**MVVM**

**Swizzling**

**4- What is made up of NSError object?**There are three parts of NSError object a domain, an error code, and a user info dictionary. The domain is a string that identifies what categories of errors this error is coming from.

**14- What’s Completion Handler ?**  
Completion handlers are super convenient when our app is making an API call, and we need to do something when that task is done, like updating the UI to show the data from the API call. We’ll see completion handlers in Apple’s APIs like dataTaskWithRequest and they can be pretty handy in your own code.

The completion handler takes a chunk of code with 3 arguments:(NSData?, NSURLResponse?, NSError?) that returns nothing: Void. It’s a closure.

15 – Difference between frame and bounds ?

The frame relates a view’s location and size in its parent View.

The bounds relate a view’s location and size using its own coordinate system.

**What’s the difference between not-running, inactive, active, background and suspended execution states?**

* **Not running:** The app has not been launched or was running but was terminated by the system.
* **Inactive:** The app is running in the foreground but is currently not receiving events. (It may be executing other code though.) An app usually stays in this state only briefly as it transitions to a different state.
* **Active:** The app is running in the foreground and is receiving events. This is the normal mode for foreground apps.
* **Background:** The app is in the background and executing code. Most apps enter this state briefly on their way to being suspended. However, an app that requests extra execution time may remain in this state for a period of time. In addition, an app being launched directly into the background enters this state instead of the inactive state.
* **Suspended:** The app is in the background but is not executing code. The system moves apps to this state automatically and does not notify them before doing so. While suspended, an app remains in memory but does not execute any code. When a low-memory condition occurs, the system may purge suspended apps without notice to make more space for the foreground app.
* **What's the difference between using a *delegate* and *notification*?**
* Both are used for sending values and messages to 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 \_delegate\_s 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.
* [**NSNotificationCenter** defaultCenter]
* postNotificationName:@"TestNotification"
* object:**self**];
* 12 – What are NSNotificationCenter and how does it work ?
* NSNotificationCenter is what Apple has provided as 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 of 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 have to know about the broadcaster.