Reading Assignment III: The Rest of Swift

Objective

Time to finish off your reading assignments for the quarter by reading the rest of the Swift Programming Language document.

Materials

- All of the reading comes from this Swift Programming Language document.

Sections to Read

Below are chapters which were in gray in previous assignments (or chapters that were not even mentioned in previous assignments).

In this document yellow means “not that important” and red means “very important.”

Concepts that are important will be discussed and usually demoed in lecture in this course. So if you read a section and your head is spinning, don’t panic until you see it come up in lecture and still don’t understand it. Many of these remaining topics (the ones in yellow) are things that you will probably not even need this quarter (but it is important to know that they exist).

Also note that the Swift Programming Language document is a reference document. You can always go back to it and read up on something. As long as you know some topic exists (e.g. optional chaining, subscripts or generics) and you have a general idea what it’s about, you will know to revisit this document when you are confronted with it or an occasion to use that feature arises.

Subscripts

Subscript Syntax
Subscript Options

Initialization

Class Inheritance and Initialization
Failable Initializers
Required Initializers

Deinitialization

This is almost never needed, but you can read up on it for completeness’ sake.

Automatic Reference Counting

Don’t freak out about this too much. Reference cycles are not that common of an occurrence. However, understanding how to keep a closure from having a strong reference to something that has a strong reference to it is important, so …

Strong Reference Cycles for Closures

Optional Chaining

Entire Chapter

Nested Types

Entire Chapter

Extensions

Don't go crazy with this! It's a cool feature, but it can be abused. Remember that readability of your code is most important. If you add too many weird extensions or extend something in a way that is non-intuitive, you will lose readers of your code.

Protocols

Entire Chapter

This is very important. It will be covered and demoed in lecture.

Generics

Entire Chapter

Access Control

Entire Chapter. We will not be creating our own frameworks this quarter, so you will never use `public`, but again, it's good to know about. However, you must always use `private` appropriately in this course.

Advanced Operators

Entire Chapter