Flutter resumen documentación

# Styling widgets

## MediaQuery

Establishes a subtree in which media queries resolve to the given data.

For example, to learn the size of the current media (e.g., the window containing your app), you can use [MediaQuery.sizeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/sizeOf.html): MediaQuery.sizeOf(context).

Querying the current media using specific methods (for example, [MediaQuery.sizeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/sizeOf.html) and [MediaQuery.paddingOf](https://api.flutter.dev/flutter/widgets/MediaQuery/paddingOf.html)) will cause your widget to rebuild automatically whenever the property you query changes.

On the other hand, querying using [MediaQuery.of](https://api.flutter.dev/flutter/widgets/MediaQuery/of.html) will cause your widget to rebuild automatically whenever any field of the [MediaQueryData](https://api.flutter.dev/flutter/widgets/MediaQueryData-class.html) changes (e.g., if the user rotates their device). Therefore, if you are only concerned with one or a few fields of [MediaQueryData](https://api.flutter.dev/flutter/widgets/MediaQueryData-class.html), prefer using the specific methods (for example: [MediaQuery.sizeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/sizeOf.html) and [MediaQuery.paddingOf](https://api.flutter.dev/flutter/widgets/MediaQuery/paddingOf.html)).

If no [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) is in scope then the series of methods like [MediaQuery.of](https://api.flutter.dev/flutter/widgets/MediaQuery/of.html) and [MediaQuery.sizeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/sizeOf.html) will throw an exception. Alternatively, the "maybe-" variant methods (such as [MediaQuery.maybeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeOf.html) and [MediaQuery.maybeSizeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeSizeOf.html)) can be used, which returns null instead of throwing if no [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) is in scope.

### Constructors

[MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery/MediaQuery.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [MediaQueryData](https://api.flutter.dev/flutter/widgets/MediaQueryData-class.html) data, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child})

Creates a widget that provides [MediaQueryData](https://api.flutter.dev/flutter/widgets/MediaQueryData-class.html) to its descendants.

const

[MediaQuery.removePadding](https://api.flutter.dev/flutter/widgets/MediaQuery/MediaQuery.removePadding.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeLeft = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeTop = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeRight = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeBottom = false, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child})

Creates a new [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) that inherits from the ambient [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) from the given context, but removes the specified padding.

factory

[MediaQuery.removeViewInsets](https://api.flutter.dev/flutter/widgets/MediaQuery/MediaQuery.removeViewInsets.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeLeft = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeTop = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeRight = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeBottom = false, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child})

Creates a new [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) that inherits from the ambient [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) from the given context, but removes the specified view insets.

factory

[MediaQuery.removeViewPadding](https://api.flutter.dev/flutter/widgets/MediaQuery/MediaQuery.removeViewPadding.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeLeft = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeTop = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeRight = false, [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html) removeBottom = false, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child})

Creates a new [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) that inherits from the ambient [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) from the given context, but removes the specified view padding.

factory

### Properties

[child](https://api.flutter.dev/flutter/widgets/ProxyWidget/child.html) → [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)

The widget below this widget in the tree.

finalinherited

[data](https://api.flutter.dev/flutter/widgets/MediaQuery/data.html) → [MediaQueryData](https://api.flutter.dev/flutter/widgets/MediaQueryData-class.html)

Contains information about the current media.

final

[hashCode](https://api.flutter.dev/flutter/widgets/Widget/hashCode.html) → [int](https://api.flutter.dev/flutter/dart-core/int-class.html)

The hash code for this object.

read-onlyinherited

[key](https://api.flutter.dev/flutter/widgets/Widget/key.html) → [Key](https://api.flutter.dev/flutter/foundation/Key-class.html)?

Controls how one widget replaces another widget in the tree.

finalinherited

[runtimeType](https://api.flutter.dev/flutter/dart-core/Object/runtimeType.html) → [Type](https://api.flutter.dev/flutter/dart-core/Type-class.html)

A representation of the runtime type of the object.

read-onlyinherited

### Methods

[createElement](https://api.flutter.dev/flutter/widgets/InheritedModel/createElement.html)() → [InheritedModelElement](https://api.flutter.dev/flutter/widgets/InheritedModelElement-class.html)<\_MediaQueryAspect>

Inflates this configuration to a concrete instance.

inherited

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

Returns a list of DiagnosticsNode objects describing this node's children.

inherited

[debugFillProperties](https://api.flutter.dev/flutter/widgets/MediaQuery/debugFillProperties.html)([DiagnosticPropertiesBuilder](https://api.flutter.dev/flutter/foundation/DiagnosticPropertiesBuilder-class.html) properties) → void

Add additional properties associated with the node.

override

[isSupportedAspect](https://api.flutter.dev/flutter/widgets/InheritedModel/isSupportedAspect.html)([Object](https://api.flutter.dev/flutter/dart-core/Object-class.html) aspect) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns true if this model supports the given aspect.

inherited

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

Invoked when a nonexistent method or property is accessed.

inherited

[toDiagnosticsNode](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toDiagnosticsNode.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html)? name, [DiagnosticsTreeStyle](https://api.flutter.dev/flutter/foundation/DiagnosticsTreeStyle.html)? style}) → [DiagnosticsNode](https://api.flutter.dev/flutter/foundation/DiagnosticsNode-class.html)

Returns a debug representation of the object that is used by debugging tools and by [DiagnosticsNode.toStringDeep](https://api.flutter.dev/flutter/foundation/DiagnosticsNode/toStringDeep.html).

inherited

[toString](https://api.flutter.dev/flutter/foundation/Diagnosticable/toString.html)({[DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.info}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

A string representation of this object.

inherited

[toStringDeep](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toStringDeep.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html) prefixLineOne = '', [String](https://api.flutter.dev/flutter/dart-core/String-class.html)? prefixOtherLines, [DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.debug}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

Returns a string representation of this node and its descendants.

inherited

[toStringShallow](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toStringShallow.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html) joiner = ', ', [DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.debug}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

Returns a one-line detailed description of the object.

inherited

[toStringShort](https://api.flutter.dev/flutter/widgets/Widget/toStringShort.html)() → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

A short, textual description of this widget.

inherited

[updateShouldNotify](https://api.flutter.dev/flutter/widgets/MediaQuery/updateShouldNotify.html)(covariant [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) oldWidget) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Whether the framework should notify widgets that inherit from this widget.

override

[updateShouldNotifyDependent](https://api.flutter.dev/flutter/widgets/MediaQuery/updateShouldNotifyDependent.html)(covariant [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) oldWidget, [Set](https://api.flutter.dev/flutter/dart-core/Set-class.html)<[Object](https://api.flutter.dev/flutter/dart-core/Object-class.html)> dependencies) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Return true if the changes between this model and oldWidget match any of the dependencies.

override

### Operators

[operator ==](https://api.flutter.dev/flutter/widgets/Widget/operator_equals.html)([Object](https://api.flutter.dev/flutter/dart-core/Object-class.html) other) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

The equality operator.

inherited

### Static Methods

[accessibleNavigationOf](https://api.flutter.dev/flutter/widgets/MediaQuery/accessibleNavigationOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns accessibleNavigationOf for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[alwaysUse24HourFormatOf](https://api.flutter.dev/flutter/widgets/MediaQuery/alwaysUse24HourFormatOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns alwaysUse for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[boldTextOf](https://api.flutter.dev/flutter/widgets/MediaQuery/boldTextOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns the boldText accessibility setting for the nearest MediaQuery ancestor or false, if no such ancestor exists.

[boldTextOverride](https://api.flutter.dev/flutter/widgets/MediaQuery/boldTextOverride.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns the boldText accessibility setting for the nearest MediaQuery ancestor or false, if no such ancestor exists.

[devicePixelRatioOf](https://api.flutter.dev/flutter/widgets/MediaQuery/devicePixelRatioOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [double](https://api.flutter.dev/flutter/dart-core/double-class.html)

Returns devicePixelRatio for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[disableAnimationsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/disableAnimationsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns disableAnimations for the nearest MediaQuery ancestor or [Brightness.light](https://api.flutter.dev/flutter/dart-ui/Brightness.html), if no such ancestor exists.

[displayFeaturesOf](https://api.flutter.dev/flutter/widgets/MediaQuery/displayFeaturesOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [List](https://api.flutter.dev/flutter/dart-core/List-class.html)<[DisplayFeature](https://api.flutter.dev/flutter/dart-ui/DisplayFeature-class.html)>

Returns displayFeatures for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[fromView](https://api.flutter.dev/flutter/widgets/MediaQuery/fromView.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [FlutterView](https://api.flutter.dev/flutter/dart-ui/FlutterView-class.html) view, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child}) → [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)

Wraps the child in a [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) which is built using data from the provided view.

[fromWindow](https://api.flutter.dev/flutter/widgets/MediaQuery/fromWindow.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child}) → [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)

Deprecated. Use [MediaQuery.fromView](https://api.flutter.dev/flutter/widgets/MediaQuery/fromView.html) instead.

[gestureSettingsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/gestureSettingsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [DeviceGestureSettings](https://api.flutter.dev/flutter/gestures/DeviceGestureSettings-class.html)

Returns gestureSettings for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[highContrastOf](https://api.flutter.dev/flutter/widgets/MediaQuery/highContrastOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns highContrast for the nearest MediaQuery ancestor or false, if no such ancestor exists.

[invertColorsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/invertColorsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns invertColorsOf for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[maybeAccessibleNavigationOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeAccessibleNavigationOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)?

Returns accessibleNavigation for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeAlwaysUse24HourFormatOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeAlwaysUse24HourFormatOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)?

Returns alwaysUse24HourFormat for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeBoldTextOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeBoldTextOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)?

Returns the boldText accessibility setting for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeDevicePixelRatioOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeDevicePixelRatioOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [double](https://api.flutter.dev/flutter/dart-core/double-class.html)?

Returns devicePixelRatio for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeDisableAnimationsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeDisableAnimationsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)?

Returns disableAnimations for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeDisplayFeaturesOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeDisplayFeaturesOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [List](https://api.flutter.dev/flutter/dart-core/List-class.html)<[DisplayFeature](https://api.flutter.dev/flutter/dart-ui/DisplayFeature-class.html)>?

Returns displayFeatures for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeGestureSettingsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeGestureSettingsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [DeviceGestureSettings](https://api.flutter.dev/flutter/gestures/DeviceGestureSettings-class.html)?

Returns gestureSettings for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeHighContrastOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeHighContrastOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)?

Returns highContrast for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeInvertColorsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeInvertColorsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)?

Returns invertColors for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeNavigationModeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeNavigationModeOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [NavigationMode](https://api.flutter.dev/flutter/widgets/NavigationMode.html)?

Returns navigationMode for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [MediaQueryData](https://api.flutter.dev/flutter/widgets/MediaQueryData-class.html)?

The data from the closest instance of this class that encloses the given context, if any.

[maybeOnOffSwitchLabelsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeOnOffSwitchLabelsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)?

Returns onOffSwitchLabels for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeOrientationOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeOrientationOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [Orientation](https://api.flutter.dev/flutter/widgets/Orientation.html)?

Returns orientation for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybePaddingOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybePaddingOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html)?

Returns viewInsets for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybePlatformBrightnessOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybePlatformBrightnessOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [Brightness](https://api.flutter.dev/flutter/dart-ui/Brightness.html)?

Returns platformBrightness for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeSizeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeSizeOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [Size](https://api.flutter.dev/flutter/dart-ui/Size-class.html)?

Returns size for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeSystemGestureInsetsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeSystemGestureInsetsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html)?

Returns systemGestureInsets for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeTextScaleFactorOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeTextScaleFactorOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [double](https://api.flutter.dev/flutter/dart-core/double-class.html)?

Deprecated. Will be removed in a future version of Flutter. Use [maybeTextScalerOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeTextScalerOf.html) instead.

[maybeTextScalerOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeTextScalerOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [TextScaler](https://api.flutter.dev/flutter/painting/TextScaler-class.html)?

Returns the [TextScaler](https://api.flutter.dev/flutter/painting/TextScaler-class.html) for the nearest [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) ancestor or [TextScaler.noScaling](https://api.flutter.dev/flutter/painting/TextScaler/noScaling-constant.html) if no such ancestor exists.

[maybeViewInsetsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeViewInsetsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html)?

Returns viewInsets for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[maybeViewPaddingOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeViewPaddingOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html)?

Returns viewPadding for the nearest MediaQuery ancestor or null, if no such ancestor exists.

[navigationModeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/navigationModeOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [NavigationMode](https://api.flutter.dev/flutter/widgets/NavigationMode.html)

Returns navigationMode for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[of](https://api.flutter.dev/flutter/widgets/MediaQuery/of.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [MediaQueryData](https://api.flutter.dev/flutter/widgets/MediaQueryData-class.html)

The data from the closest instance of this class that encloses the given context.

[onOffSwitchLabelsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/onOffSwitchLabelsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

Returns onOffSwitchLabels for the nearest MediaQuery ancestor or false, if no such ancestor exists.

[orientationOf](https://api.flutter.dev/flutter/widgets/MediaQuery/orientationOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [Orientation](https://api.flutter.dev/flutter/widgets/Orientation.html)

Returns orientation for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[paddingOf](https://api.flutter.dev/flutter/widgets/MediaQuery/paddingOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html)

Returns padding for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[platformBrightnessOf](https://api.flutter.dev/flutter/widgets/MediaQuery/platformBrightnessOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [Brightness](https://api.flutter.dev/flutter/dart-ui/Brightness.html)

Returns platformBrightness for the nearest MediaQuery ancestor or [Brightness.light](https://api.flutter.dev/flutter/dart-ui/Brightness.html), if no such ancestor exists.

[sizeOf](https://api.flutter.dev/flutter/widgets/MediaQuery/sizeOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [Size](https://api.flutter.dev/flutter/dart-ui/Size-class.html)

Returns size for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[systemGestureInsetsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/systemGestureInsetsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html)

Returns systemGestureInsets for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[textScaleFactorOf](https://api.flutter.dev/flutter/widgets/MediaQuery/textScaleFactorOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [double](https://api.flutter.dev/flutter/dart-core/double-class.html)

Deprecated. Will be removed in a future version of Flutter. Use [maybeTextScalerOf](https://api.flutter.dev/flutter/widgets/MediaQuery/maybeTextScalerOf.html) instead.

[textScalerOf](https://api.flutter.dev/flutter/widgets/MediaQuery/textScalerOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [TextScaler](https://api.flutter.dev/flutter/painting/TextScaler-class.html)

Returns the [TextScaler](https://api.flutter.dev/flutter/painting/TextScaler-class.html) for the nearest [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) ancestor or null if no such ancestor exists.

[viewInsetsOf](https://api.flutter.dev/flutter/widgets/MediaQuery/viewInsetsOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html)

Returns viewInsets for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[viewPaddingOf](https://api.flutter.dev/flutter/widgets/MediaQuery/viewPaddingOf.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html)

Returns viewPadding for the nearest MediaQuery ancestor or throws an exception, if no such ancestor exists.

[withClampedTextScaling](https://api.flutter.dev/flutter/widgets/MediaQuery/withClampedTextScaling.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, [double](https://api.flutter.dev/flutter/dart-core/double-class.html) minScaleFactor = 0.0, [double](https://api.flutter.dev/flutter/dart-core/double-class.html) maxScaleFactor = double.infinity, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child}) → [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)

Wraps the child in a [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) and applies [TextScaler.clamp](https://api.flutter.dev/flutter/painting/TextScaler/clamp.html) on the current [MediaQueryData.textScaler](https://api.flutter.dev/flutter/widgets/MediaQueryData/textScaler.html).

[withNoTextScaling](https://api.flutter.dev/flutter/widgets/MediaQuery/withNoTextScaling.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child}) → [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)

Wraps the child in a [MediaQuery](https://api.flutter.dev/flutter/widgets/MediaQuery-class.html) with its [MediaQueryData.textScaler](https://api.flutter.dev/flutter/widgets/MediaQueryData/textScaler.html) set to [TextScaler.noScaling](https://api.flutter.dev/flutter/painting/TextScaler/noScaling-constant.html).

## Padding

A widget that insets its child by the given padding.

When passing layout constraints to its child, padding shrinks the constraints by the given padding, causing the child to layout at a smaller size. Padding then sizes itself to its child's size, inflated by the padding, effectively creating empty space around the child.

### Why use a [Padding](https://api.flutter.dev/flutter/widgets/Padding-class.html) widget rather than a [Container](https://api.flutter.dev/flutter/widgets/Container-class.html) with a [Container.padding](https://api.flutter.dev/flutter/widgets/Container/padding.html) property?

There isn't really any difference between the two. If you supply a [Container.padding](https://api.flutter.dev/flutter/widgets/Container/padding.html) argument, [Container](https://api.flutter.dev/flutter/widgets/Container-class.html) builds a [Padding](https://api.flutter.dev/flutter/widgets/Padding-class.html) widget for you.

[Container](https://api.flutter.dev/flutter/widgets/Container-class.html) doesn't implement its properties directly. Instead, [Container](https://api.flutter.dev/flutter/widgets/Container-class.html) combines a number of simpler widgets together into a convenient package. For example, the [Container.padding](https://api.flutter.dev/flutter/widgets/Container/padding.html) property causes the container to build a [Padding](https://api.flutter.dev/flutter/widgets/Padding-class.html) widget and the [Container.decoration](https://api.flutter.dev/flutter/widgets/Container/decoration.html) property causes the container to build a [DecoratedBox](https://api.flutter.dev/flutter/widgets/DecoratedBox-class.html) widget. If you find [Container](https://api.flutter.dev/flutter/widgets/Container-class.html) convenient, feel free to use it. If not, feel free to build these simpler widgets in whatever combination meets your needs.

In fact, the majority of widgets in Flutter are combinations of other simpler widgets. Composition, rather than inheritance, is the primary mechanism for building up widgets.

See also:

* [EdgeInsets](https://api.flutter.dev/flutter/painting/EdgeInsets-class.html), the class that is used to describe the padding dimensions.
* [AnimatedPadding](https://api.flutter.dev/flutter/widgets/AnimatedPadding-class.html), which animates changes in [padding](https://api.flutter.dev/flutter/widgets/Padding/padding.html) over a given duration.
* [SliverPadding](https://api.flutter.dev/flutter/widgets/SliverPadding-class.html), the sliver equivalent of this widget.
* The [catalog of layout widgets](https://flutter.dev/widgets/layout/).

### Constructors

[Padding](https://api.flutter.dev/flutter/widgets/Padding/Padding.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [EdgeInsetsGeometry](https://api.flutter.dev/flutter/painting/EdgeInsetsGeometry-class.html) padding, [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)? child})

Creates a widget that insets its child.

const

### Properties

[child](https://api.flutter.dev/flutter/widgets/SingleChildRenderObjectWidget/child.html) → [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)?

The widget below this widget in the tree.

finalinherited

[hashCode](https://api.flutter.dev/flutter/widgets/Widget/hashCode.html) → [int](https://api.flutter.dev/flutter/dart-core/int-class.html)

The hash code for this object.

read-onlyinherited

[key](https://api.flutter.dev/flutter/widgets/Widget/key.html) → [Key](https://api.flutter.dev/flutter/foundation/Key-class.html)?

Controls how one widget replaces another widget in the tree.

finalinherited

[padding](https://api.flutter.dev/flutter/widgets/Padding/padding.html) → [EdgeInsetsGeometry](https://api.flutter.dev/flutter/painting/EdgeInsetsGeometry-class.html)

The amount of space by which to inset the child.

final

[runtimeType](https://api.flutter.dev/flutter/dart-core/Object/runtimeType.html) → [Type](https://api.flutter.dev/flutter/dart-core/Type-class.html)

A representation of the runtime type of the object.

read-onlyinherited

### Methods

[createElement](https://api.flutter.dev/flutter/widgets/SingleChildRenderObjectWidget/createElement.html)() → [SingleChildRenderObjectElement](https://api.flutter.dev/flutter/widgets/SingleChildRenderObjectElement-class.html)

RenderObjectWidgets always inflate to a [RenderObjectElement](https://api.flutter.dev/flutter/widgets/RenderObjectElement-class.html) subclass.

inherited

[createRenderObject](https://api.flutter.dev/flutter/widgets/Padding/createRenderObject.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [RenderPadding](https://api.flutter.dev/flutter/rendering/RenderPadding-class.html)

Creates an instance of the [RenderObject](https://api.flutter.dev/flutter/rendering/RenderObject-class.html) class that this [RenderObjectWidget](https://api.flutter.dev/flutter/widgets/RenderObjectWidget-class.html) represents, using the configuration described by this [RenderObjectWidget](https://api.flutter.dev/flutter/widgets/RenderObjectWidget-class.html).

override

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

Returns a list of DiagnosticsNode objects describing this node's children.

inherited

[debugFillProperties](https://api.flutter.dev/flutter/widgets/Padding/debugFillProperties.html)([DiagnosticPropertiesBuilder](https://api.flutter.dev/flutter/foundation/DiagnosticPropertiesBuilder-class.html) properties) → void

Add additional properties associated with the node.

override

[didUnmountRenderObject](https://api.flutter.dev/flutter/widgets/RenderObjectWidget/didUnmountRenderObject.html)(covariant [RenderObject](https://api.flutter.dev/flutter/rendering/RenderObject-class.html) renderObject) → void

A render object previously associated with this widget has been removed from the tree. The given [RenderObject](https://api.flutter.dev/flutter/rendering/RenderObject-class.html) will be of the same type as returned by this object's [createRenderObject](https://api.flutter.dev/flutter/widgets/Padding/createRenderObject.html).

inherited

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

Invoked when a nonexistent method or property is accessed.

inherited

[toDiagnosticsNode](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toDiagnosticsNode.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html)? name, [DiagnosticsTreeStyle](https://api.flutter.dev/flutter/foundation/DiagnosticsTreeStyle.html)? style}) → [DiagnosticsNode](https://api.flutter.dev/flutter/foundation/DiagnosticsNode-class.html)

Returns a debug representation of the object that is used by debugging tools and by [DiagnosticsNode.toStringDeep](https://api.flutter.dev/flutter/foundation/DiagnosticsNode/toStringDeep.html).

inherited

[toString](https://api.flutter.dev/flutter/foundation/Diagnosticable/toString.html)({[DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.info}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

A string representation of this object.

inherited

[toStringDeep](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toStringDeep.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html) prefixLineOne = '', [String](https://api.flutter.dev/flutter/dart-core/String-class.html)? prefixOtherLines, [DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.debug}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

Returns a string representation of this node and its descendants.

inherited

[toStringShallow](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toStringShallow.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html) joiner = ', ', [DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.debug}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

Returns a one-line detailed description of the object.

inherited

[toStringShort](https://api.flutter.dev/flutter/widgets/Widget/toStringShort.html)() → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

A short, textual description of this widget.

inherited

[updateRenderObject](https://api.flutter.dev/flutter/widgets/Padding/updateRenderObject.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context, covariant [RenderPadding](https://api.flutter.dev/flutter/rendering/RenderPadding-class.html) renderObject) → void

Copies the configuration described by this [RenderObjectWidget](https://api.flutter.dev/flutter/widgets/RenderObjectWidget-class.html) to the given [RenderObject](https://api.flutter.dev/flutter/rendering/RenderObject-class.html), which will be of the same type as returned by this object's [createRenderObject](https://api.flutter.dev/flutter/widgets/Padding/createRenderObject.html).

override

### Operators

[operator ==](https://api.flutter.dev/flutter/widgets/Widget/operator_equals.html)([Object](https://api.flutter.dev/flutter/dart-core/Object-class.html) other) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

The equality operator.

inherited

## theme

Applies a theme to descendant widgets.

A theme describes the colors and typographic choices of an application

Descendant widgets obtain the current theme's [ThemeData](https://api.flutter.dev/flutter/material/ThemeData-class.html) object using [Theme.of](https://api.flutter.dev/flutter/material/Theme/of.html). When a widget uses [Theme.of](https://api.flutter.dev/flutter/material/Theme/of.html), it is automatically rebuilt if the theme later changes, so that the changes can be applied.

The [Theme](https://api.flutter.dev/flutter/material/Theme-class.html) widget implies an [IconTheme](https://api.flutter.dev/flutter/widgets/IconTheme-class.html) widget, set to the value of the [ThemeData.iconTheme](https://api.flutter.dev/flutter/material/ThemeData/iconTheme.html) of the [data](https://api.flutter.dev/flutter/material/Theme/data.html) for the [Theme](https://api.flutter.dev/flutter/material/Theme-class.html).

### Constructors

[Theme](https://api.flutter.dev/flutter/material/Theme/Theme.html)({[Key](https://api.flutter.dev/flutter/foundation/Key-class.html)? key, required [ThemeData](https://api.flutter.dev/flutter/material/ThemeData-class.html) data, required [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html) child})

Applies the given theme [data](https://api.flutter.dev/flutter/material/Theme/data.html) to [child](https://api.flutter.dev/flutter/material/Theme/child.html).

const

### Properties

[child](https://api.flutter.dev/flutter/material/Theme/child.html) → [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)

The widget below this widget in the tree.

final

[data](https://api.flutter.dev/flutter/material/Theme/data.html) → [ThemeData](https://api.flutter.dev/flutter/material/ThemeData-class.html)

Specifies the color and typography values for descendant widgets.

final

[hashCode](https://api.flutter.dev/flutter/widgets/Widget/hashCode.html) → [int](https://api.flutter.dev/flutter/dart-core/int-class.html)

The hash code for this object.

read-onlyinherited

[key](https://api.flutter.dev/flutter/widgets/Widget/key.html) → [Key](https://api.flutter.dev/flutter/foundation/Key-class.html)?

Controls how one widget replaces another widget in the tree.

finalinherited

[runtimeType](https://api.flutter.dev/flutter/dart-core/Object/runtimeType.html) → [Type](https://api.flutter.dev/flutter/dart-core/Type-class.html)

A representation of the runtime type of the object.

read-onlyinherited

### Methods

[build](https://api.flutter.dev/flutter/material/Theme/build.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [Widget](https://api.flutter.dev/flutter/widgets/Widget-class.html)

Describes the part of the user interface represented by this widget.

override

[createElement](https://api.flutter.dev/flutter/widgets/StatelessWidget/createElement.html)() → [StatelessElement](https://api.flutter.dev/flutter/widgets/StatelessElement-class.html)

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

inherited

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

Returns a list of DiagnosticsNode objects describing this node's children.

inherited

[debugFillProperties](https://api.flutter.dev/flutter/material/Theme/debugFillProperties.html)([DiagnosticPropertiesBuilder](https://api.flutter.dev/flutter/foundation/DiagnosticPropertiesBuilder-class.html) properties) → void

Add additional properties associated with the node.

override

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

Invoked when a nonexistent method or property is accessed.

inherited

[toDiagnosticsNode](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toDiagnosticsNode.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html)? name, [DiagnosticsTreeStyle](https://api.flutter.dev/flutter/foundation/DiagnosticsTreeStyle.html)? style}) → [DiagnosticsNode](https://api.flutter.dev/flutter/foundation/DiagnosticsNode-class.html)

Returns a debug representation of the object that is used by debugging tools and by [DiagnosticsNode.toStringDeep](https://api.flutter.dev/flutter/foundation/DiagnosticsNode/toStringDeep.html).

inherited

[toString](https://api.flutter.dev/flutter/foundation/Diagnosticable/toString.html)({[DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.info}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

A string representation of this object.

inherited

[toStringDeep](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toStringDeep.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html) prefixLineOne = '', [String](https://api.flutter.dev/flutter/dart-core/String-class.html)? prefixOtherLines, [DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.debug}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

Returns a string representation of this node and its descendants.

inherited

[toStringShallow](https://api.flutter.dev/flutter/foundation/DiagnosticableTree/toStringShallow.html)({[String](https://api.flutter.dev/flutter/dart-core/String-class.html) joiner = ', ', [DiagnosticLevel](https://api.flutter.dev/flutter/foundation/DiagnosticLevel.html) minLevel = DiagnosticLevel.debug}) → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

Returns a one-line detailed description of the object.

inherited

[toStringShort](https://api.flutter.dev/flutter/widgets/Widget/toStringShort.html)() → [String](https://api.flutter.dev/flutter/dart-core/String-class.html)

A short, textual description of this widget.

inherited

### Operators

[operator ==](https://api.flutter.dev/flutter/widgets/Widget/operator_equals.html)([Object](https://api.flutter.dev/flutter/dart-core/Object-class.html) other) → [bool](https://api.flutter.dev/flutter/dart-core/bool-class.html)

The equality operator.

inherited

### Static Methods

[of](https://api.flutter.dev/flutter/material/Theme/of.html)([BuildContext](https://api.flutter.dev/flutter/widgets/BuildContext-class.html) context) → [ThemeData](https://api.flutter.dev/flutter/material/ThemeData-class.html)

The data from the closest [Theme](https://api.flutter.dev/flutter/material/Theme-class.html) instance that encloses the given context.