

# Flutter UI 개발 가이드

---

## Google Fonts와 RichText를 활용한 사용자 인터페이스 개발

## 학습목표

- Google Fonts 패키지를 활용한 커스텀 폰트 적용 방법 습득
- RichText 위젯을 사용한 고급 텍스트 스타일링 구현
- 다양한 플랫폼 환경에서의 UI 개발 방법 이해
- 플랫폼별 UI 최적화 및 사용자 경험 향상 전략 학습

## UI 개발 개요

- Flutter 위젯의 기본 요소를 활용한 사용자 인터페이스 구축
- 글꼴 커스터마이징을 통한 텍스트 인터페이스 개선
- 호스트 플랫폼별 최적화된 UI 구현 (Android, iOS, Web 등)
- 성능 최적화를 고려한 Flutter 기능 활용 (Lazy Loading, Caching 등)

## Google Fonts 구현

- pubspec.yaml에 google\_fonts 패키지 추가
- MaterialApp의 테마에서 글로벌 폰트 스타일 정의
- 개별 Text 위젯에 Google Fonts 스타일 적용
- 다양한 폰트 패밀리와 스타일 옵션 활용
- 폰트 파일 로딩 및 적용 시 성능 고려

# Google Fonts 예제 코드

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

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Google Fonts Demo',
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        primarySwatch: Colors.blue,
        textTheme: TextTheme(
          bodyText1:
            GoogleFonts.aBeeZee(fontSize: 30, color: Colors.deepOrange),
          bodyText2:
            GoogleFonts.aBeeZee(fontSize: 30, color: Colors.white60)),
      home: const MyHomePage(title: 'Flutter and Dart Cookbook'),
    );
  }
}
```

## RichText 구현 방법

- RichText 위젯을 사용한 다양한 텍스트 스타일 적용
- TextSpan을 활용한 텍스트 스타일 세그먼트 구성
- 폰트 크기, 색상, 굵기 등 세부 스타일링 제어
- 반응형 디자인을 위한 MediaQuery 활용

# RichText 예제 코드

```
class MyRichText extends StatelessWidget {  
  const MyRichText({Key? key}) : super(key: key);  
  
  @override  
  Widget build(BuildContext context) {  
    return RichText(  
      text: const TextSpan(  
        children: [  
          TextSpan(  
            text: 'Hello',  
            style: TextStyle(fontWeight: FontWeight.bold, fontSize: 24),  
          ),  
          TextSpan(  
            text: 'Luxembourg',  
            style: TextStyle(  
              fontWeight: FontWeight.bold, fontSize: 32, color: Colors.grey),  
            ),  
        ],  
      ),  
    );  
  }  
}
```

# 플랫폼 감지 구현

```
class PlatformCheck extends StatelessWidget {
  const PlatformCheck({Key? key}) : super(key: key);

  bool get isMobileDevice => !kIsWeb && (Platform.isIOS || Platform.isAndroid);
  bool get isDesktopDevice =>
    !kIsWeb && (Platform.isMacOS || Platform.isWindows || Platform.isLinux);
  bool get isMobileDeviceOrWeb => kIsWeb || isMobileDevice;
  bool get isDesktopDeviceOrWeb => kIsWeb || isDesktopDevice;

  @override
  Widget build(BuildContext context) {
    return Column(
      children: [
        Text('isMobileDevice: $isMobileDevice'),
        Text('isDesktopDevice: $isDesktopDevice'),
      ],
    );
  }
}
```



## 요약

- Google Fonts 패키지를 활용한 커스텀 폰트 적용 방법 습득
- RichText 위젯을 사용한 고급 텍스트 스타일링 구현
- 다양한 플랫폼 환경에서의 UI 개발 방법 이해
- 플랫폼별 UI 최적화 및 사용자 경험 향상 전략 학습

# END

---