aakarshrestha / compose-vertical-slider

Type / to search

<> Code   Issues   Pull requests   Actions   Projects   Security   Insights

compose-vertical-slider   Public

main	1 Branch	4 Tags

Go to file   t   Go to file   +   About   Code   ...

Aakar Shrestha	Upgrad...	a0f9c41 · 3 years ago	29 Commits

📁 .gradle	Upgraded the library ver...	3 years ago
📁 .idea	Upgraded the library ver...	3 years ago
📁 app	Upgraded the library ver...	3 years ago
📁 composeVerticalS...	Updated the ComposeV...	3 years ago
📁 gradle/wrapper	initial commit	3 years ago
📄 .DS_Store	initial commit	3 years ago
📄 LICENSE	Create LICENSE	3 years ago
📄 README.md	Update README.md	3 years ago
📄 build.gradle	library version 0.1.0 add...	3 years ago
📄 gradle.properties	initial commit	3 years ago

About

A small library that provides the Vertical Slider that is created in Jetpack Compose.

📖 Readme

⚖️ MIT license

〰️ Activity

☆ 15 stars

👁 1 watching

🍴 1 fork

Report repository

Releases   4

🏷 Vertical Slider - Jetpack Com...   Latest
on May 11, 2021

+ 3 releases

📄	gradlew	initial commit	3 years ago
📄	gradlew.bat	initial commit	3 years ago
📄	local.properties	initial commit	3 years ago
📄	settings.gradle	initial commit	3 years ago

📖 **README**   ⚖️ **MIT license**   ✏️   ☰

Release version: `JitPack` `0.4.0`

compose-vertical-slider

A simple vertical slider that is created in Jetpack Compose. ComposeVerticalSlider allows users to make selections from the range of values by dragging the slider in vertical axis. With screen orientation change, ComposeVerticalSlider also adjusts accordingly.

The min value that is allowed to choose is 0 and max value is 100.

Download

Add it in your root build.gradle at the end of repositories:

```
allprojects {
        repositories {
```

Packages

No packages published

Languages

● **Kotlin** 100.0%

```
            ...
            maven { url 'https://jitpack.io' }
        }
    }
```
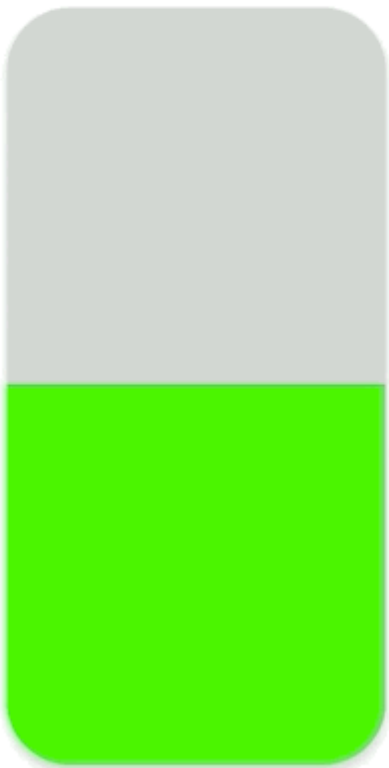
Add it in your app build.gradle:

```
dependencies {
    implementation "com.github.aakarshrestha:compose-vertical-slider:[version]"
}
```

Compose Vertical Slider demo

50

Usage

```
@Composable
fun ComposeVerticalSlider(
    state: ComposeVerticalSliderState,
    progressValue: Int? = null,
    enabled: Boolean = true,
    width: Dp = 140.dp,
    height: Dp = 300.dp,
    radius: CornerRadius = CornerRadius(80f, 80f),
    trackColor: Color = Color.LightGray,
    progressTrackColor: Color = Color.Green,
    onProgressChanged: (Int) -> Unit,
    onStopTrackingTouch: (Int) -> Unit
)
```

- @param state maintains the state of ComposeVerticalSlider.
- @param progressValue current value of the Slider. It can be null or the value can be assigned to it.
- @param enabled whether or not component is enabled and can we interacted with or not.
- @param width width of the slider
- @param height height of the slider
- @param radius corner curves of the slider
- @param trackColor that can be set to a desired color.
- @param progressTrackColor that can be set to a desired color.

- @param onProgressChanged lambda that is invoked when the slider value changes when [MotionEvent.ACTION_MOVE] is triggered.
- @param onStopTrackingTouch lambda that is invoked when the slider value changes when [MotionEvent.ACTION_UP] is triggered.

Implementation: Check out the app to see how it works.

```
@Composable
fun VerticalSlider(progressValue: Int? = null, value: (Int) ->
Unit) {

    val state = rememberComposeVerticalSliderState()

    ComposeVerticalSlider(
        state = state,
        enabled = state.isEnabled.value,
        progressValueSet = progressValue,
        onProgressChanged = {
            value(it)
        },
        onStopTrackingTouch = {
            value(it)
        }
    )
}
```

License