**What is the difference between var and let?**

The keywords “var” and “let” can declare mutable and immutable variables. The difference between “var” and “let” is that once the var variable has been initialised, it can be changed later on. However, once the let variable has been initialised, it cannot be changed.

A let variable in Swift is similar to a const variable in other programming languages, where if you were to attempt to mutate, reinitialise or reassign, it will cause a complier error.

**What is an optional?**

An optional is a variable or constant that has value or no value. The value can either be nil or not nil. Optional is executed/denoted by writing "?" after declaring a type.

**What is optional chaining vs optional binding?**

This is a process of calling and querying properties, subscripts and methods on an optional that may or may not be nil. If there's value in the optional, then the call is a success. If it is nil, however, then the call will return nil.

Optional binding is used to unwrap the optional value safely. Within a temporary constant or variable, optional value is assigned. If there’s value in the optional variable, then it will be assigned to the temporary variable.

**What are the different ways to unwrap an optional? How do they work? Are they safe?**

There are seven ways to unwrap an optional:

* Forced unwrapping – this is an unsafe way to unwrap an optional
* Implicitly unwrapping – this is unsafe in multiple cases
* Optional binding – is a safe method
* Optional chaining – is a safe method
* Nil coalescing operator – is a safe method
* Guard statement – is a safe method
* Optional pattern – is a safe method

**What is a closure?**

Closure is a type of function where is a function name is not required. It captures and stores references to any variable and constants in the context they are defined. It can be passed around like a variable.

**What is the difference between a class and a struct?**

The difference between a struct and a class is that a struct is a value type while a class is a reference type. There are two types of categories in Swift: Reference types and Value types. Value types holds a unique copy of data, such as Enum or a struct. A reference type has a single copy of data, and this type is mostly a class.

**What is the syntax '??' do?**

The syntax ‘??’ is like an else statement in Swift.

**What is a tuple?**

A tuple is a group of various values. Each value in tuple consists of different data types. For example, storing information about the name and price of a product, a tuple can be created with the value string (name) and the value float (price).

**What is Any vs AnyObject?**

Any vs AnyObject are special types that are mainly used for working with non-specific types. AnyObject represents instances of any class types, while Any represents instances of any type, such as function types and optional types.

**What is a protocol?**

Protocol is a blueprint of properties or methods which can be adopted by classes or other types. During compile time, Swift checks for protocol conformity issues, which allows developers to find bugs in the code before the program is run.