Flutter Interview Tip #1
Stateless vs Stateful Widgets

Overview

Understanding the difference between Stateless and Stateful widgets is one of the most common Flutter interview questions. These two define how your UI reacts to data and user interactions.

Stateless Widget

A StatelessWidget is immutable - once it's built, it cannot change during runtime. It is perfect for static UI components that remain constant regardless of user interaction or data changes.

Examples: Text labels, icons, logos, static layouts.

Key Traits:

- Builds only once.

- No internal state or data changes.

- Ideal for performance and static views.

Stateful Widget

A StatefulWidget is dynamic - it can rebuild its UI when its state changes. It's used when your interface reacts to inputs, animations, or data updates.

Examples: Forms, counters, animations, input fields.

Key Traits:

- Can rebuild multiple times using setState().

- Maintains internal data.

- Used for interactive or real-time updates.

Flutter Interview Tip #1
Stateless vs Stateful Widgets

Created by Gul Faraz Ahmed | Follow for more Flutter Tips & Packages

Comparison Table

Stateless Widget	Stateful Widget
Immutable - doesn't change after build	Mutable - changes dynamically
Builds only once	Can rebuild multiple times
No internal state	Has internal state data
Used for static content	Used for interactive content
Fast and efficient	Slightly heavier due to state tracking

Interview Insight

In interviews, don't just say 'Stateless is static and Stateful is dynamic.' Explain that Flutter rebuilds the widget tree - not a single widget - to maintain UI consistency efficiently.

Quick Summary

- StatelessWidget -> Use for fixed UI components.

- StatefulWidget -> Use when UI depends on user input or data.

- Keep logic separate from UI for cleaner, maintainable code.