

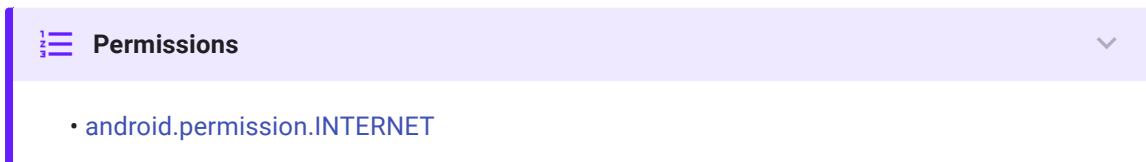
List Picker

Category	Requires	Version
User Interface	API 19, Android 4.4 - 4.4.4 KitKat	17

Overview

A button that, when clicked on, displays a list of texts for the user to choose among. The texts can be specified through the Designer or Blocks Editor by setting the `ElementsFromString` property to their string-separated concatenation (for example, *choice 1, choice 2, choice 3*) or by setting the `Elements` property to a List in the Blocks editor.

Setting property `ShowFilterBar` to true, will make the list searchable. Other properties affect the appearance of the button (`TextAlignment` , `BackgroundColor` , etc.) and whether it can be clicked on (`Enabled`).



Events

After Picking

Event to be raised after the picker activity returns its result and the properties have been filled in.

Params	
selection	Text

Before Picking

Event to raise when the button of the component is clicked or the list is shown using the Open block. This event occurs before the list of items is displayed, and can be used to prepare the list before it is shown.

Got Focus

Indicates the cursor moved over the button so it is now possible to click it.

Lost Focus

Indicates the cursor moved away from the button so it is now no longer possible to click it.

Touch Down

Indicates that the button was pressed down.

Touch Up

Indicates that a button has been released.

Methods

Button Click

Perform a button click as function.

Open

Opens the picker, as though the user clicked on it.

Set Animation Style

Allows you to set animation style. Valid (case-insensitive) values are: ChasingDots, Circle, CubeGrid, DoubleBounce, FadingCircle, FoldingCube, Pulse, RotatingCircle, RotatingPlane, ThreeBounce, WanderingCubes, Wave. If invalid style is used, animation will be removed. Position can be: top, left, right, bottom. Size can be 100.

Params	
style	<div>Text</div>
position	<div>Text</div>
size	<div>Number</div>
color	<div>Number</div>

Set Shadow

Place a blurred shadow of text underneath the text, drawn with the specified x, y, radius, color (e.g. -11, 12, 13, black).

Params	
x	<div>Number</div>
y	<div>Number</div>
radius	<div>Number</div>
color	<div>Number</div>

With Icon From Font Awesome

Show an image on the given position near to the button. You can use following words for the position: 'Left', 'Right', 'Top' or 'Bottom'. Use the padding to add space between the icon and text. Use a material icon as the button icon without uploading a image resource into your project. You can find the icon name (or code) here at <https://fontawesome.com/cheatsheet> Use as example for a heart icon just 'f004'.

Params	
position	<input type="text" value="Text"/>
icon Name	<input type="text" value="Text"/>
icon Color	<input type="text" value="Number"/>
padding	<input type="text" value="Number"/>
size	<input type="text" value="Number"/>

With Icon From Material Font

Show an image on the given position near to the button. You can use following words for the position: 'Left', 'Right', 'Top' or 'Bottom'. Use the padding to add space between the icon and text. Use a material icon as the button icon without uploading a image resource into your project. You can find the icon name (or code) here at <https://material.io/icons>

Params	
position	Text
icon Name	Text
icon Color	Number
padding	Number
size	Number

With Icon From Picture

Show an image on the given position near to the button. You can use following words for the position: 'Left', 'Right', 'Top' or 'Bottom'. Use the padding to add space between the icon and text.

Params	
position	Text
picture	Text
padding	Number
width	Number
height	Number

Properties

Background Color

Color
Default: #444444FF
☐
—
Read
Write
-
Designer
Blocks

Returns the button's background color

Border Shadow

Available as Advanced Property

Boolean **Default: *True*** — **Write** - **Designer**

Returns true if the button have a outside border shadow on click.

Elements

List — **Read** **Write** - **Blocks**

Elements property getter method

Elements From String

Text — **Write** - **Designer** **Blocks**

ElementsFromString property setter method

Enabled

Boolean **Default: *True*** — **Read** **Write** - **Designer** **Blocks**

If set, user can tap check box to cause action.

Font Bold

Boolean **Default: *False*** — **Read** **Write** - **Designer** **Blocks**

If set, button text is displayed in bold.

Font Italic

Boolean **Default: *False*** — **Read** **Write** - **Designer** **Blocks**

If set, button text is displayed in italics.

Font Size

Number Default: 14.0 — Read Write - Designer Blocks

Point size for button text.

Font Typeface

Number Default: 0 — Write - Designer

Font family for button text.

Font Typeface Import

Available as [Advanced](#) Property

Text — Write - Designer Blocks

Set a custom font.

Height

Number — Read Write - Blocks

Specifies the component's vertical height, measured in pixels.

Height Percent

Number — Write - Blocks

Specifies the component's vertical height as a percentage of the height of its parent Component.

HTML Format

Available as [Advanced](#) Property

Boolean Default: *False* — Write - Designer

If true, then this button will show html text else it will show plain text. Note: Not all HTML is supported.

Image

Text — Read Write - Designer Blocks

Image to display on button.

Item Background Color

Color Default: *#000000FF*  — Read Write - Designer Blocks

The background color of the ListPicker items.

Item Text Color

Color Default: *#FFFFFF*  — Read Write - Designer Blocks

The text color of the ListPicker items.

Rotation Angle

Available as Advanced Property

Number Default: *0.0* — Read Write - Designer Blocks

Sets the degrees that the button is rotated around the pivot point. Increasing values result in clockwise rotation.

Selection

Text — Read Write - Designer Blocks

The selected item. When directly changed by the programmer, the SelectionIndex property is also changed to the first item in the ListPicker with the given value. If the value does not appear, SelectionIndex will be set to 0.

Selection Index

Number — Read Write - Blocks

The index of the currently selected item, starting at 1. If no item is selected, the value will be 0. If an attempt is made to set this to a number less than 1 or greater than the number of items in the ListPicker, SelectionIndex will be set to 0, and Selection will be set to the empty text.

Shape

Number Default: 0 — Write - Designer

Specifies the button's shape (default, rounded, rectangular, oval). The shape will not be visible if an Image is being displayed.

Show Feedback

Available as [Advanced](#) Property

Boolean Default: True — Read Write - Designer Blocks

Specifies if a visual feedback should be shown for a button that has an image as background.

Show Filter Bar

Boolean Default: False — Read Write - Designer Blocks

Returns current state of ShowFilterBar indicating if Search Filter Bar will be displayed on ListPicker or not

Status Bar Color

Available as [Advanced](#) Property

Color Default: #2196F2FF  — Read Write - **Designer** **Blocks**

The StatusBarColor of the ListPicker.

Status Bar Visible

Available as [Advanced](#) Property

Boolean Default: *True* — Read Write - **Designer** **Blocks**

Whether to show status bar

Text

Text — Read Write - **Designer** **Blocks**

Text to display on button.

Text Alignment

Number Default: 1 — Write - **Designer**

Left, center, or right.

Text Color

Color Default: #FFFFFFF  — Read Write - **Designer** **Blocks**

Color for button text.

Title

Text — Read Write - **Designer** **Blocks**

Optional title displayed at the top of the list of choices.

Title Bar Color

Available as [Advanced](#) Property

Color Default: #03A9F3FF  — Read Write - Designer Blocks

The TitleBarColor of the ListPicker.

Title Visible

Available as [Advanced](#) Property

Boolean Default: *True* — Read Write - Designer Blocks

Whether to show title bar

Touch Color

Available as [Advanced](#) Property

Color Default: #CCCCCCFF  — Read Write - Designer Blocks

Set the buttons touch color.

Visible

Boolean Default: *True* — Read Write - Designer Blocks

Returns true iff the component is visible.

Width

Number — Read Write - Blocks

Specifies the component's horizontal width, measured in pixels.

Width Percent

Number — Write - Blocks

Specifies the component's horizontal width as a percentage of the Width of its parent Component.

Last update: January 25, 2020