

फ्लटर ऐप

यह ब्लॉग पोस्ट एक `pubspec.yaml` प्रोजेक्ट, संभवतः एक निजी ब्लॉग एप्लीकेशन, पर चर्चा करता है। दी गई फ़ाइल संरचना एक मानक `pubspec.yaml` प्रोजेक्ट सेटअप का सुझाव देती है, जिसमें प्लेटफ़ॉर्म-विशिष्ट निर्देशिकाएँ (`android`, `ios`, `linux`, `macos`, `web`) और कोर `pubspec.yaml` फ़ाइलें (`pubspec.yaml`, `pubspec.lock`) शामिल हैं। विशिष्ट विवरणों की अनुपस्थिति के कारण एक सामान्य अवलोकन की आवश्यकता है।

एक सामान्य `pubspec.yaml` प्रोजेक्ट में विजेट्स के साथ उपयोगकर्ता इंटरफ़ेस बनाना, एप्लीकेशन की स्थिति का प्रबंधन करना, उपयोगकर्ता इनपुट को संभालना और प्लेटफ़ॉर्म-विशिष्ट सुविधाओं या बाहरी `pubspec.yaml` के साथ एकीकरण करना शामिल है। `main.dart` फ़ाइल प्रवेश बिंदु के रूप में कार्य करती है, जो एप्लीकेशन के प्रारंभिक विजेट ट्री को परिभाषित करती है। `pubspec.yaml` फ़ाइल निर्भरताओं और प्रोजेक्ट मेटाडेटा का प्रबंधन करती है।

इस प्रोजेक्ट का सोर्स कोड `pubspec.yaml` फ़ाइल निर्भरताओं और प्रोजेक्ट मेटाडेटा का प्रबंधन करती है।

इस `pubspec.yaml` प्रोजेक्ट के लिए प्रमुख विचार शामिल हैं:

- **विकास वातावरण:** क्रॉस-प्लेटफ़ॉर्म विकास के लिए `pubspec.yaml` और `pubspec.lock` दोनों स्थापित सुनिश्चित करें।
- **परीक्षण:** विभिन्न प्लेटफ़ॉर्म पर एप्लीकेशन का पूरी तरह से परीक्षण करने के लिए भौतिक या वर्चुअल डिवाइस कनेक्ट करें।
- **पूर्व अनुभव:** `pubspec.yaml` और `pubspec.lock` विकास सिद्धांतों से परिचित होना फायदेमंद होगा।

फ़ाइल निर्देशिकाएँ:

```
.
├── README.md
├── analysis_options.yaml
├── android
│   ├── app
│   ├── build.gradle
│   ├── gradle
│   ├── gradle.properties
│   ├── gradlew
│   ├── gradlew.bat
│   ├── local.properties
│   ├── lzwjava_blog_android.iml
│   └── settings.gradle
├── build
│   ├── 26c07c686c162683d91db277284f9499
│   ├── app
│   ├── cache.dill.track.dill
│   ├── flutter_assets
│   ├── macos
│   ├── native_assets
│   ├── path_provider_android
│   └── web
└── ios
```

```
Flutter
Podfile
Runner
Runner.xcodeproj
Runner.xcworkspace
RunnerTests
lib
  main.dart
linux
  CMakeLists.txt
  flutter
  runner
lwjava_blog.iml
macos
  Flutter
  Podfile
  Podfile.lock
  Pods
  Runner
  Runner.xcodeproj
  Runner.xcworkspace
  RunnerTests
pubspec.lock
pubspec.yaml
test
  widget_test.dart
web
  favicon.png
  icons
  index.html
  manifest.json
windows
  CMakeLists.txt
  flutter
  runner
```

कोड:

```
import 'package:flutter/material.dart';
```

```
void main() {
```

```

runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Hello World',
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
        useMaterial3: true,
      ),
      home: const Scaffold(
        body: Center(
          child: Text('Hello World!'),
        ),
      ),
    );
  }
}

```