Mobile Programming

Dart Refresher Workshop and Flutter Setup

Dart Essentials | Environment Setup | First Flutter App

Workshop Agenda

1. Review key concepts and common mistakes in **Dart**
2. Explore **Null Safety** and data modeling
3. Review asynchronous programming (**Future / async / await**)
4. Set up the **Flutter** development environment and use `flutter doctor`
5. Troubleshoot common installation issues (Gradle, access restrictions, SDK Tools)
6. Understand the Flutter project directory structure and `pubspec.yaml`
7. Compare **Hot Reload** and **Hot Restart**
8. Review and run the workshop starter code

1. Null Safety in Practice

```dart
String title = "Mobile Programming"; // Non-nullable
String? nullableName;                // Nullable

void greetUser(String? name) {
  // Use ?? to provide a default value
  final cleanName = name?.trim() ?? "Guest student";
  print("Hello $cleanName");
}
```

- **Golden rule:** Avoid using the `!` (null assertion) operator whenever possible. Use it only when you are completely certain that the value is not null.

2. Environment Setup and Troubleshooting with Flutter Doctor

```
# Check the complete development environment status
flutter doctor -v

# Accept Android licenses
flutter doctor --android-licenses
```

Command-line Tools Error

In Android Studio, open **SDK Manager > SDK Tools** and enable `Android SDK Command-line Tools (latest)`.

Gradle Stuck During Setup

During the first run, Gradle needs to download dependencies. If access is restricted, configure a suitable DNS service (such as Shekan or Radar) or a system-wide circumvention tool.

3. Common Command-Line Interface (CLI) Commands

```
# Create a new project
flutter create mp_starter_app

# Fetch packages declared in pubspec.yaml
flutter pub get

# List available emulators and connected devices
flutter devices

# Run the project
flutter run
```

4. Hot Reload vs. Hot Restart

- **Hot Reload (key `r`):**
  - Injects code changes into the Dart virtual machine, usually in less than a second.
  - **Preserves the current application state** (existing variable values are not cleared).
- **Hot Restart (key `R`):**
  - Restarts the application from the `main()` function.
  - **Resets the state** and returns variables to their initial values. This is useful when changing `initState` or global variables.