200+

Interview Q & A in

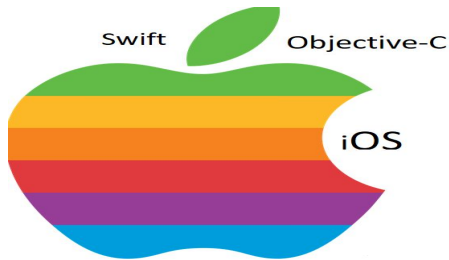# iOS Development

Objective-C

Swift

Programming

By Bandana Ojha

Frequently Asked Interview Q & A in iOS Development

Swift and Objective-C Programming

By Bandana Ojha

Introduction

The authors of this book " Frequently asked Interview Q & A in iOS development" have conducted many iOS development interviews at various companies and meticulously collected the most effective swift and objective- C programming interview notes with simple, straightforward explanations. Rather than going through comprehensive, textbook-sized swift and objective-c reference guides, this book contains only the information required immediately for iOS development interview to start their career as iOS developer. Answers of all the questions are short and to the point. This book contains 200+ questions and answers and we assure that you will get 90% frequently asked interview questions in Swift and objective c programming language , going through this book. It will clear your fundamentals, concepts and boost your confidence to appear any iOS development interview in any companies anywhere in the world whether it is telephonic or face to face.

Wishing good luck to all my readers!!!

Check out our other interview Q & A series. Like Interview Questions & Answers in Manual Testing, Selenium Testing, Mobile Testing, Core Java programming, Advance Java programming, J2EE programming, Python programming and Scala programming.

Please check this out:

Our other best-selling books

500+ Java & J2EE Interview Questions & Answers-Java & J2EE Programming

200+ Frequently Asked Interview Questions & Answers in iOS Development

200 + Frequently Asked Interview Q & A in SQL , PL/SQL, Database Development & Administration

100+ Frequently Asked Interview Questions & Answers in Scala

100+ Frequently Asked Interview Q & A in Swift Programming

100+ Frequently Asked Interview Q & A in Python Programming

100+ Frequently Asked Interview Questions & Answers in Android Development

100+ most Frequently Asked Interview Questions & Answers in Manual Testing

Frequently asked Interview Q & A in Java programming

Frequently Asked Interview Questions & Answers in J2EE

Frequently asked Interview Q & A in Mobile Testing

Frequently asked Interview Q & A in Test Automation-Selenium Testing

****

1. Explain about Swift programming language?

Swift is a general-purpose, multi-paradigm, compiled programming language developed by Apple Inc. for iOS, macOS, watchOS, tvOS, and Linux. Swift is designed to work with Apple's Cocoa and Cocoa Touch frameworks.

2. What are the advantages of Swift over Objective-C?

Following are the are the advantages of Swift over Objective-C

Swift is easier to read.

Swift is easier to maintain.

Swift is safer.

Swift is unified with memory management.

Swift requires less code.

Swift is faster.

Fewer name collisions with open source projects.

Swift support dynamic libraries.

Swift Playgrounds encourages interactive coding.

3.Explain Swift's pattern matching techniques.

Following are the swift's pattern matching techniques

Tuple patterns are used to match values of corresponding tuple types.

Type-casting patterns allow you to cast or match types.

Wildcard patterns match and ignore any kind and type of value.

Optional patterns are used to match optional values.

Enumeration case patterns match cases of existing enumeration types.

Expression patterns allow you to compare a given value against a given expression.

4. How can you define a base class in swift?

The classes are not inherited directly from the base class in the Swift. The classes defined by the developer without specifying the superclass becomes the base class automatically.

5. What is Dictionary in Swift?

Dictionary enables you to store the key-value pairs and access the value by providing the key. It is like that of the hash tables in other programming languages.

6. What is Code Coverage ?

Code coverage is a metric that helps us to measure the value of our unit tests.

7. Is Swift Object-Oriented Programming language?

Yes, Swift is an Object-Oriented Programming language.

8. What is Internal access ?

Internal access enables entities to be used within any source file from their defining module, but not in any source file outside of the module.

Internal access is the default level of access. So even though we haven't been writing any access control specifiers in our code, our code has been at an internal level by default.

9. What Is the Significance Of "?" In Swift?

The question mark (?) is used during the declaration of a property. If the property does not hold a value, the question mark (?) helps to avoiding application errors.

10 Explain what is parsing and mention which class can we use for passing of XML in iPhone?

Parsing is the process referred to access the data in the XML element. We can use class "NSXML" parser for passing XML in iPhone.

11. What are the control transfer statements in swift?

The control transfer statements in swift are:

Continue

Break

Fallthrough

Return

12. **Explain what is #import?**

#import is a C pre-processor construct to avoid multiple inclusions of the same file.

13. What is pair programming?

Pair programming is a tool to share information with junior developers. Junior and senior developer sitting side-by-side this is the best way for the junior to learn from senior developers.

14.What are blocks?

Blocks are a way of defining a single task or unit of behavior without having to write an entire Objective-C class. they are anonymous functions.

15. Why do we use availability attributes ?

Apple wants to support one system version back Availability Attributes helps us to support previous version iOS.

16. What is a category and when is it used ?

A category is a way of adding additional methods to a class without extending it. It is often used to add a collection of related methods. A common use case is to add additional methods to built-in classes in the Cocoa frameworks.

17. Difference between frame and bounds ?

The bounds of an UIView is the rectangle, expressed as a location (x,y) and size (width, height) relative to its own coordinate system (0,0).

The frame of an UIView is the rectangle, expressed as a location (x,y) and size (width, height) relative to the superview it is contained within.

18. What is MVC ?

MVC is a design pattern that stands for model view controller, this design pattern separates the data from its display, mediated by a View Controller.

19.What is Deep Linking?

Deep linking is a way to pass data to your application from any platform like, website or any other application. By tapping once on link, you can pass necessary data to your application.

20.What is Polymorphism?

Polymorphism is the ability of a class instance to be substituted by a class instance of one of its subclasses.

21. Explain Functions?

It lets you group a series all together to perform a specific task. A function once created can be used over and over in the code, and in case you find a repeating statement in the code, then the function can be the answer to avoid the repetition.

22. What is the difference between let and var in swift?

Let and var are used to define constants and variables in swift. Let is used to make a constant and var is used to create a variable. The value of constant need not be declared before compilation. It doesn't need to be known at compile time as well. The value can be assigned any time but only once and this value can be used in many places.

23. Explain what is an accessor method?

Accessor methods are methods belonging to a class that enables you to get and set the values of instance valuable contained within the class.

24. Explain what is the use of category in Objective-C?

The use of category in Objective-C is to extend an existing class by appending behavior that is useful only in certain situations. To add such extension to existing classes, objective –C provides extensions

and categories. The syntax used to define a category
is *@interface* keyword.

25. How can you make a property optional in Swift?

Declaring a question mark "?" in the code can make a property optional.

If a property doesn't hold a value, then this symbol "?" helps in avoiding the runtime errors.

26. **Mention which class are used to establish a connection between applications to the web server?**

The class used to establish connection between applications to web server are

NSURL

NSURL REQUEST

NSURL CONNECTION

27. **Explain class definition in Objective-C?**

A class definition begins with the keyword *@interface* followed by the interface (class) name, and the class body, closed by a pair of curly braces. In Objective-C, all classed are retrieved from the base class called *NSObject.* It gives basic methods like memory allocation and initialization.

29. Explain any three-shift pattern matching techniques?

Typecasting Patterns – This pattern allows you to match or cast the types.

Wildcard Patterns – This pattern matches as well as ignores any kind and type of value.

Optional Patterns – This pattern is used to match the optional values

30. How can you write a multiple line command Swift?

The multiple line comment is written in between the symbols (/*) at the start and (*/) at the end.

31.What is the disadvantage to hard-coding log statements ?

First, when you start to log, it starts to accumulate. If you hardcode something on your code it will completely "destroy" the portability of your code in a great extent. Even with a platform independent language you will not able to say, "Compile once, Run anywhere".

Second, each time we add one to the code base, we take a risk of injecting new bugs into our code.

32. What is Pointer ?

A pointer is a direct reference to a memory address. Whereas a variable act as a transparent container for a value, pointers remove a layer of abstraction and let you see how that value is stored.

33.What's the difference between using a delegate and notification?

Both are used for sending values and messages to the interested parties. A delegate is for one-to-one communication and is a pattern promoted by Apple. In delegation, the class raising events will have a property for the delegate and will typically expect it to implement some protocol. The delegating class can then call the delegates protocol methods.

Notification allows a class to broadcast events across the entire application to any interested parties. The broadcasting class doesn't need to know anything about the listeners for this event, therefore notification is very useful in helping to decouple components in an application.

34. What is the difference between the function calls and messages?

The difference between function call and message is that a function and its arguments are linked together in the compiled code, but a message and a receiving object are not linked until the program is executing and the message is sent.

35.How many UITableViewCells are allocated when you first load a UITableView? How many additional ones are allocated as you scroll through the table?

A UITableView will normally allocate just enough UITableViewCell objects to display the content visible in the table. Because of the reuseIdentifier, the UITableView will not allocate new UITableViewCell objects for each new item that scrolls into view, avoiding laggy animations.

36. Mention whether NS object is a parent class or derived class?

NS object is the parent class and consists of several instance variables and instance methods.

37. What is collection in Cocoa and Cocoa Touch?

In Cocoa and Cocoa Touch, a collection is a Foundation framework class used for storing and managing groups of objects. Its primary role is to store objects in the form of either an array, a dictionary, or a set.

38. What is delegate?

Delegation is a commonly used pattern in object-oriented programming. It is a situation where an object, instead of performing a task itself, delegates that task to another, helper object. The helper object is called the delegate.

A delegate allows one object to send messages to another object when an event happens.

A delegate is just an object that another object sends messages to when certain things happen, so that the delegate can handle app-specific details the original object wasn't designed for. It's a way of customizing behavior without sub classing.

39. What is auto-layout?

Auto Layout dynamically calculates the size and position of all the views in your view hierarchy, based on constraints placed on those views.

40.What is service extension?

The service extension allows to change the content in a notification before it is presented.

41.When do you use optional chaining vs. if let or guard ?

We use *optional chaining* when we do not really care if the operation fails, otherwise we use *if let* or *guard*. Optional chaining allows us to run code only if our optional has a value.

42. What is Responder Chain ?

A Responder Chain is a hierarchy of objects that can respond to events received.

43. What are the collection types available in Swift?

Swift Programming Language provides three main collection types called Arrays, Sets and Dictionaries. These three collections are used to store a collection of values. Dictionaries are not ordered collections which will have key-value pairs associations. These three data types are always clear about the types of values and keys associated with them. The different or wrong type key or values cannot be inserted into these collection types.

44. Explain how messaging works in Objective-C?

Messaging are not bound to method implementation until runtime in Objective-C. The compiler transforms a message expression, into a call on a messaging function, objc_msgSend(). This function connects the receiver and the name of the method mentioned in the message.

45. What is synchronized() block in objective C? What is the use of that?

The synchronized() directive locks a section of code for use by a single thread. Other threads are blocked until the thread exits the protected code.

46.Explain final keyword into a class ?

A class that is declared final cannot be sub classed.

47. What is the meaning of "retain" keyword?

Retain creates a reference from one object to another and increases the retain count of the source object.

49. What is the meaning of "assign" keyword?

Assign creates a reference from one object to another without increasing the retain count of the source object.

50. What is the meaning of "atomic" keyword?

"atomic", the synthesized setter/getter will ensure that a whole value is always returned from the getter or set by the setter, only single thread can access variable to get or set value at a time.

51. Explain what are Responder Chain and First Responder?

A ResponderChain is a hierarchy of objects that can respond to events received.

The first object in the ResponderChain is called the FirstResponder.

52. What is Keychain ?

Keychain is an API for persisting data securely in iOS App.

53. What's the difference between a xib and a storyboard?

Both are used in Xcode to layout screens (view controllers). A xib defines a single View or View Controller screen, while a storyboard shows many view controllers and shows the relationship between them.

54. Explain AVFoundation framework.

AVFoundation allows us to work on a detailed level with time-based audio-visual data. With it, we can create, edit, analyze, and re-encode media files. AVFoundation has two sets of APIs, one is video and other one is audio.

55. Explain Sequence in Swift.

Sequence is a basic type in Swift for defining an aggregation of elements that distribute sequentially in a row. All collection types are inherited from Sequence such as Array, Set, Dictionary.

56. Explain immutable and mutable objects.

Immutable objects can't be changed. However, they are just pointing to some location where stored values are made constant. You can

change that reference to other location. You can change the value of mutable objects.

57. Explain JSONEncoder and JSONDecoder in Swift4.

*JSONEncoder :*Encodable protocol to take instances of our object and turn it into data. With that data, we can store it to the files, send it to the server, whatever you need to do with it.

*JSONDecoder :*Decodable protocol, which allows us to take data and create instances of our object, populated with the data passed down from the server.

58. What is Pointer ?

A pointer is a direct reference to a memory address. Whereas a variable act as a transparent container for a value, pointers remove a layer of abstraction and let you see how that value is stored.

59. How many ways to pass data in Swift ?

There are many ways such as Delegate, KVO, Segue, NSNotification, Target-Action and Callbacks.

60. What is the difference between Synchronous & Asynchronous task ?

Synchronous: waits until the task has completed

Asynchronous: completes a task in background and can notify you when complete

61. What is Regular expressions ?

Regular expressions are special string patterns that describe how to search through a string.

62. What is operator overloading ?

Operator overloading allows you to change the way existing operators work with specific structures or classes.

63. What is generics in Swift ?

Generics create code that does not get specific about underlying data types.

64. Explain method swizzling ?

Method swizzling is a well-known practice in Objective-C and in other languages that support dynamic method dispatching.

Through swizzling, the implementation of a method can be replaced with a different one at runtime, by changing the mapping between a specific #selector(method) and the function that contains its implementation.

To use method swizzling with your Swift classes there are two requirements that you must comply with:

The class containing the methods to be swizzled must extend NSObject

The methods you want to swizzle must have the dynamic attribute

65. Explain content extension.

The content extension gives us the tools, we have in an app to design the notification.

66. Explain priority inversion and priority inheritance.

If high priority thread waits for low priority thread, this is called Priority Inversion.

If low priority thread temporarily inherits the priority of the high priority thread, this is called Priority Inheritance.

67. What is Hashable ?

Hashable allows us to use our objects as keys in a dictionary. So, we can make our custom types.

68.Which git command saves your code without making a commit ?

git stash is the command which saves our code without making commit.

69.What is Instruments?

Instrument is a powerful performance tuning tool to analyze that performance, memory footprint, smooth animation, energy usage,

leaks and file/network activity are working fine.

70.Define App Bundle.

When you build your iOS app, Xcode packages it as a bundle. A bundle is a directory in the file system that groups related resources together in one place. An iOS app bundle contains the app executable file and supporting resource files such as app icons, image files, and localized content.

71.What is Dynamic Dispatch ?

Dynamic Dispatch is the process of selecting which implementation of a polymorphic operation that's a method or a function to call at run time.

72. What is ARC ?

It stands for Automatic Reference Counting. ARC is a compiler feature that provides automatic memory management for Objective C Objects, so that developers can focus primarily on building application functionality and not worry about retain and releases.

73. How is an inout parameter different from a regular parameter?

An Inout passes by reference while a regular parameter passes by value.

74. What are the states of an iOS App?

Non-running—The app is not running.

Inactive—The app is running in the foreground, but not receiving events. An iOS app can be placed into an inactive state, for example, when a call or SMS message is received.

Active—The app is running in the foreground and receiving events.

Background—The app is running in the background and executing code.

Suspended—The app is in the background, but no code is being executed.

75. What is Webhooks ?

Webhooks allow external services to be notified when certain events happen within your repository. (push, pull-request, fork)

76.What is internal access ?

Internal access enables entities to be used within any source file from their defining module, but not in any source file outside of the module.

Internal access is the default level of access. So even though we haven't been writing any access control specifiers in our code, our code has been at an internal level by default.

77. What are subscripts ?

Subscripts are used to access information from a collection, sequence and a list in Classes, Structures and Enumerations without using a method. These subscripts are used to store and retrieve the values with the help of index without the use of separate method. To access elements via subscripts, write one or more values between square brackets after the instance name.

For example: Array elements are accessed with the help of some Array[index] and its subsequent member elements in a Dictionary instance can be accessed as some Dictionary[key].

78. What is the difference ANY and ANYOBJECT ?

Any can represent an instance of any type at all, including function types and optional types.

AnyObject can represent an instance of any class type.

79. What is forced unwrapping?

If you defined a variable as optional, then to get the value from this variable, you will have to unwrap it. Just putting an exclamation mark at the end of the variable makes it forced unwrap.

80. What is the relation between iVar and @property?

iVar is an instance variable. It cannot be accessed unless we create accessors, which are generated by @property. iVar and its counterpart @property can be of different names.

iVar can be accessed using KVC.

81.Explain KVC—KVO

KVC stands for Key-Value Coding. It's a mechanism by which an object's properties can be accessed using strings at runtime rather than having to statically know the property names at development time.

KVO stands for Key-Value Observing and allows a controller or class to observe changes to a property value. In KVO, an object can ask to be notified of any changes to a specific property, whenever that property changes value, the observer is automatically notified.

82. What is a memory leak?

A memory leak is a type of resource leak that occurs when a computer program incorrectly manages memory allocations in such a way that memory which is no longer needed is not released. In object-oriented programming, a memory leak may happen when an object is stored in memory but cannot be accessed by the running code.

83. What is Cocoa Touch?

Cocoa Touch is a UI framework for building software programs to run on iOS for the iPhone, iPod Touch, and iPad, watchOS for the Apple Watch, and tvOS for the fourth-generation Apple TV, from Apple Inc. Cocoa Touch provides an abstraction layer of iOS, the operating system for the iPhone, iPod Touch, and iPad.

84. Difference between cocoa and cocoa touch?

Cocoa is commonly referred to as the combination of the Foundation and AppKit frameworks, while Cocoa Touch is the combination of the Foundation and UIKit frameworks.

Cocoa and Cocoa Touch sit on top of other collections of frameworks to create the API stacks. The other layers are Media, Core Services and Core OS.

The big difference is the UI layer. On the Mac, you have the AppKit framework, on the iPhone, you have UIKit. The smaller differences

are in the Foundation framework

Cocoa is for Mac development; Cocoa Touch is for iOS development.

85. Explain some common execution states in iOS?

The states of the common execution can be as follows:

Not running – This state means that there is no code that is being executed and the application is completely switched off.

Inactive – This state means that the application is running in the background and is not receiving any events.

Active – This state means that the applications are running in the background and is receiving the events.

Background – This state means that the application is executing the code in the background.

Suspended – This state means that the application is in the background and is not executing.

86. What Is "protocol" On Objective C?

A protocol declares methods that can be implemented by any class. Protocols are not classes themselves. They simply define an interface that other objects are responsible for implementing. Protocols have many advantages. The idea is to provide a way for classes to share the same method and property declarations without inheriting them from a common ancestor.

87. Explain completion handler?

When our application is making an API call and we are supposed to update the UI to show the data from the API call, then Completion Handler becomes handy and is super convenient.

88.What is intrinsic content size?

Every view that contains content can calculate its intrinsic content size. The intrinsic content size is calculated by a method on every UIView instance. This method returns a CGSize instance.

89. What is .gitignore?

.gitignore is a file extension that you tell Git server about document types, and folders that you do not want to add to the project or do not want to track changes made in Git server projects.

90. What do you mean by the term "defer"?

The term "defer" is a keyword that provides a block of code that executes when the execution is leaving the current scope.

91.What is ARC and how is it different from AutoRelease?

ARC is used inside the scope, autoRelease is used outside the scope of the function.

92. What is receipt validation ?

The receipt for an application or in-app purchase is a record of the sale of the application and of any in-app purchases made from within the application. You can add receipt validation code to your application to prevent unauthorized copies of your application from running.

93. What is initialization ?

Initialization is the process of preparing an instance of a class, structure, or enumeration for use. This process involves setting an initial value for each stored property on that instance and performing any other setup or initialization that is required before the new instance is ready for use.

95. What is synthesize in Objective-C?

Once you have declared the property in objective-C, you must tell the compiler instantly by using synthesize directive. This will tell the compiler to generate a getter & setter message.

96. What is TVMLKit ?

TVMLKit is the glue between TVML, JavaScript, and native tvOS application.

97. What are the difference between MRR & ARC?

 MRR – Manual Retain Release , In MRR we explicitly manage memory by keeping track of objects we own). Using the reference

counting model.

ARC –Automatic Reference Counting. The system inserts appropriate memory management method calls for us at compile time

98. Why design patterns are important ?

Design patterns are reusable solutions to common problems in software design. They're templates designed to help you write code that is easy to understand and reuse.

99. What are the common design patterns in Cocoa?

Most common Cocoa design patterns are:

Creational: Singleton.

Structural: Decorator, Adapter, Facade.

Behavioral: Observer, and, Memento

100. What is Singleton design pattern ?

The Singleton design pattern ensures that only one instance exists for a given class and that there's a global access point to that instance. It usually uses lazy loading to create the single instance when it's needed the first time.

101.What is Facade design pattern ?

The Facade design pattern provides a single interface to a complex subsystem. Instead of exposing the user to a set of classes and their APIs, you only expose one simple unified API.

102. What is Strategy pattern?

Strategy pattern allows you to change the behavior of an algorithm at run time. Using interfaces, we can define a family of algorithms, encapsulate each one, and make them interchangeable, allowing us to select which algorithm to execute at run time.

103. What is Decorator Design Pattern ?

The Decorator pattern dynamically adds behaviors and responsibilities to an object without modifying its code. It's an

alternative to sub classing where you modify a class's behavior by wrapping it with another object.

In Objective-C there are two very common implementations of this pattern: Category and Delegation. In Swift there are also two very common implementations of this pattern: Extensions and Delegation.

104. What is Adapter Pattern ?

An Adapter allows classes with incompatible interfaces to work together. It wraps itself around an object and exposes a standard interface to interact with that object.

105. What is Observer Pattern ?

In the Observer pattern, one object notifies other objects of any state changes.

Cocoa implements the observer pattern in two ways: Notifications and Key-Value Observing (KVO).

106. What is Memento Pattern ?

In Memento Pattern saves your stuff somewhere. Later, this externalized state can be restored without violating encapsulation; that is, private data remains private. One of Apple's specialized implementations of the Memento pattern is Archiving other hand iOS uses the Memento pattern as part of State Restoration.

107.What is a category and when is it used ?

A category is a way of adding additional methods to a class without extending it. It is often used to add a collection of related methods. A common use case is to add additional methods to built-in classes in the Cocoa frameworks.

108.Distinguish between @synthesize and @dynamic in Objective –C?

@synthesize – It generates the getter and setter methods for the property.

@dynamic – It notifies the compiler that the getter and setter are implemented at some other place.

109.What is place holder constraint?

This tells the interface builder to go ahead and remove the constraints when we build and run this code. It allows the layout engine to figure out a base layout, and then we can modify that base layout at run time.

110. When should we use autorelease?

Use autorelease when you need to send a deferred release message—typically when returning an object from a method.

111.What is #keyPath() ?

Using #keyPath(), a static type check will be performed by the key-path literal string being used as a StaticString or StringLiteralConvertible. At this point, it's then checked to ensure that

- it is a thing that exists.

- it is properly exposed to Objective-C.

112. What is NSFetchRequest ?

NSFetchRequest class is used to request data fetches from the model. This tool allows you to set object filtering and sorting rules at the fetching stage, so the operation becomes several times quicker and more efficient than fetching every object first and then manually sorting or filtering the data we require.

When we know the basic properties of this class, we can easily manage requests and get a specific fetch without developing additional algorithms and crutches as everything is already implemented in Core Data.

113. Explain NSPersistentContainer ?

A container that encapsulates the Core Data stack in your application.

NSPersistentContainer simplifies the creation and management of the Core Data stack by handling the creation of the managed object model (NSManagedObjectModel), persistent store coordinator

(NSPersistentStoreCoordinator), and the managed object context (NSManagedObjectContext).

114. Explain NSFetchedResultsController ?

NSFetchedResultsController is a controller, but it's not a view controller. It has no user interface. Its purpose is to make developers' job easier by abstracting away much of the code needed to synchronize a table view with a data source backed by Core Data.

Set up an NSFetchedResultsController correctly, and your table will mimic its data source without you have to write more than a few lines of code.

115. What are fileprivate, private and public private(set) access level ?

fileprivate is accessible within the current file.

private is accessible within the current declaration.

public private(set) means getter is public, but the setter is private.

116. What is Responder Chain ?

A ResponderChain is a hierarchy of objects that can respond to events received.

117. What is Hashable ?

Hashable allows us to use our objects as keys in a dictionary. So, we can make our custom types.

118. What is bounding box?

Bounding box is a term used in geometry; it refers to the smallest measure (area or volume) within which a given set of points.

119.Please explain types of notifications.

There are two types of notifications: Remote and Local. Remote notification requires connection to a server. Local notifications don't require server connection. Local notifications happen on device.

120. What is a do-catch block?

In Swift, errors are thrown and handled inside of do-catch blocks.

121. What is Xcode?

Xcode is an integrated development environment (IDE) for macOS containing a suite of software development tools developed by Apple for developing software for macOS, iOS, watchOS, and tvOS.

122. What are Bots?

Bots are processes that Xcode Server runs to perform integrations on the current version of a project in a source code repository. An integration is a single run of a bot. Integrations consist of building, analyzing, testing, and archiving the apps (or other software products) defined in your Xcode projects. With Xcode Server able to access the source code repositories of those projects, you can configure bots to perform continuous integrations on them.

123. When to use Bots?

Bots can be configured to run in the following ways:

Every time a change is committed to the repository

On a regular schedule, such as hourly, daily, or weekly

When manually initiated

124. What is Core Data ?

Core Data is neither an ORM or object-relational mapper nor a database. It is an object graph manager which also can persist object graphs to a persistent store on a disk.

125.What are Selectors in Objective-C ?

Selectors are Objective-C's internal representation of a method name.

126.What is Code Coverage ?

Code coverage is a metric that helps us to measure the value of our unit tests.

127. What is Protocol Extensions?

We can adopt protocols using extensions as well as on the original type declaration. This allows you to add protocols to types which you

don't necessarily own.

128. Explain lazy initialization/ Lazy loading ?

Lazy initialization (also sometimes called lazy instantiation, or lazy loading) is a technique for delaying the creation of an object or some other expensive process until it is needed. When programming for iOS, this is helpful to make sure that you utilize only the memory you need.

129.What are NSNotificationCenter and how does it work ?

NSNotificationCenter is an Observer Pattern in the Cocoa library . The basic idea is that a listener registers with a broadcaster using some predefined protocol. At some later point, the broadcaster is told to notify all its listeners, where it calls some function on each of its listeners and passes certain arguments along. This allows for asynchronous message passing between two different objects that don't have to know about one another, they just must know about the broadcaster.

130.What is GraphQL ?

GraphQL is trying to solve creating a query interface for the clients at the application level. Apollo iOS is a strongly-typed, caching GraphQL client for iOS, written in Swift.

131. What is @synthesize in Objective-C ?

synthesize generates getter and setter methods for your property.

132. What is @dynamic in Objective-C ?

We use dynamic for subclasses of NSManagedObject. @dynamic tells the compiler that getter and setters are implemented somewhere else.

133.What do you do when you realize that your App is prone to crashing?

Do the following when you realize that your App is prone to crashing

First, determine the iOS version or model of the device.

Collect enough information to reproduce the issue.

If possible, acquire device logs.

Acquire tooling or create a unit test and begin debugging once you get the idea of the nature of the issue.

134. What does Yak Shaving mean ?

Yak shaving is a programing term that refers to a series of tasks that need to be performed before a project can progress to its next milestone.

135. Explain "Arrange-Act-Assert" / AAA.

AAA is a pattern for arranging and formatting code in Unit Tests. If we were to write XCTests each of our tests would group these functional sections, separated by blank lines:

Arrange all necessary preconditions and inputs.

Act on the object or method under test.

Assert that the expected results have occurred.

136. What is a managed object context ?

A managed object context represents a single object space or scratch pad, in a Core Data application.

137. What is GCD? How is it used?

GCD is the most commonly used API to manage concurrent code and execute operations asynchronously at the Unix level of the system. GCD provides and manages queues of tasks. A good example is when an App fetch data from an API, this network call should be done in a background thread and the display of the data in the view should be executed in the main thread as well as any UI updates.

138. What are the collection types available in Swift?

Swift Programming Language provides three main collection types called Arrays, Sets and Dictionaries.

These three collections are used to store a collection of values. Dictionaries are not ordered collections which will have key-value pairs associations. These three data types are always clear about

the types of values and keys associated with them. The different or wrong type key or values cannot be inserted into these collection types. This leads to less error-prone or types checking safety at runtime. The collection types Arrays, Sets, and Dictionaries are mutable. This means that the values can be changed and modified as per the requirement or the operations need to be done.

The best practice is to use and create the immutable collections instead of mutable ones. This will keep safe of runtime data. An Array always stores the collection of data of the same type. The same value can occur at different indices in the array. This means duplicates are allowed for this collection type. Arrays can be created empty or with default values or with some required values while declaring. Sets store distinct values and will not allow duplicates. The order is not guaranteed in Sets. Dictionary will also not guarantee the order of insertion which has same key types and value types for all the list of key-value pairs.

139. Define fast enumeration?

Fast enumeration is a language feature that allows you to enumerate over the contents of a collection. (Your code will also run faster because the internal implementation reduces message send overhead and increases pipe lining potential.

140. What is property in objective-C?

Property allow declared variables with specification like atomic/non-atomic or retain/assign.

141. What is the difference between Swift and Objective-c language?

Swift Programming-

In a swift programming, the variables and constants are declared before use.

In a swift programming, var keyword used for variable and let keyword for constant.

In a swift programming, no need to end code with semi-colon

In a swift programming, we do not require creating a separate interface .

In a swift programming, we can define methods in class, structure or enumeration.

Objective-C Programming-

In objective-C programming, we need to end code with semi-colon

In objective-C programming, we can declare constant as int and variable as NSString.

In objective-C programming, we do require creating a separate interface .

142. Explain what is the protocol in Objective C?

In Objective-C, a protocol is a language feature, that provides multiple inheritances in a single inheritance language. Objective C supports two types of protocol.

Ad hoc protocols known as informal protocol

Compiler protocols known as a formal protocol

143.What is encapsulation?

Encapsulation is an object-oriented design principle and hides the internal states and functionality of objects. That means the objects keep their state information private.

144. What is dependency management?

If we want to integrate open source project, add a framework from a third-party project, or even reuse code across our own products, dependency management helps us manage these relationships.

145. What is Continuous Integration?

Continuous integration allows us to get early feedback when something is going wrong during application development. There are a lot of continuous integration tools available.

146. What "app thinning" ?

The store and operating system optimize the installation of iOS, tvOS, and watchOS apps by tailoring app delivery to the capabilities of the user's device, with minimal footprint. This optimization is called app thinning which allows you to create apps that use the most device features, occupy minimum disk space, and accommodate future updates that can be applied by Apple.

147. What do you know about the concept of Nil Coalescing & Ternary Operator?

NilCoalescing & Ternary Operator can simply be interpreted as an easily return and unwrapped optional value. In the case where we are not having any value to assign, we can set zero or any default choice of our need.

148. Explain the difference between atomic and nonatomic synthesized properties.

atomic : If an object is declared as atomic then it becomes thread-safe. Thread-safe means, at a time only one thread of an instance of that class can have the control over that object.

nonatomic: It is not thread-safe. We can use the nonatomic property attribute to specify that synthesized accessors simply set or return a value directly, with no guarantees about what happens if that same value is accessed simultaneously from different threads. For this reason, it's faster to access a nonatomic property than an atomic one.

149. When to use CharlesProxy ?

If I need a proxy server that includes both complete requests and responses and the HTTP headers, then I am using CharlesProxy. With CharlesProxy, we can support binary protocols, rewriting and traffic throttling.

150. What is RGR (Red—Green—Refactor) ?

Red, Green and Refactor are stages of the TDD (Test Driven Development).

Red: Write a small amount of test code usually no more than seven lines of code and watch it fail.

Green: Write a small amount of production code. Again, usually no more than seven lines of code and make your test pass.

Refactor: Tests are passing, you can make changes without worrying. Clean up your code.

151. What is the difference CollectionViews & TableViews ?

TableViews display a list of items, in a single column, a vertical fashion, and limited to vertical scrolling only.

CollectionViews also display a list of items, however, they can have multiple columns and rows.

152.What is NotificationCenter ?

NotificationCenter is an observer pattern,
The NSNotificationCenter singleton allows us to broadcast information using an object called NSNotification.

The biggest difference between KVO and NotificationCenter is that KVOtracks specific changes to an object,
while NotificationCenter is used to track generic events.

153.What do you understand by the term App Bundle?

A bundle can simply be defined as a directory which groups up all the related resources together at one place within the file system. For example, if you build an iOS app, Xcode packages then it can be called as a bundle. Inside an iOS app bundle, it contains all the applications files which can be executable. App Bundle also supports the resource different files which include localized content, image files and app icons. These elements helps in delivering the best application performance to the iOS software.

154. What is Downcasting ?

When we're casting an object to another type in Objective-C, it's simple since there's only one way to do it. In Swift, though, there are two ways to cast—one that's safe and one that's not .

as used for upcasting and type casting to bridge type

as? used for safe casting, return nil if failed

as! used to force casting, crash if failed. should only be used when we know the downcast will succeed.

155. Explain difference between SDK and Framework ?

SDK is a set of software development tools. This set is used for creation of applications. Framework is basically a platform which is used for developing software applications. It provides the necessary foundation on which the programs can be developed for a specific platform. SDK and Framework complement each other, and SDKs are available for frameworks.

156. What is the Test-Driven Development of three simple rules ?

You are not allowed to write any production code unless it is to make a failing unit test pass.

You are not allowed to write any more of a unit test than is sufficient to fail; and compilation failures are failures.

You are not allowed to write any more production code than is sufficient to pass the one failing unit test.

157. Name the methods to achieve concurrency in iOS.

These are the methods to achieve concurrency functionality in iOS:
1.Threads
2.Dispatch Queues
3.Operation Queues

158. What is three triggers for a local notification ?

Location, Calendar, and Time Interval. A Location notification fires when the GPS on your phone is at a location or geographic region. Calendar trigger is based on calendar data broken into date components. Time Interval is a count of seconds until the timer goes off.

159. How could we get device token ?

There are two steps to get device token. First, we must show the user's permission screen, after we can register for remote notifications. If these steps go well, the system will provide device token. If we uninstall or reinstall the app, the device token would change.

160. What is AutoLayout?

AutoLayout provides a flexible and powerful layout system that describes how views and the UI controls calculates the size and position in the hierarchy.

161. Explain Property Observer.

A property observer observes and responds to changes in a property's value. With property observer, we don't need to reset the controls, every time attribute change.

162. What is CoreSpotlight?

CoreSpotlight allows us to index any content inside of our app. While NSUserActivity is useful for saving the user's history, with this API, you can index any data you like. It provides access to the CoreSpotlight index on the user's device.

163.Explain Regular expression and Responder chain?

Regular Expression – These are the special string patterns that describe how a search is performed through a string.

Responder Chain – It is a hierarchy of objects that obtain the opportunity to respond to the events.

164. Explain the procedure of passing a variable as a reference.

There are two types of variables: variable as a reference and variable as a value . The major variations between these both types of the variables are that by passing value types the variable will then generate a duplicate of the data within and whereas the reference type variable will simply point the actual data in the memory.

165. Differentiate the concepts of copy and retain?

Retaining in the sense relates to the aspect in which the total count increases by one. This states that the instance of the object will be kept in memory unless it retains count totally drops to the value of null or zero.

Whereas, the term Copy indicates that the object will be cloned with duplicate values.

166. Is multitasking function is supported by the iOS?

Multitasking function is supported by the iOS 4 & the other high-level series. These series support the app to run in the background once they are launched and they keep running unless they are terminated. This is the highly extraordinary future showcased by the iOS application platform.

167. Among the different JSON framework which one does supported by the iOS?

iOS supports SBJson framework. It is a well advanced and highly flexible framework that supports flexible functioning of APIs & and some other extra added features that make the Json handling a lot easier. SBJson is a highly advanced concept of iOS software platform.

168. Explain SiriKit Limitations.

SiriKit cannot use all app types

Not a substitute for a full app only an extension

Siri requires a consistent Internet connection to work

Siri service needs to communicate Apple Servers.

169. What do you understand by the term NSFetchRequest?

Fetch requests are extensively used for fetching predefined objects which match the exact requirement in-relation to individual values, and criteria. The main use of this NSFetchRequest is to fetch from the core data.

170. What's the difference between Android and iOS designs?

Android uses icons, iOS mostly uses text as their action button. When we finish an action, on Android you will see a checkmark icon, whereas on iOS you'll have a 'Done' or 'Submit' text at the top.

iOS uses subtle gradients, Android uses flat colors and use different tone/shades to create dimension. iOS uses bright, vivid colors—almost neon like. Android is colorful, but not as vivid.

iOS design is mostly flat, but Android's design is more dimensional. Android adds depth to their design with floating buttons, cards and so on.

iOS menu is at the bottom, Android at the top. Also, a side menu/hamburger menu is typically Android. iOS usually has a tab called 'More' with a hamburger icon, and that's usually the menu & settings rolled into one.

Android ellipsis icon (…) to show more options is vertical, iOS is horizontal.

Android uses Roboto and iOS uses San Francisco font

Frosted glass effect used to be exclusive to iOS, but you'll see a lot of them in Android too nowadays. As far as I know it's still a hard effect to replicate in Android.

Android usually has navigation button in color, whereas iOS is usually either white or gray.

171. Which Programming Languages are used for iOS Development?

The languages used for iOS development are :
1. Objective-C
2. .NET
3. C
4. HTML5
5. JavaScript
6. Swift

172. What is the difference between viewDidLoad and viewDidAppear? Which should you use to load data from a

remote server to display in the view?

viewDidLoad is called when the view is loaded, whether from a Xib file, storyboard or programmatically created in loadView. *viewDidAppear* is called every time the view is presented on the device. Which to use depends on the use case for your data. If the data is static and not likely to change then it can be loaded in viewDidLoad and cached. However, if the data changes regularly then using viewDidAppear to load it is better. In both situations, the data should be loaded asynchronously on a background thread to avoid blocking the UI.

173. Explain keywords alloc and new.

The *alloc* keyword is used to create a new memory location in the system. However, it does not initialize it. In case of *new* keyword, it also helps to create a new memory location in the system. However, it can initialize the contents unlike the alloc keyword.

174. What is a protocol in objective-C ?

A protocol is used to define a list of required optional methods that a class needs to implement. If a class adopts a protocol, it must implement all the needed methods in the protocols it adopts. It is identical to an interface in Java and to a purely virtual class in C++. Cocoa uses protocols to support inter-process communication through Objective-C messages.

175. What is the difference between release and pool drain?

The release keyword is used to free a memory location in the system which is not being utilized. The drain keyword is used to release the NSAutoreleasePool.

176. What is plist?

Plist represents property list. It is a key-value store for the application to save and retrieve persistent data values. This is specifically used for iPhone development. It is basically an XML File.

178. What is an accessor method?

Accessor methods belong to a class that enables you to get and set the values of instance valuable contained within the class.

179. What is the maximum byte-size for a push notification to Apple Server?

The maximum memory size is 256 Bytes to send a push Notification to Apple Server.

180.What is the difference SVN and Git ?

SVN relies on a centralized system for version management. It's a central repository where working copies are generated, and a network connection is required for access.

Git relies on a distributed system for version management. You will have a local repository on which you can work, with a network connection only required to synchronize.

181. Explain Web Services?

The Web Services are the application components which enables communication using open protocols. The Web Services are self – describing and self – contained. The base for development of Web Services functionality is Extensible Markup Language (XML).

182. Explain what is the main difference between the function calls and messages?

The main difference between function call and message is that a function and its arguments are linked together in the compiled code, but a message and a receiving object are not linked until the program is executing and the message is sent.

183. What is a category and when is it used?

A category is a way of adding additional methods to a class without extending it. It is often used to add a collection of related methods. A common use case is to add additional methods to built-in classes in the Cocoa frameworks.

184. How is memory management handled in iOS?

Swift uses Automatic Reference Counting (ARC). This is conceptually the same thing in Swift as it is in Objective-C. ARC keeps track of strong references to instances of classes and increases or decreases their reference count accordingly when you assign or unassign instances of classes (reference types) to constants, properties, and variables. It deallocates memory used by objects whose reference count dropped to zero. ARC does not increase or decrease the reference count of value types because, when assigned, these are copied. By default, if you don't specify otherwise, all the references will be strong references.

185. explain what the difference is between Delegate and KVO?

Both are ways to have relationships between objects. *Delegation* is a one-to-one relationship where one object implements a delegate protocol, and another uses it and sends messages to it if those methods are implemented since the receiver promised to comply to the protocol. *KVO* is a many-to-many relationship where one object could broadcast a message and one or multiple other objects can listen to it and react. KVO does not rely on protocols. KVO is the first step and the fundamental block of reactive programming (RxSwift, ReactiveCocoa, etc.)

186. Where Factory method design pattern is used?

Factory Method is used to replace class constructors, to abstract and hide objects initialization so that the type can be determined at runtime, and to hide and contain switch/if statements that determine the type of object to be instantiated.

187.What is Single Responsibility Principle (SRP) ?

The single responsibility principle is a programming principle that states that every module or class should have responsibility over a single part of the functionality provided by the software, and that responsibility should be entirely encapsulated by the class. All its services should be narrowly aligned with that responsibility.

188. What is Open/Closed Principle (OCP) ?

The open/closed principle states that software entities (classes, modules, functions, etc.) should be open for extension, but closed for modification, that is, such an entity can allow its behavior to be extended without modifying its source code.

The name open/closed principle has been used in two ways. Both ways use generalizations (for instance, inheritance or delegate functions) to resolve the apparent dilemma, but the goals, techniques, and results are different.

189. What is Liskov Substitution Principle (LSP) ?

The Liskov Substitution Principle (LSP) states that objects in a program should be replaceable with instances of their subtypes without altering the correctness of that program. It means that when you inherit from a class or an abstract class or implement an interface (protocol), your objects should be replaceable and injectable wherever that interface or class that you sub classed from ,was used. This principle is often referred to as design by contract or, as of late in the Swift community, referred to as protocol-oriented programming. The main message of this principle is that you should not violate the contract that your interfaces you subclass from promise to fulfill, and that by sub classing, those subclasses could be used anywhere that the superclass was previously used.

190. What is Interface Segregation Principle (ISP) ?

The interface-segregation principle (ISP) states that no client should be forced to depend on methods it does not use. ISP splits interfaces that are very large into smaller and more specific ones so that clients will only have to know about the methods that are of interest to them. Such shrunken interfaces are also called *role interface*s. ISP is intended to keep a system decoupled and thus easier to refactor, change, and redeploy. ISP is one of the five solid principles of object-oriented design.

191.What is Dependency Inversion Principle (DIP) ?

The dependency inversion principle refers to a specific form of decoupling software module. When following this principle, the

conventional dependency relationships established from high-level, policy-setting modules to low-level, dependency modules are reversed, thus rendering high-level modules independent of the low-level module implementation details.

The principle states:

1. High-level modules should not depend on low-level modules. Both should depend on abstractions.

2. Abstractions should not depend on details. Details should depend on abstractions.

192. How and when would you need to serialize and map data on iOS?

There are two most common scenarios where you need to serialize and map data in iOS applications:

1.Receiving or sending data in the networking layer (such as JSON or XML or something else),

2.Persisting or retrieving models in the storage layer (NSData, NSManagedObject, etc.).

193. How would you optimize scrolling performance of dynamically sized table or collection views?

Scrolling performance is a big issue with UITableViews and quite often can be very hard to get right. The main difficulty is cell height calculation. When the user scrolls, every next cell needs to calculate its content and then height before it can be displayed.

Potential solutions for scrolling performance issues could be

1.Calculate cell height yourself

2.Keep a prototype cell that you fill with content and use to calculate cell height

194. How do you manage dependencies?

Dependencies management is an important task on every iOS project. These days many dependency managers are there to

choose from: CocoaPods, Carthage, and Swift Package Manager (SPM). The most dominant and robust one is CocoaPods.

195. Explain what are the characteristics of the category?

Characteristics of category includes

1.Even if you don't have the original source code for implementation, a category can be declared for any class

2.Any methods that you define in a category will be available to all instances of the original class as well as any sub-classes for the original class

3.At runtime, there is no variation between a method appended by a category and one that is implemented by the original class

196. Explain unwind segue.

An unwind segue moves backward through one or more segues to return the user to a scene managed by an existing view controller.

197. What is HealthKit ?

HealthKit is a framework on iOS. It stores health and fitness data in a central location. It takes in data from multiple sources, which could be different devices. It allows users to control access to their data and maintains privacy of user data. Data is synced between your phone and your watch.

198.What are SpriteKit and SceneKit?

SpriteKit is a framework for easy development of animated 2D objects.

SceneKit is a framework inherited from OS X that assists with 3D graphics rendering.

SpriteKit, SceneKit, and Metal are expected to power a new generation of mobile games that redefine what iOS devices' powerful GPUs can offer.

199. What are iBeacons?

iBeacon technology allows Mobile Apps to understand their position on a micro-local scale and deliver hyper-contextual content to users

based on location. The underlying communication technology is Bluetooth Low Energy.

200. Which API would you use to write test scripts to exercise the application's UI elements?

UI Automation API is used to automate test procedures. JavaScript test scripts that are written to the UI Automation API simulate user interaction with the application and return log information to the host computer.

201. Which is the framework that is used to construct application's user interface for iOS?

The UIKit framework is used to develop application's user interface for iOS. It provides event handling, drawing model, windows, views, and controls specifically designed for a touch screen interface.

202.Which JSON framework is supported by iOS?

iOS supports SBJson framework. SBJson is a JSON parser and generator for Objective-C. It provides flexible APIs and additional control, making JSON handling easier.

203. Explain what is parsing and mention which class can we use for passing of XML in iPhone?

Parsing is the process referred to access the data in the XML element. We can use class "NSXML" parser for passing XML in iPhone.

204. Mention which class are used to establish a connection between applications to the web server?

The class used to establish connection between applications to web server are

NSURL

NSURL REQUEST

NSURL CONNECTION

205. What is IPA?

IPA represents iOS App Store Package. It has an .ipa extension which represents iPhone application archive file that stores an iPhone application. Every file is compressed with a Binary for the ARM architecture and can only be installed on an iPhone, iPad or an iPod Touch. It is mostly encrypted with Apple's Fair Play DRM Technology.

206. Explain what is dot notation?

Dot notation involves assessing an instance variable by determining a class "instance" followed by a "dot" followed in turn by the name of instance variable or property to be accessed.

207. Explain when to use NSArray and NSMutableArray?

NSArray: You will use an NS array when data in the array don't change. For example, the company name you will put in NS Array so that no one can manipulate it.

NSMutableArray: This array will be used in an array when data in an array will change. For instance, if you are passing an array to function and that function will append some elements in that array then you will choose NSMutable Array.