

Incorporating Video into Your iOS App

Introduction

This tutorial should take 20 to 30 minutes to complete. Yes, it's really quick.

Apple does not have a widget or view for playing a video in. Instead, starting a video using the **AVPlayerViewController** will cause a segue to a popup where your video will play full screen. So, you'll probably want to have a button that initiates the popup and playback of your video.

You can play a video that is **on the internet** through its URL. You can also play a video that you store as a **local file** within your app. (Just like you store other files such as images, sounds, and fonts within your app.)

Resources

1. [Building a Basic Playback App \(Apple Developer Web Site\)](#)
2. [How to play video with AVPlayerViewController \(AVKit\) in Swift](#)
3. [How to play a local video with Swift?](#)
4. [This video file that you can include directly in your app](#)

Part 1

1. Do the iOS section of Tutorial #1 above. Note that the page has instructions for both iOS and MacOS. Do only the iOS portion, about half the web page.
2. When you get to the middle of the web page, and it says "Your app is complete" and then it goes on to Mac OS, you're finished with this step.
3. Run the app and verify that the video plays when you tap the button.

Part 2

1. In the app you already made, add a second button that plays this video instead: <https://samplelib.com/lib/preview/mp4/sample-5s.mp4>.
2. You *could* make a second action method to play the video, but instead, just use the *same* action method, and distinguish between the buttons using their tags. When you tap the button with the other tag, it should trigger the playback of the alternate video URL.
3. Run the app and verify that both videos play.

Part 3

1. Download the nature.mp4 video file and drag it into your project. Make sure to check the "Add to targets" checkbox.
2. Look at the project settings (blue folder at the top of the Project navigator). Look at the Build Phases. Look at the Copy Bundle Resources. Check that nature.mp4 is in the list. (See the tutorial for pictures of how this works.)
3. Create a third button to play the local nature.mp4 file.
4. Create an action for the third button and use Tutorial #3 above to write the code. See the green checkmark for the appropriate code to write. It's best to type in this code instead of copy and paste. The code to play a local file is similar to the code for playing a remote URL. It is also similar to the code for playing audio (c.f. the Noise Maker project).
5. Run the app and verify that all three videos play properly.

