**NSLocalizedString**

**Question:** What is NSLocalizedString? What does it do? When it should be used?

**What is it?**

**Localization means Transalation. The ability of your app to support multiple language.**

**NSLocalizedString** is a Foundation macro that returns a string according to the locale of the device. It has two arguments: key, which uniquely identifies the string to be localized, and comment , a string that is used to provide sufficient context for accurate translation.

**What does it do?**

**It allows people who speaks different languages to use your app. This way your app can be promoted globally as opposed to just your local region.**

For example, if I am writing an app which have an English and Turkish version, then we should have at least two version of the localizable.strings file, one for English and another one for Turkish. The format is the same for both files.

**When it should be used?**

I would say, use it all the time to cover the languages of your target markets. You don’t have to extend it to markets that you’re are sure your app will never be used or accessed.

**Subclassing**

**Question:** What is subclassing? List the outstanding questions that you have. What do you think the answers are?

**What is subclassing?**

Subclassing is just a subclass of another class and it inherits behavior and members of its parent class.

For example, if we have a "car" class which has properties that are common to all cars such color, tires, engine, etc. It also has behaviors that are common to all cars such as drive forward, drive backwards, stop, etc.

Now for some reason we need to differentiate between different types of cars but we don't want to repeat all those properties and behaviors in each new class so we can write subclasses of the "car" class.

For example, we can have a subclasses such as "manual cars", "automatic cars", "convertibles", "electric cars", etc. All of these subclasses have some common traits inherited from the "car" class but will also have some unique characteristics that can be included in the subclass.

Using subclass also help override some default values which we may have in our parent or super class. For example, if our car class has variable called “noofcylinders” which is equal to 6. But if we create a subclass from the class car for Mazda, we are able to override the “noofcylinder” inside that subclass to 4.

**Outstanding Questions**

**Question 1**

The viewController.m file in the storyboard can contain both @interface and @implementation ONLY if the properties in that view/object is not supposed to be used by any other objects or basically private to that Object. If any of those views properties or methods are supposed to be accessible from other objects, then the properties and methods MUST be defined in the header(viewController.h) file and only implementations of those methods should exist in the implementation file .m file.

I think this is true? Do you have anything else to add to this?

**Steve, can you please give me solid example of an object/view which always declared and used privately through the viewController.m file?**

**Question 2**

I think subclasses are a good example of private declarations. For example, the overridden/special values of a subclass would only be used in that subclass, hence, we can declare the properties and method definition as well as the implemtnation of a subclass in the .m file?

Whereas the class will always be declared publicly as it is the superclass which MUST be visible to all its subclasses and not only to its own class.

**I believe this is true. What do you think Steve? If my answer is not correct, can you tell me why?**

**Question 3**

resignFirstResponder is suppose to free up a resource so it can be used by another object. In our alcolator example, I have noticed that some of the IBAction have that argument in the method and some don’t.

**WHY?**

For example, textFieldDidChange doesn’t have that argument but sliderValueDidChange does have it?

How to know when to use the regsinFirstResponder in my method?

What’s its usage based on?