

05

목록 구성

다양한 위젯

리스트 뷰

화면 스크롤 지원하기

• 스크롤이 필요한 상황

```
Column(  
  children: [  
    Container(height: 300,color: Colors.red,),  
    Container(height: 300,color: Colors.green,),  
    Container(height: 300,color: Colors.blue,),  
  ],  
)
```

▶ 실행 결과

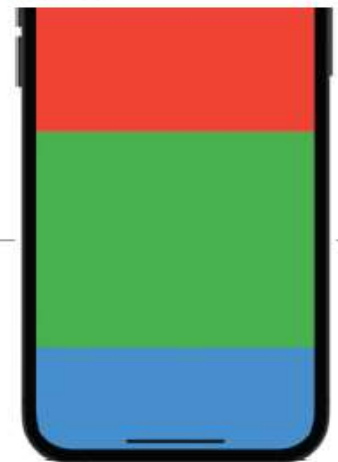


스크롤 경고

• 스크롤 지원 리스트 뷰

```
ListView(  
  children: [  
    Container(height: 300,color: Colors.red,),  
    Container(height: 300,color: Colors.green,),  
    Container(height: 300,color: Colors.blue,),  
  ],  
)
```

▶ 실행 결과



리스트 뷰

화면 스크롤 지원하기

- 리스트 뷰는 위젯을 세로로 나열할 때 주로 사용하지만 원한다면 가로로 나열할 수도 있습니다.



리스트 뷰

목록 구성하기

- 항목을 스크롤에 따라 불러오기 — `ListView.builder()`
 - `ListView.builder()` 생성자에는 `itemCount`와 `itemBuilder`라는 속성을 설정
 - `itemCount`는 리스트 뷰에 출력할 항목 수
 - `itemBuilder`는 항목을 구성하는 위젯

• 항목을 스크롤에 따라 불러오기

```
List<String> citys = [
  '서울시', '인천시', '부산시', '대구시', '대전시', '광주시', '울산시', '세종시'
];

@override
Widget build(BuildContext context) {
  return ListView.builder(
    itemCount: citys.length,
    itemBuilder: (context, index){
      return Container(
        padding: EdgeInsets.only(left: 10, top: 10),
        height: 100,
        child: Text(citys[index]),
      );
    },
  );
}
```

▶ 실행 결과



리스트 뷰

목록 구성하기

- 항목 구분자 설정하기 — `ListView.separated()`
 - `separatorBuilder` 속성에 지정하는 함수에서 구분자로 사용할 위젯을 준비해 반환하면 자동으로 반환한 위젯이 구분자로 출력

• 항목 구분자 설정하기

```
ListView.separated(  
  itemCount: citys.length,  
  itemBuilder: (context, index){  
    return Container(  
      padding: EdgeInsets.only(left: 10, top: 10),  
      height: 100,  
      child: Text(citys[index]),  
    );  
  },  
  separatorBuilder: (context, index){  
    return Divider(height: 2, color: Colors.black,);  
  },  
)
```

▶ 실행 결과



리스트 뷰

항목 구성하기 – ListTile

- ListTile은 항목을 구성하는 위젯
- ListTile 생성자에는 title, subtitle, leading, trailing 등의 속성



그림 12-1 ListTile로 항목 구성 예

• 항목 구성하기

```
ListView.separated(  
  itemCount: users.length,  
  itemBuilder: (context, index){  
    return ListTile(  
      leading: CircleAvatar(  
        radius: 25,  
        backgroundImage: AssetImage('images/big.jpeg'),  
      ),  
      title: Text(users[index].name),  
      subtitle: Text(users[index].phone),  
      trailing: Icon(Icons.more_vert),  
      onTap: (){  
        print(users[index].name);  
      },  
    );  
  },  
  separatorBuilder: (context, index){  
    return Divider(height: 2, color: Colors.black,);  
  },  
)
```

▶ 실행 결과



그리드 뷰

- 가로·세로 방향으로 나열하지 않고 한 줄에 여러 개를 함께 나열
- 그리드 뷰도 `GridView.builder()` 생성자를 제공하며 이 생성자의 `itemCount`, `itemBuilder` 속성을 이용해 항목의 개수와 위젯을 지정
- `GridView.builder()`에는 꼭 `gridDelegate` 속성을 설정
- `crossAxisCount`값이 한 줄에 함께 나와야 하는 항목의 개수

• 항목을 세로로 배치하기

```
GridView.builder(  
  itemCount: citys.length,  
  itemBuilder: (context, index){  
    return Card(  
      child: Text(citys[index]),  
    );  
  },  
  gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(crossAxisCount: 2),  
)
```

• 항목을 가로로 배치하기

```
GridView.builder(  
  ... (생략)...  
  scrollDirection: Axis.horizontal,  
  gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(crossAxisCount: 3),  
)
```

▶ 실행 결과



(기본값)



(scrollDirection: Axis.horizontal 지정)

페이지 뷰

- 페이지 뷰는 항목을 하나만 보여주며 사용자가 항목을 왼쪽이나 오른쪽으로 밀었을 때(스와이프) 차례로 나타납니다.
- PageController 객체에 맨 처음 보일 페이지를 설정
- 왼쪽 오른쪽에 있는 페이지를 현재 화면에서 어느 정도 보이게 할 것인지 설정

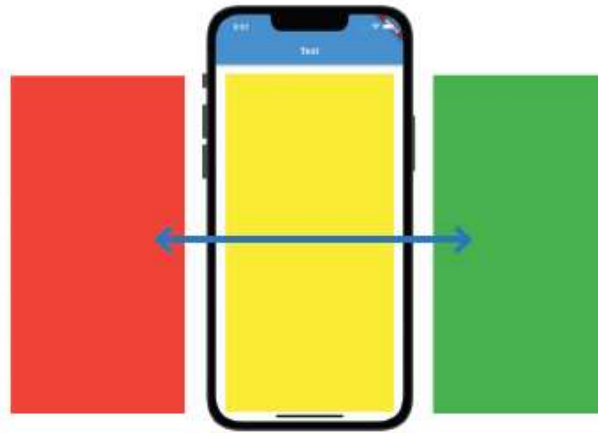


그림 12-3 페이지 뷰 동작

페이지 뷰

• 페이지 뷰 사용하기

```
PageController controller = PageController(  
  initialPage: 1,  
  viewportFraction: 0.8  
);  
@override  
Widget build(BuildContext context) {  
  return PageView(  
    controller: controller,  
    children: [  
      Container(margin: EdgeInsets.all(20), color: Colors.red,),  
      Container(margin: EdgeInsets.all(20), color: Colors.yellow,),  
      Container(margin: EdgeInsets.all(20), color: Colors.green,),  
    ],  
  );  
}
```

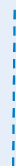
▶ 실행 결과





감사합니다

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



윌리엄 셰익스피어
William Shakespeare