Barebones CodeKata - Minimal Working Version

Step 1: Create Xcode Project

1. Open Xcode
2. Create new iOS App
3. Name: "CodeKata"
4. Interface: SwiftUI
5. Language: Swift
6. iOS 17.0+

Step 2: Replace Default Files

CodeKataApp.swift

```swift
import SwiftUI

@main
struct CodeKataApp: App {
    var body: some Scene {
        WindowGroup {
            ContentView()
        }
    }
}
```

ContentView.swift

```swift
import SwiftUI

struct ContentView: View {
    @State private var challenges: [Challenge] = Challenge.sampleData

    var body: some View {
        NavigationView {
            List(challenges) { challenge in
                NavigationLink(destination: ChallengeDetailView(challenge: challenge)) {
                    VStack(alignment: .leading, spacing: 4) {
                        Text(challenge.title)
                            .font(.headline)
                        Text(challenge.description)
                            .font(.caption)
                            .foregroundColor(.secondary)
                            .lineLimit(2)
                    }
                    .padding(.vertical, 2)
                }
            }
            .navigationTitle("CodeKata")
        }
    }
}

#Preview {
    ContentView()
}
```

Step 3: Add Basic Models

Challenge.swift

swift

```
import Foundation

struct Challenge: Identifiable {
    let id = UUID()
    let title: String
    let description: String
    let difficulty: String

    static let sampleData = [
        Challenge(
            title: "Two Sum",
            description: "Find two numbers that add up to target",
            difficulty: "Easy"
        ),
        Challenge(
            title: "Reverse String",
            description: "Reverse a string in-place",
            difficulty: "Easy"
        ),
        Challenge(
            title: "Binary Tree Traversal",
            description: "Traverse a binary tree in order",
            difficulty: "Medium"
        ),
        Challenge(
            title: "Dynamic Programming",
            description: "Solve using dynamic programming",
            difficulty: "Hard"
        )
    ]
}
```

Step 4: Add Challenge Detail View

ChallengeDetailView.swift

```
swift
```

```
import SwiftUI

struct ChallengeDetailView: View {
  let challenge: Challenge
  @State private var userCode = ""
  @State private var showingAlert = false

  var body: some View {
    ScrollView {
      VStack(alignment: .leading, spacing: 20) {
        // Challenge info
        VStack(alignment: .leading, spacing: 8) {
          Text("Difficulty: \(challenge.difficulty)")
            .font(.subheadline)
            .foregroundColor(.blue)

          Text(challenge.description)
            .font(.body)
        }

        // Code input
        VStack(alignment: .leading, spacing: 8) {
          Text("Your Solution:")
            .font(.headline)

          TextEditor(text: $userCode)
            .font(.system(.body, design: .monospaced))
            .border(Color.gray.opacity(0.3))
            .frame(minHeight: 200)
        }

        // Submit button
        Button("Submit Solution") {
```

```
                    showingAlert = true
                }
                .buttonStyle(.borderedProminent)
                .frame(maxWidth: .infinity)
            }
            .padding()
        }
        .navigationTitle(challenge.title)
        .navigationBarTitleDisplayMode(.large)
        .alert("Solution Submitted!", isPresented: $showingAlert) {
            Button("OK") { }
        } message: {
            Text("Great job! Your solution has been submitted.")
        }
    }
}

#Preview {
    NavigationView {
        ChallengeDetailView(challenge: Challenge.sampleData[0])
    }
}
```

That's It!

This gives you:

- ✅ **4 files total** (App, ContentView, Challenge, ChallengeDetailView)
- ✅ **List of coding challenges**
- ✅ **Challenge detail with code editor**
- ✅ **Navigation between screens**

- ✅ **Sample data to test with**
- ✅ **Compiles and runs immediately**

File Structure:

```
CodeKata/
├── CodeKataApp.swift
├── ContentView.swift
├── Challenge.swift
└── ChallengeDetailView.swift
```

To Test:

1. Create the Xcode project
2. Replace/add these 4 files
3. Build and run (⌘R)
4. Tap any challenge to see the detail view
5. Type code and submit

No dependencies, no complex setup, no SwiftData, no Fastlane - just pure SwiftUI basics that work out of the box.

Want to add anything specific to this minimal version?