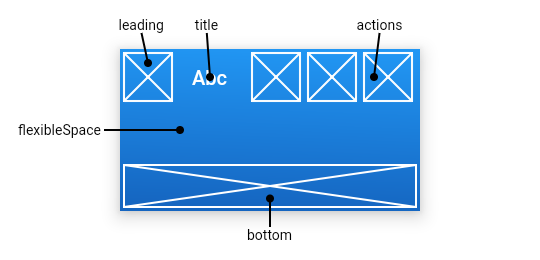
# AppBar class

A material design app bar.

An app bar consists of a toolbar and potentially other widgets, such as a [TabBar](https://docs.flutter.io/flutter/material/TabBar-class.html) and a [FlexibleSpaceBar](https://docs.flutter.io/flutter/material/FlexibleSpaceBar-class.html). App bars typically expose one or more common [actions](https://docs.flutter.io/flutter/material/AppBar/actions.html) with [IconButton](https://docs.flutter.io/flutter/material/IconButton-class.html)s which are optionally followed by a [PopupMenuButton](https://docs.flutter.io/flutter/material/PopupMenuButton-class.html) for less common operations (sometimes called the "overflow menu").

App bars are typically used in the [Scaffold.appBar](https://docs.flutter.io/flutter/material/Scaffold/appBar.html) property, which places the app bar as a fixed-height widget at the top of the screen. For a scrollable app bar, see [SliverAppBar](https://docs.flutter.io/flutter/material/SliverAppBar-class.html), which embeds an [AppBar](https://docs.flutter.io/flutter/material/AppBar-class.html) in a sliver for use in a [CustomScrollView](https://docs.flutter.io/flutter/widgets/CustomScrollView-class.html).

The AppBar displays the toolbar widgets, [leading](https://docs.flutter.io/flutter/material/AppBar/leading.html), [title](https://docs.flutter.io/flutter/material/AppBar/title.html), and [actions](https://docs.flutter.io/flutter/material/AppBar/actions.html), above the [bottom](https://docs.flutter.io/flutter/material/AppBar/bottom.html) (if any). The [bottom](https://docs.flutter.io/flutter/material/AppBar/bottom.html)is usually used for a [TabBar](https://docs.flutter.io/flutter/material/TabBar-class.html). If a [flexibleSpace](https://docs.flutter.io/flutter/material/AppBar/flexibleSpace.html) widget is specified then it is stacked behind the toolbar and the bottom widget. The following diagram shows where each of these slots appears in the toolbar when the writing language is left-to-right (e.g. English):



If the [leading](https://docs.flutter.io/flutter/material/AppBar/leading.html) widget is omitted, but the [AppBar](https://docs.flutter.io/flutter/material/AppBar-class.html) is in a [Scaffold](https://docs.flutter.io/flutter/material/Scaffold-class.html) with a [Drawer](https://docs.flutter.io/flutter/material/Drawer-class.html), then a button will be inserted to open the drawer. Otherwise, if the nearest [Navigator](https://docs.flutter.io/flutter/widgets/Navigator-class.html) has any previous routes, a [BackButton](https://docs.flutter.io/flutter/material/BackButton-class.html) is inserted instead. This behavior can be turned off by setting the [automaticallyImplyLeading](https://docs.flutter.io/flutter/material/AppBar/automaticallyImplyLeading.html) to false. In that case a null leading widget will result in the middle/title widget stretching to start.

## Sample code

**new** AppBar(

title: **new** Text('My Fancy Dress'),

actions: <Widget>[

**new** IconButton(

icon: **new** Icon(Icons.playlist\_play),

tooltip: 'Air it',

onPressed: \_airDress,

),

**new** IconButton(

icon: **new** Icon(Icons.playlist\_add),

tooltip: 'Restitch it',

onPressed: \_restitchDress,

),

**new** IconButton(

icon: **new** Icon(Icons.playlist\_add\_check),

tooltip: 'Repair it',

onPressed: \_repairDress,

),

],

)

See also:

* [Scaffold](https://docs.flutter.io/flutter/material/Scaffold-class.html), which displays the [AppBar](https://docs.flutter.io/flutter/material/AppBar-class.html) in its [Scaffold.appBar](https://docs.flutter.io/flutter/material/Scaffold/appBar.html) slot.
* [SliverAppBar](https://docs.flutter.io/flutter/material/SliverAppBar-class.html), which uses [AppBar](https://docs.flutter.io/flutter/material/AppBar-class.html) to provide a flexible app bar that can be used in a [CustomScrollView](https://docs.flutter.io/flutter/widgets/CustomScrollView-class.html).
* [TabBar](https://docs.flutter.io/flutter/material/TabBar-class.html), which is typically placed in the [bottom](https://docs.flutter.io/flutter/material/AppBar/bottom.html) slot of the [AppBar](https://docs.flutter.io/flutter/material/AppBar-class.html) if the screen has multiple pages arranged in tabs.
* [IconButton](https://docs.flutter.io/flutter/material/IconButton-class.html), which is used with [actions](https://docs.flutter.io/flutter/material/AppBar/actions.html) to show buttons on the app bar.
* [PopupMenuButton](https://docs.flutter.io/flutter/material/PopupMenuButton-class.html), to show a popup menu on the app bar, via [actions](https://docs.flutter.io/flutter/material/AppBar/actions.html).
* [FlexibleSpaceBar](https://docs.flutter.io/flutter/material/FlexibleSpaceBar-class.html), which is used with [flexibleSpace](https://docs.flutter.io/flutter/material/AppBar/flexibleSpace.html) when the app bar can expand and collapse.
* [material.google.com/layout/structure.html#structure-toolbars](https://material.google.com/layout/structure.html" \l "structure-toolbars)

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)
* [StatefulWidget](https://docs.flutter.io/flutter/widgets/StatefulWidget-class.html)
* AppBar

Implements

* [PreferredSizeWidget](https://docs.flutter.io/flutter/widgets/PreferredSizeWidget-class.html)

## Constructors

[**AppBar**](https://docs.flutter.io/flutter/material/AppBar/AppBar.html)({[Key](https://docs.flutter.io/flutter/foundation/Key-class.html) key, [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html) leading, [bool](https://docs.flutter.io/flutter/dart-core/bool-class.html) automaticallyImplyLeading: true, [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html) title, [List](https://docs.flutter.io/flutter/dart-core/List-class.html)<[Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html)> actions, [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html) flexibleSpace, [PreferredSizeWidget](https://docs.flutter.io/flutter/widgets/PreferredSizeWidget-class.html) bottom, [double](https://docs.flutter.io/flutter/dart-core/double-class.html) elevation: 4.0, [Color](https://docs.flutter.io/flutter/dart-ui/Color-class.html) backgroundColor, [Brightness](https://docs.flutter.io/flutter/services/Brightness-class.html) brightness, [IconThemeData](https://docs.flutter.io/flutter/widgets/IconThemeData-class.html) iconTheme, [TextTheme](https://docs.flutter.io/flutter/material/TextTheme-class.html) textTheme, [bool](https://docs.flutter.io/flutter/dart-core/bool-class.html) primary: true, [bool](https://docs.flutter.io/flutter/dart-core/bool-class.html) centerTitle, [double](https://docs.flutter.io/flutter/dart-core/double-class.html) titleSpacing: NavigationToolbar.kMiddleSpacing, [double](https://docs.flutter.io/flutter/dart-core/double-class.html) toolbarOpacity: 1.0, [double](https://docs.flutter.io/flutter/dart-core/double-class.html) bottomOpacity: 1.0 })

Creates a material design app bar. [[...]](https://docs.flutter.io/flutter/material/AppBar/AppBar.html)

## Properties

[**actions**](https://docs.flutter.io/flutter/material/AppBar/actions.html) → [List](https://docs.flutter.io/flutter/dart-core/List-class.html)<[Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html)>

Widgets to display after the title widget. [[...]](https://docs.flutter.io/flutter/material/AppBar/actions.html)

*final*

[**automaticallyImplyLeading**](https://docs.flutter.io/flutter/material/AppBar/automaticallyImplyLeading.html) → [bool](https://docs.flutter.io/flutter/dart-core/bool-class.html)

Controls whether we should try to imply the leading widget if null. [[...]](https://docs.flutter.io/flutter/material/AppBar/automaticallyImplyLeading.html)

*final*

[**backgroundColor**](https://docs.flutter.io/flutter/material/AppBar/backgroundColor.html) → [Color](https://docs.flutter.io/flutter/dart-ui/Color-class.html)

The color to use for the app bar's material. Typically this should be set along with brightness, iconTheme, textTheme. [[...]](https://docs.flutter.io/flutter/material/AppBar/backgroundColor.html)

*final*

[**bottom**](https://docs.flutter.io/flutter/material/AppBar/bottom.html) → [PreferredSizeWidget](https://docs.flutter.io/flutter/widgets/PreferredSizeWidget-class.html)

This widget appears across the bottom of the app bar. [[...]](https://docs.flutter.io/flutter/material/AppBar/bottom.html)

*final*

[**bottomOpacity**](https://docs.flutter.io/flutter/material/AppBar/bottomOpacity.html) → [double](https://docs.flutter.io/flutter/dart-core/double-class.html)

How opaque the bottom part of the app bar is. [[...]](https://docs.flutter.io/flutter/material/AppBar/bottomOpacity.html)

*final*

[**brightness**](https://docs.flutter.io/flutter/material/AppBar/brightness.html) → [Brightness](https://docs.flutter.io/flutter/services/Brightness-class.html)

The brightness of the app bar's material. Typically this is set along with backgroundColor, iconTheme, textTheme. [[...]](https://docs.flutter.io/flutter/material/AppBar/brightness.html)

*final*

[**centerTitle**](https://docs.flutter.io/flutter/material/AppBar/centerTitle.html) → [bool](https://docs.flutter.io/flutter/dart-core/bool-class.html)

Whether the title should be centered. [[...]](https://docs.flutter.io/flutter/material/AppBar/centerTitle.html)

*final*

[**elevation**](https://docs.flutter.io/flutter/material/AppBar/elevation.html) → [double](https://docs.flutter.io/flutter/dart-core/double-class.html)

The z-coordinate at which to place this app bar. This controls the size of the shadow below the app bar. [[...]](https://docs.flutter.io/flutter/material/AppBar/elevation.html)

*final*

[**flexibleSpace**](https://docs.flutter.io/flutter/material/AppBar/flexibleSpace.html) → [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html)

This widget is stacked behind the toolbar and the tabbar. It's height will be the same as the app bar's overall height. [[...]](https://docs.flutter.io/flutter/material/AppBar/flexibleSpace.html)

*final*

[**iconTheme**](https://docs.flutter.io/flutter/material/AppBar/iconTheme.html) → [IconThemeData](https://docs.flutter.io/flutter/widgets/IconThemeData-class.html)

The color, opacity, and size to use for app bar icons. Typically this is set along with backgroundColor, brightness, textTheme. [[...]](https://docs.flutter.io/flutter/material/AppBar/iconTheme.html)

*final*

[**leading**](https://docs.flutter.io/flutter/material/AppBar/leading.html) → [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html)

A widget to display before the title. [[...]](https://docs.flutter.io/flutter/material/AppBar/leading.html)

*final*

[**preferredSize**](https://docs.flutter.io/flutter/material/AppBar/preferredSize.html) → [Size](https://docs.flutter.io/flutter/dart-ui/Size-class.html)

A size whose height is the sum of [kToolbarHeight](https://docs.flutter.io/flutter/material/kToolbarHeight-constant.html) and the [bottom](https://docs.flutter.io/flutter/material/AppBar/bottom.html) widget's preferred height. [[...]](https://docs.flutter.io/flutter/material/AppBar/preferredSize.html)

*final*

[**primary**](https://docs.flutter.io/flutter/material/AppBar/primary.html) → [bool](https://docs.flutter.io/flutter/dart-core/bool-class.html)

Whether this app bar is being displayed at the top of the screen. [[...]](https://docs.flutter.io/flutter/material/AppBar/primary.html)

*final*

[**textTheme**](https://docs.flutter.io/flutter/material/AppBar/textTheme.html) → [TextTheme](https://docs.flutter.io/flutter/material/TextTheme-class.html)

The typographic styles to use for text in the app bar. Typically this is set along with brightness backgroundColor, iconTheme. [[...]](https://docs.flutter.io/flutter/material/AppBar/textTheme.html)

*final*

[**title**](https://docs.flutter.io/flutter/material/AppBar/title.html) → [Widget](https://docs.flutter.io/flutter/widgets/Widget-class.html)

The primary widget displayed in the appbar. [[...]](https://docs.flutter.io/flutter/material/AppBar/title.html)

*final*

[**titleSpacing**](https://docs.flutter.io/flutter/material/AppBar/titleSpacing.html) → [double](https://docs.flutter.io/flutter/dart-core/double-class.html)

The spacing around title content on the horizontal axis. This spacing is applied even if there is no leading content or actions. If you want title to take all the space available, set this value to 0.0. [[...]](https://docs.flutter.io/flutter/material/AppBar/titleSpacing.html)

*final*

[**toolbarOpacity**](https://docs.flutter.io/flutter/material/AppBar/toolbarOpacity.html) → [double](https://docs.flutter.io/flutter/dart-core/double-class.html)

How opaque the toolbar part of the app bar is. [[...]](https://docs.flutter.io/flutter/material/AppBar/toolbarOpacity.html)

*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

[**createState**](https://docs.flutter.io/flutter/material/AppBar/createState.html)( → \_AppBarState

Creates the mutable state for this widget at a given location in the tree. [[...]](https://docs.flutter.io/flutter/material/AppBar/createState.html)

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

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

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

Add additional properties associated with the node. [[...]](https://docs.flutter.io/flutter/widgets/Widget/debugFillProperties.html)

*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*