Theory Assignments:

Q. 1. Explain the benefits of using Flutter over other cross-platform frameworks.

A. This feature allows Flutter to write the code once, maintain, and can run on multiple platforms like IOS, ANDROID, WEB DESKTOP. It saves the time, effort, and money of the developers.

Q. 2. Describe the role of Dart in Flutter. What are its advantages for mobile development?

A. It enables Flutter to achieve high performance, cross-platform compatibility and a fast development cycle through features like HOT RELOAD. Dart strengths lie in its ability to be compiled to native code(AOT) for production and its just-in-time(JOT) compilation for faster development cycle.

Q. 3. Outline the steps to set up a Flutter development environment.

A. **1. Install Flutter SDK**
Windows, macOS, or Linux

1. Go to the official Flutter website: https://flutter.dev/docs/get-started/install
2. Download the appropriate Flutter SDK zip file for your OS.
3. Extract the zip file to a desired location (e.g., `C:\src\flutter` on Windows or `~/development/flutter` on macOS/Linux).
4. **Add Flutter to the system path**:
    - Windows: Add `flutter\bin` to the PATH environment variable.
    - macOS/Linux: Add `export PATH="$PATH:pwd/flutter/bin"` to your shell startup file (`.zshrc`, `.bashrc`, etc.)

---

2. Run **`flutter doctor`**
This command checks your environment and displays any missing dependencies.
3. Install Required Dependencies
Based on `flutter doctor` output:

- Install **Git** (if not already installed).
- Install **Android Studio** or another IDE with Flutter support.
- Accept Android licenses (run `flutter doctor --android-licenses`).
- Install **Xcode** on macOS for iOS development.
- Install **Chrome** for web development (optional).

4. Set Up an Editor
Flutter works well with:

- **Android Studio**
    - Install Flutter and Dart plugins from the plugin marketplace.
- **Visual Studio Code**
    - Install Flutter and Dart extensions from the Extensions view.

---

5. Set Up Emulators/Devices

- **Android Studio**: Create Android Virtual Devices (AVDs) via AVD Manager.
- **Xcode**: Use the iOS Simulator (macOS only).
- Or connect a physical device via USB with developer mode enabled.

Q. 4. Describe the basic Flutter app structure, explaining `main.dart`, the `main` function, and the widget tree.

A. 1. `main.dart`

- This is the **entry point** of every Flutter app.
- Located in the `lib/` directory.
- Contains the `main()` function and the root of the widget tree.

`lib/main.dart`
→ This file bootstraps the app and defines its basic structure.

---

2. The `main()` Function

The `main()` function is the starting point for any Dart/Flutter program. It calls `runApp()` to initialize the app and attach it to the screen.

```
dart
CopyEdit
void main() {
  runApp(MyApp());
}
```

- `runApp()` takes a **widget** and makes it the root of the app.
- `MyApp` is usually a custom class extending `StatelessWidget` or `StatefulWidget`.

---

3. Widget Tree

Flutter apps are built using a **widget tree**, a hierarchical structure of widgets.

Example:

```
dart
CopyEdit
class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'My Flutter App',
      theme: ThemeData(primarySwatch: Colors.blue),
      home: MyHomePage(),
    );
  }
}
```

Breakdown:

- **`MaterialApp`**: A widget that wraps the entire app with Material Design.
- **`home:`**: The default screen shown on app launch.
- **`MyHomePage`**: A custom widget, usually a screen, like a homepage or dashboard.

Widget Tree Visualization

```
scss
CopyEdit
MyApp (StatelessWidget)
└── MaterialApp
    └── MyHomePage (StatefulWidget)
        └── Scaffold
            ├── AppBar
            └── Body (e.g., Center, Column, Text)
```

- Widgets are **nested** inside each other.
- UI is built by composing smaller widgets into larger structures.

➢ **Stateless vs Stateful Widgets**

- **StatelessWidget**: UI that doesn't change over time.
- **StatefulWidget**: UI that can update in response to user input, timers, or other events.