Module - 1

1. Benefits of Using Flutter Over Other Cross-Platform Frameworks:

- One Codebase for All Platforms: Write once and run on Android, iOS, web, desktop — no need to write separate code for each.
- Fast Development: "Hot Reload" lets you see changes instantly without restarting the app.
- Beautiful UI: Comes with lots of built-in widgets that make it easy to create nice-looking apps.
- High Performance: Flutter apps run fast because they are compiled to native code.
- Strong Community & Backing: Made by Google and has a large, helpful developer community.
- Customizable Everything: You can create your own widgets easily, which gives you full control over the design.

---

2. Role of Dart in Flutter & Its Advantages:

- **What is Dart?**
- Dart is the programming language used to write Flutter apps.
- **Why Dart is Good for Mobile Development:**
  - <u>Fast:</u> Dart compiles to native code, which helps Flutter apps run smoothly.
  - <u>Easy to Learn:</u> Dart has a clean and simple syntax, similar to JavaScript or Java.

- ○ <u>Hot Reload Support</u>: Dart works perfectly with Flutter's hot reload feature.
- ○ <u>Built for UI</u>: Dart was designed for building user interfaces, so it's perfect for Flutter.

---

3. Steps to Set Up a Flutter Development Environment:

1. <u>Download Flutter from the official site:</u> https://flutter.dev
2. Extract the Flutter SDK to a folder on your computer.
3. Add Flutter to PATH (so you can use it in terminal/command prompt).
4. Install a Code Editor like Visual Studio Code or Android Studio.
5. Install Flutter and Dart Plugins in your editor.
6. Run flutter doctor in the terminal to check for missing tools (like Android SDK).
7. Set Up an Emulator or use a physical phone for testing.
8. Create Your First App using flutter create my_app.

---

4. Basic Flutter App Structure:

- main.dart: This is the main file of your app. It's where your app starts.
- <u>main() Function:</u>
  - ○ It's the starting point of every Dart program.
  - ○ Example:
  - ○ void main() {

- ○ 　runApp(MyApp());
- ○ }
- **Widget Tree:**
  - ○ Everything in Flutter is a widget (buttons, text, layouts, etc.).
  - ○ Widgets are arranged in a tree (some widgets contain other widgets).
  - ○ Example:
  - ○ class MyApp extends StatelessWidget {
  - ○ 　@override
  - ○ 　Widget build(BuildContext context) {
  - ○ 　　return MaterialApp(
  - ○ 　　　home: Scaffold(
  - ○ 　　　　appBar: AppBar(title: Text('Hello')),
  - ○ 　　　　body: Center(child: Text('Welcome to Flutter')),
  - ○ 　　　），
  - ○ 　　);
  - ○ 　}
  - 　}