Flutter Interview Guide (2025 Edition)

By ChatGPT | Comprehensive Q&A; for Flutter Developers

Beginner Level

■ What is Flutter?

➡■ Flutter is an open-source UI framework by Google for building natively compiled applications for mobile, web, and desktop from a single codebase using Dart.

■ What is Dart and why does Flutter use it?

➡■ Dart is an object-oriented language developed by Google. Flutter uses Dart because it compiles to native code and supports hot reload, fast development, and high performance.

■ What is the difference between StatelessWidget and StatefulWidget?

➡■ StatelessWidget does not change once built, while StatefulWidget maintains mutable state that can change during the widget's lifecycle.

■ What is Hot Reload?

➡■ Hot Reload allows developers to instantly see changes in the app without restarting it, retaining the app state.

Intermediate Level

■ What is Provider in Flutter?

➡■ Provider is a state management solution that allows widgets to efficiently rebuild when data changes, using ChangeNotifier as the base.

■ What is FutureBuilder?

➡■ FutureBuilder builds a widget based on the latest snapshot of a Future. It’s commonly used for handling asynchronous API calls.

■ How do you improve app performance in Flutter?

➡■ Use const constructors, minimize widget rebuilds, avoid unnecessary layouts, use efficient state management, and profile with DevTools.

Advanced Level

■ What are Platform Channels?

➡■ Platform Channels are used for communication between Flutter and native Android/iOS code using method calls.

■ Explain BLoC architecture.

➡■ BLoC (Business Logic Component) separates business logic from UI using Streams and Sinks, promoting a reactive data flow.

■ What is Navigator 2.0?

➡■ Navigator 2.0 provides a declarative approach to navigation, allowing full control over navigation stack and URL-based routing.

Dart Language Deep Dive

■ What are Mixins in Dart?

➡■ Mixins allow code reuse in multiple class hierarchies without inheritance. They use the 'with' keyword.

■ Explain async, await, and Future in Dart.

➡■ Future represents a value that will be available later. async marks a function asynchronous, and await pauses execution until the Future completes.

■ What is Null Safety in Dart?

➡■ Null safety ensures variables cannot contain null unless explicitly declared nullable with '?', preventing runtime null errors.

Practical / Scenario-Based

■ How do you implement real-time chat in Flutter?

➡■ Use WebSockets or Socket.IO for live data exchange, maintaining a Stream of messages updated in real-time.

■ How to handle offline data in Flutter?

➡■ Use local storage solutions like Hive, SQLite, or SharedPreferences and sync with the server when back online.

■ How to handle large lists efficiently?

➡■ Use ListView.builder for lazy loading, pagination, and caching images to prevent jank.