# StatelessWidget

A widget that does not require mutable state.

# StatefulWidget

A widget that has mutable state.

State is information that (1) can be read synchronously when the widget is built and (2) might change during the lifetime of the widget.

# SingleChildScrollView

A box in which a single widget can be scrolled.

This widget is useful when you have a single box that will normally be entirely visible, for example a clock face in a time picker, but you need to make sure it can be scrolled if the container gets too small in one axis (the scroll direction).

# ListView

A scrollable list of widgets arranged linearly.

ListView is the most commonly used scrolling widget. It displays its children one after another in the scroll direction. In the cross axis, the children are required to fill the ListView.

By default, ListView will automatically pad the list's scrollable extremities to avoid partial obstructions indicated by MediaQuery's padding. To avoid this behavior, override with a zero padding property.

import 'package:flutter/material.dart';

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

/// This Widget is the main application widget.

class MyApp extends StatelessWidget {

static const String \_title = 'Flutter Code Sample';

@override

Widget build(BuildContext context) {

return MaterialApp(

title: \_title,

home: MyStatelessWidget(),

);

}

}

/// This is the stateless widget that the main application instantiates.

class MyStatelessWidget extends StatelessWidget {

MyStatelessWidget({Key key}) : super(key: key);

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Expanded Row Sample'),

),

body: Center(

child: ListView(

padding: const EdgeInsets.all(8),

children: <Widget>[

Container(

height: 50,

color: Colors.amber[600],

child: const Center(child: Text('Entry A')),

),

Container(

height: 50,

color: Colors.amber[500],

child: const Center(child: Text('Entry B')),

),

Container(

height: 50,

color: Colors.amber[100],

child: const Center(child: Text('Entry C')),

),

],

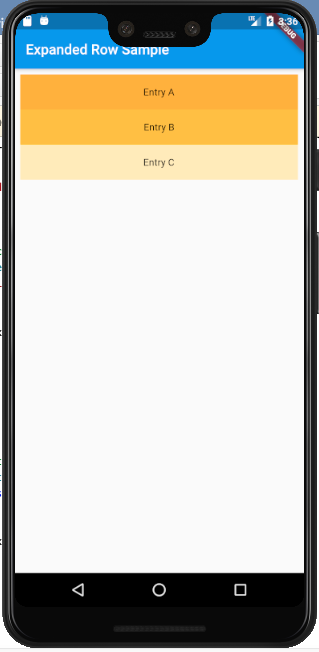
),

),

);

}

}



This example mirrors the previous one, creating the same list using the ListView.builder constructor. Using the IndexedWidgetBuilder, children are built lazily and can be infinite in number.

import 'package:flutter/material.dart';

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

/// This Widget is the main application widget.

class MyApp extends StatelessWidget {

static const String \_title = 'Flutter Code Sample';

@override

Widget build(BuildContext context) {

return MaterialApp(

title: \_title,

home: MyStatelessWidget(),

);

}

}

/// This is the stateless widget that the main application instantiates.

class MyStatelessWidget extends StatelessWidget {

MyStatelessWidget({Key key}) : super(key: key);

final List<String> entries = <String>['A', 'B', 'C'];

final List<int> colorCodes = <int>[600, 500, 100];

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Expanded Row Sample'),

),

body: Center(

child: ListView.builder(

padding: const EdgeInsets.all(8),

itemCount: entries.length,

itemBuilder: (BuildContext context, int index) {

return Container(

height: 50,

color: Colors.amber[colorCodes[index]],

child: Center(child: Text('Entry ${entries[index]}')),

);

}),

),

);

}

}

This example continues to build from our the previous ones, creating a similar list using ListView.separated. Here, a Divider is used as a separator.

import 'package:flutter/material.dart';

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

/// This Widget is the main application widget.

class MyApp extends StatelessWidget {

static const String \_title = 'Flutter Code Sample';

@override

Widget build(BuildContext context) {

return MaterialApp(

title: \_title,

home: MyStatelessWidget(),

);

}

}

/// This is the stateless widget that the main application instantiates.

class MyStatelessWidget extends StatelessWidget {

MyStatelessWidget({Key key}) : super(key: key);

final List<String> entries = <String>['A', 'B', 'C'];

final List<int> colorCodes = <int>[600, 500, 100];

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Expanded Row Sample'),

),

body: Center(

child: ListView.separated(

padding: const EdgeInsets.all(8),

itemCount: entries.length,

itemBuilder: (BuildContext context, int index) {

return Container(

height: 50,

color: Colors.amber[colorCodes[index]],

child: Center(child: Text('Entry ${entries[index]}')),

);

},

separatorBuilder: (BuildContext context, int index) =>

const Divider(),

),

),

);

}

}

A ListView of 3 amber colored containers with sample text and a Divider
between each of them.

# Complete ListView Example

import 'dart:convert';

import 'dart:typed\_data';

import 'package:flutter/material.dart';

import './AppConstants.dart';

import './TextViews.dart';

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

/// This Widget is the main application widget.

class MyApp extends StatelessWidget {

static const String \_title = 'Flutter Code Sample';

@override

Widget build(BuildContext context) {

return MaterialApp(

title: \_title,

home: MyStatelessWidget(),

);

}

}

/// This is the stateless widget that the main application instantiates.

class MyStatelessWidget extends StatelessWidget {

MyStatelessWidget({Key key}) : super(key: key);

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('ListView'),

),

body: Padding(

padding: const EdgeInsets.fromLTRB(

AppConstants.smallPadding,

AppConstants.smallPadding,

AppConstants.smallPadding,

0.0,

),

child: Column(

mainAxisAlignment: MainAxisAlignment.start,

children: <Widget>[

Padding(

padding: const EdgeInsets.only(bottom: AppConstants.smallPadding),

child: HeadingText(text: "Items"),

),

ListView.builder(

shrinkWrap: true,

itemCount: 2 + 1,

itemBuilder: (context, index) {

if (index == 2) {

return Padding(

padding: const EdgeInsets.only(bottom: AppConstants.smallPadding),

child: GestureDetector(

onTap: () {

},

child: Container(

child: CreateListingTile(),

decoration: BoxDecoration(

border: Border.all(

color: Colors.blueAccent,

width: 2.0

),

borderRadius: BorderRadius.circular(5),

),

),

),

);

} else {

return Padding(

padding: const EdgeInsets.only(bottom: AppConstants.smallPadding),

child: GestureDetector(

onTap: () {

},

child: Container(

child: ListingTile(),

decoration: BoxDecoration(

border: Border.all(

color: Colors.blue,

width: 1.0

),

borderRadius: BorderRadius.circular(5),

),

),

),

);

}

},

),

],

),

),

);

}

}

class ListingTile extends StatefulWidget {

ListingTile({Key key}) : super(key: key);

@override

ListingTileState createState() => ListingTileState();

}

class ListingTileState extends State<ListingTile> {

@override

void initState() {

\_image = MemoryImage(bytes);

super.initState();

}

@override

Widget build(BuildContext context) {

return ListTile(

leading: Padding(

padding: const EdgeInsets.only(left: AppConstants.tinyPadding),

child: RegularText(text: "Item"),

),

trailing: AspectRatio(

aspectRatio: 3 / 2,

child: (\_image == null)

? Container()

: Container(

decoration: BoxDecoration(

image: DecorationImage(

image: NetworkImage('https://flutter.github.io/assets-for-api-docs/assets/widgets/owl-2.jpg'),

fit: BoxFit.fitHeight,

),

),

),

),

contentPadding: EdgeInsets.all(AppConstants.tinyPadding),

);

}

}

class CreateListingTile extends StatelessWidget {

CreateListingTile({Key key}) : super(key: key);

@override

Widget build(BuildContext context) {

return Container(

height: MediaQuery.of(context).size.height / 12,

child: Row(

children: <Widget>[

Padding(

padding: const EdgeInsets.only(

left: AppConstants.tinyPadding,

right: AppConstants.tinyPadding),

child: Icon(Icons.add),

),

Expanded(child: RegularText(text: "Create another item"))

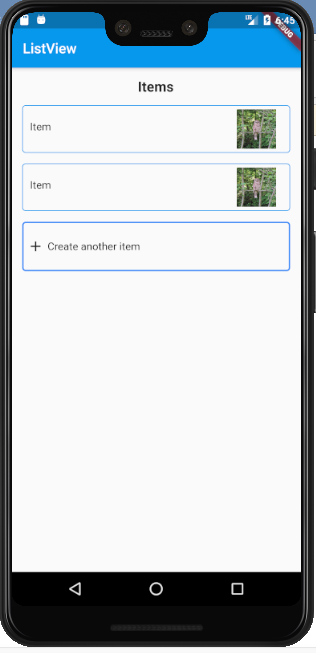
],

),

);

}

}



# GridView

A scrollable, 2D array of widgets.

The main axis direction of a grid is the direction in which it scrolls (the scrollDirection).

This example demonstrates how to create a GridView with two columns. The children are spaced apart using the crossAxisSpacing and mainAxisSpacing properties.

import 'package:flutter/material.dart';

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

/// This Widget is the main application widget.

class MyApp extends StatelessWidget {

static const String \_title = 'Flutter Code Sample';

@override

Widget build(BuildContext context) {

return MaterialApp(

title: \_title,

home: MyStatelessWidget(),

);

}

}

/// This is the stateless widget that the main application instantiates.

class MyStatelessWidget extends StatelessWidget {

MyStatelessWidget({Key key}) : super(key: key);

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('ListView'),

),

body: GridView.count(

primary: false,

padding: const EdgeInsets.all(20),

crossAxisSpacing: 10,

mainAxisSpacing: 10,

crossAxisCount: 2,

children: <Widget>[

Container(

padding: const EdgeInsets.all(8),

child: const Text("He'd have you all unravel at the"),

color: Colors.teal[100],

),

Container(

padding: const EdgeInsets.all(8),

child: const Text('Heed not the rabble'),

color: Colors.teal[200],

),

Container(

padding: const EdgeInsets.all(8),

child: const Text('Sound of screams but the'),

color: Colors.teal[300],

),

Container(

padding: const EdgeInsets.all(8),

child: const Text('Who scream'),

color: Colors.teal[400],

),

Container(

padding: const EdgeInsets.all(8),

child: const Text('Revolution is coming...'),

color: Colors.teal[500],

),

Container(

padding: const EdgeInsets.all(8),

child: const Text('Revolution, they...'),

color: Colors.teal[600],

),

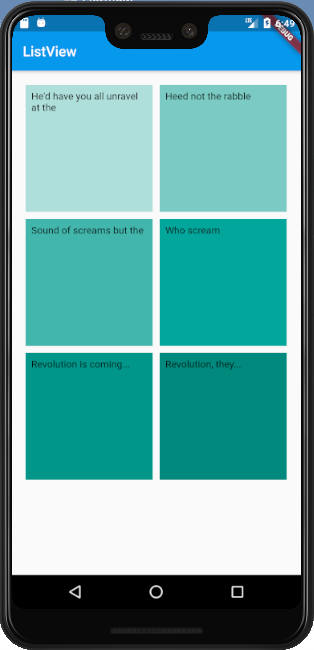
],

),

);

}

}



## GridView.Builder

import 'package:flutter/material.dart';

import './AppConstants.dart';

import './TextViews.dart';

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

/// This Widget is the main application widget.

class MyApp extends StatelessWidget {

static const String \_title = 'Flutter Code Sample';

@override

Widget build(BuildContext context) {

return MaterialApp(

title: \_title,

home: MyStatelessWidget(),

);

}

}

/// This is the stateless widget that the main application instantiates.

class MyStatelessWidget extends StatelessWidget {

MyStatelessWidget({Key key}) : super(key: key);

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('GridView'),

),

body: Padding(

padding: const EdgeInsets.fromLTRB(

AppConstants.smallPadding,

AppConstants.smallPadding,

AppConstants.smallPadding,

0.0,

),

child: Column(

mainAxisAlignment: MainAxisAlignment.start,

children: <Widget>[

Padding(

padding: const EdgeInsets.fromLTRB(

0.0, 0.0, 0.0, AppConstants.mediumPadding),

child: TextField(

decoration: InputDecoration(

hintText: 'Search',

contentPadding: EdgeInsets.all(5.0),

prefixIcon: Icon(Icons.search),

border: OutlineInputBorder(

borderSide: BorderSide(

color: Colors.grey,

width: 2.0,

),

),

),

style: TextStyle(

fontSize: 20.0,

color: Colors.black,

),

),

),

Expanded(

child: GridView.builder(

itemCount: 3,

shrinkWrap: false,

gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(

crossAxisCount: 2,

crossAxisSpacing: AppConstants.tinyPadding,

mainAxisSpacing: AppConstants.smallPadding,

childAspectRatio: 3 / 4,

),

itemBuilder: (context, index) {

return GestureDetector(

onTap: () {

/\*Navigator.pushNamed(

context,

ViewPostingPage.routeName,

arguments: posting,

);\*/

},

child: PostingGridTile(),

);

},

),

),

],

),

),

);

}

}

class PostingGridTile extends StatefulWidget {

PostingGridTile({Key key}) : super(key: key);

@override

State<PostingGridTile> createState() => \_PostingGridTileState();

}

class \_PostingGridTileState extends State<PostingGridTile> {

@override

Widget build(BuildContext context) {

return Column(

mainAxisAlignment: MainAxisAlignment.spaceAround,

crossAxisAlignment: CrossAxisAlignment.start,

children: <Widget>[

AspectRatio(

aspectRatio: 3 / 2,

child: Container(

decoration: BoxDecoration(

image: DecorationImage(

image: NetworkImage(

'https://flutter.github.io/assets-for-api-docs/assets/widgets/owl-2.jpg'),

fit: BoxFit.fill,

),

),

),

),

HeadingText(

text: 'type' + " - " + 'location',

fontSize: AppConstants.tinyFontSize,

),

HeadingText(

text: 'name',

fontSize: AppConstants.smallFontSize,

),

Text("\$100 / night"),

HeadingText(

text: "5/5 stars",

fontSize: AppConstants.tinyFontSize,

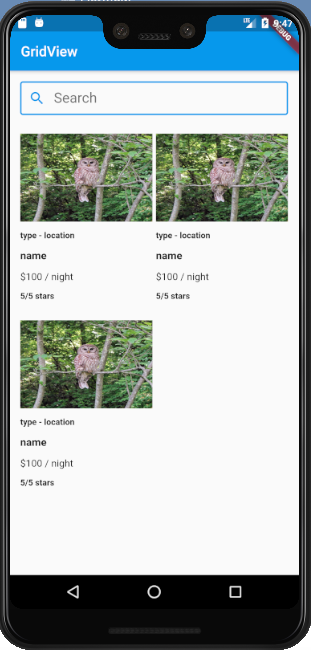
),

],

);

}

}



# PageView

A scrollable list that works page by page.

Each child of a page view is forced to be the same size as the viewport.

import 'package:flutter/material.dart';

import './AppConstants.dart';

import './TextViews.dart';

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

/// This Widget is the main application widget.

class MyApp extends StatelessWidget {

static const String \_title = 'Flutter Code Sample';

@override

Widget build(BuildContext context) {

return MaterialApp(

title: \_title,

home: MyStatelessWidget(),

);

}

}

/// This is the stateless widget that the main application instantiates.

class MyStatelessWidget extends StatelessWidget {

MyStatelessWidget({Key key}) : super(key: key);

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('PageView'),

),

body: Padding(

padding: const EdgeInsets.fromLTRB(

AppConstants.smallPadding,

AppConstants.smallPadding,

AppConstants.smallPadding,

0.0,

),

child: Column(

mainAxisAlignment: MainAxisAlignment.start,

children: <Widget>[

Container(

width: MediaQuery.of(context).size.width,

height: MediaQuery.of(context).size.height / 2.2,

child: PageView.builder(

itemCount: 3,

itemBuilder: (context, index) {

return PostingGridTile();

}),

),

Expanded(

child: Container(),

),

Padding(

padding: EdgeInsets.only(

top: AppConstants.mediumPadding,

bottom: AppConstants.mediumPadding),

child: MaterialButton(

onPressed: () {},

child: Text('Book Now!'),

minWidth: double.infinity,

height: MediaQuery.of(context).size.height / 15,

color: Colors.blue,

),

),

/\*Expanded(

child: GridView.builder(

itemCount: 3,

shrinkWrap: false,

gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(

crossAxisCount: 2,

crossAxisSpacing: AppConstants.tinyPadding,

mainAxisSpacing: AppConstants.smallPadding,

childAspectRatio: 3 / 4,

),

itemBuilder: (context, index) {

return GestureDetector(

onTap: () {

/\*Navigator.pushNamed(

context,

ViewPostingPage.routeName,

arguments: posting,

);\*/

},

child: PostingGridTile(),

);

},

),

),\*/

],

),

),

);

}

}

class PostingGridTile extends StatefulWidget {

PostingGridTile({Key key}) : super(key: key);

@override

State<PostingGridTile> createState() => \_PostingGridTileState();

}

class \_PostingGridTileState extends State<PostingGridTile> {

@override

Widget build(BuildContext context) {

return Column(

mainAxisAlignment: MainAxisAlignment.spaceAround,

crossAxisAlignment: CrossAxisAlignment.center,

children: <Widget>[

AspectRatio(

aspectRatio: 3 / 2,

child: Container(

decoration: BoxDecoration(

image: DecorationImage(

image: NetworkImage(

'https://flutter.github.io/assets-for-api-docs/assets/widgets/owl-2.jpg'),

fit: BoxFit.fill,

),

),

),

),

HeadingText(

text: 'type' + " - " + 'location',

fontSize: AppConstants.tinyFontSize,

),

HeadingText(

text: 'name',

fontSize: AppConstants.smallFontSize,

),

Text("\$100 / night"),

HeadingText(

text: "5/5 stars",

fontSize: AppConstants.tinyFontSize,

),

],

);

}

}

