# Container class

A convenience widget that combines common painting, positioning, and sizing widgets.

A container first surrounds the child with [padding](https://docs.flutter.io/flutter/widgets/Container/padding.html) (inflated by any borders present in the [decoration](https://docs.flutter.io/flutter/widgets/Container/decoration.html)) and then applies additional [constraints](https://docs.flutter.io/flutter/widgets/Container/constraints.html) to the padded extent (incorporating the width and height as constraints, if either is non-null). The container is then surrounded by additional empty space described from the [margin](https://docs.flutter.io/flutter/widgets/Container/margin.html).

During painting, the container first applies the given [transform](https://docs.flutter.io/flutter/widgets/Container/transform.html), then paints the [decoration](https://docs.flutter.io/flutter/widgets/Container/decoration.html) to fill the padded extent, then it paints the child, and finally paints the [foregroundDecoration](https://docs.flutter.io/flutter/widgets/Container/foregroundDecoration.html), also filling the padded extent.

Containers with no children try to be as big as possible unless the incoming constraints are unbounded, in which case they try to be as small as possible. Containers with children size themselves to their children. The width, height, and [constraints](https://docs.flutter.io/flutter/widgets/Container/constraints.html) arguments to the constructor override this.

## Layout behavior

See [BoxConstraints](https://docs.flutter.io/flutter/rendering/BoxConstraints-class.html) for an introduction to box layout models.

Since [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) combines a number of other widgets each with their own layout behavior, [Container](https://docs.flutter.io/flutter/widgets/Container-class.html)'s layout behavior is somewhat complicated.

tl;dr: [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) tries, in order: to honor [alignment](https://docs.flutter.io/flutter/widgets/Container/alignment.html), to size itself to the [child](https://docs.flutter.io/flutter/widgets/Container/child.html), to honor the width, height, and [constraints](https://docs.flutter.io/flutter/widgets/Container/constraints.html), to expand to fit the parent, to be as small as possible.

More specifically:

If the widget has no child, no height, no width, no [constraints](https://docs.flutter.io/flutter/widgets/Container/constraints.html), and the parent provides unbounded constraints, then [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) tries to size as small as possible.

If the widget has no child and no [alignment](https://docs.flutter.io/flutter/widgets/Container/alignment.html), but a height, width, or [constraints](https://docs.flutter.io/flutter/widgets/Container/constraints.html) are provided, then the [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) tries to be as small as possible given the combination of those constraints and the parent's constraints.

If the widget has no child, no height, no width, no [constraints](https://docs.flutter.io/flutter/widgets/Container/constraints.html), and no [alignment](https://docs.flutter.io/flutter/widgets/Container/alignment.html), but the parent provides bounded constraints, then [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) expands to fit the constraints provided by the parent.

If the widget has an [alignment](https://docs.flutter.io/flutter/widgets/Container/alignment.html), and the parent provides unbounded constraints, then the [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) tries to size itself around the child.

If the widget has an [alignment](https://docs.flutter.io/flutter/widgets/Container/alignment.html), and the parent provides bounded constraints, then the [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) tries to expand to fit the parent, and then positions the child within itself as per the [alignment](https://docs.flutter.io/flutter/widgets/Container/alignment.html).

Otherwise, the widget has a [child](https://docs.flutter.io/flutter/widgets/Container/child.html) but no height, no width, no [constraints](https://docs.flutter.io/flutter/widgets/Container/constraints.html), and no [alignment](https://docs.flutter.io/flutter/widgets/Container/alignment.html), and the [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) passes the constraints from the parent to the child and sizes itself to match the child.

The [margin](https://docs.flutter.io/flutter/widgets/Container/margin.html) and [padding](https://docs.flutter.io/flutter/widgets/Container/padding.html) properties also affect the layout, as described in the documentation for those properties. (Their effects merely augment the rules described above.) The [decoration](https://docs.flutter.io/flutter/widgets/Container/decoration.html) can implicitly increase the [padding](https://docs.flutter.io/flutter/widgets/Container/padding.html) (e.g. borders in a [BoxDecoration](https://docs.flutter.io/flutter/painting/BoxDecoration-class.html) contribute to the [padding](https://docs.flutter.io/flutter/widgets/Container/padding.html)); see [Decoration.padding](https://docs.flutter.io/flutter/painting/Decoration/padding.html).

## Sample code

This example shows a 48x48 green square (placed inside a [Center](https://docs.flutter.io/flutter/widgets/Center-class.html) widget in case the parent widget has its own opinions regarding the size that the [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) should take), with a margin so that it stays away from neighboring widgets:

**new** Center(

child: **new** Container(

margin: **const** EdgeInsets.all(10.0),

color: **const** Color(0xFF00FF00),

width: 48.0,

height: 48.0,

),

)

This example shows how to use many of the features of [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) at once. The [constraints](https://docs.flutter.io/flutter/widgets/Container/constraints.html) are set to fit the font size plus ample headroom vertically, while expanding horizontally to fit the parent. The [padding](https://docs.flutter.io/flutter/widgets/Container/padding.html) is used to make sure there is space between the contents and the text. The color makes the box teal. The [alignment](https://docs.flutter.io/flutter/widgets/Container/alignment.html) causes the [child](https://docs.flutter.io/flutter/widgets/Container/child.html) to be centered in the box. The [foregroundDecoration](https://docs.flutter.io/flutter/widgets/Container/foregroundDecoration.html) overlays a nine-patch image onto the text. Finally, the [transform](https://docs.flutter.io/flutter/widgets/Container/transform.html) applies a slight rotation to the entire contraption to complete the effect.

**new** Container(

constraints: **new** BoxConstraints.expand(

height: Theme.of(context).textTheme.display1.fontSize \* 1.1 + 200.0,

),

padding: **const** EdgeInsets.all(8.0),

color: Colors.teal.shade700,

alignment: Alignment.center,

child: **new** Text('Hello World', style: Theme.of(context).textTheme.display1.copyWith(color: Colors.white)),

foregroundDecoration: **new** BoxDecoration(

image: **new** DecorationImage(

image: **new** NetworkImage('https://www.example.com/images/frame.png'),

centerSlice: **new** Rect.fromLTRB(270.0, 180.0, 1360.0, 730.0),

),

),

transform: **new** Matrix4.rotationZ(0.1),

)

See also:

* [AnimatedContainer](https://docs.flutter.io/flutter/widgets/AnimatedContainer-class.html), a variant that smoothly animates the properties when they change.
* [Border](https://docs.flutter.io/flutter/painting/Border-class.html), which has a sample which uses [Container](https://docs.flutter.io/flutter/widgets/Container-class.html) heavily.
* [Ink](https://docs.flutter.io/flutter/material/Ink-class.html), which paints a [Decoration](https://docs.flutter.io/flutter/painting/Decoration-class.html) on a [Material](https://docs.flutter.io/flutter/material/Material-class.html), allowing [InkResponse](https://docs.flutter.io/flutter/material/InkResponse-class.html) and [InkWell](https://docs.flutter.io/flutter/material/InkWell-class.html) splashes to paint over them.
* The [catalog of layout widgets](https://flutter.io/widgets/layout/).

Inheritance

* [Object](https://docs.flutter.io/flutter/dart-core/Object-class.html)
* [Diagnosticable](https://docs.flutter.io/flutter/foundation/Diagnosticable-class.html)
* [DiagnosticableTree](https://docs.flutter.io/flutter/foundation/DiagnosticableTree-class.html)
* [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html)
* [StatelessWidget](https://docs.flutter.io/flutter/widgets/StatelessWidget-class.html)
* Container

## Constructors

[**Container**](https://docs.flutter.io/flutter/widgets/Container/Container.html)({[Key](https://docs.flutter.io/flutter/foundation/Key-class.html) key, [AlignmentGeometry](https://docs.flutter.io/flutter/painting/AlignmentGeometry-class.html) alignment, [EdgeInsetsGeometry](https://docs.flutter.io/flutter/painting/EdgeInsetsGeometry-class.html) padding, [Color](https://docs.flutter.io/flutter/dart-ui/Color-class.html) color, [Decoration](https://docs.flutter.io/flutter/painting/Decoration-class.html) decoration, [Decoration](https://docs.flutter.io/flutter/painting/Decoration-class.html) foregroundDecoration, [double](https://docs.flutter.io/flutter/dart-core/double-class.html) width, [double](https://docs.flutter.io/flutter/dart-core/double-class.html) height, [BoxConstraints](https://docs.flutter.io/flutter/rendering/BoxConstraints-class.html) constraints, [EdgeInsetsGeometry](https://docs.flutter.io/flutter/painting/EdgeInsetsGeometry-class.html) margin, [Matrix4](https://docs.flutter.io/flutter/vector_math_64/Matrix4-class.html) transform, [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html) child })

Creates a widget that combines common painting, positioning, and sizing widgets. [[...]](https://docs.flutter.io/flutter/widgets/Container/Container.html)

## Properties

[**alignment**](https://docs.flutter.io/flutter/widgets/Container/alignment.html) → [AlignmentGeometry](https://docs.flutter.io/flutter/painting/AlignmentGeometry-class.html)

Align the child within the container. [[...]](https://docs.flutter.io/flutter/widgets/Container/alignment.html)

*final*

[**child**](https://docs.flutter.io/flutter/widgets/Container/child.html) → [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html)

The child contained by the container. [[...]](https://docs.flutter.io/flutter/widgets/Container/child.html)

*final*

[**constraints**](https://docs.flutter.io/flutter/widgets/Container/constraints.html) → [BoxConstraints](https://docs.flutter.io/flutter/rendering/BoxConstraints-class.html)

Additional constraints to apply to the child. [[...]](https://docs.flutter.io/flutter/widgets/Container/constraints.html)

*final*

[**decoration**](https://docs.flutter.io/flutter/widgets/Container/decoration.html) → [Decoration](https://docs.flutter.io/flutter/painting/Decoration-class.html)

The decoration to paint behind the child. [[...]](https://docs.flutter.io/flutter/widgets/Container/decoration.html)

*final*

[**foregroundDecoration**](https://docs.flutter.io/flutter/widgets/Container/foregroundDecoration.html) → [Decoration](https://docs.flutter.io/flutter/painting/Decoration-class.html)

The decoration to paint in front of the child.

*final*

[**margin**](https://docs.flutter.io/flutter/widgets/Container/margin.html) → [EdgeInsetsGeometry](https://docs.flutter.io/flutter/painting/EdgeInsetsGeometry-class.html)

Empty space to surround the decoration and child.

*final*

[**padding**](https://docs.flutter.io/flutter/widgets/Container/padding.html) → [EdgeInsetsGeometry](https://docs.flutter.io/flutter/painting/EdgeInsetsGeometry-class.html)

Empty space to inscribe inside the decoration. The child, if any, is placed inside this padding. [[...]](https://docs.flutter.io/flutter/widgets/Container/padding.html)

*final*

[**transform**](https://docs.flutter.io/flutter/widgets/Container/transform.html) → [Matrix4](https://docs.flutter.io/flutter/vector_math_64/Matrix4-class.html)

The transformation matrix to apply before painting the container.

*final*

[*hashCode*](https://docs.flutter.io/flutter/dart-core/Object/hashCode.html) → [int](https://docs.flutter.io/flutter/dart-core/int-class.html)

The hash code for this object. [[...]](https://docs.flutter.io/flutter/dart-core/Object/hashCode.html)

*read-only, inherited*

[*key*](https://docs.flutter.io/flutter/widgets/Widget/key.html) → [Key](https://docs.flutter.io/flutter/foundation/Key-class.html)

Controls how one widget replaces another widget in the tree. [[...]](https://docs.flutter.io/flutter/widgets/Widget/key.html)

*final, inherited*

[*runtimeType*](https://docs.flutter.io/flutter/dart-core/Object/runtimeType.html) → [Type](https://docs.flutter.io/flutter/dart-core/Type-class.html)

A representation of the runtime type of the object.

*read-only, inherited*

## Methods

[**build**](https://docs.flutter.io/flutter/widgets/Container/build.html)([BuildContext](https://docs.flutter.io/flutter/widgets/BuildContext-class.html) context) → [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html)

Describes the part of the user interface represented by this widget. [[...]](https://docs.flutter.io/flutter/widgets/Container/build.html)

[**debugFillProperties**](https://docs.flutter.io/flutter/widgets/Container/debugFillProperties.html)([DiagnosticPropertiesBuilder](https://docs.flutter.io/flutter/foundation/DiagnosticPropertiesBuilder-class.html) properties) → void

[*createElement*](https://docs.flutter.io/flutter/widgets/StatelessWidget/createElement.html)( → [StatelessElement](https://docs.flutter.io/flutter/widgets/StatelessElement-class.html)

Creates a [StatelessElement](https://docs.flutter.io/flutter/widgets/StatelessElement-class.html) to manage this widget's location in the tree. [[...]](https://docs.flutter.io/flutter/widgets/StatelessWidget/createElement.html)

*inherited*

[*debugDescribeChildren*](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/debugDescribeChildren.html)( → [List](https://docs.flutter.io/flutter/dart-core/List-class.html)<[DiagnosticsNode](https://docs.flutter.io/flutter/foundation/DiagnosticsNode-class.html)>

Returns a list of [DiagnosticsNode](https://docs.flutter.io/flutter/foundation/DiagnosticsNode-class.html) objects describing this node's children. [[...]](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/debugDescribeChildren.html)

*@protected, inherited*

[*noSuchMethod*](https://docs.flutter.io/flutter/dart-core/Object/noSuchMethod.html)([Invocation](https://docs.flutter.io/flutter/dart-core/Invocation-class.html) invocation) → dynamic

Invoked when a non-existent method or property is accessed. [[...]](https://docs.flutter.io/flutter/dart-core/Object/noSuchMethod.html)

*inherited*

[*toDiagnosticsNode*](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/toDiagnosticsNode.html)([String](https://docs.flutter.io/flutter/dart-core/String-class.html) name, [DiagnosticsTreeStyle](https://docs.flutter.io/flutter/foundation/DiagnosticsTreeStyle-class.html) style }) → [DiagnosticsNode](https://docs.flutter.io/flutter/foundation/DiagnosticsNode-class.html)

Returns a debug representation of the object that is used by debugging tools and by [toStringDeep](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/toStringDeep.html). [[...]](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/toDiagnosticsNode.html)

*inherited*

[*toString*](https://docs.flutter.io/flutter/foundation/Diagnosticable/toString.html)([DiagnosticLevel](https://docs.flutter.io/flutter/foundation/DiagnosticLevel-class.html) minLevel: DiagnosticLevel.debug }) → [String](https://docs.flutter.io/flutter/dart-core/String-class.html)

Returns a string representation of this object.

*inherited*

[*toStringDeep*](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/toStringDeep.html)([String](https://docs.flutter.io/flutter/dart-core/String-class.html) prefixLineOne: '', [String](https://docs.flutter.io/flutter/dart-core/String-class.html) prefixOtherLines, [DiagnosticLevel](https://docs.flutter.io/flutter/foundation/DiagnosticLevel-class.html) minLevel: DiagnosticLevel.debug }) → [String](https://docs.flutter.io/flutter/dart-core/String-class.html)

Returns a string representation of this node and its descendants. [[...]](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/toStringDeep.html)

*inherited*

[*toStringShallow*](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/toStringShallow.html)([String](https://docs.flutter.io/flutter/dart-core/String-class.html) joiner: ', ', [DiagnosticLevel](https://docs.flutter.io/flutter/foundation/DiagnosticLevel-class.html) minLevel: DiagnosticLevel.debug }) → [String](https://docs.flutter.io/flutter/dart-core/String-class.html)

Returns a one-line detailed description of the object. [[...]](https://docs.flutter.io/flutter/foundation/DiagnosticableTree/toStringShallow.html)

*inherited*

[*toStringShort*](https://docs.flutter.io/flutter/widgets/Widget/toStringShort.html)( → [String](https://docs.flutter.io/flutter/dart-core/String-class.html)

A short, textual description of this widget.

*inherited*

## Operators

[*operator ==*](https://docs.flutter.io/flutter/dart-core/Object/operator_equals.html)(dynamic other) → [bool](https://docs.flutter.io/flutter/dart-core/bool-class.html)

The equality operator. [[...]](https://docs.flutter.io/flutter/dart-core/Object/operator_equals.html)

*inherited*