

Assignment VI:

Persistent Image Gallery

Objective

The goal of this assignment is to understand `FileManager`, `URL`, `Codable`, `UIDocument` and `UIDocumentBrowserViewController` and to learn how to use an iOS API entirely on your own from the documentation.

This assignment uses the code you built for Assignment 5, but you may want to start a new Xcode project from scratch (so you can use the Document template).

This assignment must be submitted using [the submit script described here](#) by the start of lecture next Wednesday (i.e. before lecture 16). You may submit it multiple times if you wish. Only the last submission before the deadline will be counted.

Be sure to review the Hints section below!

Also, check out the latest in the Evaluation section to make sure you understand what you are going to be evaluated on with this assignment.

Materials

- Your code from last week will be required.
-

Required Tasks

1. Image Gallery documents in your application must now be persistent.
2. You can entirely remove the `UITableView` support you added last week.
3. Instead, use the `UIDocumentBrowserViewController` to support the manipulation of Image Gallery documents (naming them, opening them, creating them, moving them around, etc.).
4. Use `Codable` to represent your document using `JSON` or `Property List` format.
5. Cache the images in all of your Image Galleries locally using `URLCache` up to a reasonable limit of file system usage (explain what limit you chose in comments in your code somewhere).
6. Your application should work on both iPad (full functionality) and iPhone (rearranging the images in documents only, not adding images).
7. You are required to get this application running on an actual device this week (i.e. not just in a simulator).

Hints

1. Fewer hints this week because it's time to start being able to figure stuff out on your own as you get close to starting on your final project.
2. Dragging in a `UICollectionView` is disabled on iPhone by default. Set `dragInteractionEnabled` to `true` to enable it.
3. You are intentionally being asked to do something that was not even mentioned in lecture (`URLCache`).
4. The Simulators do not appear to reliably support documents on the local device, so log into iCloud on your Simulator(s) and use iCloud Drive as the storage location to test your application.

Things to Learn

Here is a partial list of concepts this assignment is intended to let you gain practice with or otherwise demonstrate your knowledge of.

1. Codable
2. FileManager
3. URL
4. UIDocument
5. UIDocumentBrowserViewController
6. URLCache
7. Running on an iOS device (rather than the simulator)

Evaluation

In all of the assignments this quarter, writing quality code that builds without warnings or errors, and then testing the resulting application and iterating until it functions properly is the goal.

Here are the most common reasons assignments are marked down:

- Project does not build.
- One or more items in the Required Tasks section was not satisfied.
- A fundamental concept was not understood.
- Project does not build without warnings.
- Code is visually sloppy and hard to read (e.g. indentation is not consistent, etc.).
- Violates MVC.
- UI is a mess. Things should be lined up and appropriately spaced to “look nice.”
- Improper object-oriented design including proper use of value types versus reference types.
- Improper access control (i.e. `private` not used appropriately).
- Your solution is difficult (or impossible) for someone reading the code to understand due to lack of comments, poor variable/method names, poor solution structure, long methods, etc.

Often students ask “how much commenting of my code do I need to do?” The answer is that your code must be easily and completely understandable by anyone reading it. You can assume that the reader knows the iOS API, but should not assume that they already know your (or any) solution to the assignment.

Extra Credit

We try to make Extra Credit be opportunities to expand on what you've learned this week. Attempting at least some of these each week is highly recommended to get the most out of this course. How much Extra Credit you earn depends on the scope of the item in question.

If you choose to tackle an Extra Credit item, mark it in your code with comments so your grader can find it.

1. Set a sensible thumbnail image for your documents. It probably doesn't make sense to take an actual snapshot of your entire collection view! Do something else.
2. Make your documents open from the Files application.
3. Support dragging in `UIImage`s that don't have a valid image URL. Store the dragged-in `UIImage` into the filesystem and create your own URL to it. It's fine if the images can't be seen on other devices (when the document is stored in iCloud Drive). Be careful to remember that an absolute path URL (i.e. that starts with `/`) does not work from launch to launch of your application. You must always start by asking the `FileManager` for the URL of a sandbox directly and build *relative* paths from there. You'll have to account for this in your document data structure.