

Decorator



Karoly Nyisztor

DEVELOPER

@knyisztor www.leakka.com

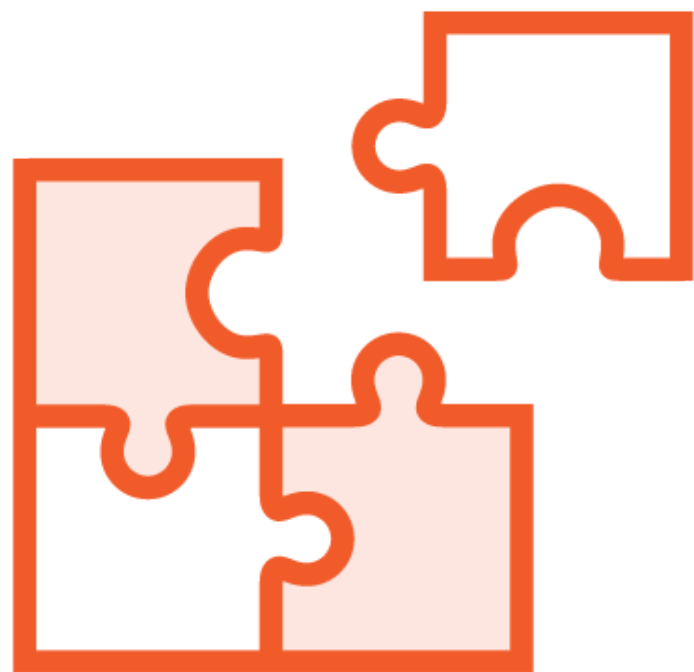
Overview

Motivation

UIColor Extensions

RoundedLabel

Motivation



Add new behavior dynamically

- Extends functionality without modifying existing type

Decorator can be implemented via

- Object wrapping
- Swift extensions

Decorator

Attaches additional responsibilities to objects dynamically, without modifying the behavior of other instances of the same class.

Demo

UIColor Extension

RoundedLabel

Summary

The Decorator Pattern:

- Allows changing the behavior of objects
- Extends type functionality without modifying the type's code
- Implement via wrapping or use Swift extensions

Pitfalls

- Replace existing functionality
- Change the original purpose of a type