

Question 1:- How to call api using nsurl session

Ans.

```
let url = "https://itunes.apple.com/search?term="+searchString URLSession.shared.dataTask(with:
URL.init(string: url)!){(data,response,err) in if let responsedata = data{ DispatchQueue.main.async
{ completion(responsedata) } } }.resume()
```

Question2:- Consider this code

```
let d = ["john": 23, "james": 24, "vincent": 34, "louis": 29]

let x = d.sort{ $0.1 < $1.1 }.map{ $0.0 }
```

What is the type of x? And what is its value?

Ans:- x is of type [String] and of value ["john", "james", "louis", "vincent"].

Question3:- The following code snippet results in a compile time error:

```
struct IntStack {

    var items = [Int]()

    func add(x: Int) {

        items.append(x) // Compile time error here.

    }

}
```

Ans:- Structures are value types. By default, the properties of a value type cannot be modified from within its instance methods.

However, you can optionally allow such modification to occur by declaring the instance methods as 'mutating'; e.g.:

```
struct IntStack {

    var items = [Int]()

    mutating func add(x: Int) {

        items.append(x) // All good!
```

```
}
```

```
}
```

Question4:- What will this code print and why?

```
var thing = "cars"
```

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

```
}
```

```
thing = "airplanes"
```

```
closure()
```

Ans:- i love car

Question 5:- How to declare protocol write Example

Ans:

```
protocol Greet {
```

```
    // blueprint of property
```

```
    var name: String { get }
```

```
    // blueprint of a method
```

```
    func message()
```

```
}
```

```
// conform class to Greet protocol
```

```
class Employee: Greet {
```

```
    // implementation of property
```

```
    var name = "Perry"
```

```
// implementation of method

func message() {

    print("Good Morning", name)

}

}

var employee1 = Employee()

employee1.message()
```

Question 6: List down three ways in which we can append two arrays in ios Swift.

Let us consider that the two arrays are declared as follows:

```
var firstArray = ["Sonal", "Rahul"]

let secondArray = ["Nawaz", "Riya"]
```

Ans:- Using the method "append(contentsOf:)"- In this method, the contents of the second array are copied into the first array.

```
firstArray.append(contentsOf: secondArray)
```

- Using the "+=" operator - In this method also, the contents of the second array are copied into the first array.

```
firstArray += secondArray
```

- Appending two arrays by using the "+" operator and adding the result to a new array -

```
let thirdArray = firstArray + secondArray
```

