1. Which kind of technology you have worked in your project.
2. Tell me about your role responsibilities in your.
3. Do you know iOS app submission process. How many type certificate is here in iOS.
4. Which type of security you following in your projects. And how did you implement it.
5. Difference between Swift and objective C programming language.
6. Which design pattern your are following indoor project.
7. App Delegate life cycle. Which is the first method called when app launch.
8. Difference between class and structure.
9. Basic concept of OOPS programming.
10. What is the difference between delegate and protocol?
11. What is the difference between method & function?
12. What is object mapping?
13. What is a protocol associated type?
14. What is In out parameters?
15. Escaping vs Non-Escaping closures.
16. Socket.IO
17. APNS work flow
18. What king of third party use
19. What is object graph in core data?
20. What is data persistance?
21. How could one simulate an NSManagedObject (dynamic properties)
22. High Order Functions(Map,FlatMap,ComapctMap)
23. How can I update my location every 100 meters?
24. What is a singleton design pattern? write code to declare a singleton
25. Difference between ARC and MRC
26. What is the difference between a serial and a concurrent queue?
27. **What will this code print :**

```
int a=0;
int b=0;

NSLog(@"step 1 value of a : %d, value of b : %d",a++,++b);
NSLog(@"step 2 value of a : %d, value of b : %d",++a,++b);
NSLog(@"step 3 value of a : %d, value of b : %d",++a,b++);
NSLog(@"step 4 value of a : %d, value of b : %d",a++,b++);
```

28. **What will this code print and why?**

```
var thing = "cars"

let closure = { [thing] in
  print("I love \(thing)")
}
```

thing = "airplanes"

closure()

29. Here's a model of a thermometer as a class and a struct. The compiler will complain about the last line. Why does it fail to compile?

```
public class ThermometerClass {
  private(set) var temperature: Double = 0.0
  public func registerTemperature(_ temperature: Double) {
    self.temperature = temperature
  }
}

let thermometerClass = ThermometerClass()
thermometerClass.registerTemperature(56.0)

public struct ThermometerStruct {
  private(set) var temperature: Double = 0.0
  public mutating func registerTemperature(_ temperature:
Double) {
    self.temperature = temperature
  }
}

let thermometerStruct = ThermometerStruct()
thermometerStruct.registerTemperature(56.0)
```

30. Here's a function to divide two optional doubles. There are three preconditions to verify before performing the actual division:

- **The dividend must contain a non nil value.**
- **The divisor must contain a non nil value.**
- **The divisor must not be zero.**

```
func divide(_ dividend: Double?, by divisor: Double?) -> Double?
{

  if dividend == nil {
    return nil
  }
  if divisor == nil {
    return nil
  }
  if divisor == 0 {
    return nil
  }
  return dividend! / divisor!
}
```

31. The following code has a compile time error. Can you spot it and explain why it happens? What are some ways you could fix it?

```
struct Kitten {
}

func showKitten(kitten: Kitten?) {
  guard let k = kitten else {
    print("There is no kitten")
  }
  print(k)
}
```

32. WAP a program finds three consecutive odd or even numbers in an array.
33. find the maximum number of in array.
34. duplicate even elements in an array. [2,4,1,5,2,3,8,9,10,4]