArray *array = [[NSArray alloc]
initWithObjects:@"One
@"Two", @"Three", @"Four", nil];
array = [array]
sortedArrayUsingSelector:@selector(compare:)
NSLog(@"Sorted Array: %@", array);
O/P:
Sorted Array:
               Four,
               One,
               Three,
                Two
```

How to check the object existence in an Array?

```
NSArray *array = [[NSArray alloc]
initWithObjects:@"One",
@"Two", @"Three", @"Four", nil];
if ([array containsObject:@"One"])
{
    NSLog(@"One is existing in array");
else
    NSLog(@"One is not existing in array");
O/P:
One is existing in array
```

What is fast enumeration?

Fast enumeration is process of iterating through the collection in a faster way.

Fast enumeration is an alternate way of iterating through collection object.

```
Syntax:
for (<#type *object#> in <#collection#>) {
   <#statements#>
Ex:
NSArray *array = [[NSArray alloc] initWithObjects:@"One"
@"Two", @"Three", @"Four", nil];
for (id anArrayObj in array)
    NSLog(@"Array object is : %@", anArrayObj);
```

NSArray vs NSMutableArray?

NSArray is constant array. We can not modify the contents of the NSArray object once it is declared.

NSMutableArray is variable array. We can modify the contents of the NSMutableArray at any point of time.

What is the output of the below statement?

```
NSString *a = nil;
NSMutableArray *array = [[NSMutableArray alloc]
initWithObjects:@"One", @"Two", a, @"Three", @"Four",
@"one", nil];
NSLog(@"Count is: %i", array.count);
O/P:
Count is : 2
```

NOTE: Array ends with nil. Here a contains nil. So the array is terminated by a.

Can we store primitive values in an Array?

No, Array is capable of holding only objects not primitive datatype values.

What is the retain count of the object a after adding it to Array?

```
Fruit *a = [[Fruit alloc] init];
NSArray *array = [[NSArray alloc] initWithObjects:a, @"Two",
@"Three", nil];
NSLog(@"retain count of a: %i", a.retainCount);

O/P:
retain count of a: 2

NOTE: Adding an object to collection increments the retain
```

count of the object.

Output?

```
NSString *a = @"Hi";
NSArray *array = [[NSArray alloc] initWithObjects:a,
@"Two", @"Three", nil];
a = @"Hello";
NSLog(@"retain count of a: %lu", a.retainCount);
NSLog(@"%@", array);
0/P:
    Hi,
    Two,
    Three
```

NOTE: Constants are copied

Output?

```
NSMutableString *a = [[NSMutableString alloc]
initWithString:@"Hi"];
NSArray *array = [[NSArray alloc] initWithObjects:a,
@"Two", @"Three", nil];
[a appendString:@" World"];
NSLog(@"retain count of a: %lu", a.retainCount);
NSLog(@"%@", array);
0/P:
retain count of a: 2
   "Hi World",
   Two,
   Three
Note: Objects are referenced
```

What is Dictionary?

Dictionary is a collection of key-value pairs.

Keys and values must be an Object.

Dictionary doesn't allow duplicate keys, but can hold duplicate values.



Does Dictionary allows duplicate keys?
No

Does Dictionary allows duplicate keys?
YES

How to access the value in Dictionary?

```
NSDictionary *aDict = [[NSDictionary alloc]
initWithObjectsAndKeys:@"Obj1",
@"Key1", @"Obj2",@"Key2", @"Obj3",@"Key3",
@"Obj4",@"Key4", nil];

NSLog(@"Accessing value for Key2: %@", [aDict valueForKey:@"Key2"]);

O/P:
Accessing value for Key2: Obj2
```

How to add new key-value pair to Dict?

```
NSMutableDictionary *aDict = [[NSMutableDictionary
alloc] initWithObjectsAndKeys:@"Obj1",
@"Key1",@"Obj2",@"Key2", nil];

[aDict setValue:@"Obj3" forKey:@"Key3"];
```

What is NSSet

NSSet is a un-ordered collection of non-duplicate elements.

It is better in performance compare to NSArrays when checking the existence of an object.

NSSet vs NSCountedSet

NSSet doesn't allow duplicates.
NSCountedSet allows duplicate elements. It maintains a counter for each object.

```
NSSet *aSet = [[NSSet alloc] initWithObjects:@"One", @"Two", @"Three",
@"One",
nil];
NSLog(@"set elements: %@", aSet);
0/P:
{(
   One,
   Three,
   Two
)}
NSCountedSet *acSet = [[NSCountedSet alloc] initWithObjects:@"One",
@"Two", @"Three", @"One", nil];
NSLog(@"set elements: %@", acSet);
0/P:
One [2],
Two [1],
Three [1]
```

What is Protocol?

```
Protocol is a set of methods declarations.
Protocol contains two kinds of methods.
i. required and ii. optional
Syntax:
@protocol ProtocolName <NSObject>
// Method declarations
@end
Ex:
@protocol ColorChangeDelegate <NSObject>
@required
-(void)colorChanged;
@optional
-(void)colorChangedToRed;
-(void)colorChangedToGreen;
-(void)colorChangedToBlue;
@end
```

What is the default method kind in a protocol?

By default all methods are required methods

```
@protocol ColorChangeDelegate <NSObject>
-(void)colorChanged;
@end
```

the -(void)colorChanged; is required method

How to use / adopt protocol?

```
@interface Fruit : NSObject<ProtocolName>
@end
#import "Fruit.h"
@implementation Fruit
-(void)colorChanged
    NSLog(@"Color changed");
-(void)colorChangedToRed
    NSLog(@"Color changed to red");
-(void)colorChangedToGreen
    NSLog(@"Color changed to green");
-(void)colorChangedToBlue
    NSLog(@"Color changed to blue");
@end
```

required vs optional methods?

required methods are expected to be implemented in adopting class (Fruit class)

Optional methods are not mandatory to implement in adopting class

Can we adopt multiple protocols?

Yes. We can adopt multiple protocols for same class.



Advantages of Protocols?

Multiple inheritance and Delegation can be achieved using Protocols

Can we declare instance variables in Protocol?

No. Protocols doesn't allow to declare instance variable and properties.



What is formal protocol?

It is an alias name for the protocol.

What is informal protocol?

It is an alias name for the category.

or

It is a category on NSObject.

Can we have a class and protocol name as same?

Yes

Ex: NSObject is a class and Protocol.



What is Category?

Category is the process of extending class without sub classing (Inheritance).

We can extend the class for which we don't have implementation

What are the disadvantages of Category?

- Some times you unintentionally override the existing methods. So that it looses the actual functionality.
- We can't declare properties and instance variables in category

Write syntax of the Category?

```
@interface className (categoryName)
// Additional method declarations
@end
@implementation className (categoryName)
// Additional method implementations
@end
```

File Names:

```
ClassName + CategoryName.h
ClassName + CategoryName.m
```

Write an example of Category?

```
@interface NSString (Wrap)
-(NSString*)wrapStingInQuotes:(NSString*)aString;
@end
@implementation NSString (Wrap)
-(NSString*)wrapStingInQuotes:(NSString*)aString
    return [NSString stringWithFormat:@"\" %@ \"",
aString];
@end
Here, NSString class doesn't contain the
wrapStringInQuotes: method. We have extended the class
by using category.
```

Category vs Subclassing(Inheritance)

We can get the extended functionality with same class instance	We must create sub class object to get the extended functionality
We can not declare properties and instance variables in Category	We can declare instance variables and properties in Subclassing
File names are ClassName+CategoryName.h ClassName+CategoryName.m	File names are Subclass.h Subclass.m
You can override the existing methods	You can override existing methods in Subclassing also
Helps to divide the class into multiple parts	Helps to add new functionality to a class
Categories are the best solution for adding new behaviours	Subclassing is the best way for customising existing class

Reach me @iPhoneDev1990@gmail.com

What are the disadvantages of the Category?

- i. We can not declare instance variables and properties in Category.
- ii. Unintentionally overriding the existing functionality, causes to loose the actual functionality.



What is the output of the below code?

```
for (int j = 0; j < 10; j++)
    static int i = 1;
    NSLog(@"%i",i);
    i++;
0/P:
1 2 3 4 5 6 7 8 9 10
```

Note: Static variables are initialised only once

What is selector? How to create selector variable?

```
Selector is a pointer to a method
Syntax:
SEL aSelector = @selector(methodName);
```

How to execute a method which is pointed by the selector?

```
SEL aSelector = @selector(methodName);
[receiver performSelector:aSelector];
```



How to call a method after some delay?

```
SEL aSelector = @selector(methodName);
[receiver performSelector:aSelector withObject:nil
afterDelay:3];
```

How to check a method is implemented by a specific class or not? SEL aSelector = @selector(methodName); if ([self respondsToSelector:aSelector]) NSLog(@"methodName method is implemented in receiver's class"); else NSLog(@"methodName method isn't implemented in receiver's class");

How to achieve multiple inheritance in Objective-C?
Using protocols we can achieve multiple inheritance

in iOS

What is Circular Dependency?

Circular dependency is a situation where Two classes are imported in each other class.

```
Ex:
#import "Two.h"
@interface One : NSObject

@end

#import "One.h"
@interface Two : NSObject

@end
```

Here class Two is imported in class One and vice versa. So that compiler fails to link them

What is forward class declaration?

Forward class declaration is process of avoiding circular dependency.

Use @class pre-compiler directive in any one class to avoid circular dependency

```
@interface One : NSObject
@end

#import "One.h"
@interface Two : NSObject
@end
```

@class Two;

@class specifies to compiler that the class Two is existing, don't check for Class Two.

iOS

What is AppDelegate?

AppDelegate is a class which handles application life cycle events.



What is the super class of AppDelegate?

UIResponder is the super class of AppDelegate

```
#import <UIKit/UIKit.h>
@interface AppDelegate : UIResponder <UIApplicationDelegate>
@property (strong, nonatomic) UIWindow *window;
@end
```

What does UIApplicationDelegate contains?

UIApplicationDelegate is a protocol, it contains all Application Lifecycle event methods.

Some of them are below

```
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(nullable NSDictionary
*)launchOptions NS_AVAILABLE_IOS(3_0);
- (void)applicationDidBecomeActive:(UIApplication *)application;
- (void)applicationWillResignActive:(UIApplication *)application;
- (void)applicationWillTerminate:(UIApplication *)application;
- (void)applicationDidEnterBackground:(UIApplication *)application
NS AVAILABLE IOS(4 0);
- (void)applicationWillEnterForeground:(UIApplication *)application
NS_AVAILABLE_IOS(4_0);
```

What are the Application States?

Application States:

NotRunning State: App just installed or killed

Active State: App is in foreground and receiving events

InActive State: App is in foreground and received an SMS or Call etc

Background State: App is in background and receiving events

Suspended State: App is in background and not receiving events

What is the first method gets called when you launch the app?

```
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(nullable NSDictionary
*)launchOptions NS_AVAILABLE_IOS(3_0)
{
}
```

How many windows an App contains?

Every iOS Application contains only one base Window.



What is ViewController?

ViewController controls the view lifecycle. ViewController has few set of methods which handles view lifecycle

```
- (void)viewDidLoad;
- (void)viewWillAppear:(B00L)animated;
- (void)viewDidAppear:(B00L)animated;
- (void)viewWillDisappear:(B00L)animated;
- (void)viewDidDisappear:(B00L)animated;
- (void)viewWillUnload;
- (void)viewDidUnload;
```

When the viewDidLoad method called?

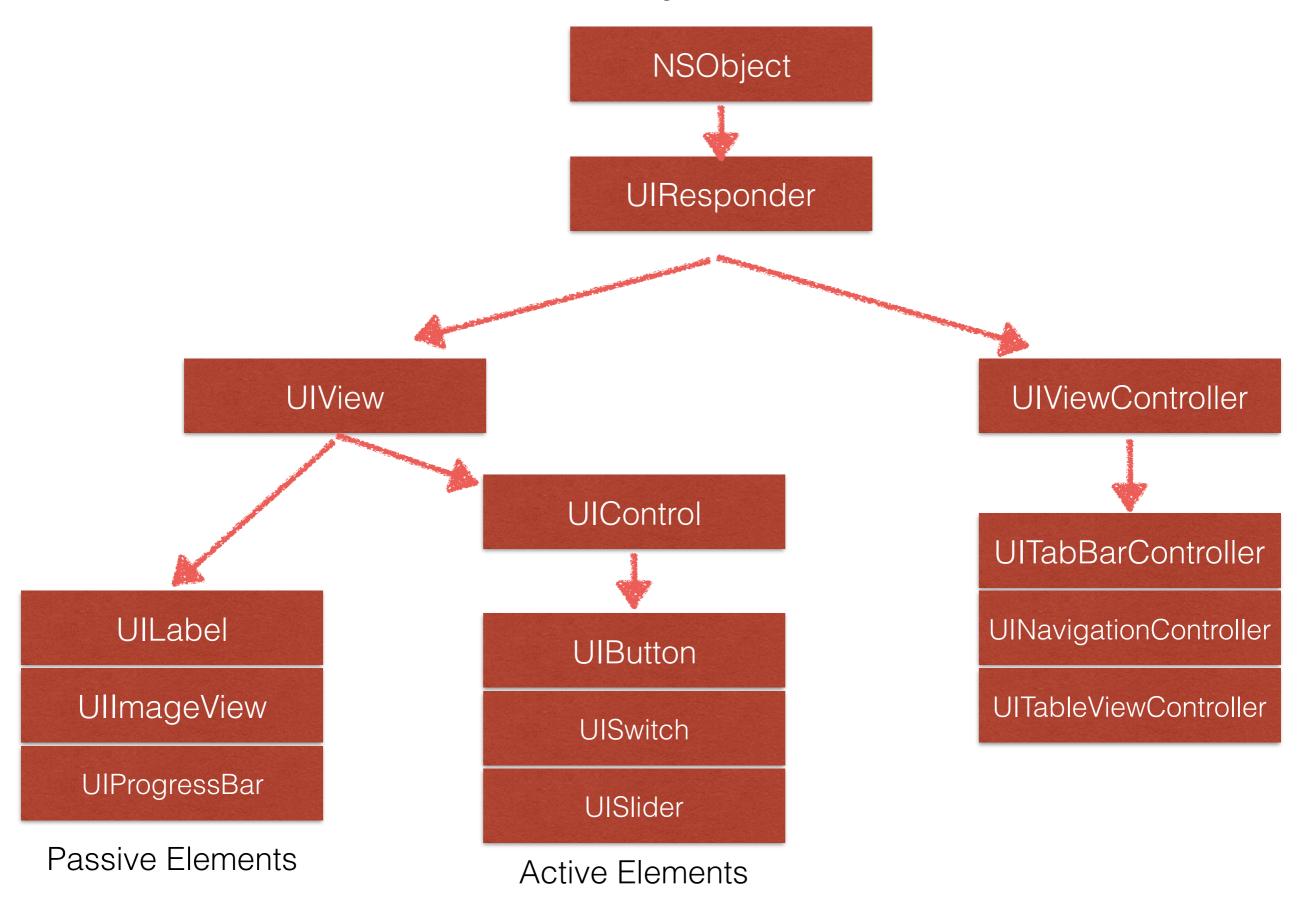
viewDidLoad method gets called when the view controller's view is accessed / loaded.

This method gets called only once when the ViewController's view is loaded.

What is Storyboard?

Storyboard is a file where we design user interface and flow of the Application

What is the Hierarchy of the UIControls?



What is the hierarchy of the UIButton?

UIButton -> UIControl -> UIView -> UIResponder->NSObject

What is IBOutlet and IBAction?

IBOutlet is a macro(preprocessor directive), it helps us to connect a storyboard UI Element with your controller. So that we can refer the storyboard UI element in your controller.

IBAction is a macro, it helps us to attach an action/method to an storyboard UI element. Based on the user interactions on that element, the action will be triggered.

IBOutlets and IBActions are replaced by void at runtime.

How to create a UIButton Programatically?

```
UIButton *aButton = [UIButton
buttonWithType:UIButtonTypeRoundedRect];
aButton.frame = CGRectMake(10, 10, 100, 30);
[aButton setTitle:@"Click Me"
forState:UIControlStateNormal];
[aButton addTarget:self action:@selector(clickedMe:)
forControlEvents:UIControlEventTouchUpInside];
[self.view addSubview:aButton];
-(void)clickedMe:(id)sender
    NSLog(@"Button clicked");
What is ":(id)sender" ?
sender is an UI element to which we have connected this
action.
Here, aButton is sender, since we have attached the
clickedMe:(id)sender method to aButton.
```

How to add a view to another view

using

- (void)addSubview:(UIView *)view;

How to customise a UIView?

We can customise a view by subclassing a UIView class.

```
@interface CustomView : UIView
// New properties and methods
@end
```

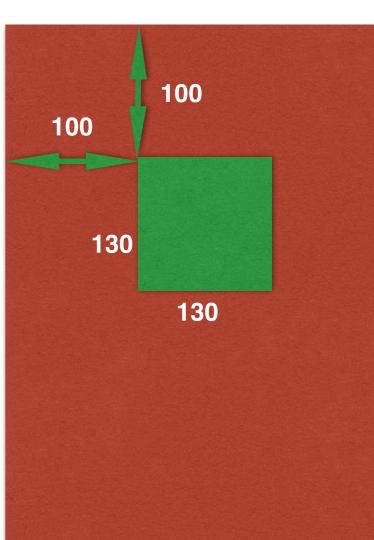
What is the difference between Frame and Bounds?

Frame is the Origin and Size of a View with respect to it's superview.

Bounds is just Size of the view, the origin is always (0,0).

Frame of green view is (x,y,width, height) i.e (100, 100, 130, 130)

Bounds of the green view is (0,0, width, height)
i.e (0,0,130,130)

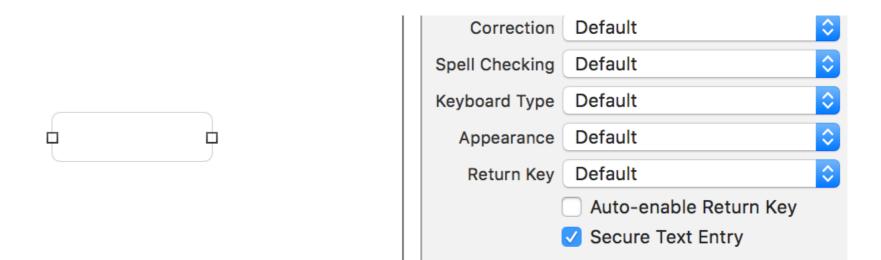


How to create a circular ImageView?

```
UIImageView *imageView = [[UIImageView alloc]
initWithFrame:CGRectMake(100, 100, 100, 100)];
imageView.layer.cornerRadius =
imageView.frame.size.width / 2;
imageView.layer.masksToBounds = NO;
[self.view addSubview:imageView];
```

How to enter secure text in UITextField?

Select the textfield and enable secure text entry in storyboard as below.



Programatically:

textField.secureTextEntry = YES;

How to hide the keyboard when user tap on return key?

```
- (B00L)textFieldShouldReturn:(UITextField *)textField
{
    [textField resignFirstResponder];
    return YES;
```



How to allow only 10 characters in UITextField?

```
- (BOOL)textField:(UITextField *)textField
shouldChangeCharactersInRange: (NSRange) range
replacementString:(NSString *)string
    if (string.length == 0) {
        return YES;
    if (textField.text.length >=9)
        return YES;
    else
        return NO;
```

How to allow only numbers in a TextField

```
- (B00L)textField:(UITextField *)textField
shouldChangeCharactersInRange: (NSRange) range
replacementString:(NSString *)string
    if (string.length == 0) {
        return YES;
    unichar aChar = [string characterAtIndex:0];
    if (aChar >= 48 \&\& aChar <= 57)
        return YES;
    else
        return NO;
```

How to display multiline text in UILabel?

```
In Autolayout System:
aLabelObject.numberOfLines = 0;
Non Auto-layout:
int requiredNumberOfLines = 3;
aLabelObject.numberOfLines = requiredNumberOfLines
```

What are the two protocols of UITableView?

UITableViewDelegate UITableViewDatasource



UITableViewDelegate vs UITableViewDataSource

UITableViewDelegate and UITableViewDataSource are the protocols.

UITableViewDelegate contains the methods which are triggered based on the user interactions

- i. Selections
- ii. DeSelections
- iii. Row Heights etc

UITableViewDataSource contains the methods which provides metadata for the TableView like

- i. Number of rows
- ii. Number of sections
- iii. Cell for each row

Write any 5 methods of UITableViewDelegate?

```
- (void)tableView:(UITableView *)tableView
didSelectRowAtIndexPath:(NSIndexPath *)indexPath;
- (void)tableView:(UITableView *)tableView
didDeselectRowAtIndexPath:(NSIndexPath *)indexPath
NS_AVAILABLE_IOS(3_0);
- (CGFloat)tableView:(UITableView *)tableView
heightForRowAtIndexPath: (NSIndexPath *)indexPath;
- (CGFloat)tableView:(UITableView *)tableView
heightForHeaderInSection: (NSInteger) section;
- (CGFloat)tableView:(UITableView *)tableView
heightForFooterInSection: (NSInteger) section;
- (nullable UIView *)tableView:(UITableView *)tableView
viewForHeaderInSection:(NSInteger)section;
```

Write any 5 methods of UITableViewDataSource?

```
- (NSInteger)tableView:(UITableView *)tableView
numberOfRowsInSection:(NSInteger)section;
- (UITableViewCell *)tableView:(UITableView *)tableView
cellForRowAtIndexPath:(NSIndexPath *)indexPath;
- (NSInteger)numberOfSectionsInTableView: (UITableView)
*)tableView;
- (nullable NSString *)tableView:(UITableView *)tableView
titleForHeaderInSection: (NSInteger) section;
- (nullable NSString *)tableView:(UITableView *)tableView
titleForFooterInSection: (NSInteger) section;
- (BOOL)tableView:(UITableView *)tableView canEditRowAtIndexPath:
(NSIndexPath *)indexPath;
- (BOOL)tableView:(UITableView *)tableView canMoveRowAtIndexPath:
(NSIndexPath *)indexPath;
- (nullable NSArray<NSString *> *)sectionIndexTitlesForTableView:
(UITableView *)tableView TVOS PROHIBITED;
```

What are the two required methods of DataSource?

```
- (NSInteger)tableView:(UITableView *)tableView
numberOfRowsInSection:(NSInteger)section;
```

- (UITableViewCell *)tableView:(UITableView
*)tableView cellForRowAtIndexPath:(NSIndexPath
*)indexPath;

Write the implementation of the CellForRowAtIndexPath Method?

```
-(UITableViewCell *)tableView:(UITableView *)tableView
cellForRowAtIndexPath:(NSIndexPath *)indexPath
    static NSString *reuseIdentifier = @"Cell";
    UITableViewCell *cell = [tableView
dequeueReusableCellWithIdentifier:reuseIdentifier];
    if (cell == nil)
        cell = [[UITableViewCell alloc]
initWithStyle:UITableViewCellStyleDefault
reuseIdentifier:reuseIdentifier];
    return cell;
```

What is static in static NSString *reuseIdentifier = @ "Cell"?

The CellForRowAtIndexPath: method gets called once per row. Creating reuseIdentifier object many times causes the memory wastage. So declaring as a static makes sure the object is initialised only once.



What is dequeueReusableCellWithIdentifier:?

dequeueReusableCellWithIdentifier: returns a free cell if it is existing with given identifier in a queue.

How to set the title and image to UITableViewCell?

```
// Setting text
cell.textLabel.text = @"Setting title";
//Setting image
cell.imageView.image = [UIImage
imageNamed:@"image1.png"];
//Setting sub title
cell.detailedTextLabel.text = @"Sub Title";
//Setting accessory view
cell.accessoryType =
UITableViewCellAccessoryTypeCheckMark;
```

How to refresh a TableView?

[tableView reloadData];

How to refresh specific cells in UITableView?

```
[self.tableView beginUpdates];
[self.tableView
reloadRowsAtIndexPaths:@[indexPathOfYourCell]
withRowAnimation:UITableViewRowAnimationNone];
[self.tableView endUpdates];
```

How to customise UITableViewCell?

Use UITableViewCell subclassing to customise the UITableviewCell

or

Use PrototypeCell with Tags

How to show the delete button on left swipe on Cell?

Implement following Delegate and Datasource methods

```
-(BOOL)tableView:(UITableView *)tableView canEditRowAtIndexPath:
(NSIndexPath *)indexPath
    return YES;
-(void)tableView:(UITableView *)tableView commitEditingStyle:
(UITableViewCellEditingStyle)editingStyle forRowAtIndexPath:
(NSIndexPath *)indexPath
    NSLog(@"User deleted a row");
-(UITableViewCellEditingStyle)tableView:(UITableView *)tableView
editingStyleForRowAtIndexPath:(NSIndexPath *)indexPath
    return UITableViewCellEditingStyleDelete;
```

How to display multiline text in UITableView?

- Drag and drop a UILabel object on Prototype cell and set Leading, Trailing, Top and Bottom constraints.
- Set 0 as number of lines for the UILabel object.
- Set the estimatedRowHeight property for a tableview.
 tableView.estimatedRowHeight = 50;

```
• Implement the below delegate method and return
-(CGFloat)tableView:(UITableView *)tableView
heightForRowAtIndexPath:(NSIndexPath *)indexPath
{
    return UITableViewAutomaticDimension;
}
```

TableView vs CollectionView

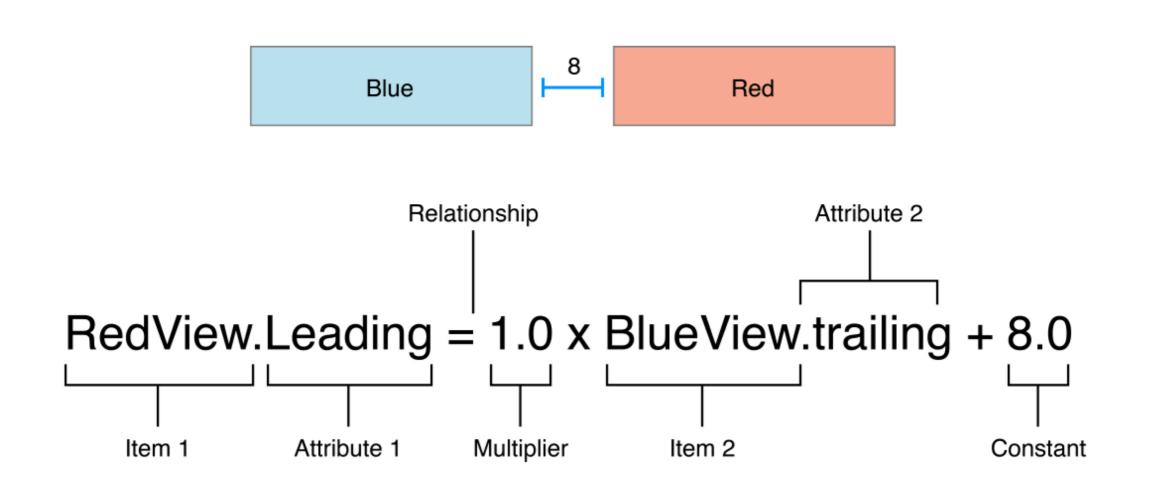
Contains only Rows	Contains Rows and Columns
Comprised of UITableViewCells	Comprised of UICollectionViewCells
Use TableView when you want to display a list	Use Collection view when you want to display items in a Grid fashion
	CollectionView supports different layouts (Circular, Grid, Tiled)

What is AutoLayout?

AutoLayout is collection of constraints those are used to design dynamic user interface

What is AutoLayout Equation?

aView.attribute = anotherView.attribute * Multiplier + constant



What is Constraint?

Constraint is a linear equation between a view and it's superview or surrounding views. Auto layout can be achieved using multiple constraints.



How to write constraint programatically?

What is Intrinsic Content Size?

Intrinsic content size is size of actual content in a view.

Ex: UILabel, UIImageView, UITextField etc contains intrinsic content size.



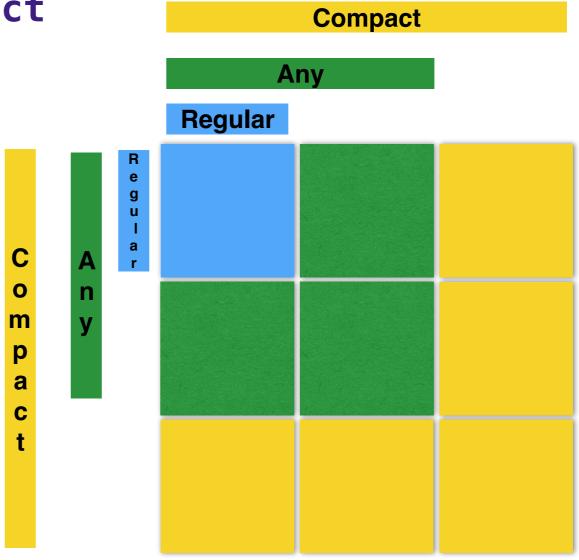
Actual size is size of blue view.
Intrinsic content size is the size of the green view.

What are the size classes we have?

Adoptive size classes having 3 size classes (Regular, Any and Compact) and their combinations.

iPhones: wRegular, hAny
All Devices: wAny, hAny

All iPads:wCompact, hCompact

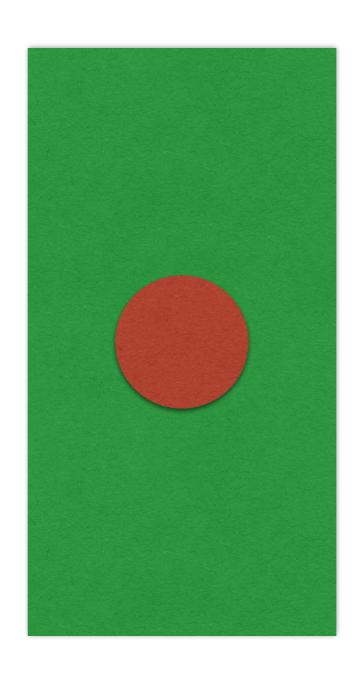


Which size class is used to design an Universal App?

Use wAny, hAny size classes to support for iPhones and iPADs.

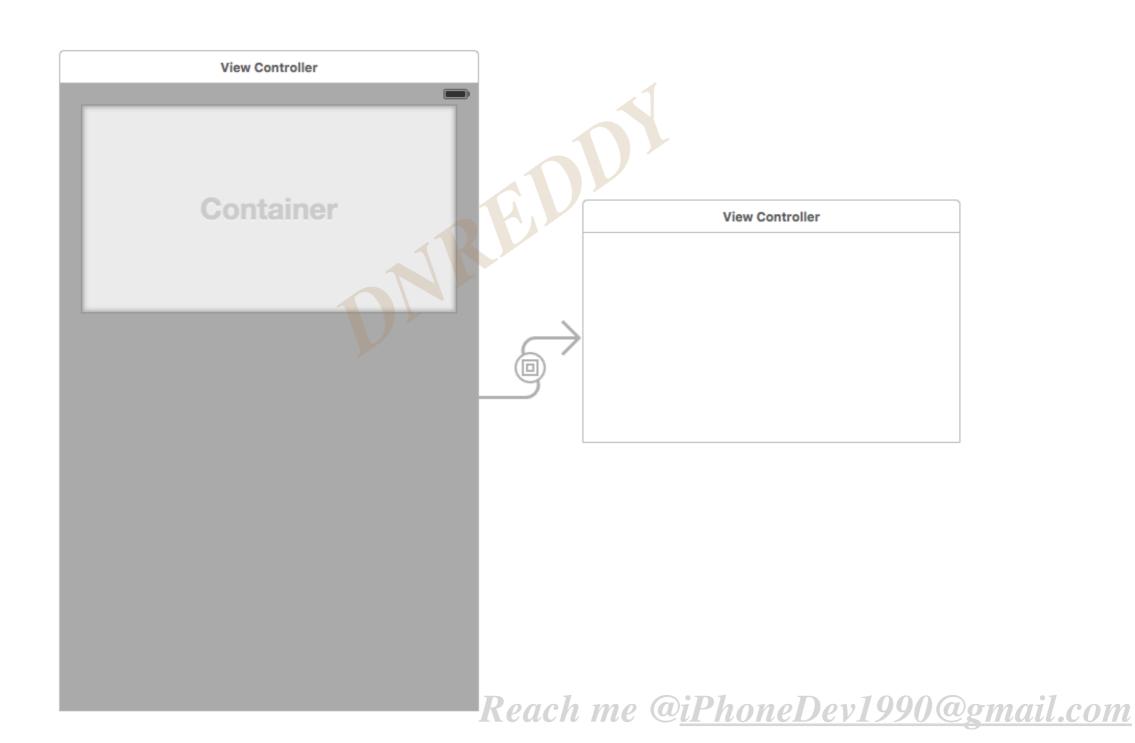


How to set the constrains for the below layout?



What is ContainerView?

UIContainerView is a class which provides a view that holds a view controller as a subview in ViewController.



What is StackView?

StackView is a View which is capable of displaying the items in an Horizontal or Vertical fashion.

StackView needs less constraints compared to AutoLayout

AutoLayout can be achieved with minimal constraints using StackView.

StackViews can be nested

StackView contains following attributes

Attributes

In the Attributes inspector, set the following stack view attributes:

Stack	Axis	Alignment	Distribution	Spacing
Stack View	Vertical	Fill	Fill	8

AutoLayout vs StackView

Collection of Constraints	Simpler way to achieve AutoLayout
Introduced in iOS 5	Introduced in iOS9
Difficult to achieve complex layouts	Simpler to achieve complex layouts

How to move from One VC to Another VC?

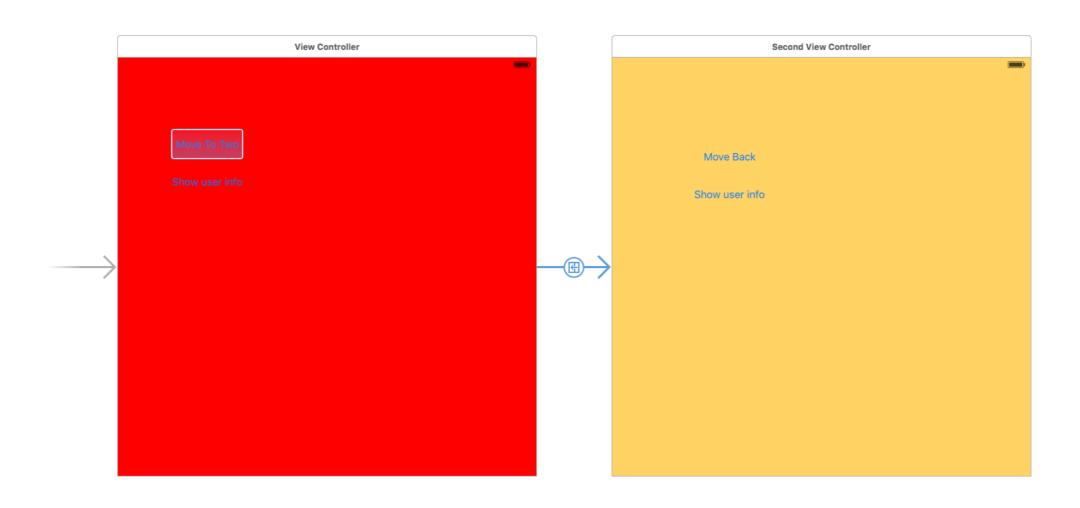
```
Way-1:
UIStoryboard *mainStoryboard = [UIStoryboard
storyboardWithName:@"Main" bundle:nil];
DestinationViewController *destinationViewController =
[mainStoryboard
instantiateViewControllerWithIdentifier:@"DestinationViewStoryboardI
D"1:
[self presentViewController:destinationViewController animated:YES
completion:nil];
Way-2: Using UINavigationController
UIStoryboard *mainStoryboard = [UIStoryboard
storyboardWithName:@"Main" bundle:nil];
DestinationViewController *destinationViewController =
[mainStoryboard
instantiateViewControllerWithIdentifier:@"DestinationViewStory
boardID"];
[self navigationController
pushViewController:destinationViewController animated:YES];
```

How to pass data between VCs?

```
Way-1:
UIStoryboard *mainStoryboard = [UIStoryboard
storyboardWithName:@"Main" bundle:nil];
DestinationViewController *destinationViewController = [mainStoryboard
instantiateViewControllerWithIdentifier:@"DestinationViewStoryboardID"
];
destinationViewController.stringObject = @"Data to be passed";
[self presentViewController:destinationViewController animated:YES
completion:nil];
Way-2: Using UINavigationController
UIStoryboard *mainStoryboard = [UIStoryboard
storyboardWithName:@"Main" bundle:nil];
DestinationViewController *destinationViewController =
[mainStoryboard
instantiateViewControllerWithIdentifier:@"DestinationViewStorybo
ardID"];
destinationViewController.stringObject = @"Data to be passed";
[self.navigationController
pushViewController:destinationViewController animated:YES];
```

What is Segue?

A segue defines a transition between two view controllers in your app's storyboard file.



What is Unwind segue?

Unwind segue is a way of calling a method in SourceViewController when moving back from Destination ViewController to SourceViewController.

It is used to transfer data back to SourceViewController when dismissing / Popping a DestinationViewController.

What is Navigation Controller?

- UINavigationController is a stack of ViewControllers
- We can perform Push and Pop operations on the stack
- We can identify number of ViewControllers in a Stack
- Easy to navigate from One VC to another VC.

How to get all view controllers in NavigationController stack?

```
NSArray *allViewControllerInStack =
[self.navigationController viewControllers];
```

How to push a ViewController in NavigationController?

```
[self.navigationController
pushViewController:destinationViewController
animated:YES];
```



How to pop a View Controller form NavController?

[self.navigationController popViewControllerAnimated:YES];

How to move to RootViewController in NavController?

```
[self.navigationController
popToRootViewControllerAnimated:YES];
```



UINavigationController vs UITabBarController?

UINavigationController is a stack of view controllers UITabBarController is a collection of View Controllers

How to display HTML Content in iOS?

```
i. Displaying HTML String
UIWebView *webView;
NSString *htmlString = @"<html>This is a HTML String!</html>";
[webView loadHTMLString:htmlString baseURL:nil];
ii. Displaying HTML String
NSString *filePath = [[NSBundle mainBundle]
pathForResource:@"TermsAndCondition" ofType:@"html"];
NSURL *url = [NSURL fileURLWithPath:filePath];
NSURLRequest *request = [NSURLRequest
requestWithURL:url];
[webView loadRequest:request];
iii. Loading a URL
NSURL *url = [NSURL URLWithString:@"https://
www.url.com"];
NSURLRequest *request = [NSURLRequest
requestWithURL:url];
[webView loadRequest:request];
```

What are the UIWebViewDelegate methods?

```
- (BOOL)webView:(UIWebView *)webView shouldStartLoadWithRequest:(NSURLRequest
*)request navigationType:(UIWebViewNavigationType)navigationType;
- (void)webViewDidStartLoad:(UIWebView *)webView;
- (void)webViewDidFinishLoad:(UIWebView *)webView;
- (void)webView:(UIWebView *)webView didFailLoadWithError:(nullable NSError *)error;
```

How to identify hyperlink clicks on UIWebView?

```
- (BOOL)webView:(UIWebView *)webView
shouldStartLoadWithRequest:(NSURLRequest *)request
navigationType:(UIWebViewNavigationType)navigationType
{
    // Use navigationType to identify the type of
    navigation like
    Link clicked / Refresh / Form Submitted etc
}
```

How to open an External Safari app in iOS?

```
NSURL *url = [NSURL URLWithString:@"https://www.url.com"];
[[UIApplication sharedApplication] openURL:url];
```

What are the ways to persistence the data in iOS?

- NSUserDefaults
- KeychainAccess
- PropertyList
- SQLite
- CoreData



What is NSUserDefaults?

NSUserDefaults is a class that allows simple storage of different data types. It is ideal for small bits of information you need to persist between app launches or device restarts.

What is KeychainAccess?

A keychain is an encrypted container that securely stores small chunks of data on behalf of apps and secure services.

Keychain Services provides secure storage of passwords, keys, certificates, and notes for one or more users. A user can unlock a keychain with a single password, and any Keychain Services—aware application can then use that keychain to store and retrieve passwords.

NSUserDefaults vs KeychainAccess

NSUserDefaults has no encryption mechanism	Keychain Access has encryption mechanism called AES-128.
Used to store small amount of application preferences data	Used to store sensitive information such as Server credentials, passwords, Certificates, credit card information
Data in UserDefaults will be lost once you uninstall the app	Data in Keychains will be persisted even after uninstalling the app
Data in UserDefaults can't be accessed from other application	Data in keychains can be accessed from other applications using Keychain Group. If two application are developed from same developer / Organisation.

What is the encryption mechanism used in Keychain Access?

AES-128 encryption

Reference: https://www.apple.com/business/docs/
ios_Security_Guide.pdf



Can we access the data saved in Keychains in another App?

Keychain items can only be shared between apps using Keychain Group if these group of applications are developed from same developer group.

Keychain Access internally uses Application GroupId and Bundle Id to identify the data belongs to an Application.

Keychain internally uses SQLite database to store data.

What is PropertyList?

Property List is a file which is capable of holding data in Key-Value pairs.

The route objects in a Property List are NSArray or NSDictionary.

What is Application Sandbox? How to get the path?

Sandbox is a secure place where the iOS App's resources like preferences, network resources and Database files are saved.

Keeping the resource files in sandbox makes sure that other applications can't have the access to the resources.

```
NSString *documentsPath =
[NSSearchPathForDirectoriesInDomains(NSDocumentDirectory,
NSUserDomainMask, YES) objectAtIndex:0];
```

```
[[[NSFileManager defaultManager] URLsForDirectory:NSDocumentDirectory
inDomains:NSUserDomainMask] lastObject];
```

How to find a file path of App Folder section?

```
NSString *filePath = [[NSBundle mainBundle]
pathForResource:@"TermsAndCondition" ofType:@"html"];
```

What is SQLite?

- SQLite is a Light weight Embedded Relational Database Management System.
- SQLite is developed in C-Language by Richard Hipp in 2000.
- It runs with in the app itself. So it doesn't require extra active server.
- It is very fast and reliable
- It is fully supported by apple and used internally in iOS and MAC OS X
- SQLite Supports Tables, Views, Triggers and Indexes.
- SQLite supports SQL standard queries.
- Use iOS native sqlite.tdb library to integrate SQLite in iOS.
- SQLite follows ACID Properties

How to integrate SQLite in iOS?

These are the steps involved in SQLite Database integration.

```
i. Open Database (sqlite3_open())ii. Prepare Query (sqlite3_prepare_v2())iii. Execute Query (sqlite3_step())iv. Clean the Query (sqlite3_finalise())v. Close Database (sqlite3_close())
```

Is SQLite thread safe?

YES



What is CoreData?

CoreData is a framework to persist the data in the Application.

CoreData manages data in relational entity-attribute modal

is CoreData thread safe?

NO

```
is NSManagedObjectID thread safe? YES
```

```
How to access the instance of NSManagedObject from another thread?
```

Ans: use

NSManagedObjectID *objectID = [managedObject objectID];
to access the NSManagedObject from other threads.

Explain CoreData stack components?

CoreData contains stack of components.

<u>ManagedObject</u>: Managed Object is an Entity (Table) record in CoreData structure

<u>ManagedObjectModal</u>: Managed Object Modal is the Schema of Database. It is logical relationships between Entities.

<u>ManagedObjectContext</u>: ManagedObjectContext is the one where all data base operations are performed. We can say it is a scratch pad. Every application by default contains 1 ManagedObjectContext. We can have multiple MOCs.

PersistenceStore: PersistenceStore is a place where the data is stored.

<u>PersistenceStoreCoordinator</u>: PersistenceStoreCoordinator is the mediator between Persistence store and ManagedObjectContext.

What are the delete rules of CoreData?

There are 4 delete rules:

- NoAction: Deleting the source doesn't have any affect in destination
- **Nullify:** In Inverse relationship, deleting the Source nullify the relationship between Destination to Source.
- Cascade: Deleting the source deletes all the destination records associate to that Source record
- **Deny:** In One-One or One-Many relationship, Deletion of Source is denied by the Destination. (Note: If Destination pointing to Source records).

Ref: https://cocoacasts.com/core-data-relationships-and-delete-rules/

What is NSFetchedResultsController?

You use a fetched results controller to efficiently manage the results returned from a Core Data fetch request to provide data for a UITableView object. or

A controller which is used to efficiently presenting the data returned by the CoreData fetch request on TableView.

```
NSManagedObjectContext *context = [self getManagedObjectContext];
NSFetchRequest *fetchRequest = [[NSFetchRequest alloc] init];
// Configure the request's entity, and optionally its predicate.
NSSortDescriptor *sortDescriptor = [[NSSortDescriptor alloc]
initWithKey:@"<#Sort key#>" ascending:YES];
NSArray *sortDescriptors = [[NSArray alloc]
initWithObjects:sortDescriptor, nil];
[fetchRequest setSortDescriptors:sortDescriptors];
NSFetchedResultsController *controller = [[NSFetchedResultsController
alloc]
initWithFetchRequest:fetchRequest
                                          managedObjectContext:context
                                          sectionNameKeyPath:nil
                                          cacheName:@"<#Cache name#>"];
NSError *error;
BOOL success = [controller performFetch:&error];
```

What are the NSFetchedResultsController Notifications?

- NSManagedObjectContextObjectsDidChangeNotification: this notification is posted when one of the managed objects of the managed object context has changed
- NSManagedObjectContextWillSaveNotification: this notification is posted
 before the managed object context performs a save operation
- NSManagedObjectContextDidSaveNotification: this notification is posted
 after the managed object context performs a save operation

What is NSPredicate?

Predicate is logical conditions used to filter over collection or CoreData Entities

```
Ex:
NSMutableArray *names = [@[@"Nick", @"Ben", @"Adam", @"Melissa"]
mutableCopy];

NSPredicate *bPredicate = [NSPredicate predicateWithFormat:@"SELF
beginswith[c] 'b'"];

NSArray *beginWithB = [names
filteredArrayUsingPredicate:bPredicate]; // @"Ben"
```

What is Parent and Child context relationship?

Process of performing CoreData operations on multiple threads is known as Concurrency.

All UI related core data operations must be performed on Main ManagedObjectContext

Perform Non-UI related core data operation on Child (Private Context) context. Ex: Parsing and Saving the JSON data, Inserting data into core data...

Use NSMainQueueConcurrencyType and NSPrivateQueueConcurrencyType to differentiate Main and Private Contexts

NSMainQueueConcurrencyType is specifically for use with your application interface and can only be used on the main queue of an application.

The NSPrivateQueueConcurrencyType configuration creates its own queue upon initialization and can be used only on that queue. Because the queue is private and internal to the NSManagedObjectContext instance, it can only be accessed through the performBlock: and the performBlockAndWait: methods.

When to use Multiple Managed Object Contexts?

```
NSArray *jsonArray = ...; //JSON data to be imported into
Core Data
NSManagedObjectContext *moc = ...; //Our primary context
the main queue
NSManagedObjectContext *private = [[NSManagedObjectCont
alloc] initWithConcurrencyType:NSPrivateQueueConcurrencyType];
[private setParentContext:moc];
[private performBlock:^{
for (NSDictionary *jsonObject in jsonArray) {
NSManagedObject *mo = ...; //Managed object that matches
incoming JSON structure
//update MO with data from the dictionary
NSError *error = nil;
if (![private save:&error]) {
NSLog(@"Error saving context: %@\n%@", [error localizedDescription], [error userInfo]);
abort();
[moc performBlockAndWait:^{
NSError *error = nil;
if (![moc save:&error]) {
NSLog(@"Error saving context: %@\n%@", [error localizedDescription], [error userInfo]);
abort();
}];
}];
```

Ref: https://code.tutsplus.com/tutorials/core-data-from-scratch-concurrency--cms-22131

What is the Default persistence store in CoreData

SQLite is the default persistence store used in core data.

What are the persistence stores available in CoreData?

```
XML (Slower)
SQLite
Binary
In-Memory
```

What is the extension of CoreDataModal?

.xcdatamodeld is the extension of CoreData modal file.

What is CoreData migration or Versioning?

The process of managing the modified ManagedObjectModal is known as CoreData migration.

Modifications to the ManagedObjectModal could be Adding new entities Adding new attributes Changing the relationships and datatypes of the attributes

CoreData migration are of two types i. Light weight Migration ii. Heavy Migration

Light Weight Migration & Heavy Migration

Perform Light Weight Migration when adding new Entities or Attributes and relationships

Perform Heavy Migration when changing the relationship types or attribute types.

How to perform Light Weight Migration?

```
Select the existing CoreDataModal (xcdatamodeld) ->
Editor -> Add Modal Version
Add changes to the new Modal
Modify the AppDelete.m file persistenceStoreCoordinator
method as follows.
NSDictionary *options = @{
NSMigratePersistentStoresAutomaticallyOption : @(YES),
NSInferMappingModelAutomaticallyOption : @(YES)
};
   if (![_persistentStoreCoordinator
addPersistentStoreWithType:NSSQLiteStoreType configuration:nil
URL:storeURL options:options error:&error]) {
       NSLog(@"Unresolved error %@, %@", error, [error userInfo]);
       abort();
```

How to preload data in to CoreData?



What is Multithreading and what are the advantages?

Multithreading or Concurrency is the process of executing multiple tasks simultaneously.

Advantages:

- i. Proper usage of available resources
- ii. Better Performance
- iii. Faster Execution
- iv. Tasks are executed on background threads so the UI will not be blocked.

How to achieve the multi threading in iOS?

```
Multithreading can be achieved in two ways in iOS. i. Dispatch Queues (Grand Central Dispatch) ii. NSOperationQueue
```

What are the default queues available in GCD?

- GrandCentralDispatch follows the Queue Data Structure.
- It contains 1 Serial Queue and 4 concurrent queues.
- We can also create our own serial and concurrent queues.
- Using the provided Serial and concurrent queues are recommended
- 4 Concurrent queues are differentiated using Priority values.

```
Priority values are:
i. High (DISPATCH_QUEUE_PRIORITY_HIGH)
ii. Default (DISPATCH_QUEUE_PRIORITY_DEFAULT)
iii.Low (DISPATCH_QUEUE_PRIORITY_LOW)
iv. Background(DISPATCH_QUEUE_PRIORITY_BACKGROUND)
```

Dispatch Queues

Serial

Concurrent

Guaranteed for both:

- Tasks follow FIFO (First In First Out). Starting in the same order.
 - Tasks are represented as blocks of code.
- Only 1 task to be executed at a time.
- Serialized access to shared resource.

System Queues:

Only one queue which is the main queue

let mainQueue = dispatch_get_main_queue()

- No guarantee how many tasks are being executing at a time.
- No guarantee about execution time.
- No guarantee about order of execution.

System Queues:

4 queues called global dispatch queues.

- DISPATCH_QUEUE_PRIORITY_HIGH
- DISPATCH QUEUE PRIORITY DEFAULT
- DISPATCH QUEUE PRIORITY LOW
- DISPATCH_QUEUE_PRIORITY_BACKGROUND

They only differ in priority. HIGH is the highest priority and BACKGROUND is the lowest

let queue = dispatch_get_global_queue
 (DISPATCH_QUEUE_PRIORITY_HIGH, 0)

Creating your own queues

You can create any number of serial queues. The first param is a unique label to your queue.

```
let newSerialQueue =
    dispatch_queue_create("com.example.serialQueue",
    DISPATCH_QUEUE_SERIAL)
```

Creating your own queues

You can create any number of concurrent queues but you don't have to as you already have 4 global queues to use.

let newConcurrentQueue =
 dispatch_queue_create("com.example.concurrentQueue",
 DISPATCH_QUEUE_CONCURRENT)

Submitting tasks to queue

dispatch_async(queue) { () -> Void in
 // your task goes here
}

Serial Queue vs Concurrent Queue

Serial queue executes the tasks one after another	Concurrent queue executes the tasks simultaneously
Slower executing	Faster Execution
There is only one serial queue in any application. That is main queue	There are 4 concurrent queues. Those are differentiated by priority levels
Execution starts in order of tasks added to queue	Execution starts in order of tasks added to queue
Order of task completion is predictable (or) Execution completes in the order of tasks added to queue	Order of task completion is non- predictable (or) Execution completion is random.
<pre>dispatch_queue_t serialQueue = dispatch_get_main_queue();</pre>	<pre>dispatch_queue_t concurrentQueue = dispatch_get_global_queue(DISPATCH_QUEUE_PRIO RITY_BACKGROUND, 0);</pre>

How to execute multiple tasks using dispatch queues?

```
-(void)executeTasksConcurrently
   dispatch_queue_t concurrentQueue = dispatch_get_global_queue(DISPATCH_QUEUE_PRIORITY_DEFAULT,
0);
   dispatch_async(concurrentQueue, ^{
       [self taskOne];
   });
   dispatch_async(concurrentQueue, ^{
       [self taskTwo];
   });
}
-(void)taskOne
     for (int i = 0; i \le 1000; i++)
          NSLog(@"%i",i);
}
-(void)taskTwo
{
     for (int i = 10000; i \le 20000; i++)
          NSLog(@"%i",i);
```

What kind of operations should perform on main thread?
UI Operations must be performed on main thread.

Get main queue using

dispatch_queue_t mainQueue = dispatch_get_main_queue();



What is Operation Queue?

- Operation Queues are alternative to Dispatch Queues.
- Operation queues are wrapper around GCD.
- Operation Queues are written in Objective-C. So that we can interact with Operation Queues in Object Oriented fashion
- NSOperationQueue is the class which provides the concurrency mechanism.
- NSOperationQueues are highly asynchronous.
- NSOperation queue doesn't follow FIFO.
- Each task is represented as NSOperation Object.

GCD vs NSOperationQueue

GCD is C-Language API	OperationQueues are Obj-C API
GCD has Serial and Concurrent Queues	Operation Queues are highly asynchronous
Operations/tasks can be executed serially using serial queues	<pre>Use [operationQueue setMaxConcurrentOperationCount:1]; to execute tasks serially using OperationQueues</pre>
Tasks can not be cancelled once it starts execution	cancel, pause and resume operations can be performed on OperationQueues
We can not set dependences between tasks	We can add dependencies between tasks
GCD is faster when compared to Operation Queues	Use NSOperationQueues when you want to set the dependencies and perform cancel, pause and resume operations

How to execute multiple tasks using Operation queues?

```
-(void)executeTasksConcurrentlyUsingOperationQueues
{
    NSOperationQueue *operationQueue = [[NSOperationQueue alloc] init];
    NSInvocationOperation *opetaionOne = [[NSInvocationOperation alloc]
initWithTarget:self selector:@selector(taskOne) object:nil];
    NSInvocationOperation *opetaionTwo = [[NSInvocationOperation alloc]
initWithTarget:self selector:@selector(taskTwo) object:nil];
    [operationQueue addOperation:opetaionOne];
    [operationQueue addOperation:opetaionTwo];
-(void)taskOne
    for (int i = 0; i \le 1000; i++)
        NSLog(@"%i",i);
-(void)taskTwo
    for (int i = 10000; i \le 20000; i++)
        NSLog(@"%i",i);
}
```

NSOperation

NSOperation is an **abstract class**, so that you can't directly create instances of NSOperation.

It has two sub classes:

NSBlockOperation – Use this class to initiate operation with one or more **blocks**. The operation itself can contain more than one block and the operation will be considered as finish when all blocks are executed.

NSInvocationOperation – Use this class to initiate an operation that consists of invoking a **selector** on a specified object.

How & When to sub class NSOperation class?

Subclass NSOperation when you are performing concurrency operation.



Ref: https://developer.apple.com/reference/foundation/operation

When to use GCD over OperationQueue?

GCD is faster compare to Operation Queues.
Use Operation Queues when you are not able to achieve the functionality using GCD such as
i. Setting Dependencies

ii. Cancel, pause and resume operations



What is Design Pattern? What are the design patterns used in iOS?

Modal View Controller Modal View View Modal Singleton Delegation Observer Target-Action Factory



What is singleton? How to create a singleton object?

Singleton is the process of making sure only one instance of a class existing in entire application life cycle.

Advantage: Data Sharing across application is very easy.

```
Syntax:
+(User*)sharedUser
{
    static User *user = nil;
    static dispatch_once_t onceToken;
    dispatch_once(&onceToken, ^{
        user = [[User alloc] init];
    });
    return user;
}
```

Exiting Examples:

```
[UIApplication sharedApplication];
[NSNotificationCenter defaultCenter];
[NSFileManager defaultManager];
[NSURLSession sharedSession];
```

What is Delegation? How to achieve it?

Delegation is one of the design patterns.

Delegation is process of working an object as helper to another object.

Delegation can be achieved using Protocol



Can Delegate be retained?

Delegate can't be retained.

Retain Cycle occurs if Delegate is retained.

Use weak property attribute to avoid retain cycle as follows.

Ex:

```
@property (nonatomic, weak) id<DelegateName> delegate;
@property (nonatomic, assign) id<UITableViewDelegate> delegate;
```

What is NSNotificationCenter?

NSNotificationCenter is a process of Broadcasting a message to multiple receivers.

Use NSNotificationCenter when you want to notify an event to multiple objects.

NSNotificationCenter follows Observer design pattern.

What is Block? Write Syntax?

```
Syntax:
ReturnType (^blockName)(Datatype, Datatype,...) = ^(Datatype
agr1, Datatype arg2...,)
   //Block Implementation
// Block Calling
blockName(valueOne, value2..,);
Ex:
void (^simpelBlock)() = ^{
    NSLog(@"This is simple block with no input args and no
output");
};
void (^imageDownload)(NSData*, NSError*) = ^(NSData*
imageData, NSError* error)
    if (error){
        NSLog(@"Failed to download image");
    }else
        NSLog(@"Image downloaded successfully");
};
// Calling a block
imageDownload(imageData, nil);
```

How to use self in block?

```
Don't use self directly in block.
Using self in block occurs retain cycle.
Use __weak self in block to avoid retain cycle.

_weak ViewController *weakSelf = self;
void (^aBlock)(NSData*, NSError*) = ^(NSData* data, NSError*
*Error)
{
    [weakSelf someMethod];
};
```

What is __block keyword?

__block is different kind of storage class like auto, extern etc...

Primitive datatype variables can't be changed within the block, use __block to allow modification of variable within the block.

```
__block int aVar = 10;
void (^Sample)(void) = ^{
    aVar = 20;
};
```

Ref: http://stackoverflow.com/a/7081005

Delegation Vs NotificationCenter Vs Blocks

Delegation is one to one communication	NotificationCenter is one to many communication	Blocks are alternative
	Use Notification Center to notify many objects about some event	laniect analit samel
	NotificationCenter uses Observer design pattern	

What is Web service?

It is a way of exchanging the information over network.

SOAP vs REST

Simple Object Access Protocol	Representational state transfer protocol
SOAP sends data in only XML format	REST can send data in JSON, XML or even Text format
SOAP a bit heavy weight	REST is light weight
Data received for SOAP request will be only XML	Data received for REST request will be JSON or XML
SOAP is not best suitable for Mobile Apps since it is heavy weight	REST is best suitable for Mobile Apps since it is light weight

NSURLSession vs NSURLConnection

NSURLSession supports Background uploads and downloads	NSURLConnection doesn't support Background uploads and Downloads
NSURLSession is capable of submitting multiple requests on one shared session	NSURLConnection creates different individual session for each request
NSURLSession supports private browsing	NSURLConnection doesn't support private browsing

Reach me @iPhoneDev1990@gmail.com

How to check the response is XML or JSON from Web Service?

Check the Content-Type attribute of response object to identify the response type xml or JSON.



How to parse the XML data?

NSXMLParser and NSXMLParserDelegate to parse the XML Data. NSXMLParser *xmlParser = [[NSXMLParser alloc] initWithData:xmlData]; xmlParser.delegate = self; // NSXMLParserDelegate [xmlParser parse]; //Implement NSXMLParserDelegate methods - (void)parserDidStartDocument:(NSXMLParser *)parser; - (void)parser:(NSXMLParser *)parser didStartElement:(NSString *)elementName namespaceURI:(nullable NSString *)namespaceURI qualifiedName:(nullable NSString *)qName attributes:(NSDictionary<NSString *, NSString *> *)attributeDict; - (void)parser:(NSXMLParser *)parser foundCharacters:(NSString *)string;

- (void)parser:(NSXMLParser *)parser didEndElement:(NSString *)elementName
 namespaceURI:(nullable NSString *)namespaceURI qualifiedName:(nullable
 NSString *)qName;
- (void)parserDidEndDocument:(NSXMLParser *)parser;
- (void)parser:(NSXMLParser *)parser parseErrorOccurred:(NSError *)parseError;

How to parse JSON Data?

```
NSDictionary *jsonResponse = [NSJSONSerialization
JSONObjectWithData:jsonData
options:NSJSONReadingAllowFragments error:error];
```

How to cache the responses of Web services?

```
[NSMutableURLRequest
requestWithURL:[NSURL URLWithString:@"Some URL"]
cachePolicy:NSURLRequestReturnCacheDataElseLoad
timeoutInterval:60];
```



How to provide security for the data transmitted over network?

AppTransportSecurity + HTTPS(SSL/TLS)
OAuth2.0

What are the Encryption mechanisms available for iOS?

base64encoding

AES-128

AES-256

SHA-1



How to provide security for the data saved locally?

Keychain Access CoreData SQLCipher



What is lazy loading?

Lazy loading is the process of creating objects when the object is needed exactly instead of creating all at once.

Synchronous request vs Asynchronous request

Only one request can be sent at given point of time. You need to wait until you get the response for the first request	You can send multiple request continuously without waiting for the responses
Order of execution is predictable	Order of execution is not predictable

How to create Custom Delegate?

TODO



What are the advantages of blocks?

TODO

Delegate vs Blocks

TODO

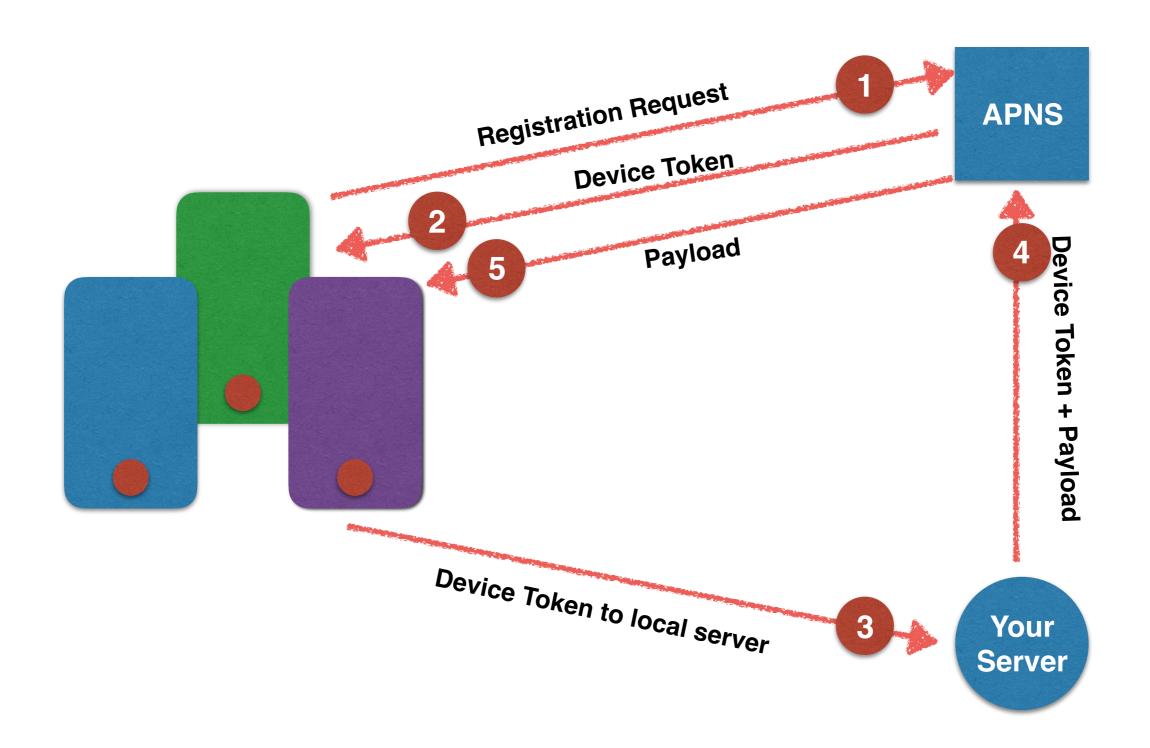
What is PushNotification?

PushNotification is the process alerting user by sending some information(notification) to the device from the APNs(Apple Push Notification service).

How the push notification works?

- i. App(Device) sends a registration request to APNs.
- ii. APNs sends a DeviceToken(64 bit) to the App(Device).
- iii. Send the Device Token to your Local Server.
- iv. Local server sends a payload (message) along with Device Token to APNs.
- v. APNs identifies the Device based on the DeviceToken and pushes the Message to the App(Device).

What is PushNotification Architecture?



What are the steps involved in setup APNs (till iOS9)?

- 1. Generate Push Enabled Certificates and provisioning profile.
- 2. Install them on the system
- 3. Request for authorisation from user for notification types (Alerts, Badges and Sounds)

```
UIUserNotificationType types = (UIUserNotificationTypeAlert | UIUserNotificationTypeBadge |
UIUserNotificationTypeSound);
UIUserNotificationSettings *settings = [UIUserNotificationSettings settingsForTypes:types
categories:nil];
[[UIApplication sharedApplication] registerUserNotificationSettings:settings];
4. Implement required AppDelegate protocol methods
-(void)application:(UIApplication *)application didRegisterUserNotificationSettings:
(UIUserNotificationSettings *)notificationSettings
    if (notificationSettings.types != UIUserNotificationTypeNone)
        [[UIApplication sharedApplication] registerForRemoteNotifications];
-(void)application:(UIApplication *)application
didRegisterForRemoteNotificationsWithDeviceToken:(NSData *)deviceToken
{
   NSLog(@"Device Token Received: Send it to your server");
-(void)application:(UIApplication *)application didReceiveRemoteNotification:(NSDictionary
*)userInfo fetchCompletionHandler:(void (^)(UIBackgroundFetchResult))completionHandler
{
   NSLog(@"Payload: %@", userInfo);
}
```

What are the steps involved in setup APNs (in iOS 10)?

- 1. Generate Push Enabled Certificates and provisioning profile.
- 2. Install them on the system
- 3. Enable Push Notifications in Capabilities
- 4. Add UserNotification.framework to your project
- 5. Instantiate UNUserNotificationCenter object and set Delegate.
- 5. Request for authorisation from user for notification types (Alerts, Badges and Sounds)
- 6. Register for Remote Notification on appDelegate Object.

```
UNUserNotificationCenter *center = [UNUserNotificationCenter currentNotificationCenter];
center.delegate = self;
[center requestAuthorizationWithOptions:(UNAuthorizationOptionAlert |
UNAuthorizationOptionSound | UNAuthorizationOptionBadge) completionHandler:^(BOOL
granted, NSError * _Nullable error) {
    if (granted)
    [[UIApplication sharedApplication] registerForRemoteNotifications];
}];
```

7. Implement required UIApplicationDelegate and UNUserNotificationCenterDelegate methods as in next slide.

```
-(void)application:(UIApplication *)application
didRegisterForRemoteNotificationsWithDeviceToken:(NSData *)deviceToken
   NSLog(@"Device Token Received: Send it to your server");
//Called when a notification is delivered to a foreground app.
-(void)userNotificationCenter:(UNUserNotificationCenter *)center
willPresentNotification:(UNNotification *)notification withCompletionHandler:
(void (^)(UNNotificationPresentationOptions options))completionHandler{
    NSLog(@"User Info : %@", notification.request.content.userInfo);
    completionHandler(UNAuthorizationOptionSound | UNAuthorizationOptionAlert
 UNAuthorizationOptionBadge);
}
//Called to let your app know which action was selected by the
user for a given notification.
-(void)userNotificationCenter:(UNUserNotificationCenter *)center
didReceiveNotificationResponse:(UNNotificationResponse *)response
withCompletionHandler:(void(^)())completionHandler{
    NSLog(@"User Info : %@",response.notification.request.content.userInfo);
    completionHandler();
```

What is Silent Notification?

Silent notifications improve the user experience by helping you keep your app up-to-date, even when it is not running. Silent notifications give you a way to wake up your app periodically so that it can refresh its data in the background.

Advantage: When apps do not run for extended periods of time, their data can become outdated. When the user finally launches the app again, the outdated data must be replaced, which can cause a delay in using the app.

The sending of a silent notification requires a special configuration of the notification's payload. If your payload is not configured properly, the notification might be displayed to the user instead of being delivered to your app in the background. In your payload, make sure the following conditions are true:

The payload's **aps** dictionary must include the **content-available** key with a value of **1**.

The payload's aps dictionary must not contain the alert, sound, or badge keys.

NOTE: Enable Remote Notifications in Background Mode settings.

What is the structure of the Push Notification Payload?

```
Sample 1:
  "aps": { "alert": "Message received from Bob"},
  "customData" : [ "bang", "whiz" ]
Sample 2:
  "aps" : {
    "alert" : {
       "title": "Game Request",
       "body": "Bob wants to play poker",
       "action-loc-key": "PLAY"
    "badge" : 5
  },
  "acme1": "bar",
  "acme2" : [ "bang", "whiz" ]
```

What is the structure of the Push Notification Payload?

```
Sample 3:
  "aps" : {
     "alert": "You got your emails.",
    "badge" : 9,
     "sound": "bingbong.aiff"
  "acme1": "bar",
  "acme2": 42
Silent Notification Payload:
  "aps" : {
    "content-available": 1
  "acme1": "bar",
  "acme2": 42
```

What is the payload size of Push notification?

Prior to iOS 8: 256 bytes

iOS 8: 2KB

iOS 9+: 4KB



Q: What is the lifetime of undelivered Notifications?

Ans: 28 Days

Q: What is the Queue name used by APNs?

Ans: QoS(Quality of Service) queue.

Q: What happens if the device is offline when sending the pushNotification?

Ans: APNs maintains the Queue to store the undelivered Notification

Q: Does QoS retains all undelivered Notifications?

Ans: No, QoS retains only the latest message for an application.

Q: What is the protocol used by the APNs to send Push Notifications?

Ans: HTTP/2

Q: What is the security Mechanism used in APNs?

Ans: TLS (Transport Layer Security)

What are the best practices to use APNs?

- Do not send sensitive information
- Do not completely depend on APNs (Push Delivery is not guarantee)

What are the security best practices in iOS?

- Store sensitive information on Keychains
- Don't store the credentials on Disk
- Use HTTPS protocols to connecting servers.
- Use custom encryption to store the data
- Clear all non-required information on logout

How to get the current date of the device?

NSDate *currentDate = [NSDate date];

How to find device metadata?

```
NSLog(@"Name: %@", [UIDevice currentDevice].name);
//Dev's iPhone

NSLog(@"Modal: %@", [UIDevice currentDevice].model);
//iPhone

NSLog(@"OS: %@", [UIDevice currentDevice].systemName);
// iPhone OS

NSLog(@"OS Version%@", [UIDevice currentDevice].systemVersion);
// Version 9.3
```

What is UDID?

UDID is a universal device identifier that is unique for device. It is a 32 big alpha numeric string

How to find device orientation?

[[UIDevice currentDevice] orientation];

Xcode 7 vs Xcode 8

Having only 5 templates	Having 7 templates Sticker Pack Application & iMessage Application are new
Uses swift 2	Uses Swift 3
Supports iOS9 and till iPhone 6s and 6s Plus	Supports iOS10 and iPhone 7 and 7 Plus
Doesn't support Autoresizing	Supports Autoresizing and Autolayout
3 size classes	2 Size classes
No configuration Pane	Device configuration pane is added

Reach me @iPhoneDev1990@gmail.com

What is the framework used to write unit test cases?



How to prepare an ipa?

Product -> Archive

How to share the app with clients?

- i. iTunes Connect (Test flight)
- ii. Airwatch
- iii. Betabuilder
- iv. Test fairy



What is the differences between Ad-hoc vs Appstore distribution?



What is an Enterprise app? How to distribute it?

How to distribute an iOS app to AppStore?



How to create static library in iOS?

How to create Framework in iOS?

What are your roles and responsibilities?

Assigning the tasks to Junior members in a team
Guiding junior members of a Team
Updating the JIRA
Attending Scrum calls
Picking up JIRA items
Involving in Coding for providing better solutions
Delivering the build to QA and Client
Version Control Management

What are the challenges you faced?

Handling the applications on our Own Handling multiple application at same point of time within given deadline

Maintaining compatibility when OS Version changes Handling Bad network conditions
Getting successes in Appstore Approval Process
Achieving the Designs which are not Apple friendly

What is the very interesting feature you have developed?



What is Stack, Queue and LinkedList Data structures?



What makes app rejection under review?



How the Scrum calls happen?



What are the different sensors available in iPhone?

Accelerometer
Ambient light sensor
Barometer
Compass
Gyroscope
Pressure Sensitive display
Proximity sensor

How to improve the performance of your iOS App?

Caching

Performing heavy tasks on Background threads

Lazy loading

Pre dumping the static data

Using proper sized images

Avoid using heavy weight XIB files

Using compression techniques when transmitting the data over

network

Using proper collections for data processing

Reusing the expensive objects and reuseIdentifiers

Use autorelease pools when dealing with iterative objects allocations

How to optimise the iOS App memory usage

Eliminate Memory Leaks
Use lazy loading
Use SQLite or CoreData for larger data



Why iOS App size is bigger then Android app?



How to achieve 3D Touch?

What is the workflow you follow to design Apps?



How to save the data in iCloud?



What is InAppPurchase? How to achieve it?



What is Apple Pay? How to achieve it in our Applications?



What is iOS App Extension?

What is iBeacons?

What is responder chain?

How do you identify the crashes of production App?

Crashlytics



What are the crash reporting frameworks available?

Crashlytics
Apphance
BugSense
Crittercism
TestFlight . . .



How to identify the route cause of a crash from crash log?



What are the app analytical frameworks you know?

Flurry
Google Analytics
Omniture
Localytics



Enterprise App Process

Each team member should have his or her own Apple ID and development certificate. Don't share signing identities between team members

certificate expires for 3 years and provisioning profile expires in 1 year If you don't distribute your app using a Mobile Device Management (MDM) system, users can install your app using the iOS App file but then they will need to manually trust your organization to launch the app

What are the Frameworks you have used?

UIKit Foundation MapKit AVFoundation MessageUI AddressBOOK **AddressBOOKUI** CoreData CoreLocation MediaPlayer StoreKit **Social UserNotifications SystemConfiguration**



How to check the internet connection status?

Use Apple provided Reachability class to check the status of the network



How to get notified the Reachability Status change?

Register for kReachabilityChangedNotification to notify the reachability status changes.



What are the 3rd party Frameworks / Libraries you have used?

The frameworks which you have used which are not provided by the iOS SDK.

- CorePlot (for Graphs design)
- AFNetworking (Web Services)
- ASIHTTPRequest (Web Services)
- YTVideoPlayer (Playing youtube Videos)
- SDWebImage (Downloading images)
- OCMock
- Facebook
- Twitter
- LinkedIn
- Swinject
- SwiftyJSON
- Swiftlint

What is Cocoapods and what are the alternatives?

Cocoapods is a Dependency Manager where all dependent libraries or frameworks are listed and those frameworks are linked dynamically.

Alternatives:

- i. Carthage
- ii. Swift Package Manager (SPM)

What is App thinning?

App thinning is a process of downloading the app with required resources of the app based on the device instead of downloading other devices model resources.

- i. Use Assets Catalog
- ii. Enable Bit Code

iii. Enable On Demand Resources (Download different sets of images based on the game levels)

What is App Slicing?

Slicing is the process of creating and delivering variants of the app bundle for different target devices.



App Icons sizes

	Device Resolution	App Icons	Spot Light	Settings	Launch Screens
iPhone Xs Max	(414x896)*3				
iPhone Xs	(375x812)*3				
iPhone XR	(375x667)*2				
iPhone X	(375x812)*3				
iPhone 6+, 6S+, 7+	(414x736)*3	180x180 or (60x60)*3	120x120 (40x40)*3	87x87 (29x29)*3	1080x1920
iPhone 6, 6s, 7	(375x667)*2	120x120 (60x60)*2	80x80 (40x40)*2	58x58 (29x29)*2	750x1334
iPhone 5, 5C, 5S, SE	(320x568)x2	120x120 (60x60)*2	80x80 (40x40)*2	58x58 (29x29)*2	640x1136
iPhone 4,4S	(320x480)*2	120x120 (60x60)*2	80x80 (40x40)*2	58x58 (29x29)*2	640x960
AppStore		1024x1024			

Reach me @iPhoneDev1990@gmail.com

What are the iOS Application Design considerations?

Ref: http://www.sourcefuse.com/mobile-app-development-considerations/

- 1. Usability (Native or Hybrid)
- 2. Continuous improvement (Analytics and Crash Handling)
- 3. Performance and Scalability
- 4. Maintainability(Proper Documentation)
- 5. Engagement (Support and Notifications)
- 6. Security (Permissions required by app, Local data Security and data over network security)
- 7. Following the Apple's User Interface and Coding Guide lines

How Native apps are different from Hybrid Apps



How do you achieve Single Sign on (SSO) in iOS?

Using Keychain Access Group we can achieve SSO in iOS.



Ref: https://www.captechconsulting.com/blogs/a-single-sign-on-pattern-for-enterprise-ios-applications

How iOS is very secure?

iOS Apps are very secure due to following factors:

- Secure Booting
- Passcode: earlier 4 Digits now 6 digits
- Touch ID: Touch ID information stored in Secure Enclave. No one can access it
- Encryption: Uses AES256 for Passcode and Touch ID
- **Keychains:** Database for storing Secure information and to access from multiple applications which are developed by same Developer Group or Organisation
- App Security: Every application must be signed with Apple-issued Certificate
- Network Security: Transport Layer Security(TLS)
- Two Factor Authentication: Apple ID + Password & OTP Verification when logging in in another devices.

Ref: https://en.wikipedia.org/wiki/IOS#Security

How to implement Touch ID Authentication?

Framework: Local Authentication

Given Access to 3rd party Apps from: iOS8



How to implement Touch ID Authentication?

```
let context = LAContext()
var error: NSError?
if context.canEvaluatePolicy(
    LAPolicy.deviceOwnerAuthenticationWithBiometrics,
    error: &error) {
    // TouchID is available on the device
} else {
    // TouchID is not available on the device
    // No scanner or user has not set up TouchID
}
```

Algorithms and Programming

Ref: http://javarevisited.blogspot.in/2015/06/top-20-array-interview-questions-and-answers.html

Write a program to find the prime numbers between 1-N numbers?

Hint: A number which has only two divisible factors (i.e 1 and itself)

```
void primeNumbersBetween1toN()
{
    int n;
    printf("Enter value for n:");
    scanf("%i", &n);
    for (int i = 2; i<=n; i++)
        BOOL isPrime = YES;
        for (int j = 2; j \le i/2; j++)
            if (i % j == 0)
                isPrime = NO;
                continue;
           (isPrime) {
            printf("%i is prime number \n", i);
```

```
Write a Program to find out the factorial of a given number?
Ex: 5! = 5*4*3*2*1 = 120
void factorialOfAGivenNumber()
    int n;
    printf("Enter value for n:");
    scanf("%i", &n);
    int fact = 1;
    for (int i = 1; i <= n; i++)
         fact *= i;
    printf("Factorial is: %i", fact);
```

Check the given number is Armstrong number or not?

```
void isArmstrongNumber(int aNumber)
    int reminder = 0;
    int finalNumber = 0;
    int temp = aNumber;
    while (temp != 0)
        reminder = temp % 10;
        temp = temp / 10;
        finalNumber = finalNumber + reminder * reminder * reminder;
    }
    if (aNumber == finalNumber) {
        printf("%i is armstrong number.", aNumber);
    }else
        printf("%i is not an armstrong number.", aNumber);
Ex: 153 , 370, 371, 407
```

```
Write a program reverse a number?
                     Ex: 123 ==> 321
void reverseANumber(int aNumber)
    int reversedNumber = 0;
    int reminder = 0;
    while (aNumber)
        reminder = aNumber % 10;
        reversedNumber = reversedNumber * 10 +
reminder;
        aNumber = aNumber / 10;
    printf("Reversed number is: %i", reversedNumber);
```

Write a program to display fibonacci series? Hint: 0,1,1,2,3,5,8,13 etc

```
void displayFibonacci(int aNumber)
    int previous = 1, lastPrevious = 0;
    int current = 0;
    for (int i = 0; i<= aNumber; i++)</pre>
        if (i <= 1)
           current = i
        else
            current = previous + lastPrevious;
            lastPrevious = previous;
            previous = current;
        printf("%i", current);
```

How to find out missing elements between 1 to 100?

Calculate sum of all elements in an Array and compare the sum with sum of 100 numbers.

int missingElement == 100(100-1)/2 - sum of Array elements;



Write a program check the given element is in an Sorted Array or not in Efficient way?

What is the complexity of Binary Search?

Ans: O(logN)

What is the time complexity for sequential search?

Ans: O(n)

How to find largest and smallest elements in an Array?

Sort and pick the top and bottom elements



Write a program to find pairs whose sum is equal to given number?

```
void printPairsWhoseSumIsGiven()
    int givenNumber = 10;
    int a[10] = \{1,-2,7,6,-12,9,-5,5,4,3\};
    for (int i = 0; i \le 9; i + + )
        for (int j = i; j \le 9; j++)
            if (a[i] + a[j] == givenNumber)
                 printf("(%i, %i)", a[i], a[j]);
```

How to find out the largest difference in an Array?

```
int a[] = \{1,2,3,4,5,6,7,8,9,10,1,2,3,4,5,6,7,8,9,10\};
int rightLargest = 0, leftSmallest = 0;
int size = sizeof(a) / 4;
int diff = 0;
int max = 0;
for (int i = size; i >= 0; i--)
    if (a[i] > max)
        max = a[i];
    if ((max - a[i]) > diff)
        diff = max - a[i];
    NSLog(@"Diff is: %i", diff);
```

Remove duplicate elements in an Array

```
void removeDuplicateElementsInAnArray()
    int a[SIZE] = \{1,2,33,5,33,4,1,2\}; // 1,2,33,5,4
    int n = 8;
    for (int i = 0; i \le n; i++)
    {
        for (int j = i+1; j < n; j++)
            if (a[i] == a[j])
                NSLog(@"Found duplicate element: %i", a[i]);
                for (int k = j; k < n-1; k++)
                    a[k] = a[k+1];
                n--;
    }
    for (int i = 0; i < n; i++)
    {
        NSLog(@"%i", a[i]);
```

How to find out second largest element in an Array without sorting?

```
int a[] = \{1,2,3,4,1,-1,5\};
int size = sizeof(a) / sizeof(int);
int l = a[0];
int sl = 0;
for (int i = 2; i < size; i++)
    if (a[i] > l)
        sl = l;
        l = a[i];
    else if (a[i] < sl)</pre>
        sl = a[i];
printf("Largest: %i", l);
printf("Second Largest: %i", sl);
```

How to find contiguous sub array with maximum sum in an array of positive and negative number?

TODO



How to merge sorted array?

```
void mergeTwoSortedArrays()
    int a[] = \{1,3,5,6\};
    int b[] = \{2,3,4,\};
    int s1 = sizeof(a) / sizeof(int);
    int s2 = sizeof(b) / sizeof(int);
    int c[s1+s2];
    // {1,2,3}
    int k = 0;
    int j = 0;
    if (s1 >= s2)
        int s = (s1>s2)?s1:s2;
        for (int i = 0; i < s;)
            if (a[i] < b[j])
                c[k] = a[i];
                k++;
                i++;
            else
                c[k] = b[j];
                k++;
                j++;
            if(j==s2)
                c[k]=a[i];
                i++;
                k++;
        }
        for (int i = 0; i < 7; i++)
            printf("%i \n", c[i]);
```

Thanks