

03

위젯 트리

UI Architecture

화면을 구성하는 위젯

- 위젯(widget)은 화면에 보일 뷰(view)를 설명하는 객체
- 화면과 관련된 모든 것이 위젯
- runApp() 함수는 전달받은 위젯을 위젯 트리(widget tree)의 루트로 만듭니다.

• runApp() 함수로 루트 위젯 등록하기

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

void main() {
  runApp(
    Center(
      child: Text(
        'Hello, world!',
      ),
    ),
  );
}
```

화면을 구성하는 위젯

• 화면 구성 예

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

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(
          title: Text('Test'),
        ),
        body: Center(child: GestureDetector(child: Text('HelloWorld'))),
      ),
    );
  }
}
```

▶ 실행 결과



화면을 구성하는 위젯

- •MaterialApp: 머티리얼 디자인 적용
 - •Scaffold: 화면 구조 설계
 - •AppBar: 화면 위쪽 앱바 구성
 - •Text: 앱바의 제목
 - •Center: 가운데 정렬
 - •GestureDetector: 사용자 이벤트 처리
 - •Text: 본문에 문자열 출력
-
- 플러터의 위젯은 모두 Widget의 자식 클래스
 - •Object → DiagnosticableTree → Widget
 - •Object → DiagnosticableTree → Widget → StatelessWidget → Text
 - •Object → DiagnosticableTree → Widget → RenderObjectWidget → SingleChildRenderObjectWidget → Align → Center



화면을 구성하는 위젯

- 선언형 프로그래밍으로 화면을 구성한다
 - 선언형은 화면 구성 정보만 작성
 - 프레임워크가 알아서 API를 이용해 화면을 출력
- 위젯은 불변이다
 - 객체를 생성한 후 상태를 바꿀 수 없습니다.
 - 화면을 새 데이터로 갱신하려면 새로운 위젯 객체를 만들어야 합니다.

위젯 트리 알아보기

위젯의 트리 구조

- 화면은 위젯을 여러 개 조합해서 구성

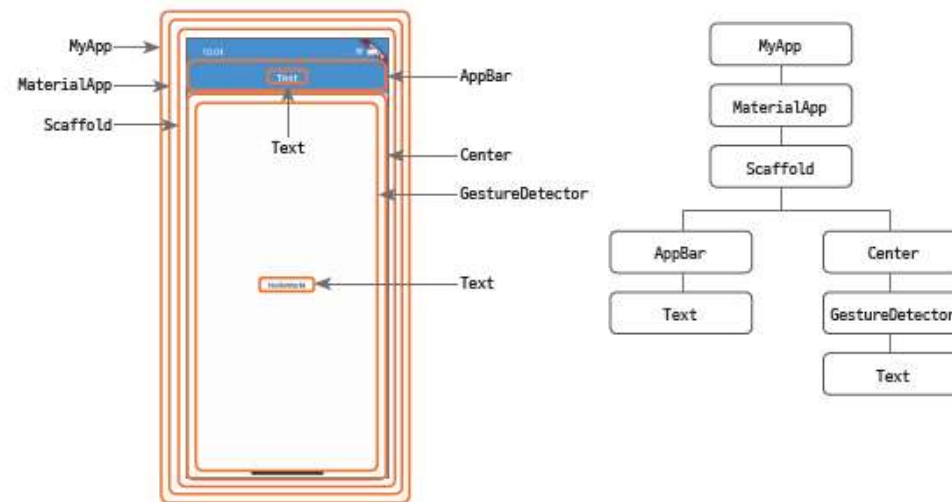


그림 8-1 위젯 구성과 트리 구조

위젯 트리 알아보기

화면을 구성하는 3개의 트리 구조

- 위젯의 트리 구조, 엘리먼트 트리(element tree)와 렌더 트리(render tree)
- 엘리먼트 트리는 ComponentElement와 RenderObjectElement 객체로 구성
- Component Element 객체는 트리 구조에서 다른 객체를 포함하는 역할
- 실제 화면에 출력할 정보는 RenderObjectElement

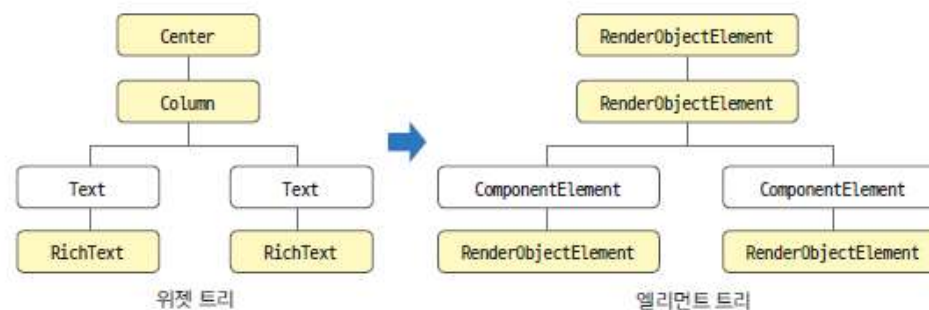
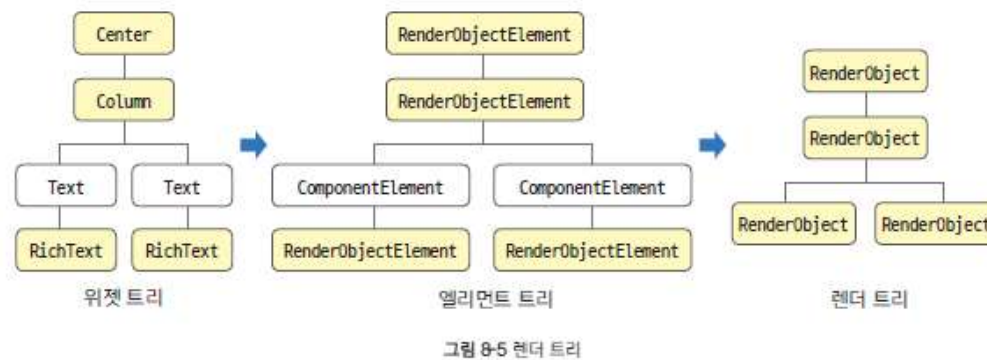


그림 8-4 엘리먼트 트리

위젯 트리 알아보기

화면을 구성하는 3개의 트리 구조

- 렌더 트리는 실제 화면에 출력할 정보를 가지는 RenderObjectElement에 해당하는 객체로만 구성





감사합니다

단단히 마음먹고 떠난 사람은
산꼭대기에 도착할 수 있다.
산은 올라가는 사람에게만 정복된다.



윌리엄 셰익스피어
William Shakespeare