1. What is flutter? features and benefits of flutter?

Flutter is a versatile framework developed by Google in 2015 that allows developers to build applications for Android, iOS, web, and desktop using a single codebase. Its key feature is its low development time, made possible by tools like **hot reload** and **hot restart**, which enable developers to see real-time changes without restarting the app.  
The main benefit of Flutter is its ability to create visually attractive applications with high performance while ensuring platform consistency. Additionally, it offers an extensive set of pre-built widgets and integrates well with various platforms, making development efficient and seamless.

1. What is dart? why should we use dart as the programming language?

Dart is a modern programming language designed by Lars and Kasper and developed by Google. It is object-oriented and similar to Java in terms of syntax, making it easier for developers with Java experience to adapt quickly.  
Dart is specifically optimized for UI development with frameworks like Flutter. Its fast compilation to native code and support for asynchronous programming make it an ideal choice for building high-performance, cross-platform applications.

1. How is the basic program of dart written?

Dart programs typically start with the main.dart file, located in the lib folder of a Flutter project. The program execution begins with the main() function, which serves as the entry point of the application. Other functions and methods can be defined and called within this structure, ensuring a modular and organized codebase.

1. What do you mean by widgets?

Widgets are the fundamental building blocks used to construct the user interface (UI) of a Flutter app. They define the visual and functional elements of the app. Examples of widgets include **Container**, **Column**, **Row**, **ListView**, **ElevatedButton**, and many more.  
Widgets can be classified as **stateless** or **stateful**, depending on whether their state changes during runtime.

1. What do you mean by stateless widgets?

A **stateless widget** is a widget that remains immutable throughout its lifecycle. Once it is built, its appearance and properties cannot be dynamically updated. Stateless widgets are typically used for UI elements that do not require user interaction or data changes.  
For example, a static text label or an icon can be implemented as a stateless widget.

1. What is stateful widgets?

A **stateful widget** is a widget that can update its appearance dynamically based on user interaction or changes in data. It holds a mutable "state" object, and whenever the state is updated, the widget rebuilds itself to reflect the changes.  
Stateful widgets are commonly used for features like forms, animations, or any element that involves interactivity.

1. what is the structure of files in flutter?

The file structure in Flutter is well-organized and includes the following:

* The **entry point** is main.dart, located in the lib folder. This file contains the main() function, where the application execution begins.
* All dependencies are managed in the pubspec.yaml file under the dependencies: section. This file is also used to configure assets and packages.
* To include assets like images or fonts, a dedicated **assets folder** should be created at the root of the project. The assets must also be listed in the pubspec.yaml file to make them accessible in the project.