

Declarative and Imperative programming are styles of code. Declarative is what you see in SwiftUI (there are still some things in SwiftUI that use imperative coding). It's a lot more simple, easier to read, and much more manageable. Imperative is more detailed, so you have to give it step-by-step instructions for everything. People often prefer Declarative more because it's a lot easier to read and causes fewer headaches. SwiftUI uses declarative programming, so it will automatically update the UI when the state changes. For Imperative in UIKit, you must manually update UI elements when data changes. In Imperative, you manually create and configure views; for Declarative, you just describe what the UI should look like.

An analogy for this would be: if you were at a restaurant and ordered a meal, the restaurant would figure out how to cook it, what to use, and how to serve it – that would be declarative programming. Imperative would be like you are the chef, figuring out how to do each action.

//What is Declarative programming?

//How does it compare to imperative?

//Why do people not often prefer it?

//Declarative is just a different style of code making

//What is a analogy(New one) for imperative and declarative.?

//How does it relate to SwiftUI?

//How can you write code that matches this pattern?