Mock interview question

1. What is function

Function is block of the codes, It is to achieve the particular task

1. What are function types

It has parameter with return,

It has parameter without return

It has no parameter with return

It has no parameter without return

**Optional Chaining**

1. What is optional chaining

It does querying values from the chains of optional. This can be properties

1. What is IS Keyword ?

It is check operator, It is for checking the values

1. What is As keyword ?

It is for changing data type into another data type

1. What is any

Any data type you can think in Swift

1. What is Any object?

This is instance of the any reference type

1. What is nested types?

It is type within another type

1. What do we need for nested type?

Because Enum without case can creating constants in the application. So, It won’t allow to initialize object

**Class and Structs**

1. What does Instance means ?

Instance is a concrete occurrences of any object

1. What is property and behaviors?

Property is variable data and behaviors are method

1. What is difference between in Struct and Class

Class are reference types, Struc and struct ares value type

Class can be inherited, Struct can not be inherited

Class has init and de-init, But Struct only has init

1. When can we use instance method?

Instance method can be accessed by only object

1. What is declaration of struct ?

If variables doesn’t assign data, It will automatically create initializers include mem-wise in init struct

1. What is object ?

Its instance of a class, struct, enum. It has state and behaviors

1. When to use structs,

Choose struct by default for crating you module

And, struct is good for mult-thread environment

1. What is dead-lock?

There are two tread, try to access one resource at the same time

1. What is mutating?

If a struct has a variable property but the instance of struct was created as constants. So, when we want to change property inside of method. Then we need a mutating keyword

1. What is Identity Operators

It is used to check whether two constants or variable refer to the same single instance or not

**Properties**

1. What is property

It is defined state of an object , It associate struct, Class, enum

1. What are types of property?

There is Stored property, It means we can put some data inside it directly. It will hold the data

There is computed property, It doesn’t store a data but It will calculate its value based on the variables and inputs and return it

1. What is lazy property

This is type of the property . It won’t have value until we access values

1. What are get and set?

Both are for computed property, when we read value, get will be executed,. When we assign some data, then set property will be executed

1. What happens if we put computed property inside set?

When we create an instance, it will get called recursively it will ultimately crashes

1. What is type property ?

Property belongs to a type itself and common to all instances

1. What is Type property and method?

No object creating is required and accessible using the type name itself

1. What is property observer?

**There are specific code bock which will get called when you setting some value inside of property**

1. What are types of properties observers

There is will-set :this codes will be called just before the value is getting set

There is did set, this code will be called after the value is set

1. What are different between instance and type method

Instance property can be access by object only

Type property can be access without object

**Error Handling**

1. What is error handling?
2. What is throw keyword

It will catch the error, It is indicating that function, method, or initializer can throw an error. Only throwing error can propagate errors.

1. What is handling Error using Do-Catch

We use do-catch statement to handle errors by running a block of code.

1. What is try?
2. It is handle an error by converting it an optional value

**Closure**

1. What is closure?

A : It is the groups the code that can be passed (assigned) from one var to another and also can be passed inside a function

1. How we write closure

A: We use Type Annotation var myClosure : () -> Void

1. What are types of closures

There are non-escaping closure and escaping closure

1. What are examples of taking time a lots

Reading/Writing, Networking Task, Downloading, Audio/Video processing, Computing a very set of data

1. What is escaping closure

It will get called sometimes later and return immediately from the function.

1. What is default

It will automatically contains default before we assign new variable

1. What is non-escaping closure

It is default, It will wait for the task to get executed. This block the thread on which it is running if it is long running tasks. //. It will wait for the task to get executed // 전체가 실행될 때 까지, 기다릴 것이다 (끝날 때까지 기다림)

1. What is Asynchronous Tasks or operations?

It will not wait for the task to get completed and will execute/return instantly. This will notify the completion of task sometimes later (Multi-Tasking possible)

1. What is Synchronous task?

It will wait for the task to get completed and then proceed with the next statement or operations

1. What is statues of non-escaping closure?

It will wait the task to get executed

1. What DispatchQueue.main.asyncAfter means in project?

It is example of long running scenario. And, it requires escaping closure. In this function, since handler is escaping closure, It won’t wait for this execution, I will get called later and return immediately from the function

1. What can be Escaping and Non-escaping closure both can be trailing closure

Both

1. What is trailing closure?

It is a present at the end of a function as it’s last parameters

1. What is meaning of default?

Not assigned value

1. What is usage of closure

It is instance function, It is simple, So, we don’t have to create function

1. What is reference type?

The data share the same address, implicitly create instance, After we copy some data, it refer into single instance of data

1. What is Collection? – It provide a way to hold certain kind of values in container
2. What is 3 types of collections : Array, Dictionaires, Sets
3. What is value type – **It create independent instance with its unique copy**
4. What is SDLC : It is software development life cycle.
5. What is enum?

This is help to organize group a group related value

1. This allows you to attach some additional information with each which you define and it’s dynamic in nature
2. What is relationship with async and escaping

Async and escaping it won’t wait task

1. Check the theory related lazy stored property?

This is type of property which It will initialize when we access it. If it is not accessed then, it won’t intizlied

1. Why do we need lazy stored property?

So, we don’t have to initialize all data. If data is unnecessary, then we don’t init it. So, we can save data an

1. What is **computed property**

The value is depending on some other value, it doesn’t store the value directly. Mostly value is calculated based on inputs and return types

1. What is the enum features

This allows you to attach some additional information with each enum which you use dynamically

1. What is raw values

When raw value is given, each case assigned specific cases. Raw value is case of specific identity

**Inheritance**

1. What is inheritance?

Inheriting the properties, method, and characteristics

1. What is logic of Inheritance

Child class is inheriting from parent class

1. What is parent class?

The parent class is get inherited

1. What are types of inheritance

Single level – One class is inheriting from another class

Multi level – Multi class is inheriting from another class

Multiple is not possible inSwift

Hierarchical – One class is inheriting from multiple child class

1. What is super class

It is a parent class, The class that you define without any super class will be it’s super class

1. What is private value?

The value can not be accessible outside of class

1. What is **over-riding?**

It helps you providing your own implementation by overriding the existing implementation

Also, It use override keyword.

1. How override in stored property?

In case of stored property, only property observer can be over-riding to provide you overriding characteristics

1. How do we prevent over-riding in swift

We put final keyword in front of the class

1. What does final keyword do?

Final class, methods, variables restrict overriding.

1. What is Subscript ?

Subscript allows you to access member elements of collection(Sets, Array, Dictionary)- group of data directly, It works with collections

1. What is singleton?

It means you are allow you to create only object above class that two from within class

1. **Child is inherits from the parent class**

**Init**

1. What is **init**?

Init is process of preparing your instance. It can be used with setting up initial value to all non-optional property

1. What is designated init?

It is primary Init, It is a special method used for initialization which return the instance any time

1. When you assign a value inside a stored property in the initializer or when you provide default value inside the stored then property observers are not called
2. What is Custom init?

We can custom init? Init has Any types of parameters

1. What is default init?

It will provide defaults value to all its stored properties

1. What is initializers delegations – It is calling other init to initializer the instance as a part of instance initlization process
2. What is designated Initializers?

It is responsible for creating an instance all the stored properties and creating an instance. This must be called in order to create instance(Primary). Every class must have on designated init

1. What is convenience Init?

This is init which is calling the designated init to create instance. We create it as per our convinient. These are secondary init

1. Rule for init

Designated init of any child class must call it’s immediately parent initializers

A Convenience init must call another init from same class

A Convenience init must or ultimately call the designated init

1. Check process :

All properties introduced by the class must be initilized before calling the super class designated init

Designated initlizer must call its superclass designated initizer before assigning/modifying besides inheritance property

Convenience init must call another intiaters before making any changes any property it means

Any instance method cannot be called from init until first phase of initlizationis done

1. What is fail-able init?

Init which can fail to initialize and can return nil

1. What required init?

Every subclass need to implement the requirement init

1. What is the init?

Deallocation of a class instance. This means once you are done with the instance then it will get released from the memory

1. When does de-init

The scope of the object is this complete playground so it will never get out of memory. It will get deallocated

ARC

1. **What is ARC**

Automatic Recycle counting is Swift System of tracking of memory

1. What is Two-Phase Intilizations

Class initializations in Swift is two-phase process: The first phase. It will assigned each stored property initial value by class that introduced it.

Each class given the property opportunity to customize its stored properties further before new instance is considered for use

**Protocol**

1. What is **Protocol** ?

It is blueprints of methods, properties, Initializers

It will create rule or require into specific types

1. **Def: Reference Type(Class) and Value type(Struct, Enum) confirms protocol**
2. **Properties and Behaviors in the Property, Can not have body scope**
3. **When class confirms the protocol and protocol has init, how do we access init?**

**Use required init**

1. **Does class care about the protocol Read only or Set only?**

**It doesn’t care at all**

1. **What are requirements of as types**

**As a return type, as a parameters inside methods or initializations, methods**

**As a type of contains vars any properties**

**As aa type you are storing inside collections**

1. **What does class bound mean?**

**A protocol can be confirmed by a class**

1. **What is protocol composition – Combining multiple protocols into one protocol it can**

**Arc**

1. **What is ARC ?**

**It is automatic reference counting to track and manage your app’s memory usage (If we handle multiple applications, it will crush applications**

1. **What is access control?**

**It restricts access to parts of your code from source files and module**

1. **Does Arc only work reference type?**

**Yes, it doesn’t work with struct**

1. What is unowned and weak?

We use Unowned when we are sure data is not nil

We use Weak value can be nil, Weak allows nil value

Generic Value

1. What is generic Codes?

It is type of cod that a multiple scenario with all data types. It enables you to write flexible, reusable functions and type that can work with any type

1. What does Associates types do?

An associated type gives a placeholder name to a type that’s used as part of protocol. assoicate types are specified with asooicated types of keywords

**Control Flow**

1. **What is control flow?**

**It provides variety of control flow**

**For-In loops, while loop, repeat while,**

1. **What is fall through**

**Keep executing next scope of codes**

1. **Control transfer statements?**

**Change order in which is executed**

**There are return, continue, break,return , throw**

1. **What is MVC**

**It is MVC framework**

**It is architectural pattern that seperates an application in to three main model**

**Model Components corresponding to all the data related logic that suer works with view. It is used for UI logic**

**Controller Acts as interface between model and view components to process all the business logic manipulate data using the model component and interect with views to render the final output**

**Patterns works**

**Users use controller, controller manipulate data to model, model updates values, then user sees it finally**