

# **Komponen dalam Mendesain Aplikasi**

**TI2137 - Pengembangan Aplikasi Mobil  
Front-End**



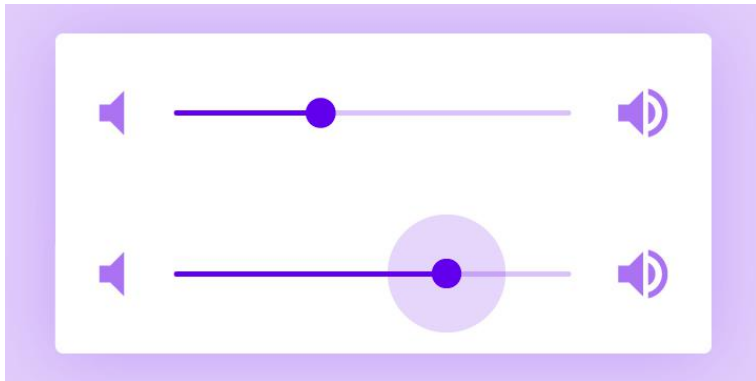
# Team Teaching

- Sunaryo Winardi, S.Kom., M.TI.
- **Sio Jurnal**is Pipin, S.Kom., M.Kom.
- M. Taufiq Hidayat Pohan, S.Kom.
- Richy Rotuahta Saragih, S.Kom.

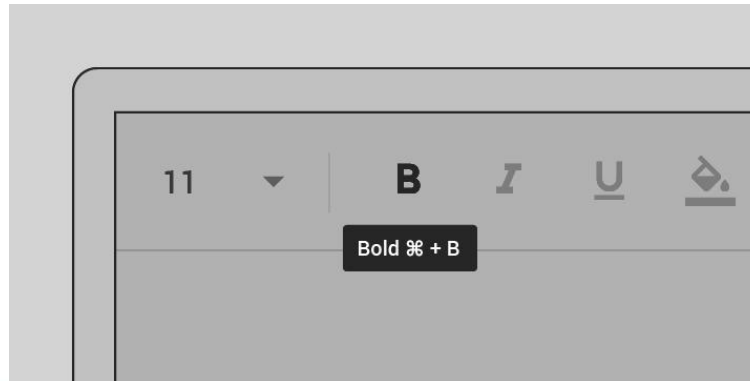


MIKROSKIL

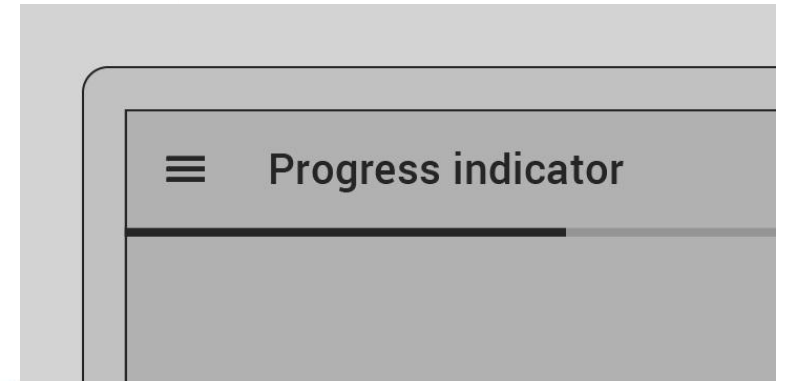
# Komponen dalam Mendesain Aplikasi Part. 6



**Sliders**



**ToolTips**



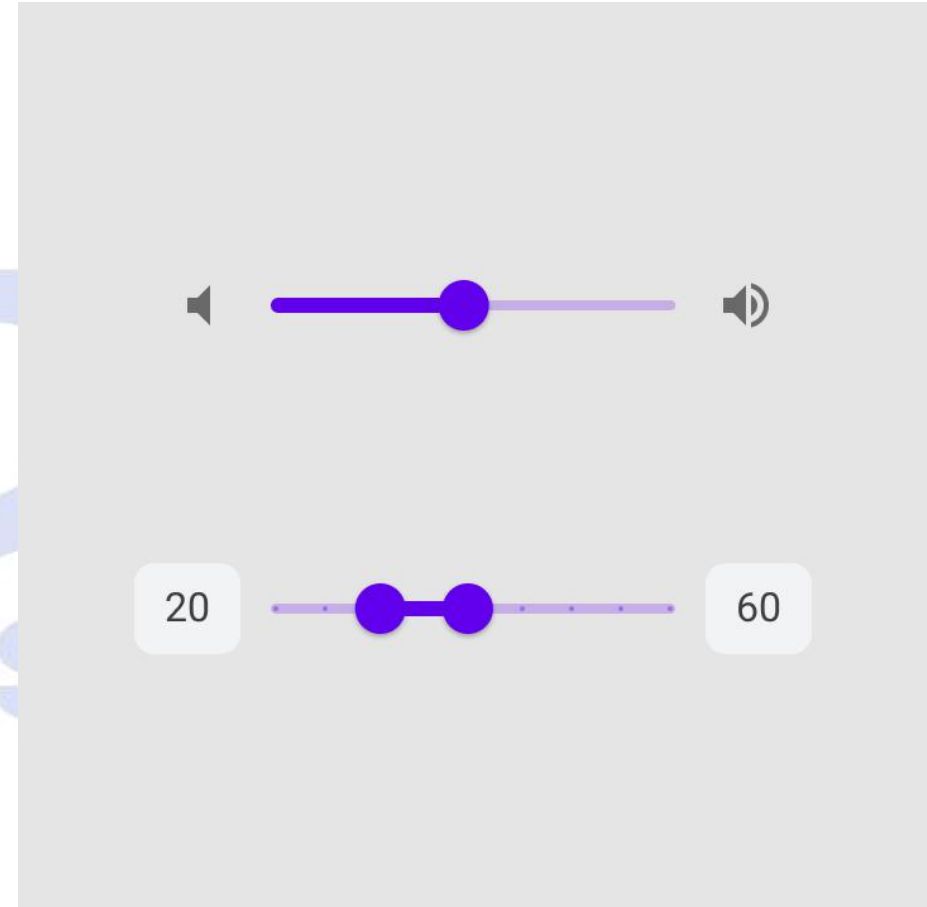
**Progress Indicator**

# Sliders

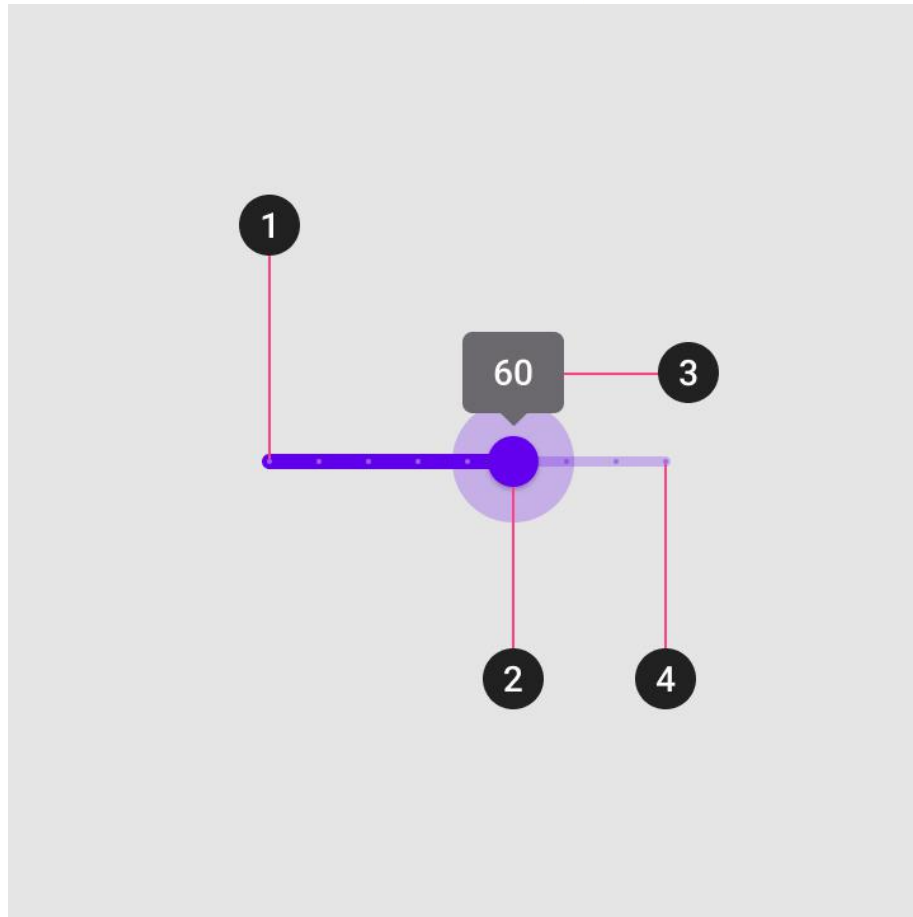


# Sliders adalah...

- Rentang nilai di sepanjang Bar
- User dapat memilih single value
- Memiliki icon di kedua sisi (awal dan akhir Bar)
- Perubahan yang dibuat dengan sliders bersifat langsung
- Slider menampilkan current state



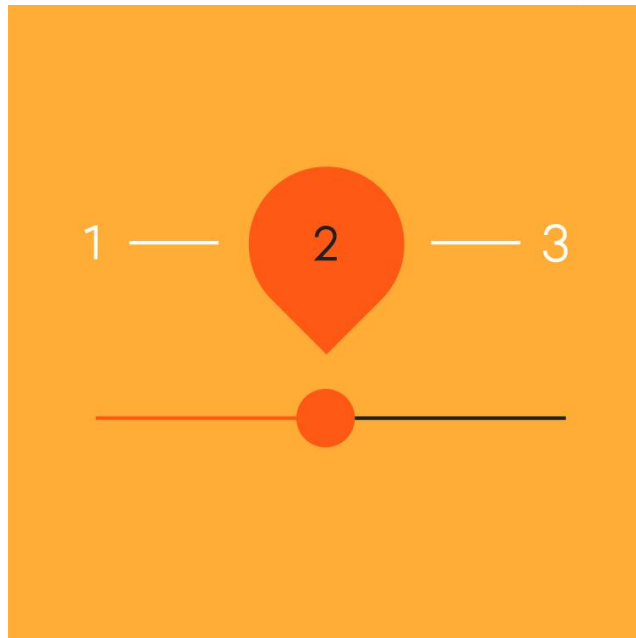
# Anatomi pada Cards



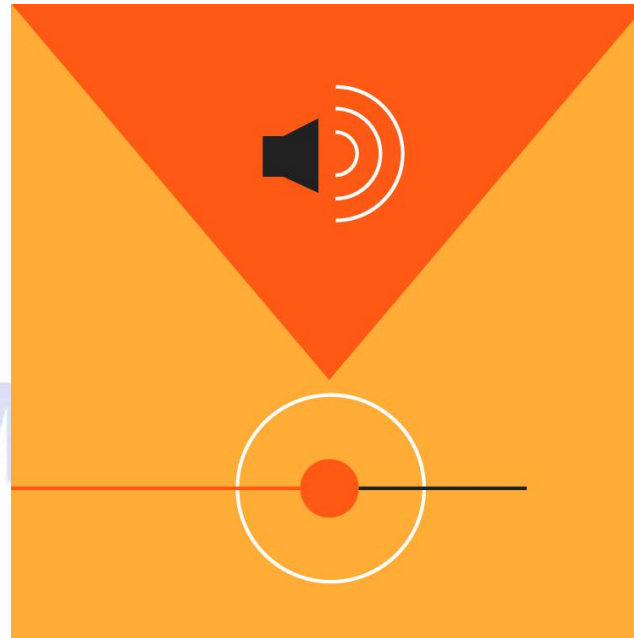
- 1. Track**
- 2. Thumb**
- 3. Value label (optional)**
- 4. Tick mark (optional)**

# Principle of Sliders

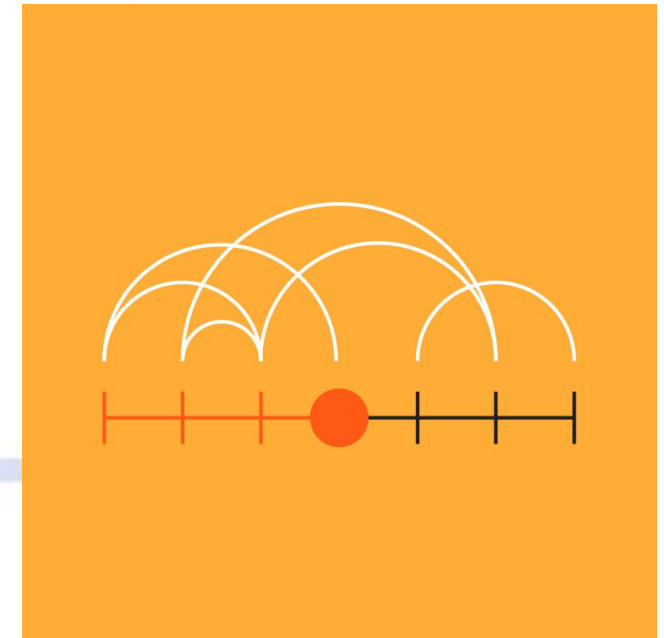
## Adjustable



## Immediate

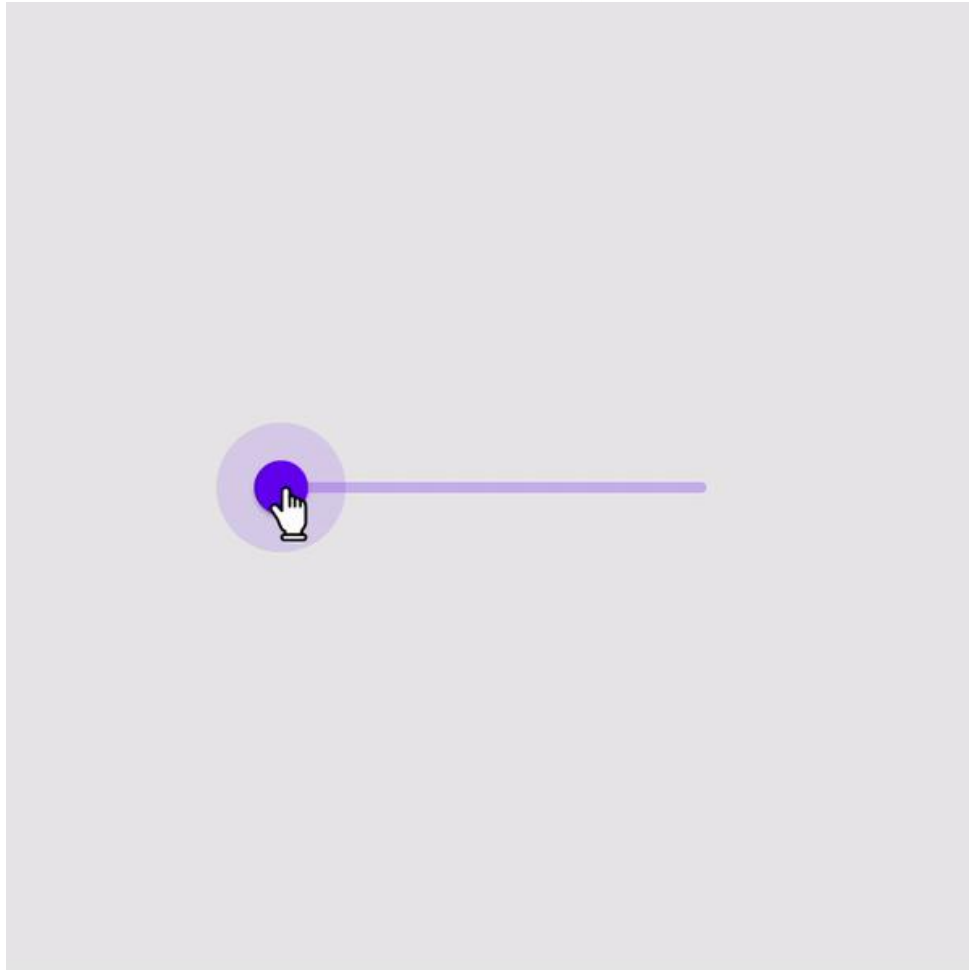


## Accessible

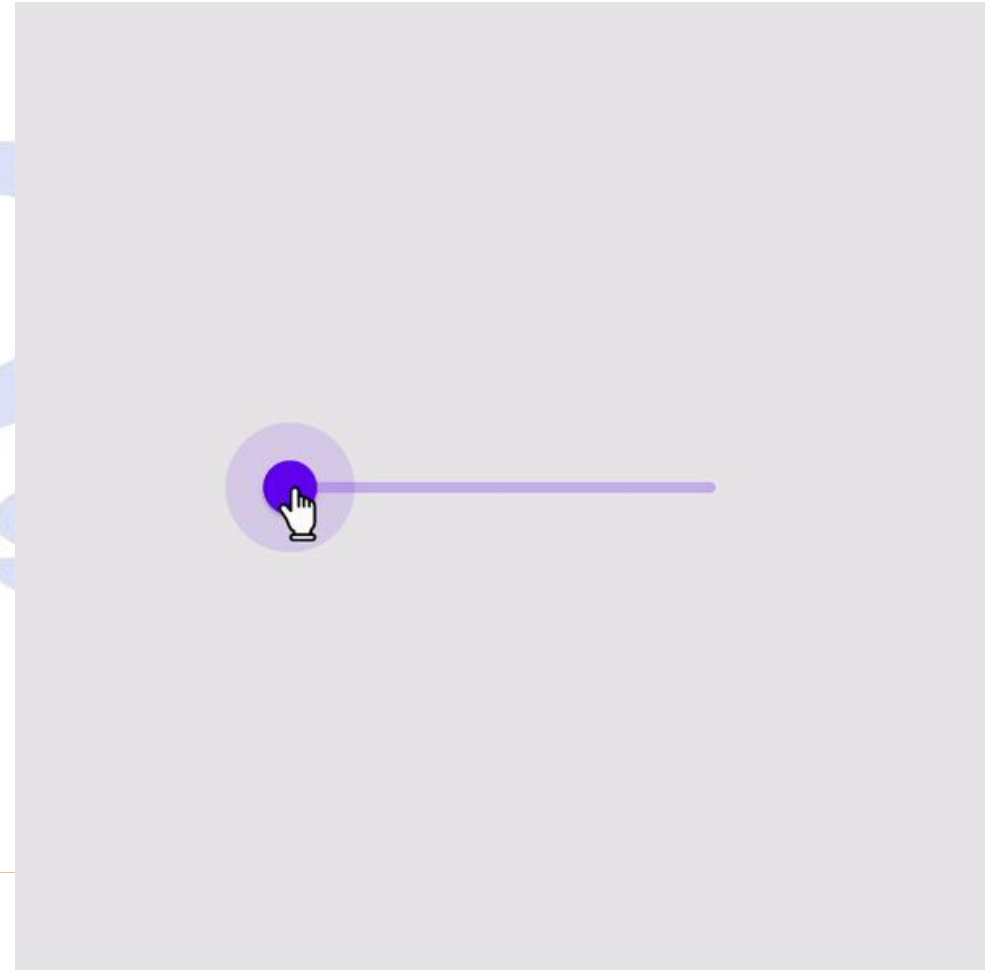


# Continuous sliders

**Click and drag**



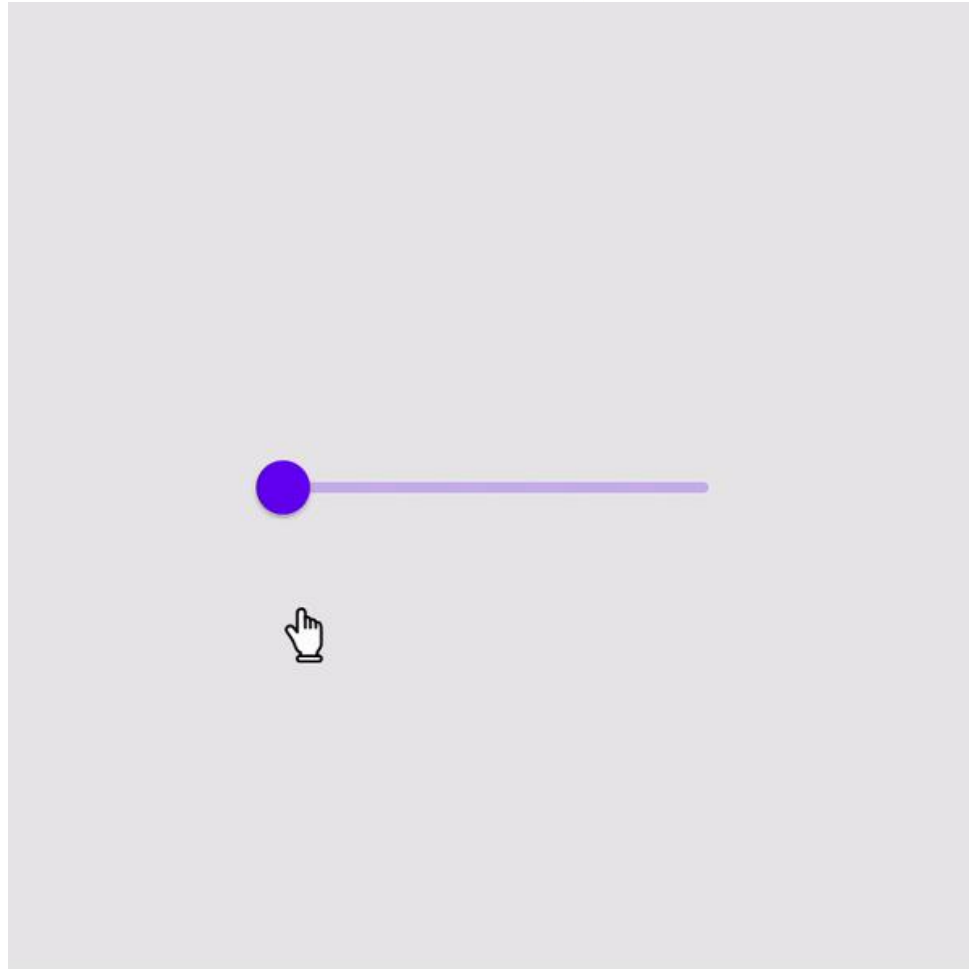
Click jump



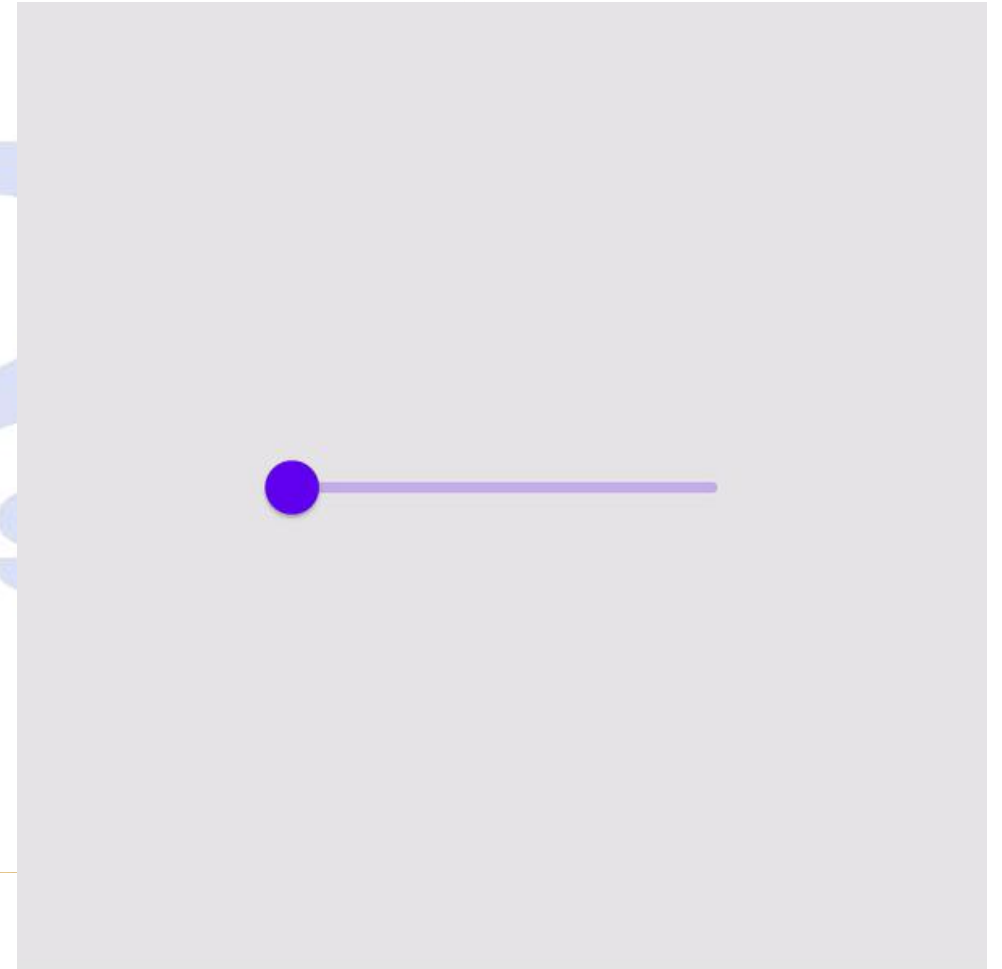


# Continuous sliders

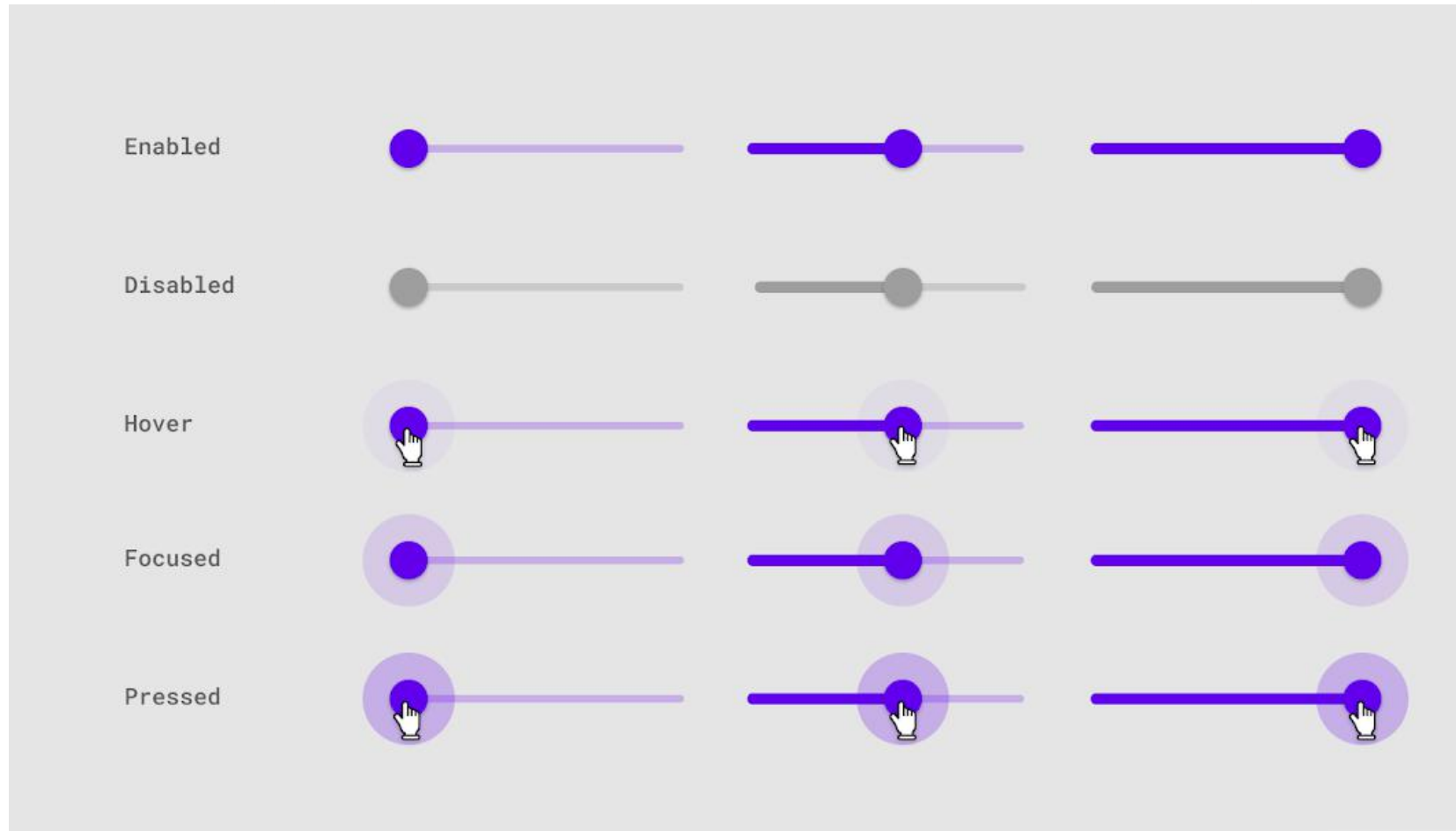
**Click and arrow**



Tab and arrow

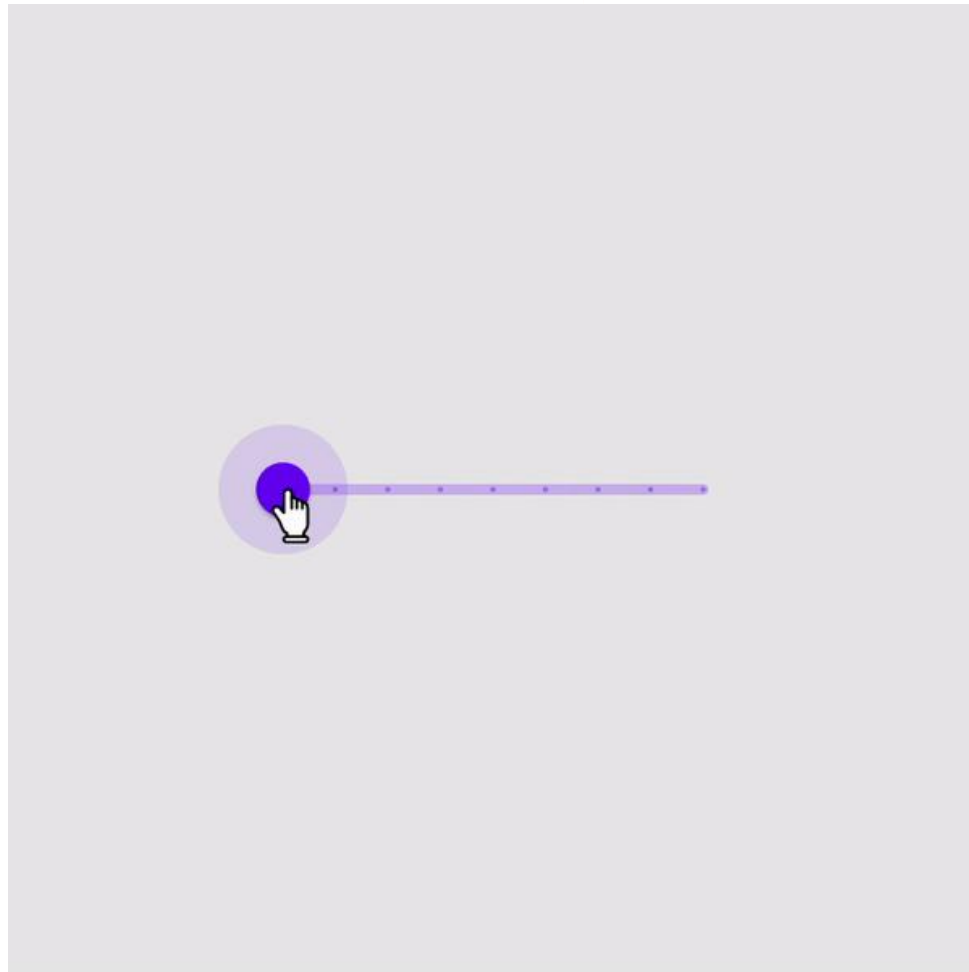


# States

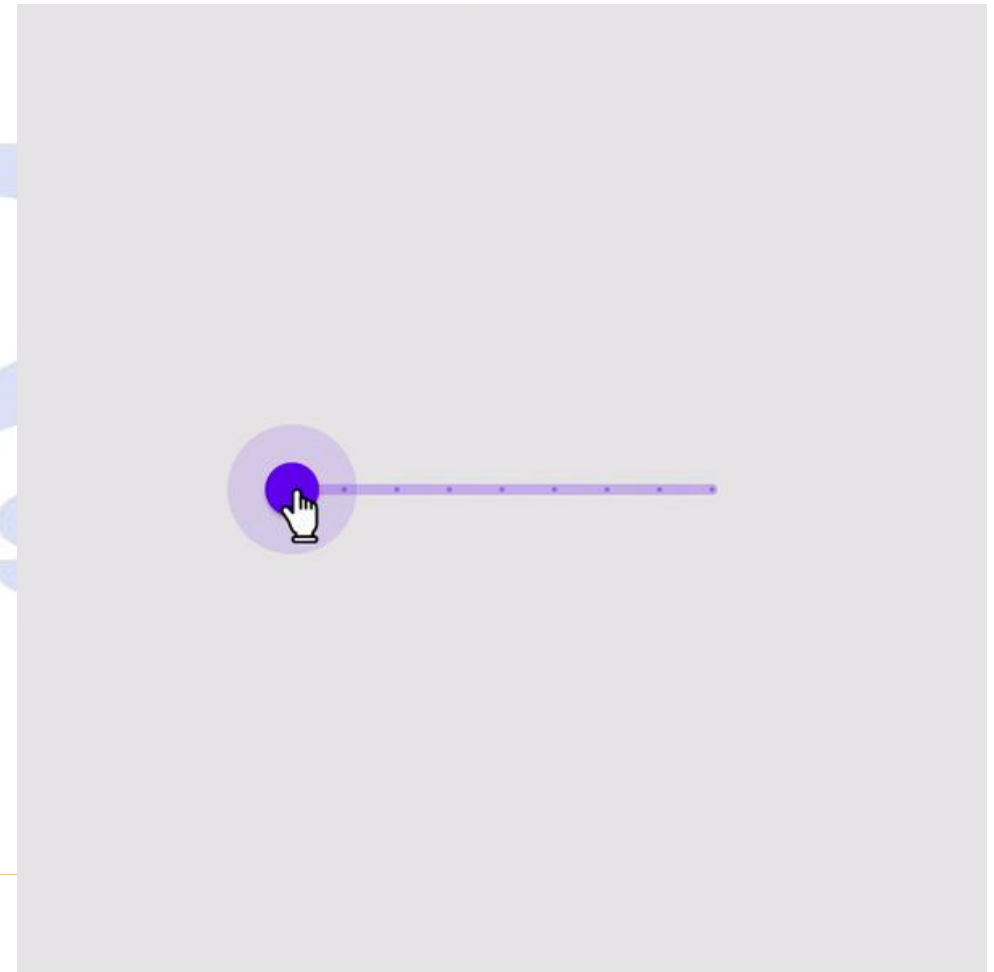


# Discrete sliders

**Click and drag**

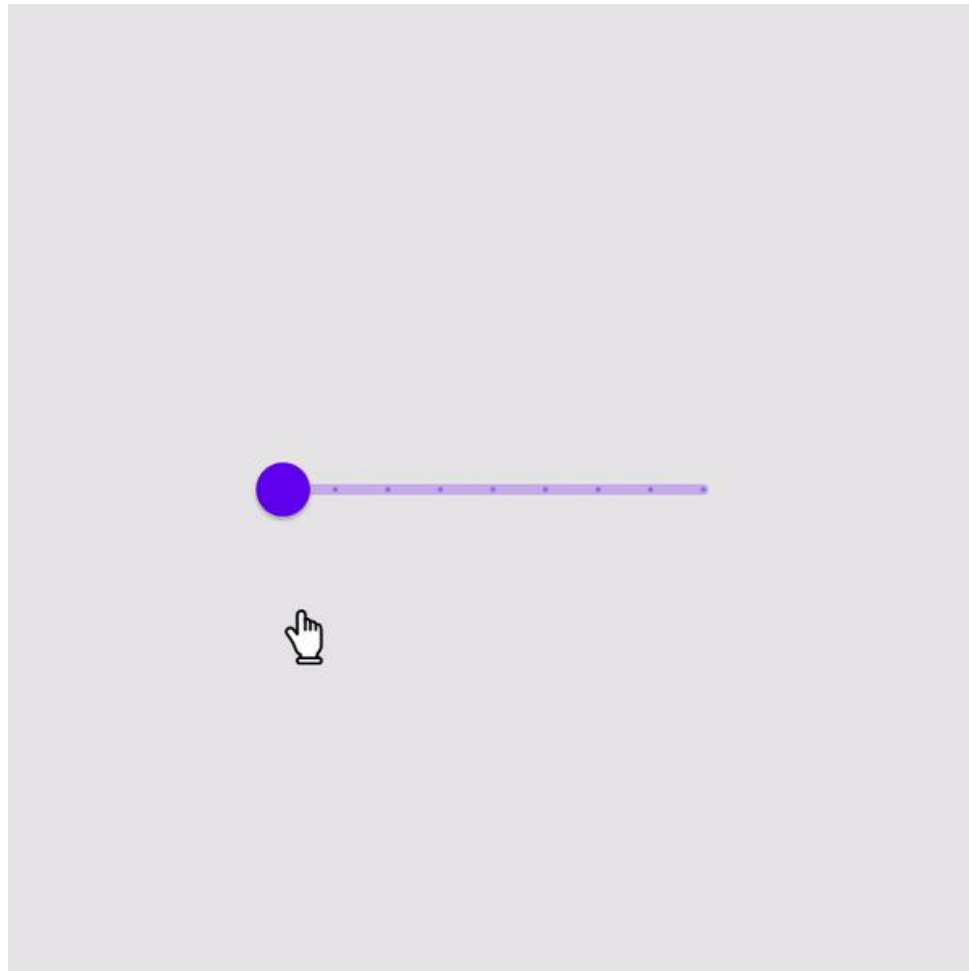


Click jump

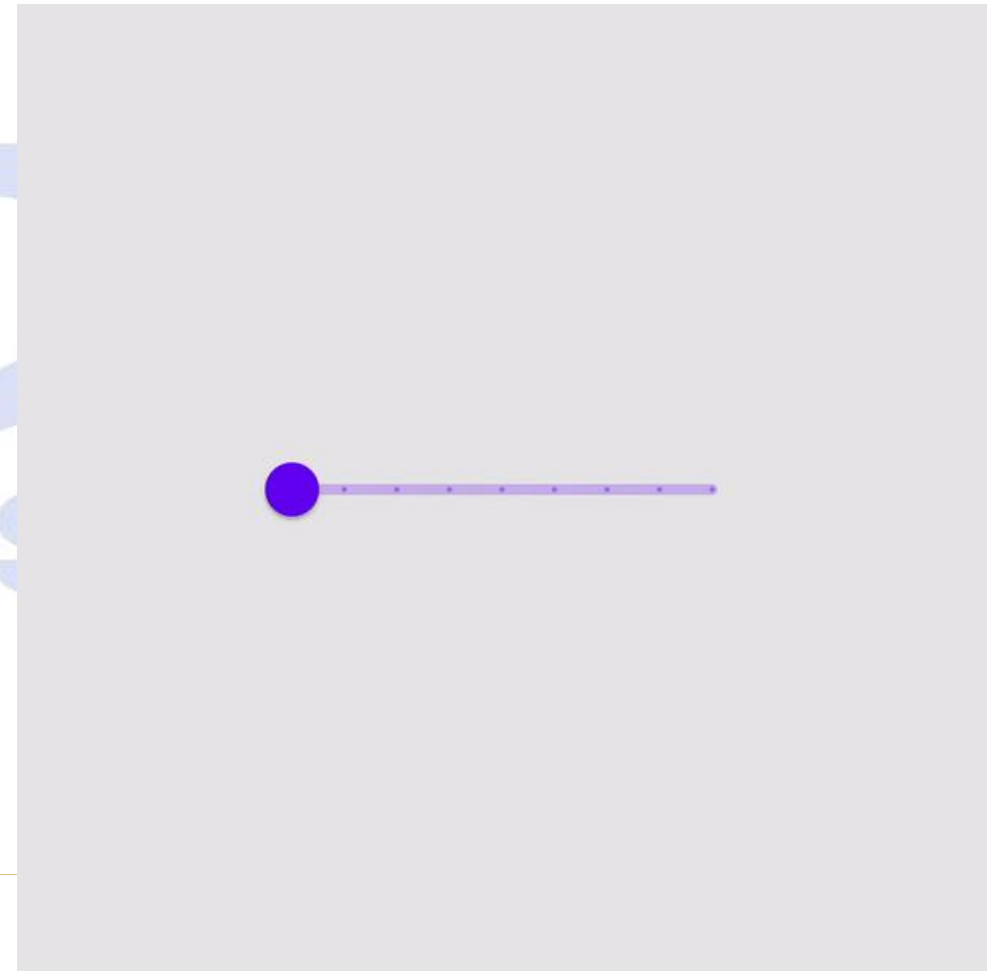


# Discrete sliders

Click and arrow

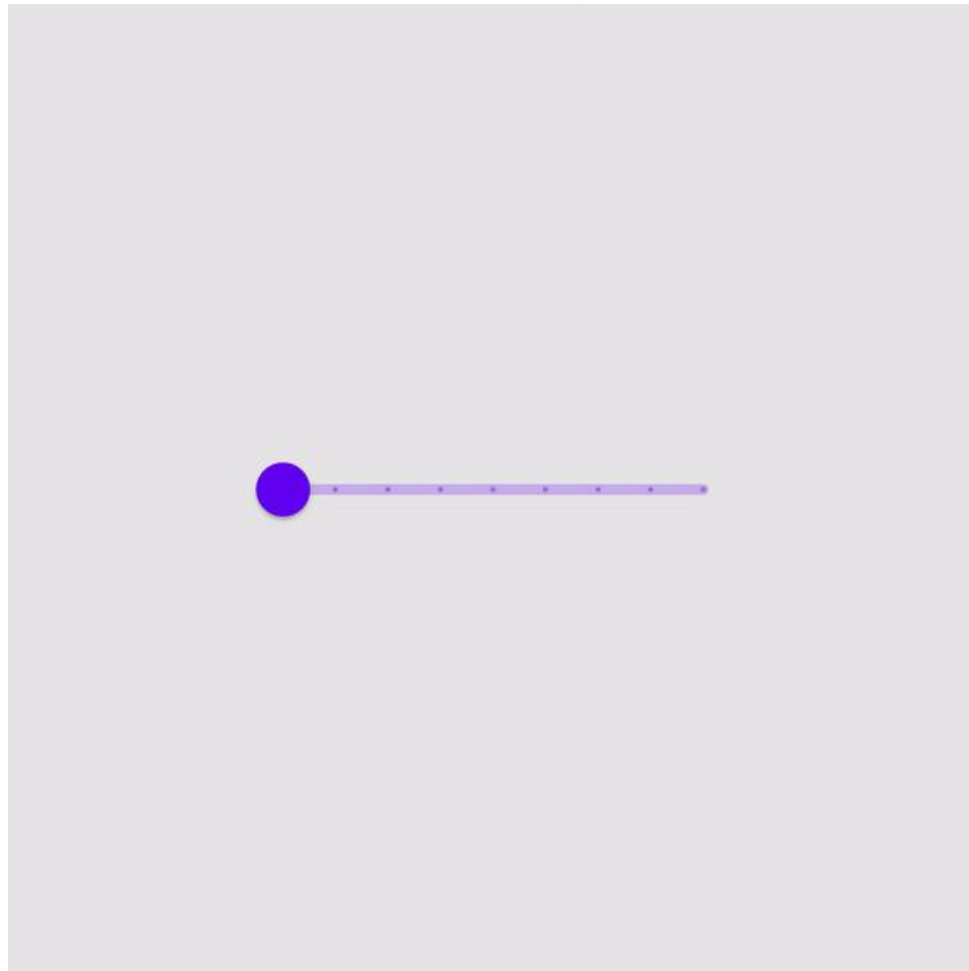


Tab and arrow



# Discrete sliders

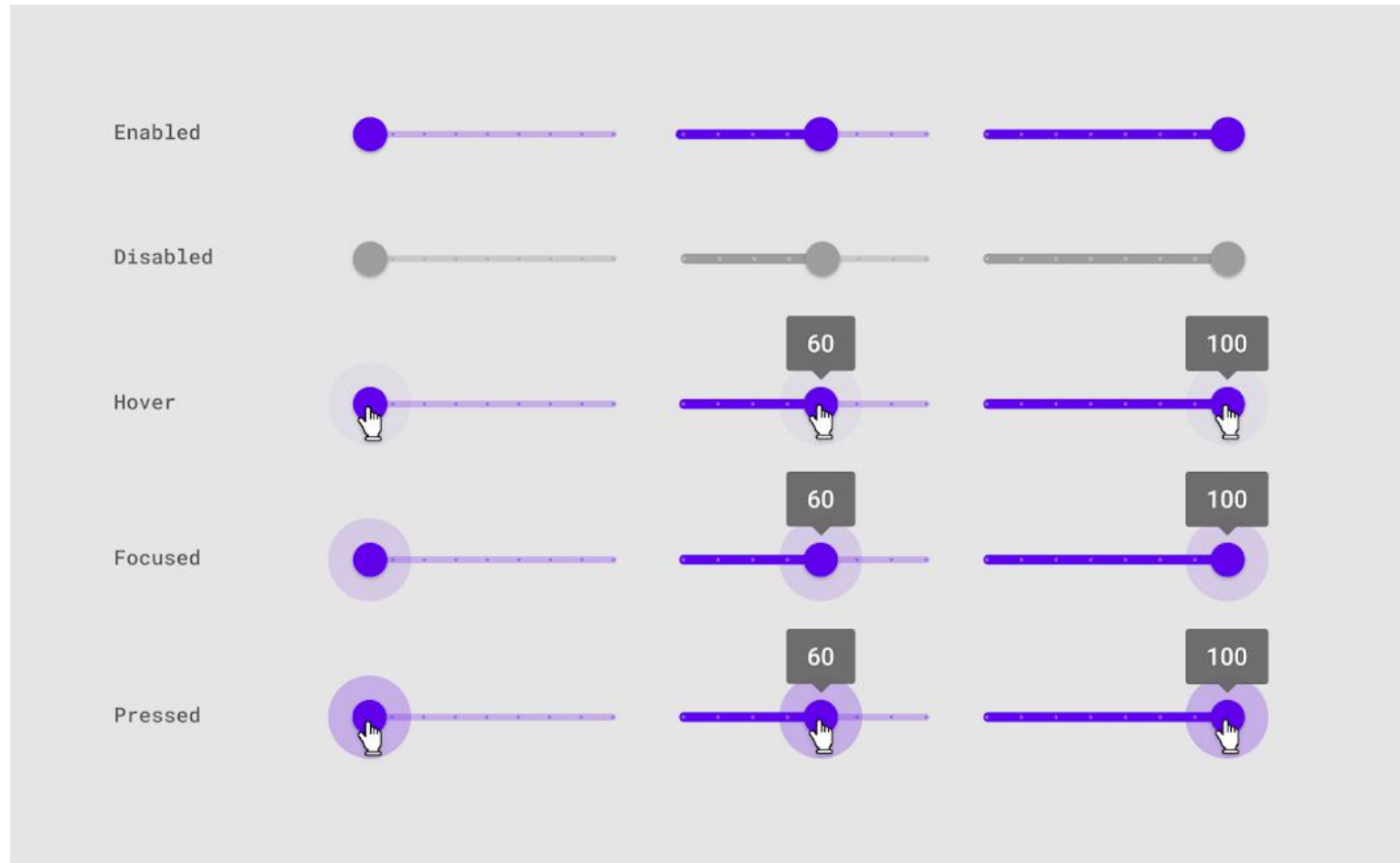
Tick marks (Optional)



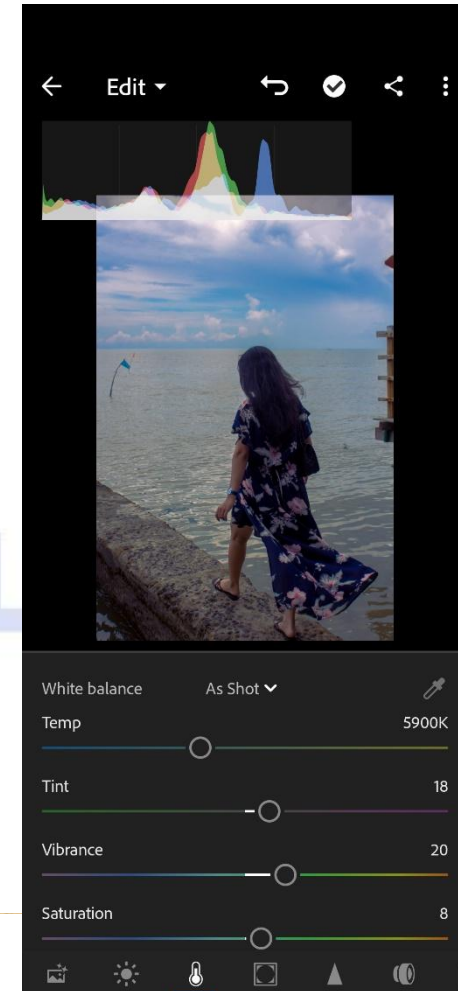
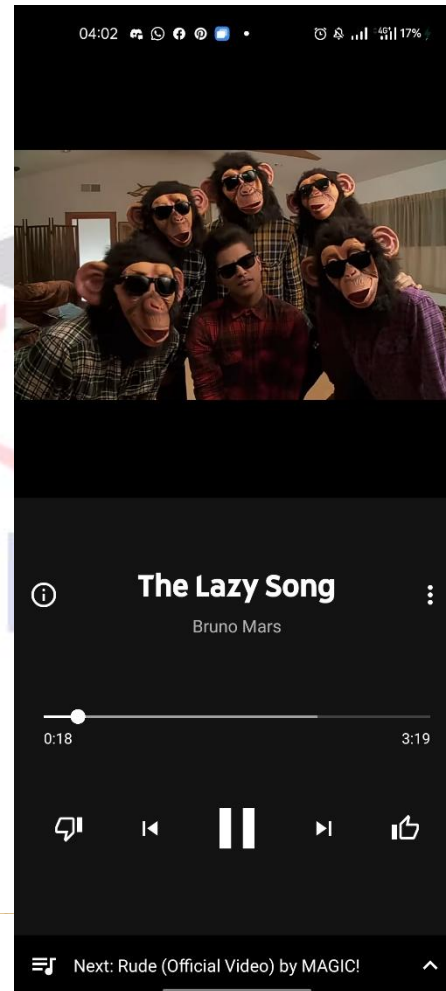
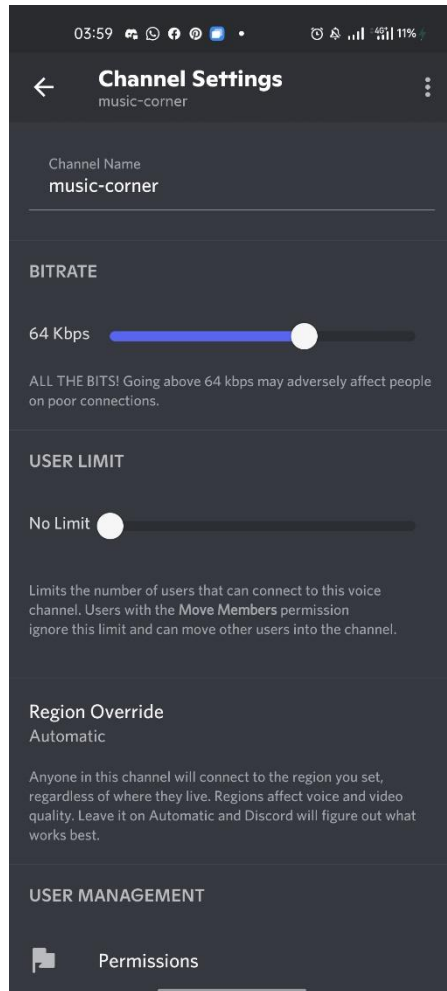
Value entry field (Optional)



# States



# Contoh Implementasi Sliders



# Penerapan di Flutter

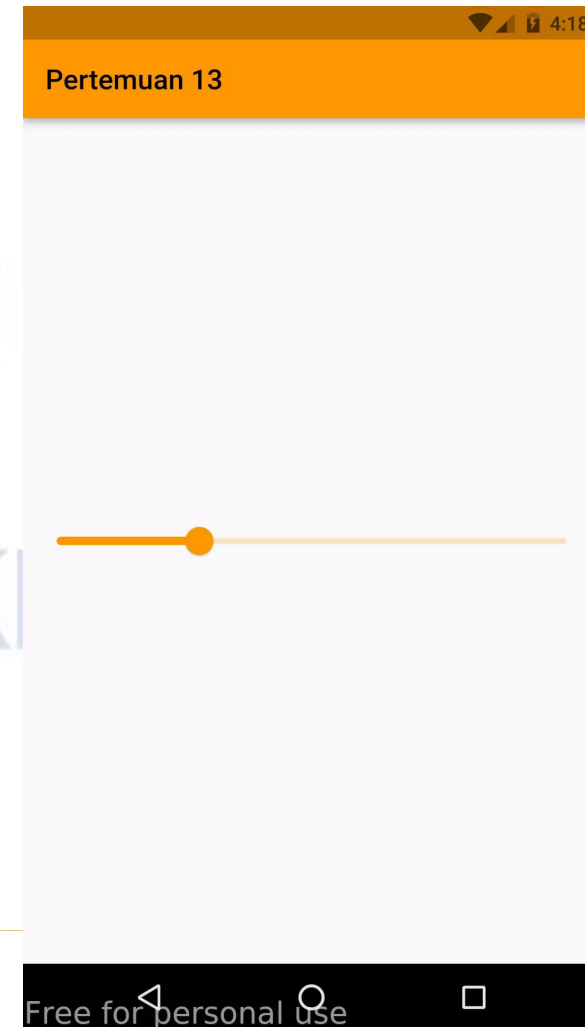




# Continuous single point slider

```
double _sliderValue = 20;

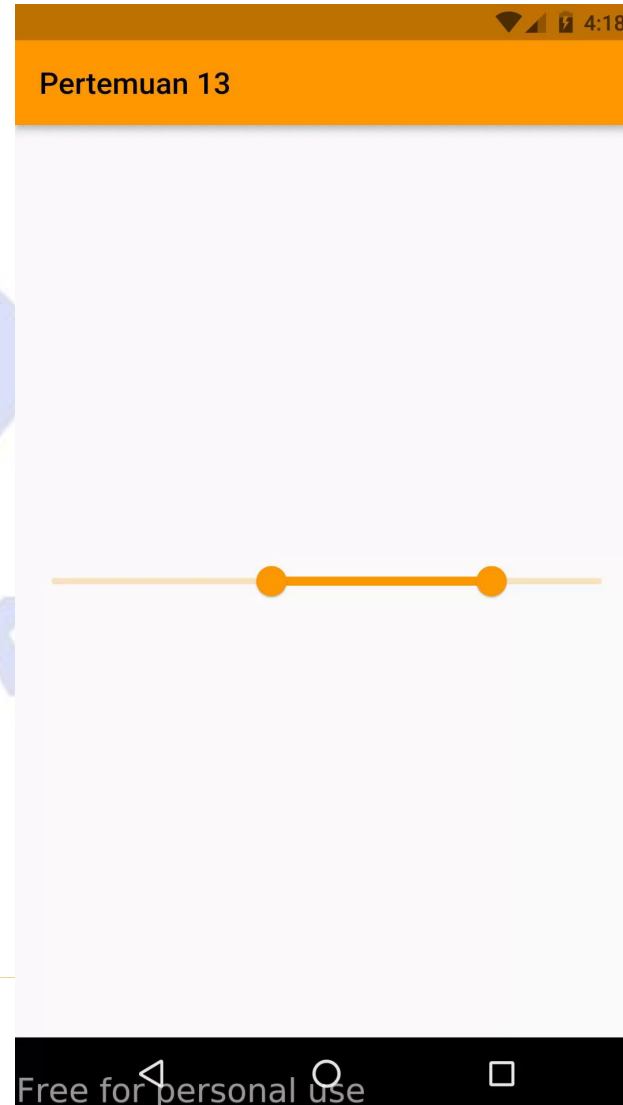
Slider(
  value: _sliderValue,
  min: 0,
  max: 100,
  label: _sliderValue.round().toString(),
  onChanged: (value) {
    setState(() {
      _sliderValue = value;
    });
  },
),
```



# Continuous range slider

```
RangeValues _rangeSliderDiscreteValues = const RangeValues(40, 80);

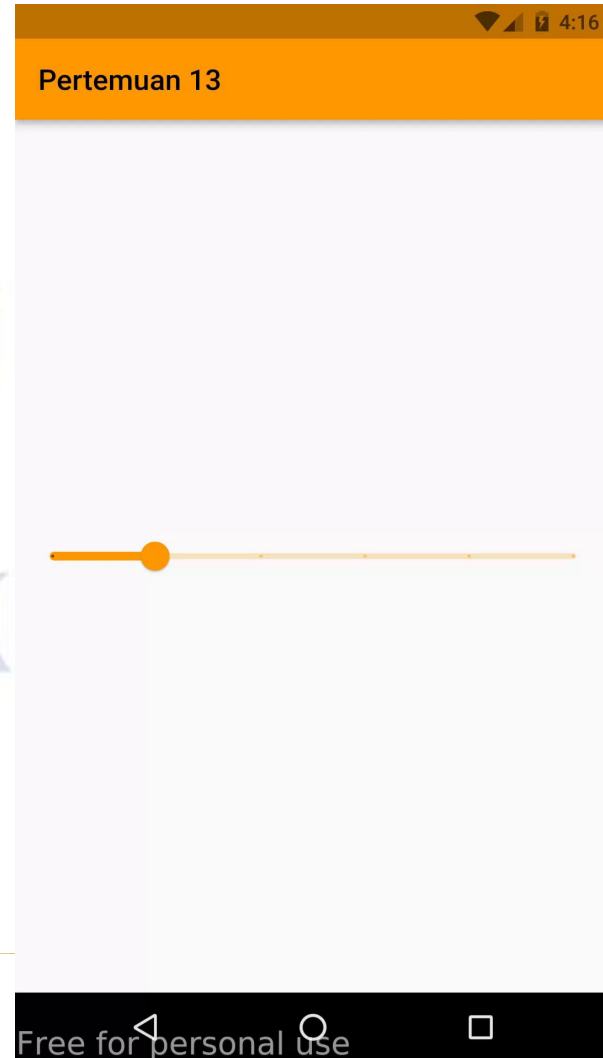
RangeSlider(
  values: _rangeSliderDiscreteValues,
  min: 0,
  max: 100,
  labels: RangeLabels(
    _rangeSliderDiscreteValues.start.round().toString(),
    _rangeSliderDiscreteValues.end.round().toString(),
  ),
  onChanged: (values) {
    setState(() {
      _rangeSliderDiscreteValues = values;
    });
  },
),
```



# Discrete single point slider

```
double _sliderDiscreteValue = 20;

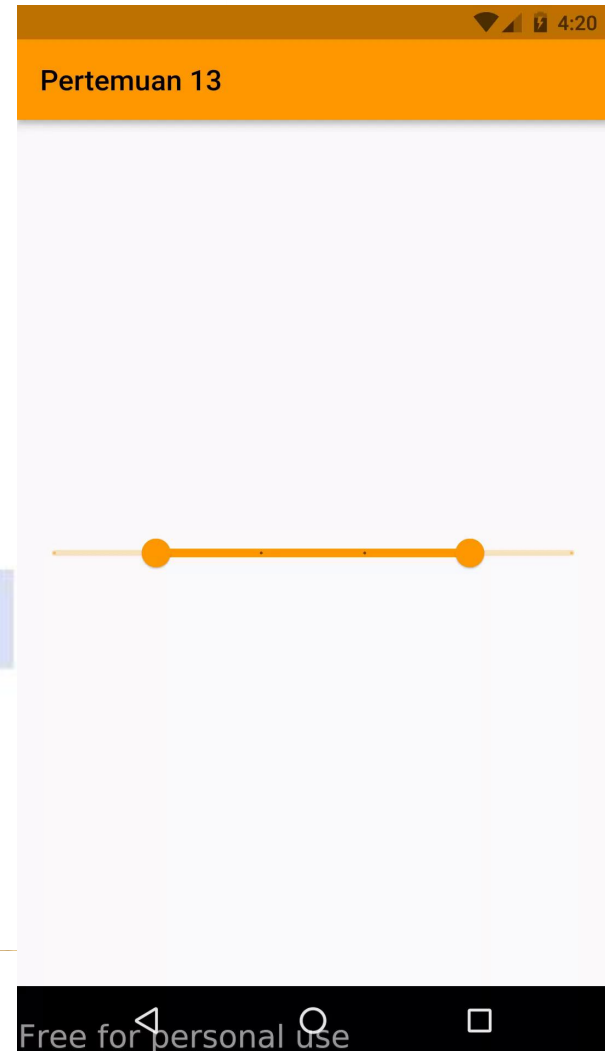
Slider(
  value: _sliderDiscreteValue,
  min: 0,
  max: 100,
  divisions: 5,
  label: _sliderDiscreteValue.round().toString(),
  onChanged: (value) {
    setState(() {
      _sliderDiscreteValue = value;
    });
  },
),
```



# Discrete range slider

```
RangeValues _rangeSliderDiscreteValues = const RangeValues(40, 80);

RangeSlider(
  values: _rangeSliderDiscreteValues,
  min: 0,
  max: 100,
  divisions: 5,
  labels: RangeLabels(
    _rangeSliderDiscreteValues.start.round().toString(),
    _rangeSliderDiscreteValues.end.round().toString(),
  ),
  onChanged: (values) {
    setState(() {
      _rangeSliderDiscreteValues = values;
    });
  },
),
```

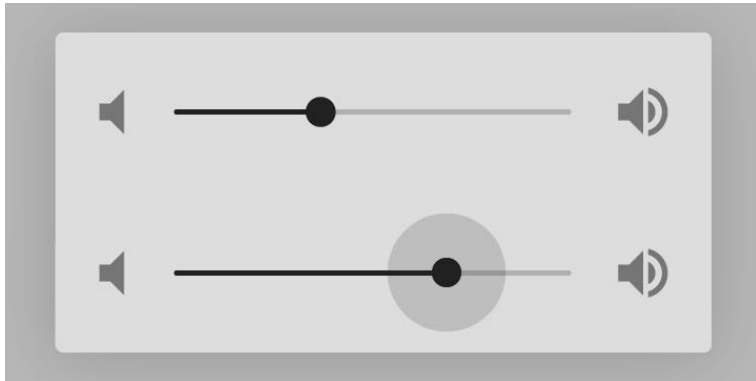


# Property di Sliders

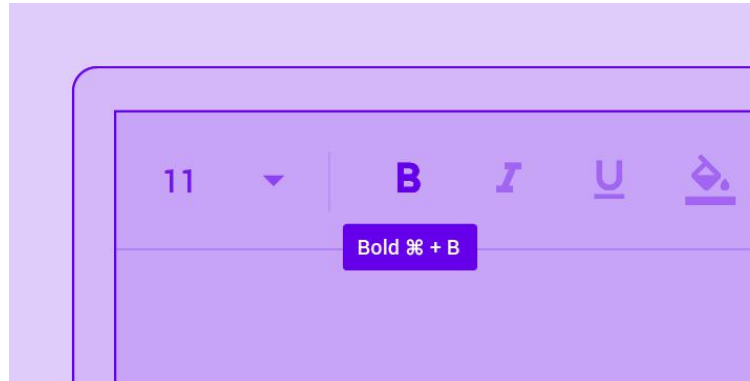
Property	Class/DataType	Deskripsi
value	double	Inisialisasi nilai awal dan hasil perubahan
min	Color?	Set value minimal
max	double	Set value maksimal
label	Color?	Set Warna shadow
divisions	int?	Membagi nilai max secara diskrit
activeColor	Color?	Warna saat sliders aktif
inactiveColor	Color?	Warna saat sliders tidak aktif
onChange	ValueChanged<double >?	Dipanggil saat ada perubahan value
onChangeStart	ValueChanged<double >?	Dipanggil saat user mulai melakukan perubahan
onChangeEnd	ValueChanged<double >?	Dipanggil saat user selesai melakukan perubahan

Property lain dapat dilihat [disini](#) dan [disini](#)

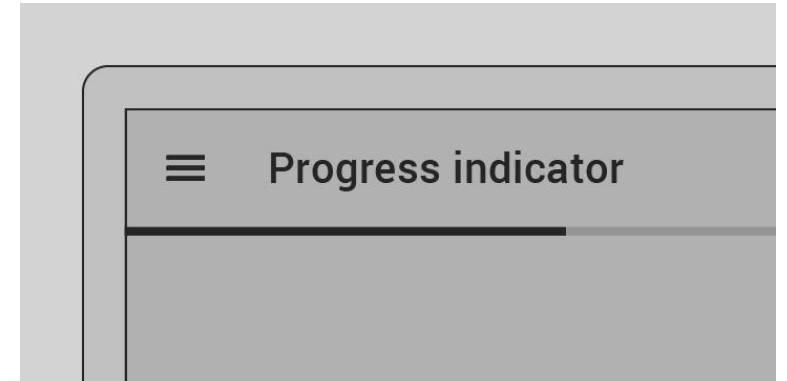
# Komponen dalam Mendesain Aplikasi Part. 6



Sliders



ToolTips



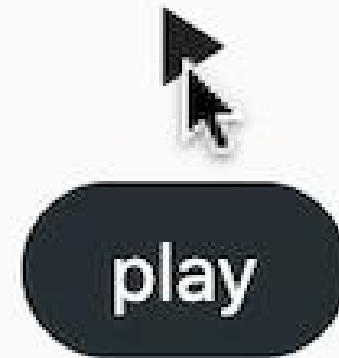
Progress Indicator

# ToolTips



# ToolTips adalah...

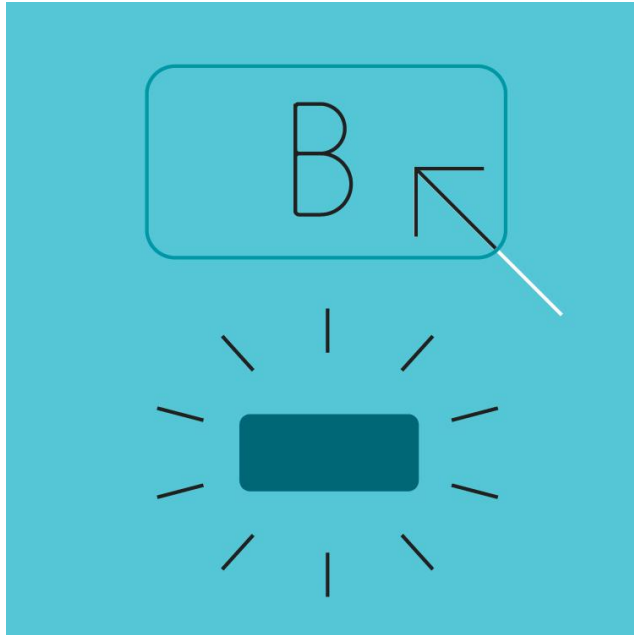
- Widget yang menampilkan teks informatif saat pengguna **mengarahkan** kursor, **fokus**, atau **tap** elemen
- Saat **diaktifkan**, tooltips menampilkan **label teks** yang **mengidentifikasi** elemen, seperti **deskripsi** fungsinya



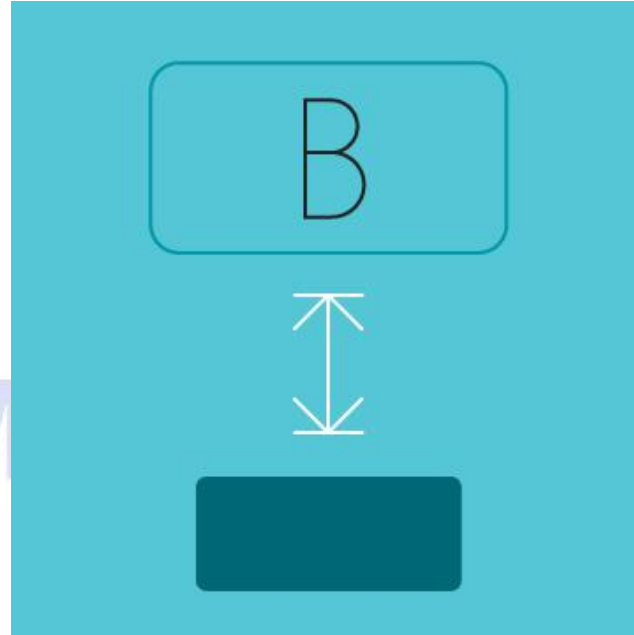


# Principle of ToolTips

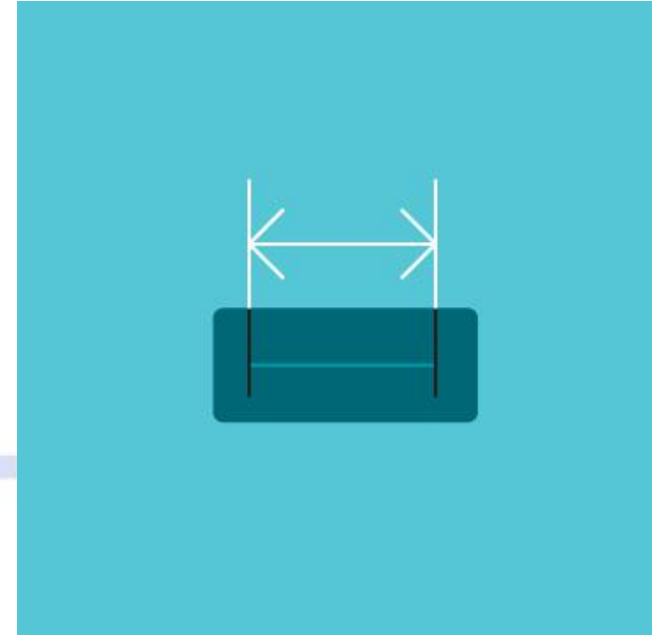
**Transient**



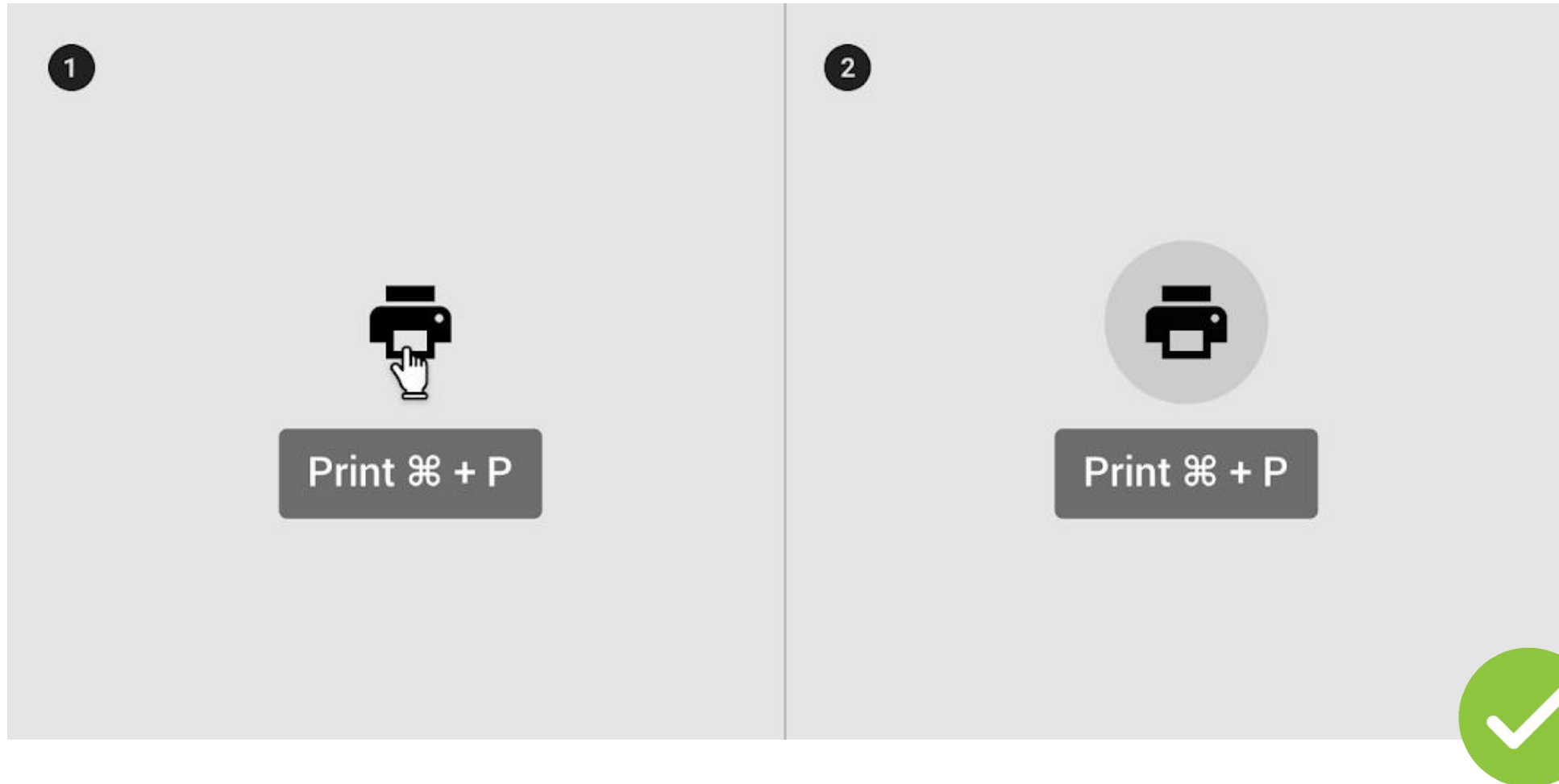
**Transient**



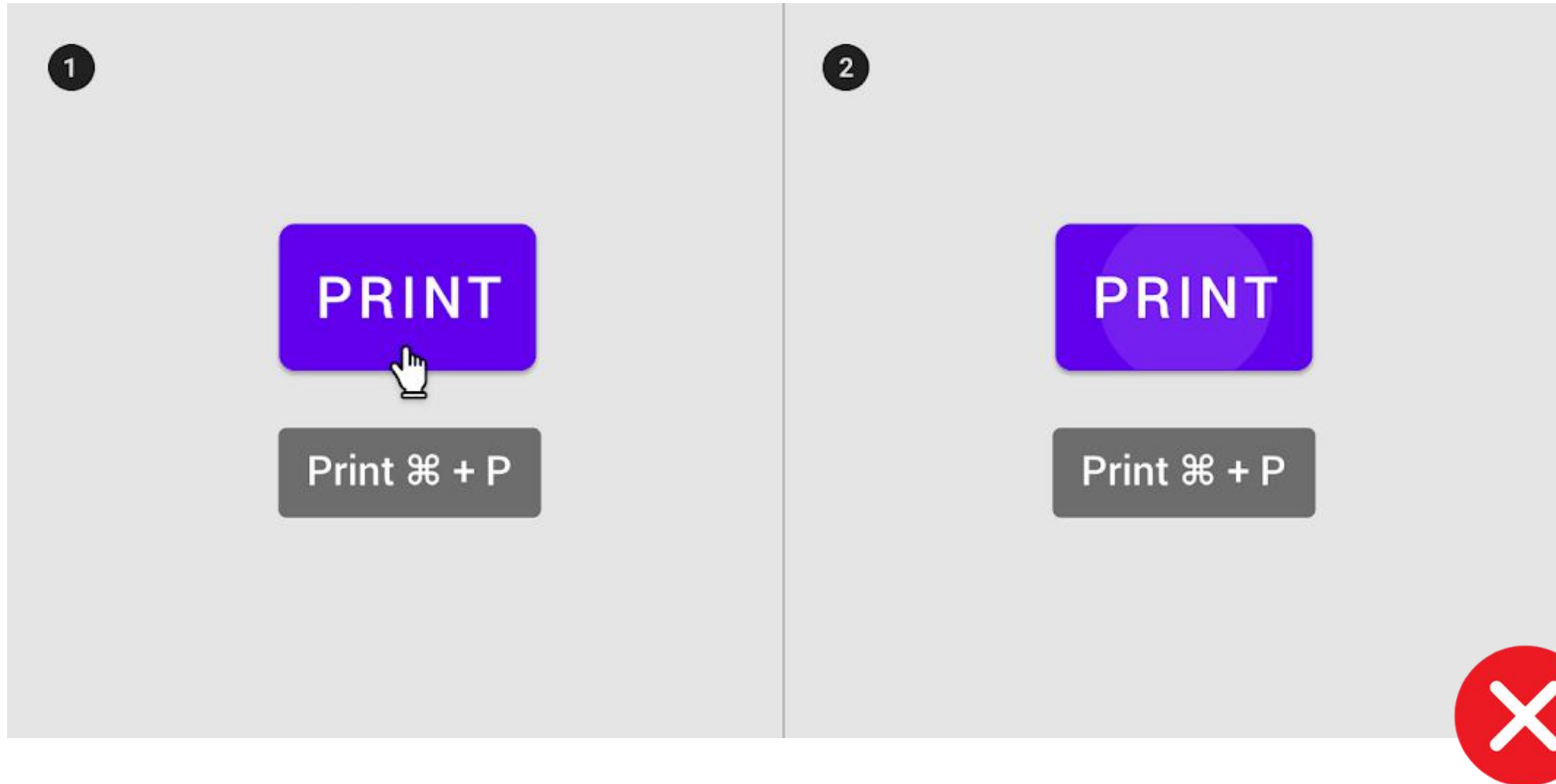
**Succinct**



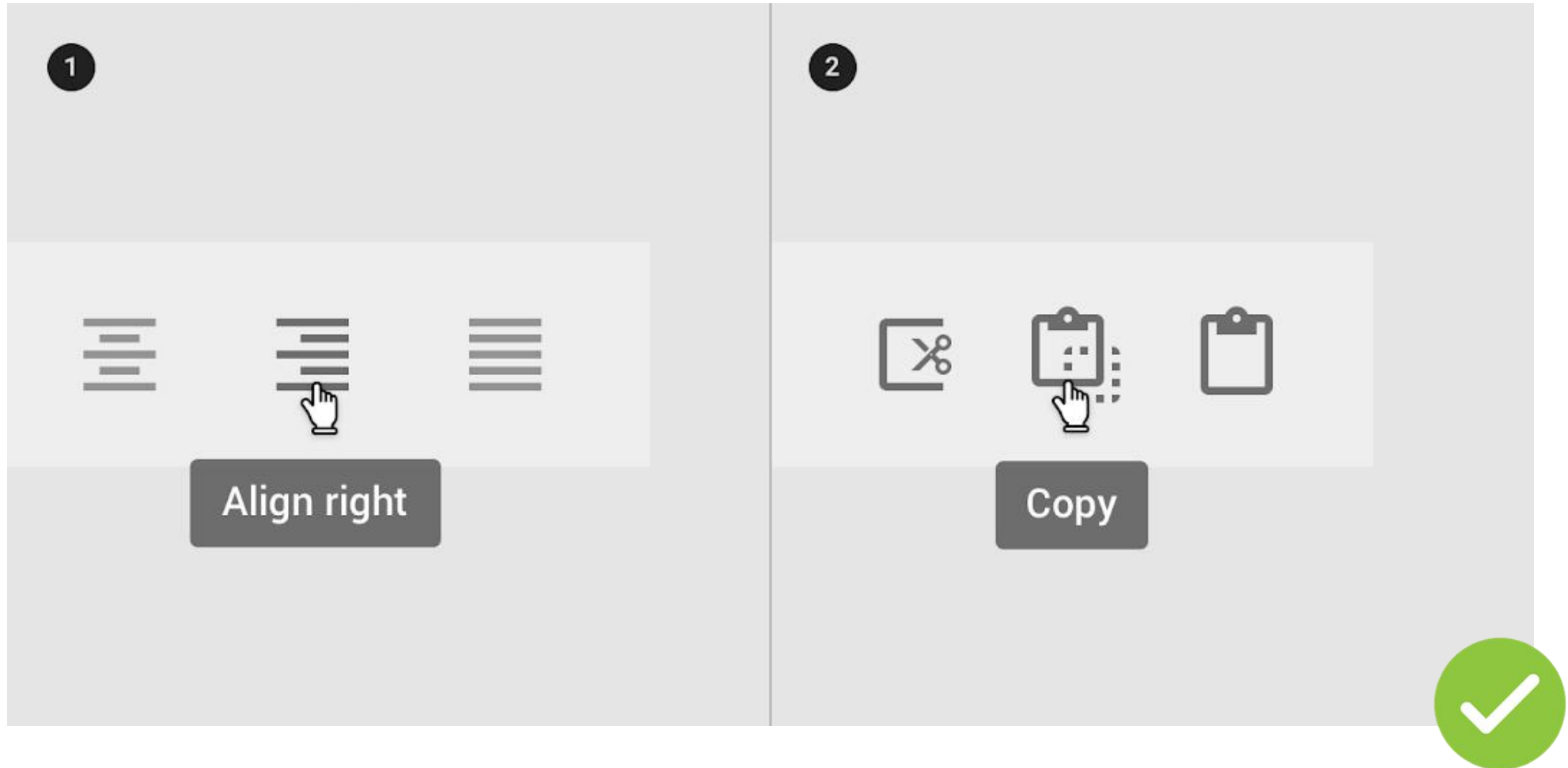
# Placement of Tooltips



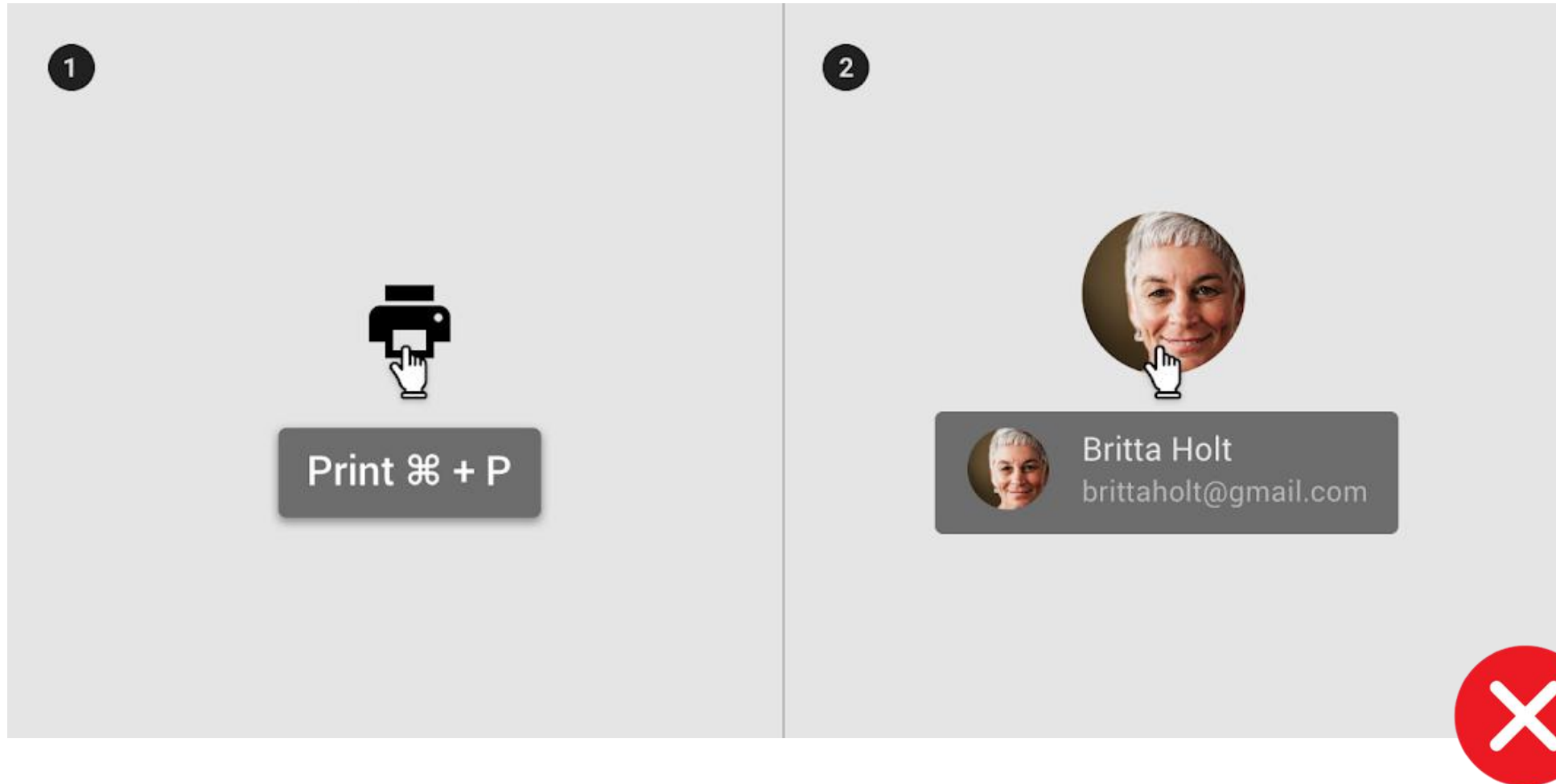
# Placement of Tooltips



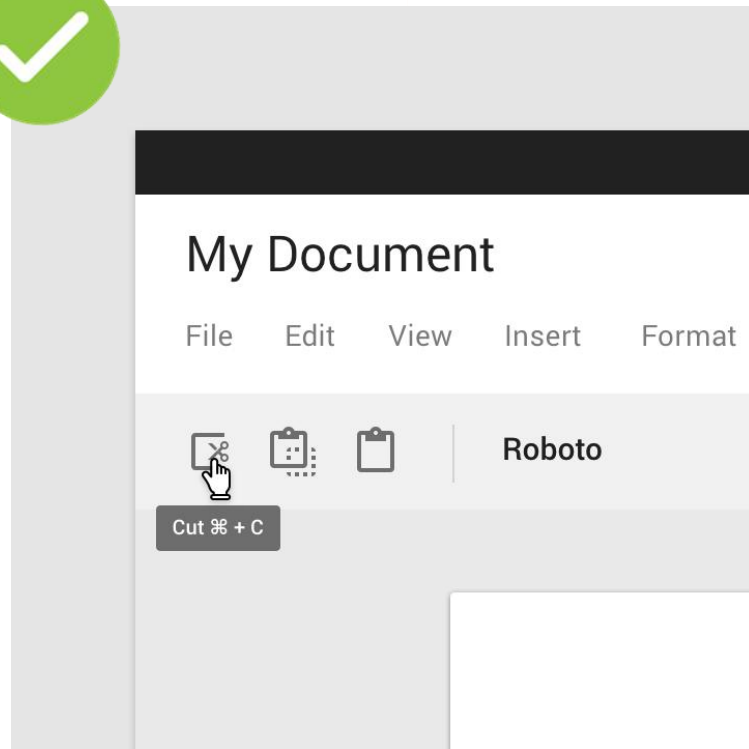
# Placement of Tooltips



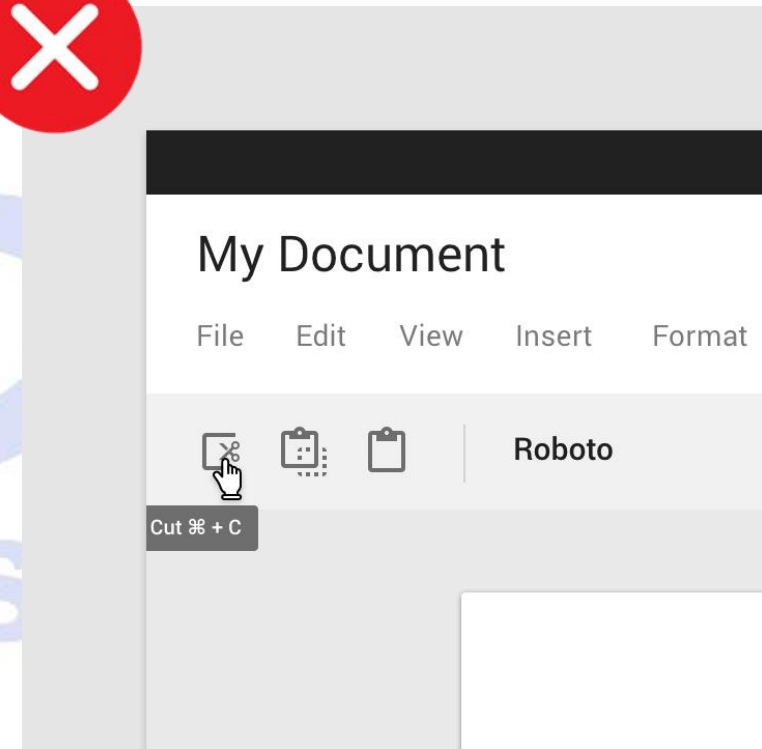
# Placement of Tooltips



# Placement of Tooltips

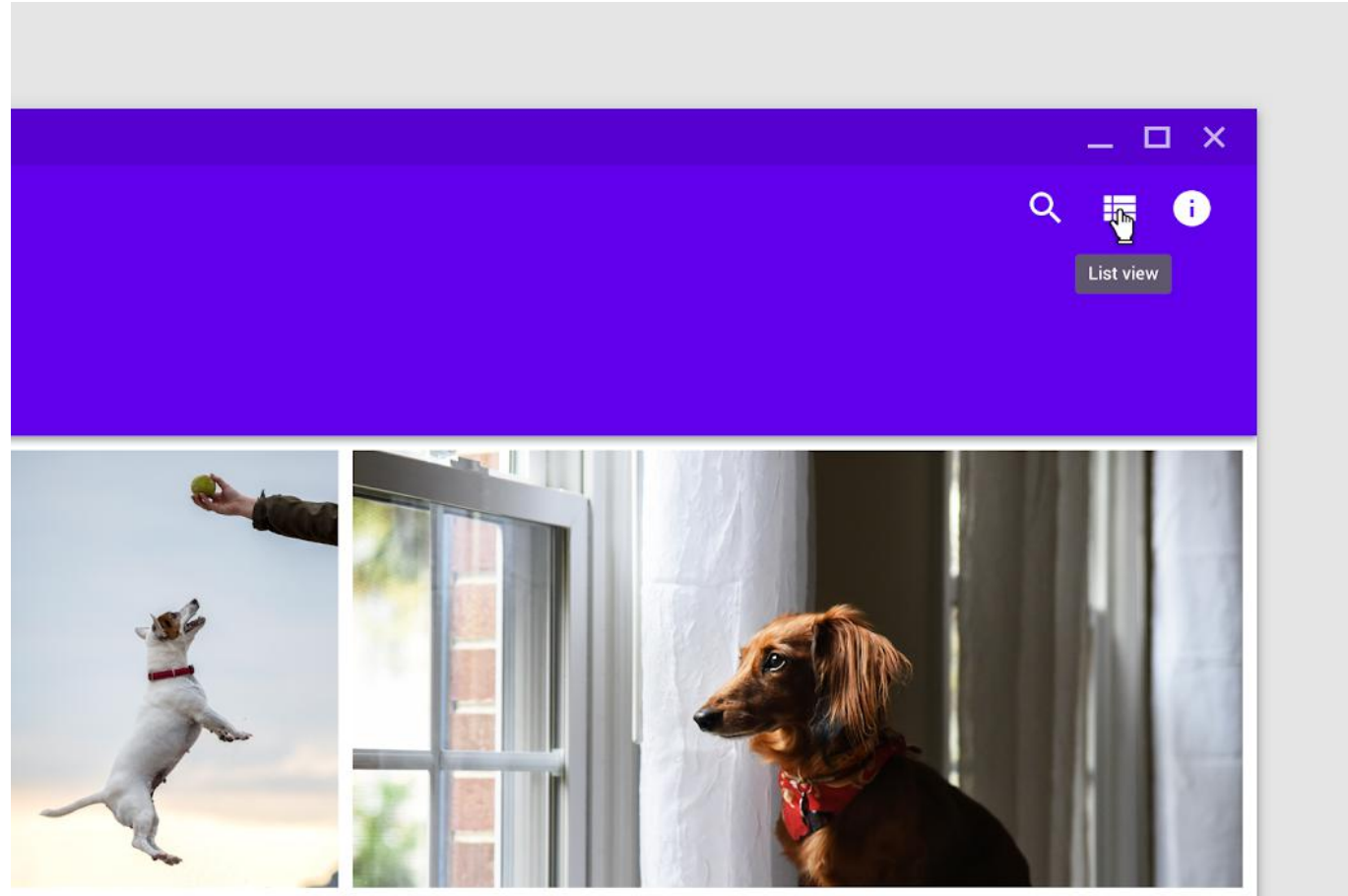


Jaga agar posisi tooltip tetap terlihat

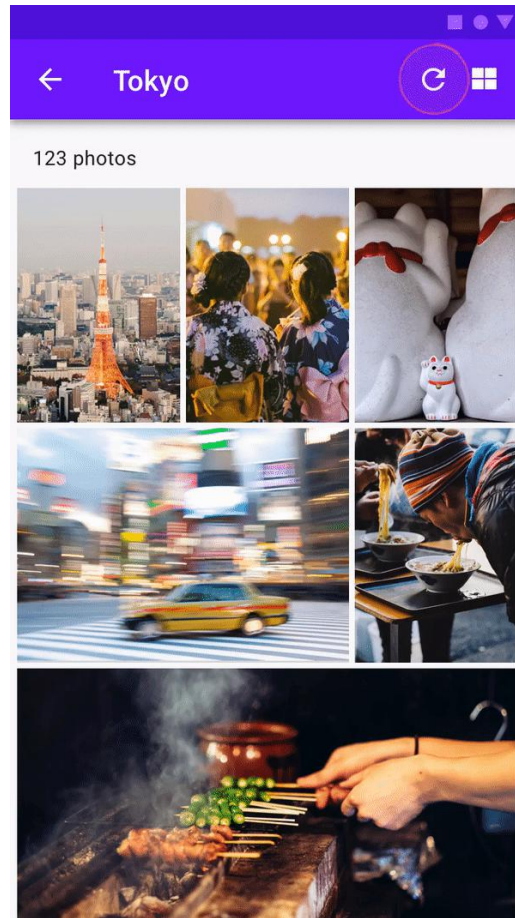


Jangan memotong tooltips

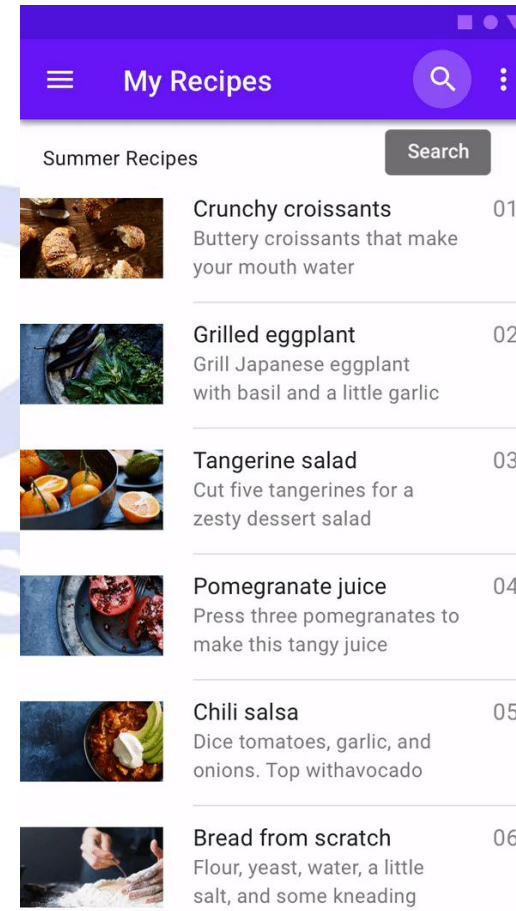
# Placement of Tooltips (Desktop)



# Placement of Tooltips (Mobile)



Tooltip ditampilkan melalui tekan lama



Tooltip dipanggil melalui fokus  
(**khusus Android**)



# Behavior

1. Tooltip ditampilkan saat :
  - Mengetuk dan menahan elemen atau komponen layar (di ponsel)
  - Mengarahkan kursor ke atasnya (desktop)
2. Terus tampilkan tooltip selama pengguna menekan lama atau mengarahkan kursor ke elemen

MIKROSKIL

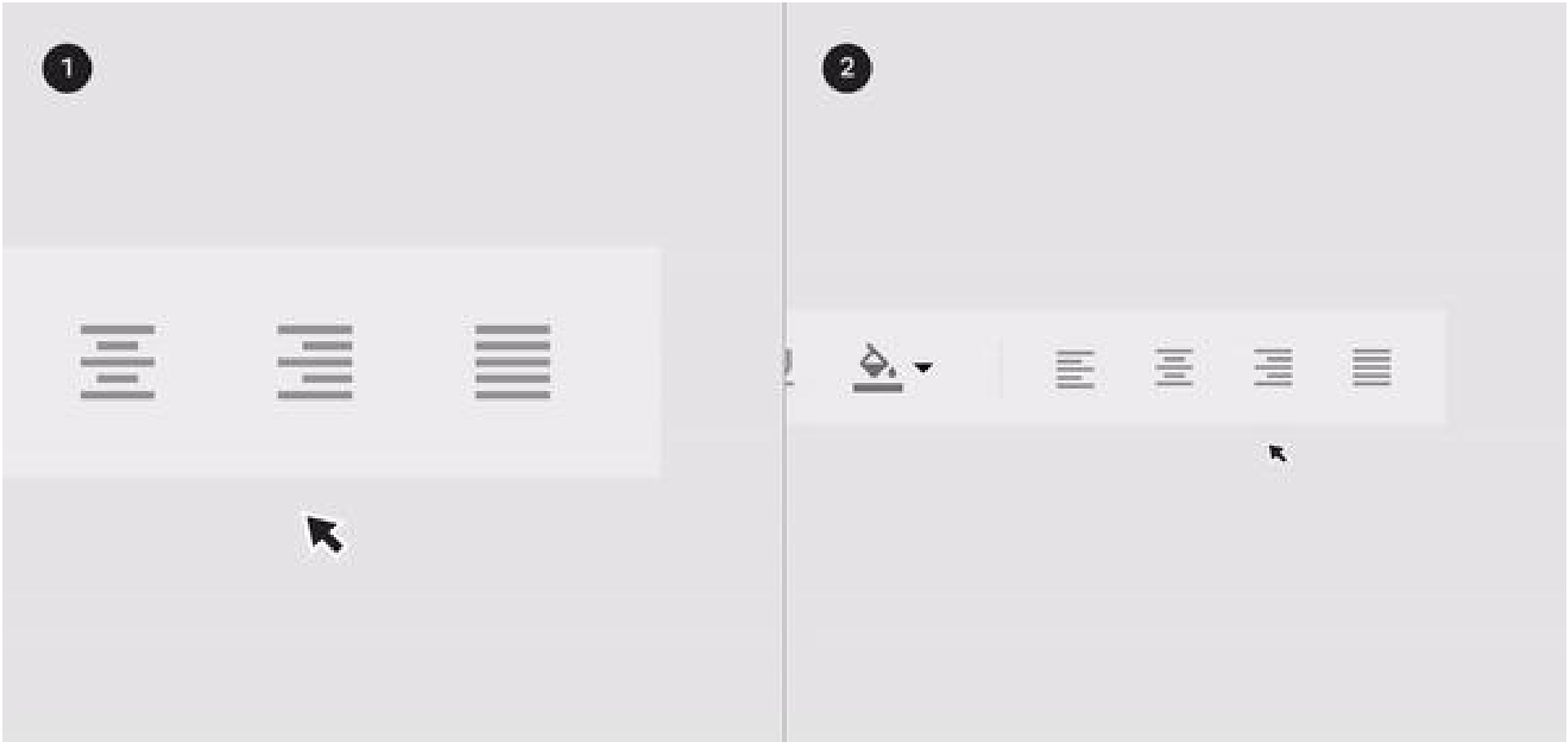
# Behavior (Timing)

- Tampilkan **tooltip** selama **1,5 detik**.
- Jika pengguna **melakukan tindakan lain** sebelum waktu tersebut berakhir, **tooltip akan hilang**.



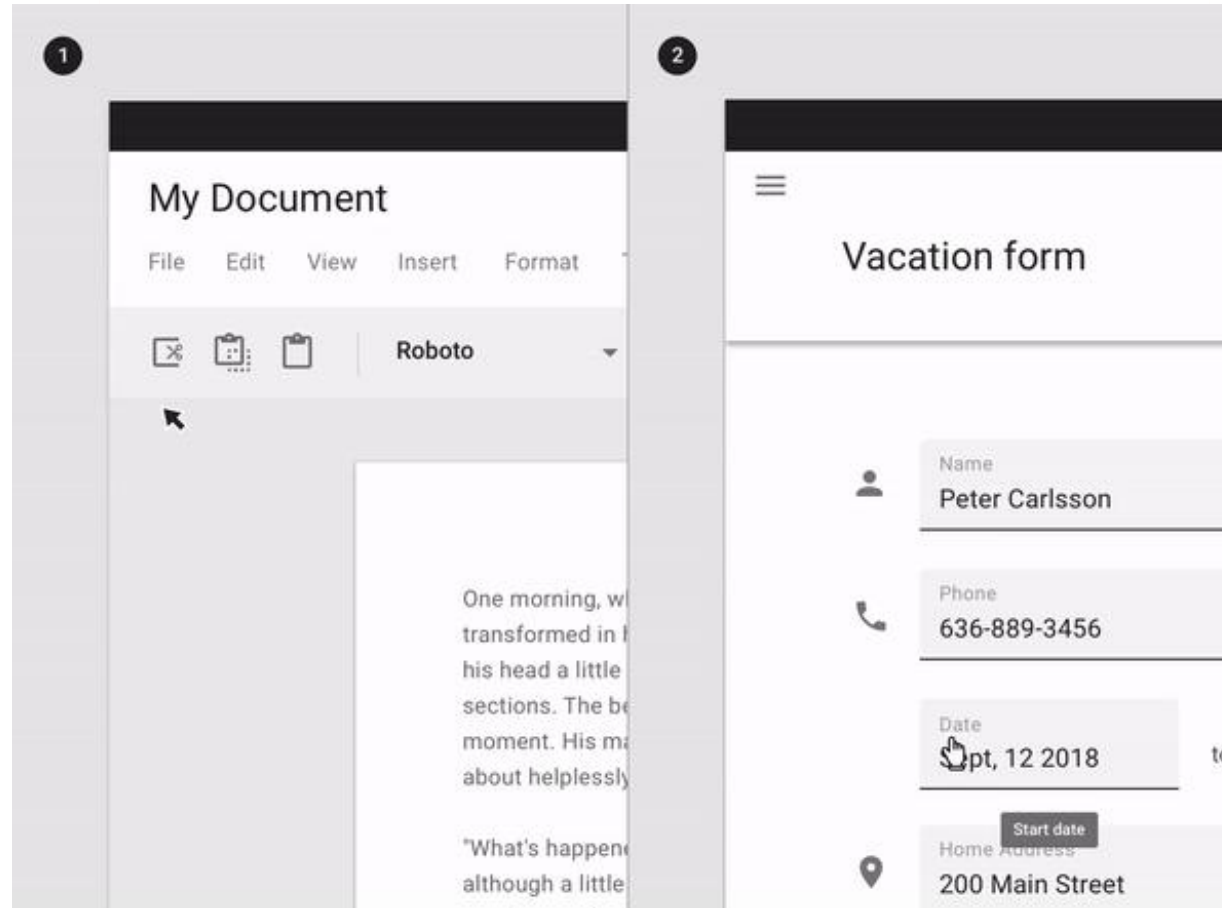
MIKROSKIL

# Behavior (Transition)



Tooltips menggunakan pola **transisi fade** untuk masuk dan keluar dari layar

# Behavior (Transition)



1. Tooltip **menghilang** saat kursor bergerak **keluar** dari target sentuh
2. Tooltip **tetap ada** saat kursor bergerak **di dalam** target

# Penerapan di Flutter



# Simple Tooltip

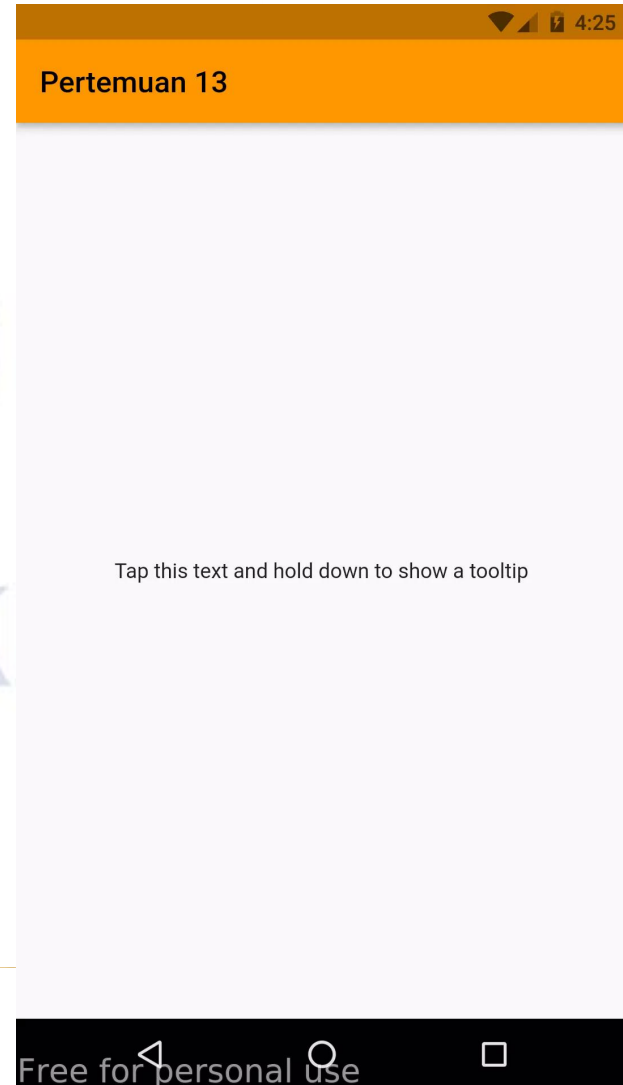


```
Center(  
  child: Tooltip(  
    message: 'play',  
    child: Icon(Icons.play_arrow),  
  ),  
),
```



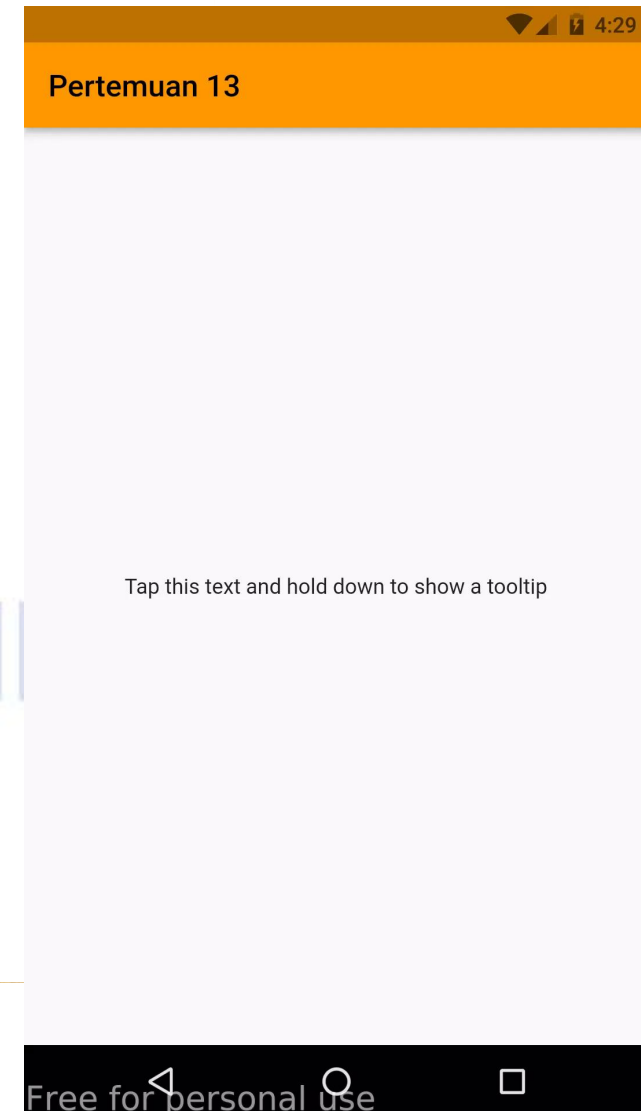
# Tooltip using by tap and hold

```
Tooltip(  
  richMessage: TextSpan(  
    text: 'I am a rich tooltip. ',  
    style: TextStyle(color: Colors.red),  
    children: <InlineSpan>[  
      TextSpan(  
        text: 'I am another span of this rich tooltip',  
        style: TextStyle(fontWeight: FontWeight.bold),  
      ),  
    ],  
  ),  
  child: Text('Tap this text and hold down to show a tooltip.'),  
);
```



# Tooltip with custom style

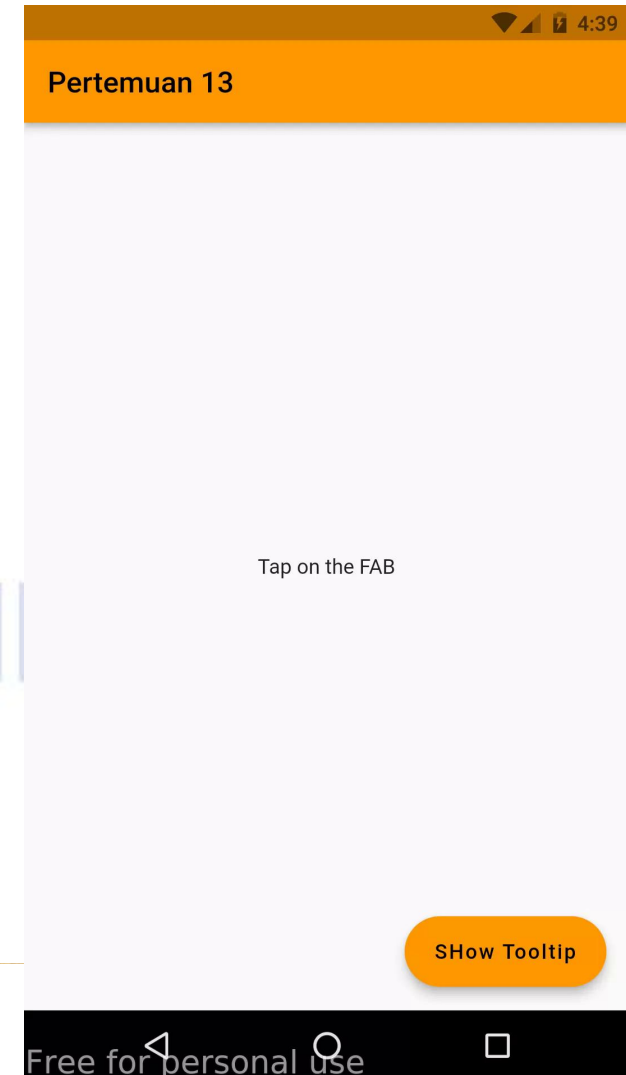
```
Tooltip(  
  message: 'I am a Tooltip',  
  decoration: BoxDecoration(  
    borderRadius: BorderRadius.circular(25),  
    gradient: const LinearGradient(  
      colors: <Color>[  
        Colors.amber, Colors.red,  
      ],  
    ),  
  ),  
  height: 50,  
  padding: const EdgeInsets.all(8.0),  
  preferBelow: false,  
  textStyle: const TextStyle(  
    fontSize: 24,  
  ),  
  showDuration: const Duration(seconds: 2),  
  waitDuration: const Duration(seconds: 1),  
  child: const Text('Tap this text and hold down to show a tooltip.'),  
);
```





# Tooltip using by manual tap

```
Scaffold(  
  appBar: AppBar(title: Text(title)),  
  body: Center(  
    child: Tooltip(  
      key: tooltipkey,  
      triggerMode: TooltipTriggerMode.manual,  
      showDuration: const Duration(seconds: 1),  
      message: 'I am a Tooltip',  
      child: const Text('Tap on the FAB'),  
    ),  
  ),  
  floatingActionButton: FloatingActionButton.extended(  
    onPressed: () {  
      tooltipkey.currentState?.ensureTooltipVisible();  
    },  
    label: const Text('Show Tooltip'),  
  ),  
);
```

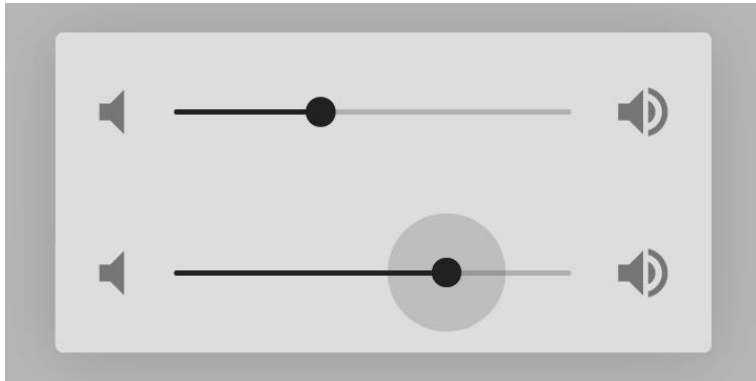


# Property di Tooltip

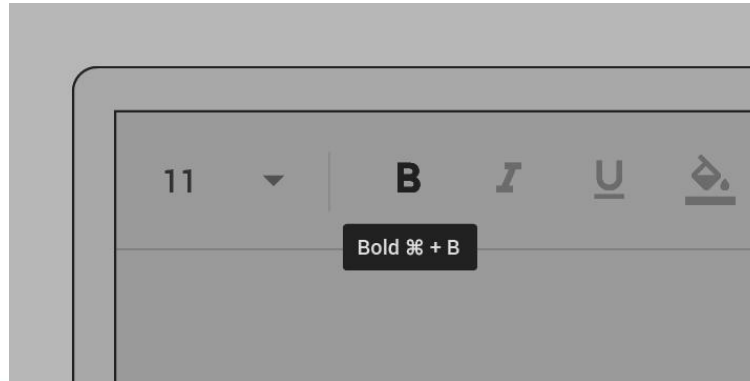
Property	Class/DataType	Deskripsi
child	Widget?	Widget yang digunakan untuk single child
decoration	Decoration?	Mengatur dekorasi dan style pada container
height	double?	Set tinggi tooltip
richMessage	InlineSpan?	Menampilkan teks pesan dalam tooltip
showDuration	Duration?	Set durasi tooltip ditampilkan setelah kursor keluar objek
textStyle	TextStyle	Mengatur style pada tulisan dalam tooltip
triggerMode	TooltipTriggerMode?	Mengatur jenis trigger untuk menampilkan tooltip
verticalOffset	double?	Mengatur jarak antara widget dengan tooltip
waitDuration	Duration	Set durasi tooltip tampil ketika di trigger

Property lain dapat dilihat [disini](#) dan [disini](#)

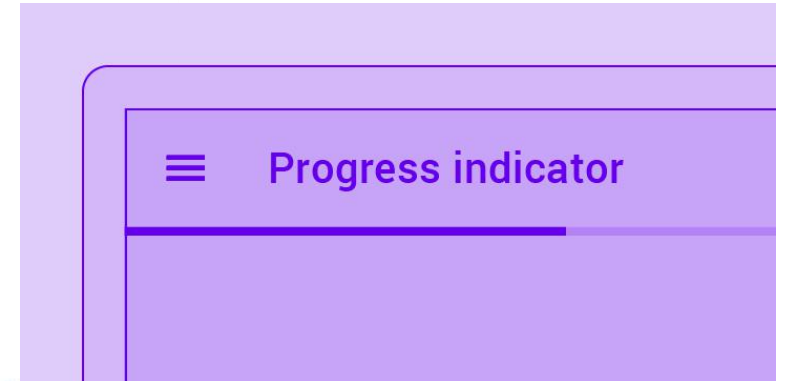
# Komponen dalam Mendesain Aplikasi Part. 6



Sliders



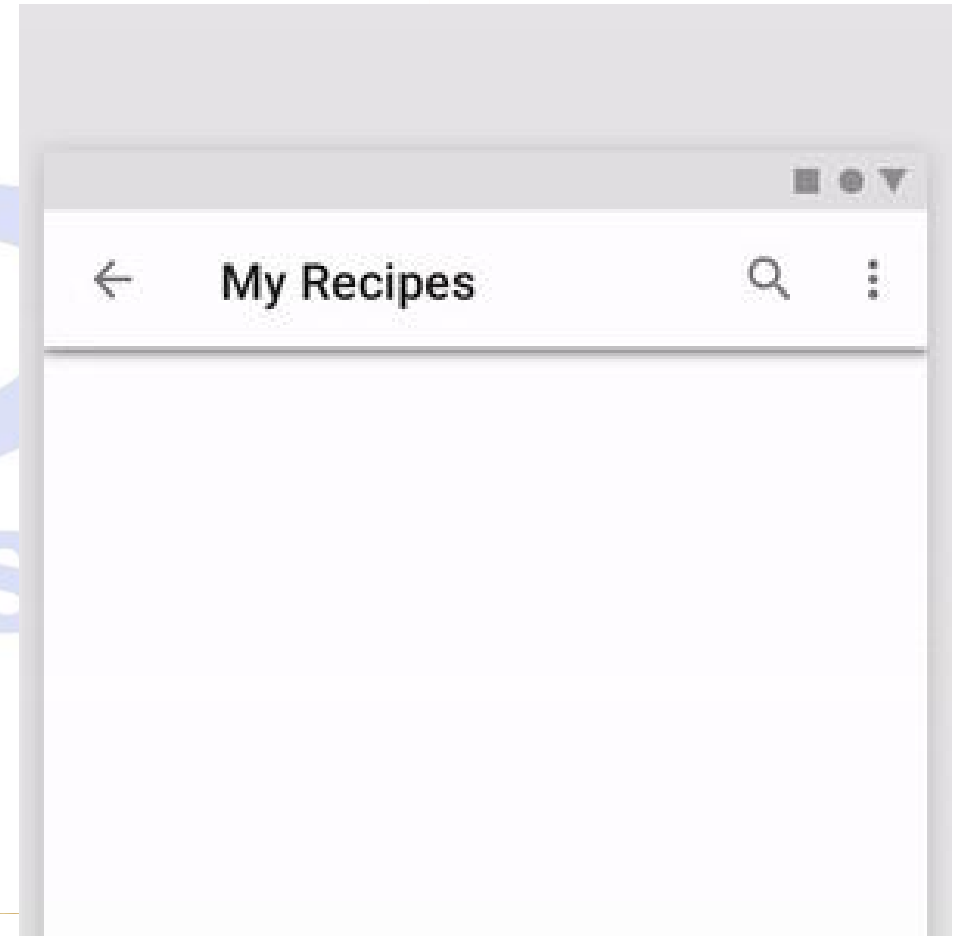
ToolTips



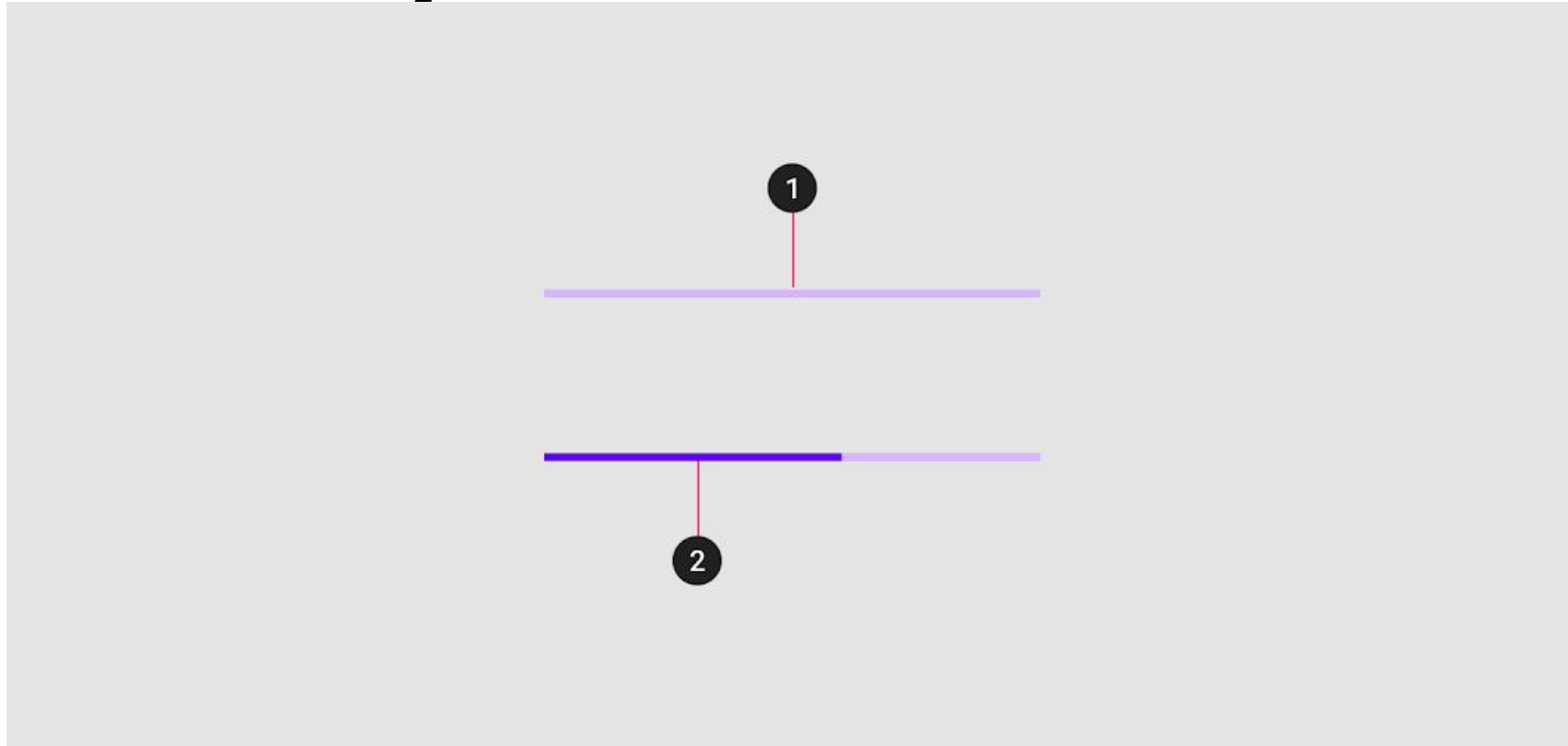
Progress Indicator

# ProgressIndicator adalah...

- Widget yang menyatakan waktu tunggu yang tidak ditentukan atau menampilkan lamanya proses
- Memberi tahu pengguna tentang status proses yang sedang berlangsung



# Anatomy (Linear progress indicators)



**1. Track**

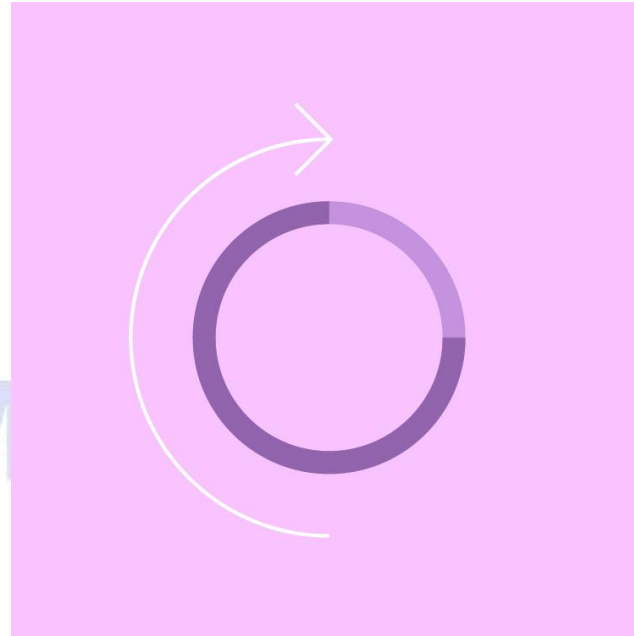
**2. Indicator**

# Principle of ProgressIndicator

**Informative**



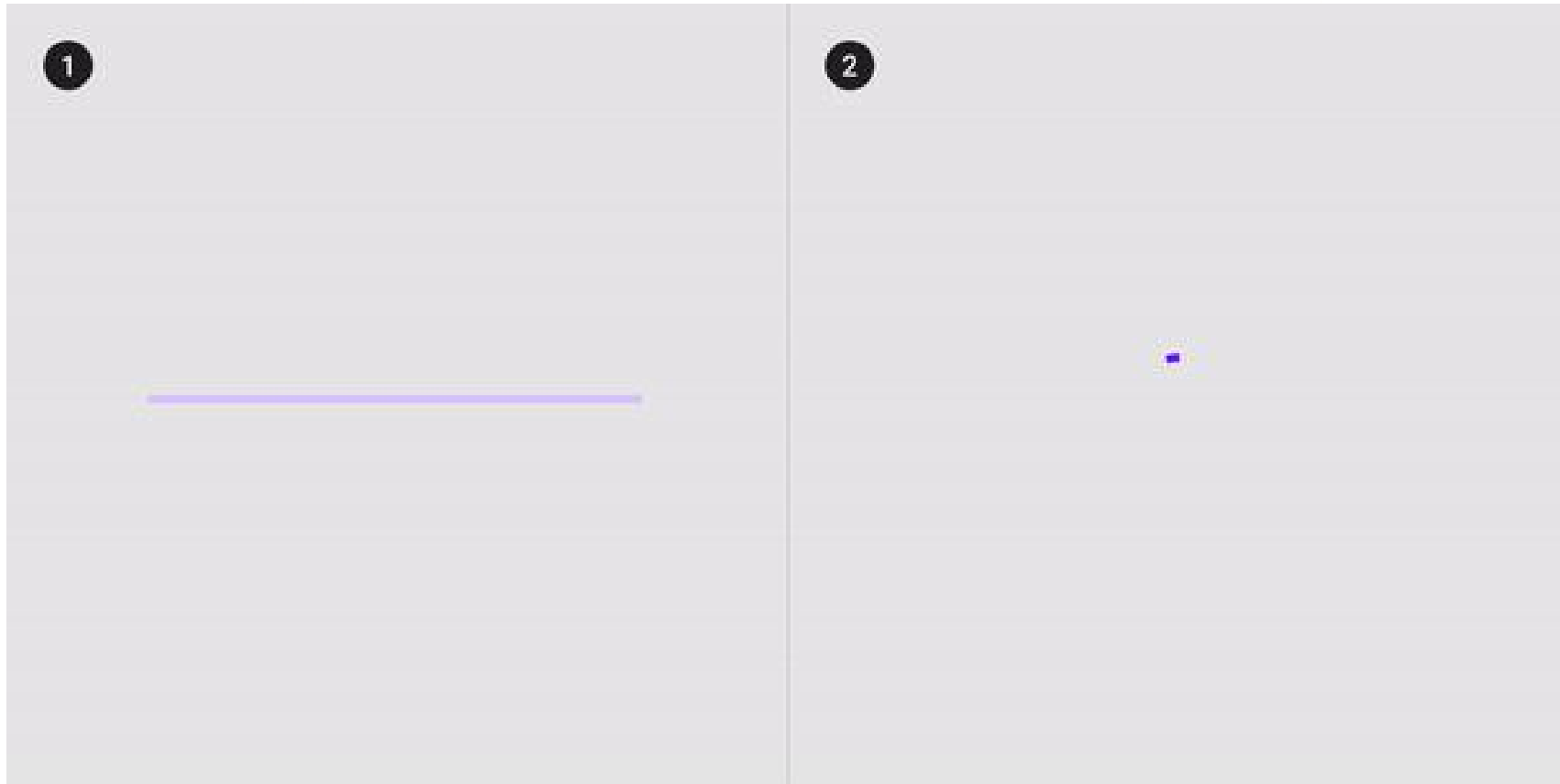
**Animated**



**Consistent**



# Jenis ProgressIndicator



**1. Linear indicator**

**2. Circular indicator**

# Jenis ProgressIndicator

## Determinate indicators

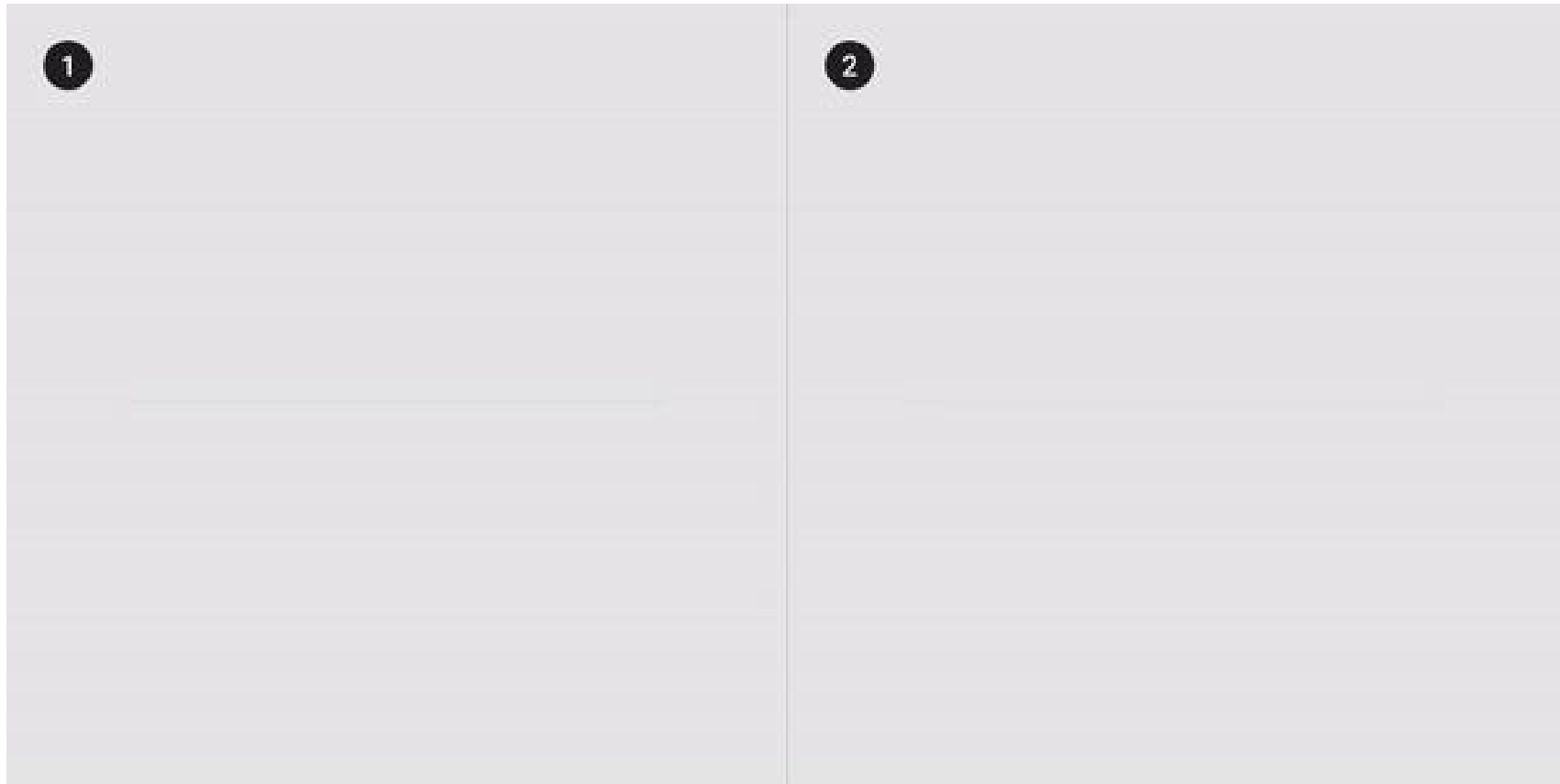
- Menampilkan berapa **lama** suatu proses akan berlangsung.
- Digunakan ketika tingkat penyelesaian proses **dapat dideteksi**

## Indeterminate indicators

- Menunjukkan jumlah waktu tunggu yang **tidak ditentukan**.
- Digunakan ketika **kemajuan tidak terdeteksi**
- Digunakan ketika **tidak perlu** untuk menunjukkan berapa **lama suatu aktivitas akan**



# Jenis ProgressIndicator



**1. Determinate  
indicators**

**2. Indeterminate  
indicators**

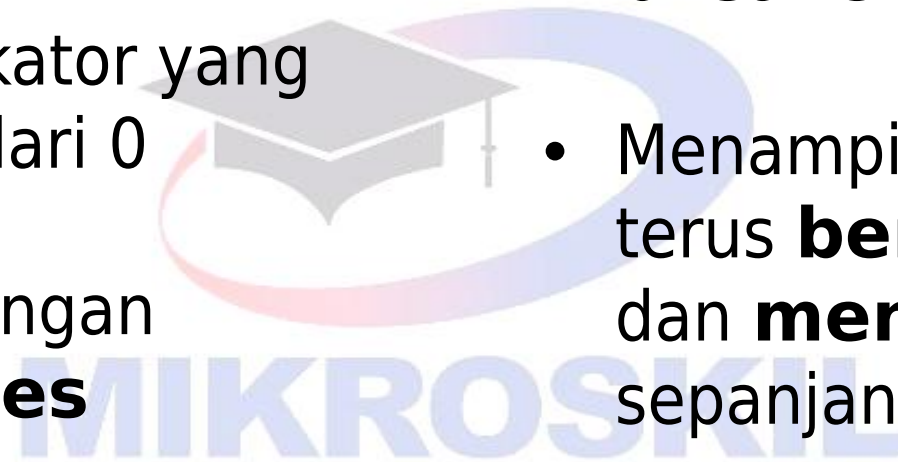
# Jenis ProgressIndicator (Linear)

## Determinate indicators

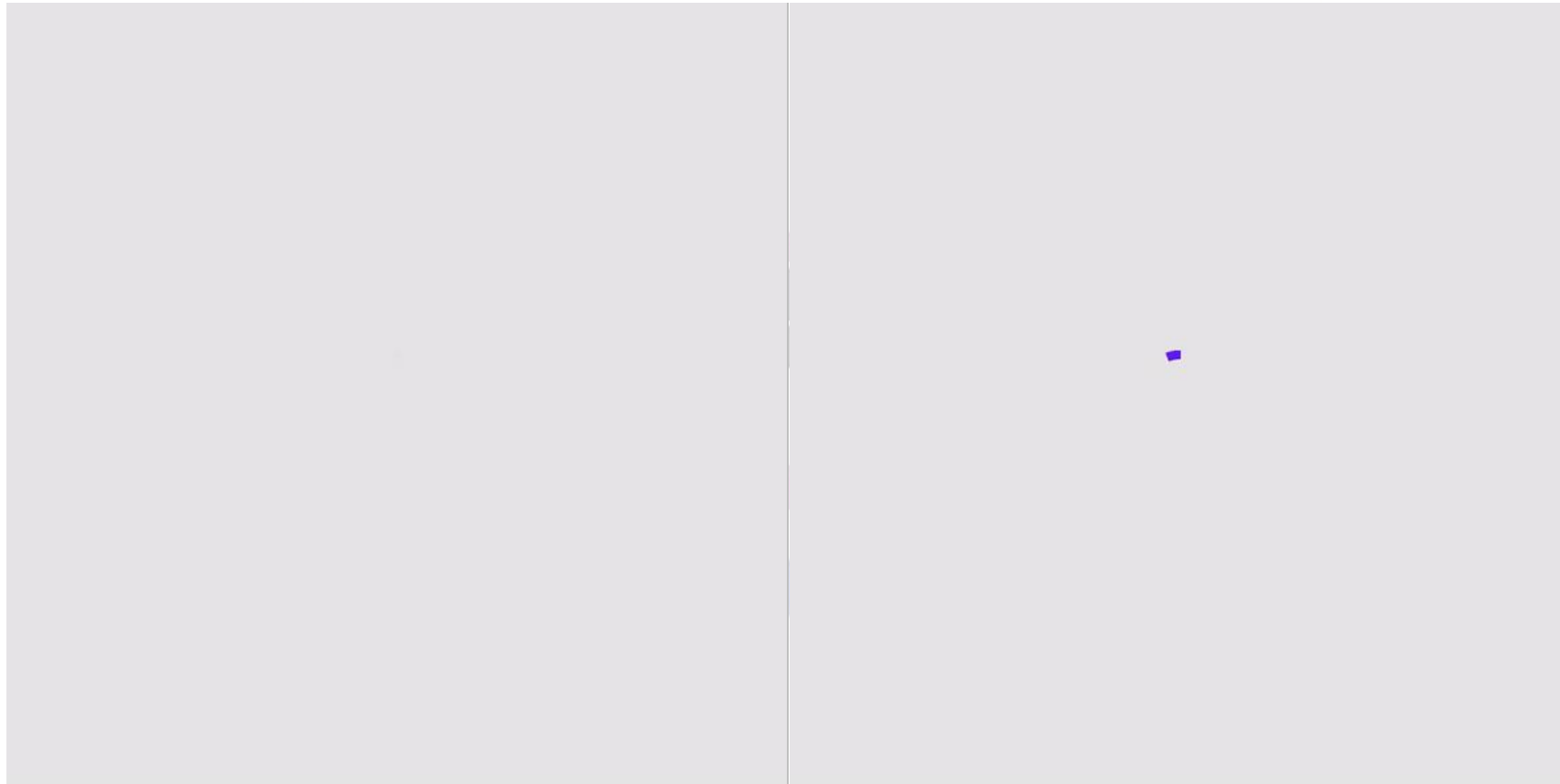
- Menampilkan indikator yang bertambah lebar dari 0 hingga 100%.
- Lebar **selaras** dengan **kemajuan proses**

## Indeterminate indicators

- Menampilkan indikator yang terus **bertambah lebar** dan **menyusut** di sepanjang lintasan **hingga proses selesai**



# Jenis ProgressIndicator



**1. Determinate  
indicators**

**2. Indeterminate  
indicators**

# Jenis ProgressIndicator (Circular)

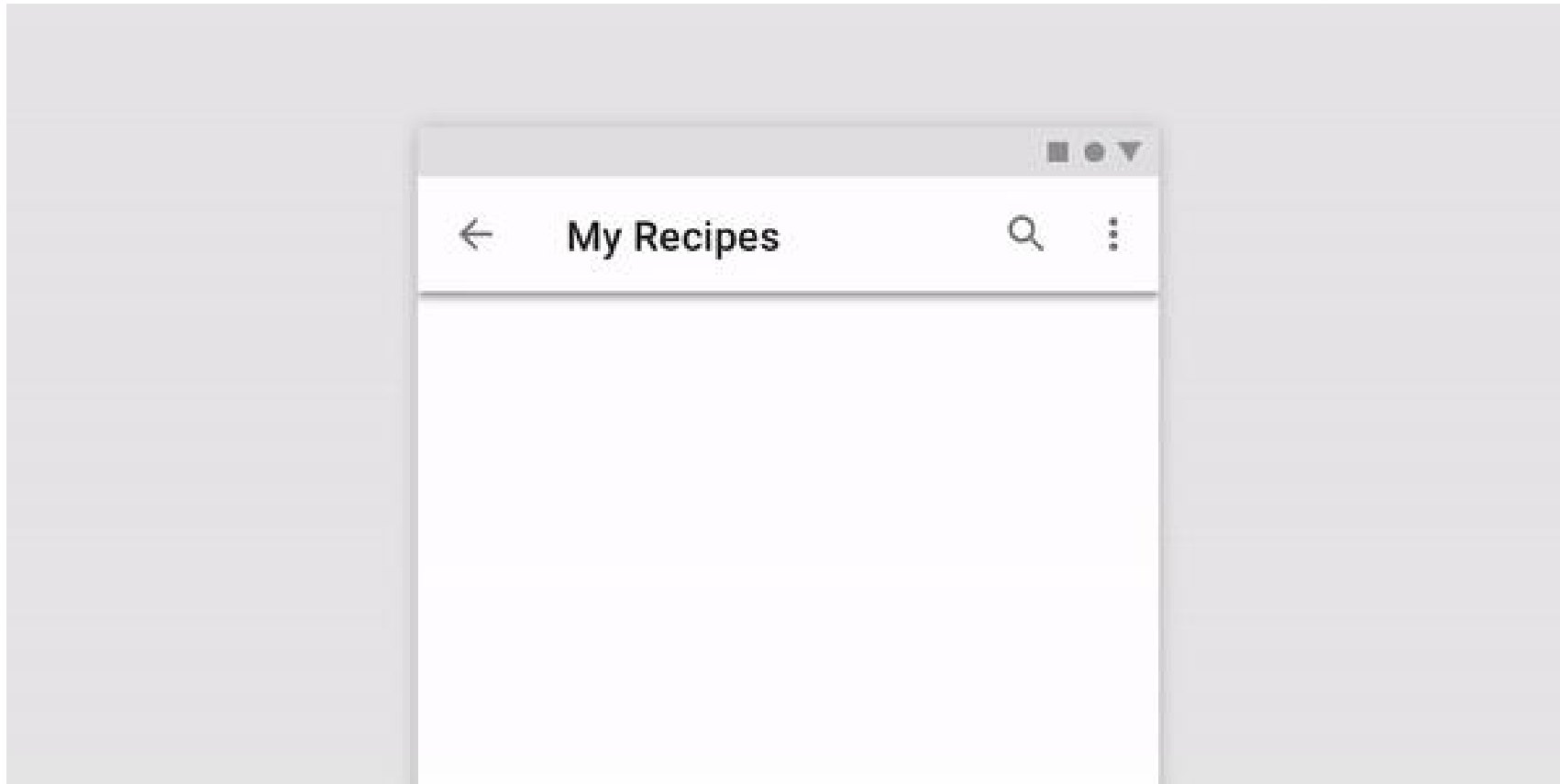
## Determinate indicators

- Mengisi track melingkar yang bertambah lebar dari 0 hingga 360°
- Lebar **selaras** dengan **kemajuan proses**

## Indeterminate indicators

- Menampilkan indikator yang terus **bertambah lebar** dan **menyusut** di sepanjang invisible track **hingga proses selesai**

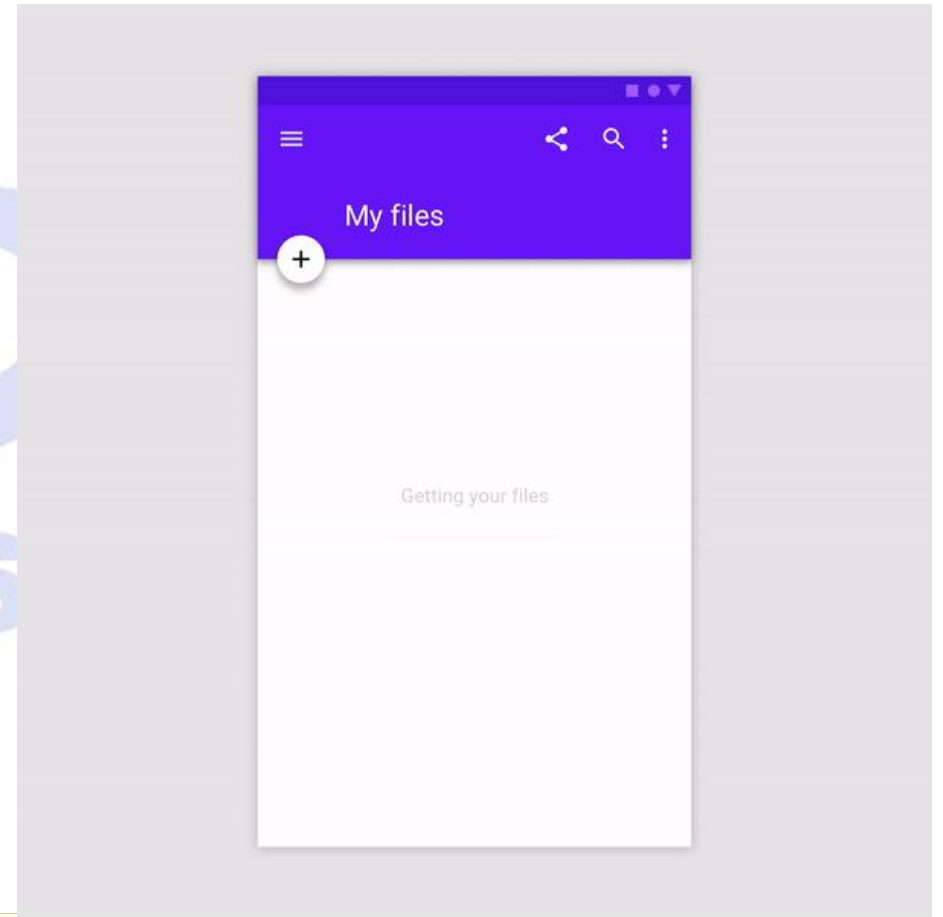
# Jenis ProgressIndicator



Linear progress indicator changes from indeterminate to determinate

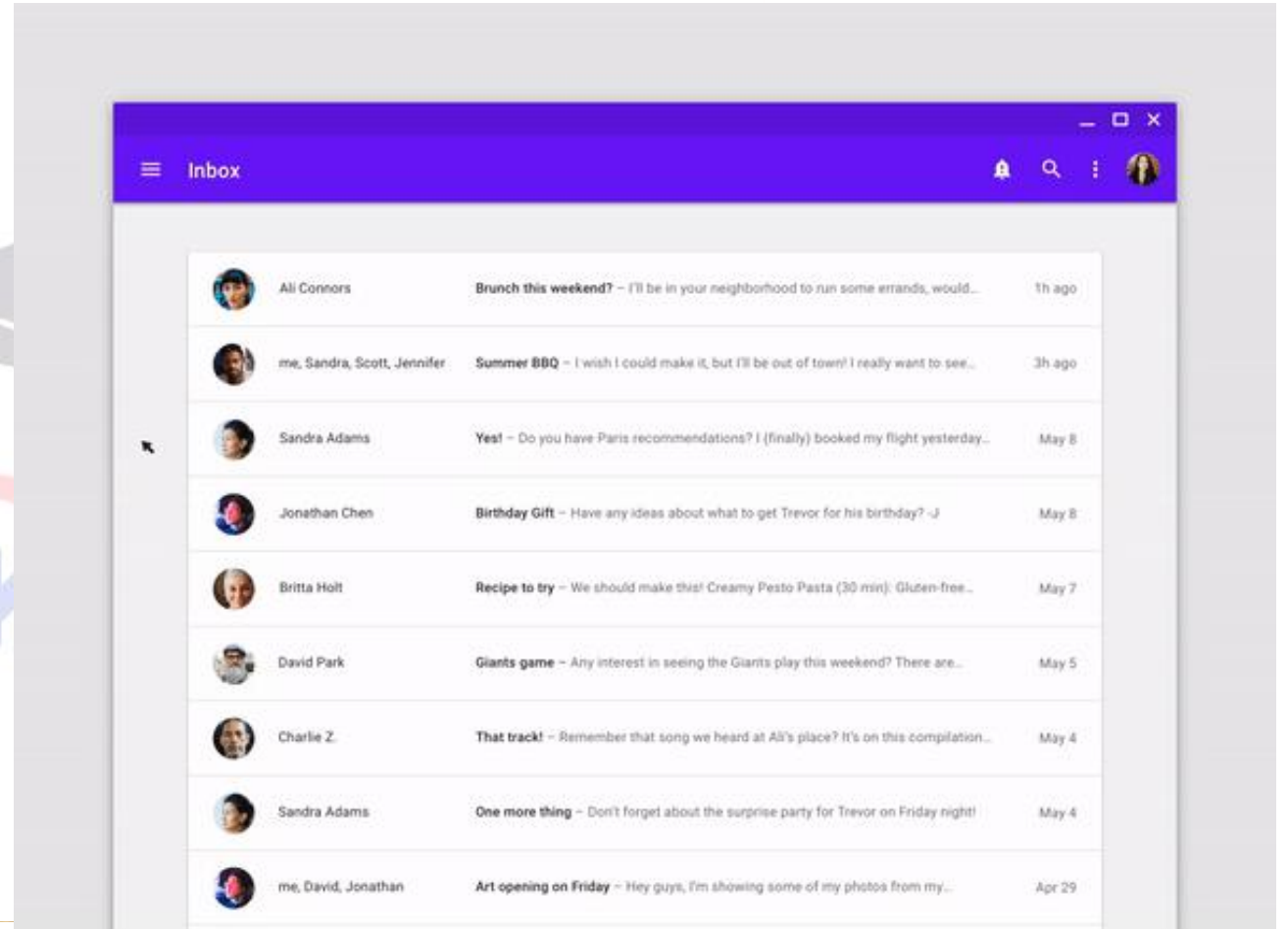
# Placement (Linear ProgressIndicator)

- Linear Indicator di **tengah layar** untuk loading **semua konten** layar
- Linear Indicator dapat digunakan pada **container**, seperti **Card**.



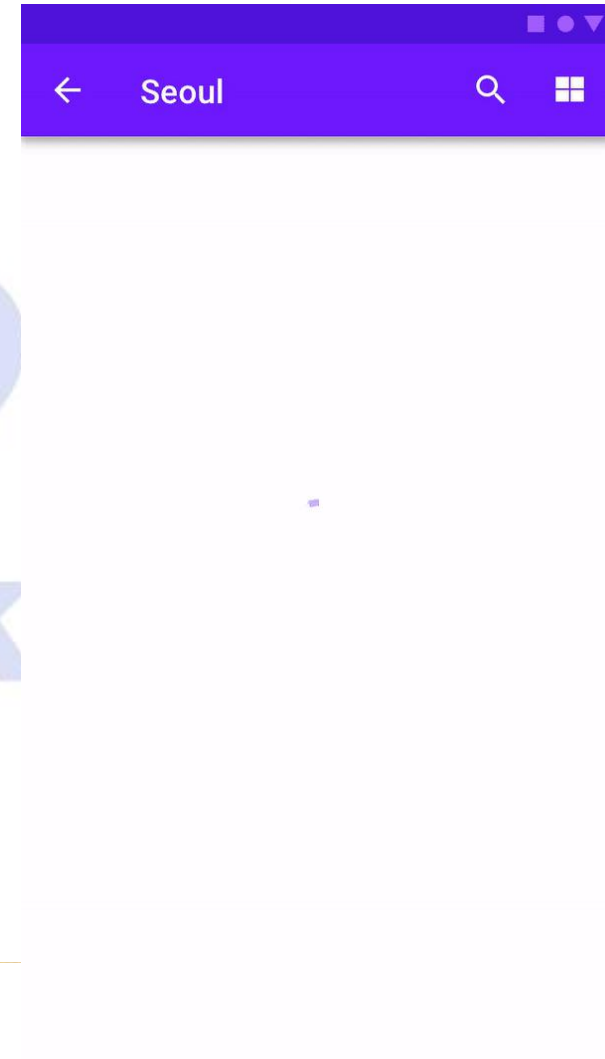
# Placement (Linear ProgressIndicator)

Dapat menempatkan  
Linear Indicator di  
sepanjang tepinya di  
**tepi border**  
container



# Placement (Circular ProgressIndicator)

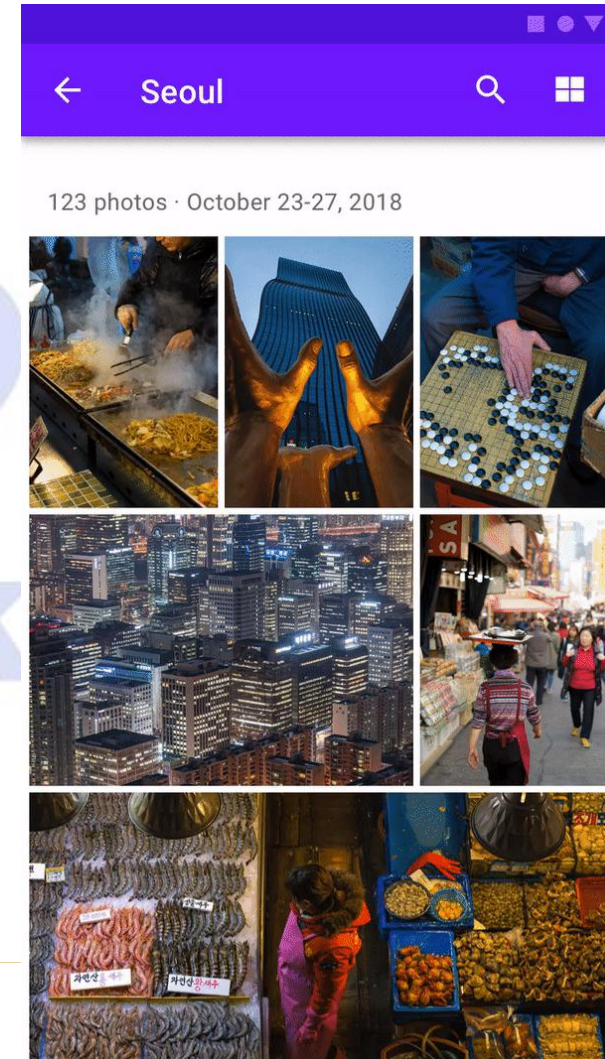
- Saat berada **di tengah** layar, menunjukkan **loading awal** konten



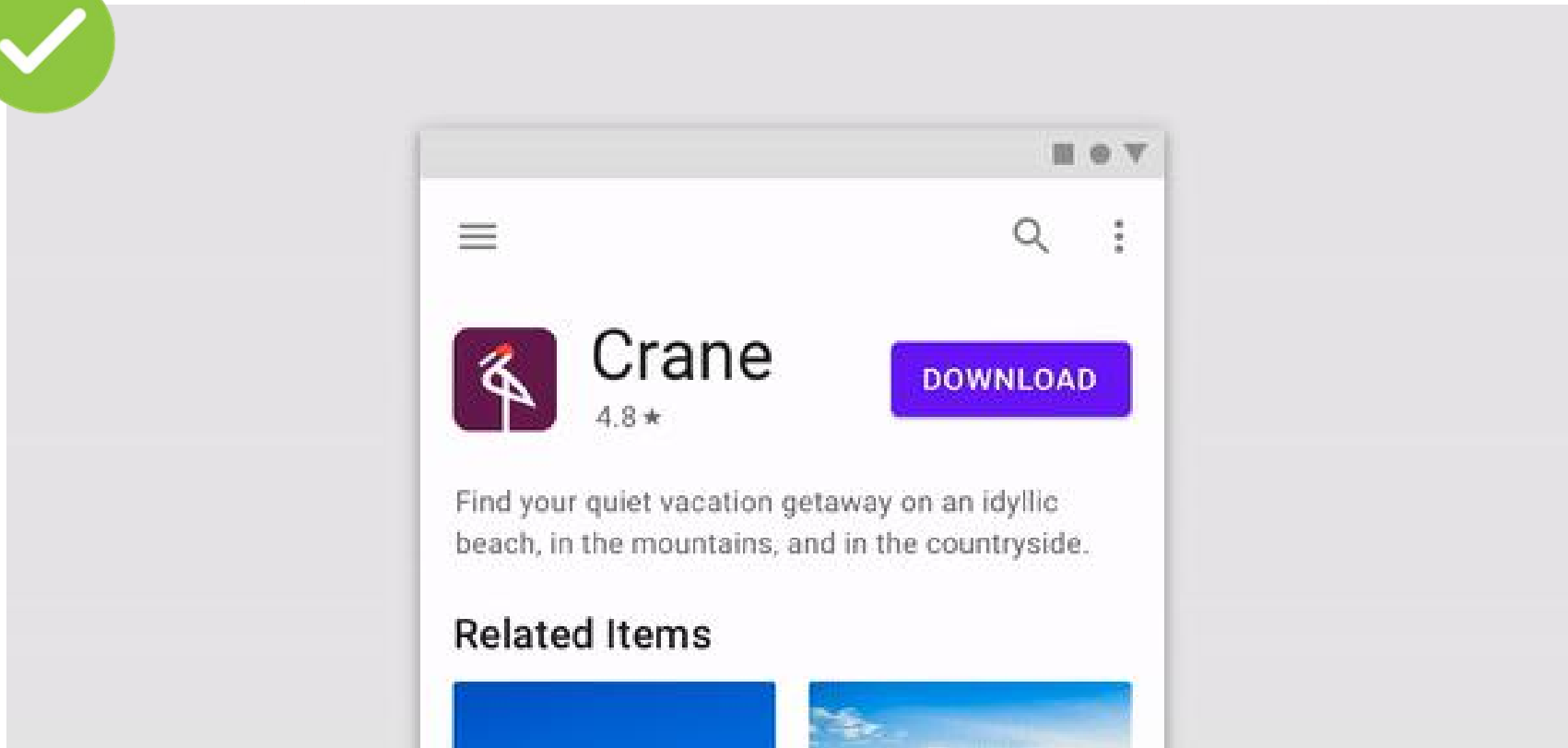


# Placement (Circular ProgressIndicator)

- Ketika berada **di bawah** atau **di atas** konten, menunjukkan loading **konten yang baru**.

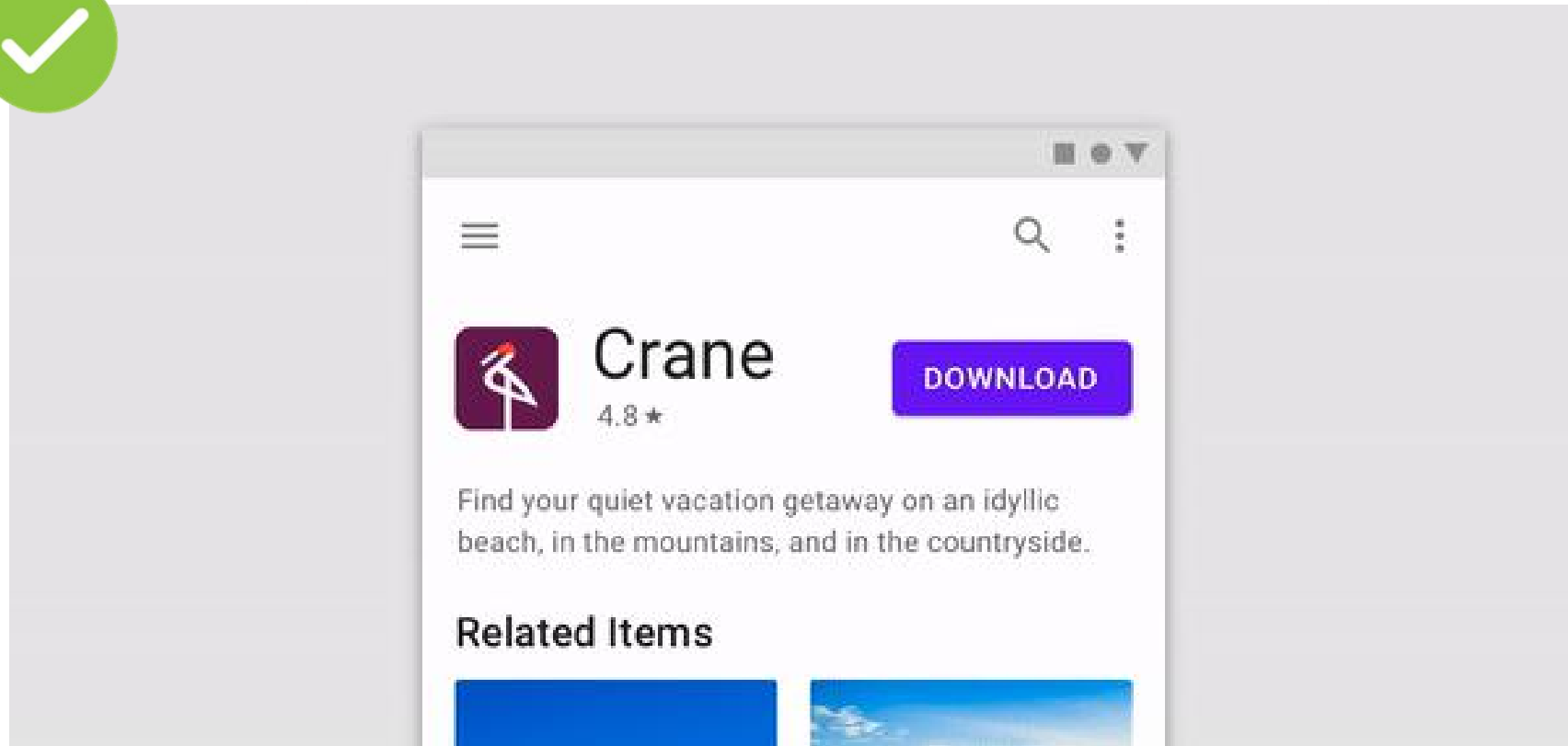


# Integrasi dengan actions



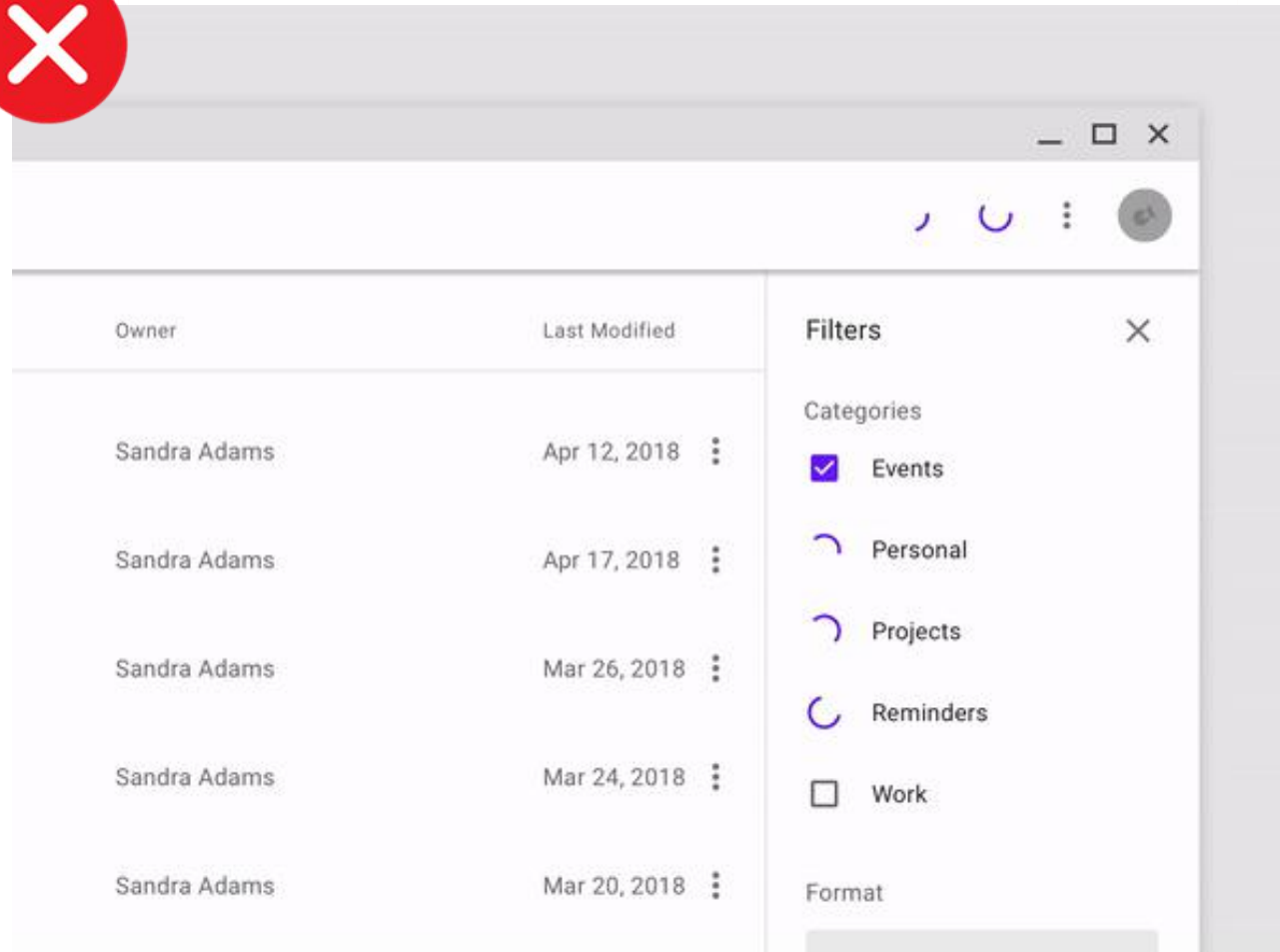
Gunakan circular indicator untuk **aktivitas singkat** dan **tidak pasti (<5 detik)** yang **membutuhkan perhatian** pengguna.

# Integrasi dengan actions



Gunakan circular indicator untuk **aktivitas singkat** dan **tidak pasti (<5 detik)** yang **membutuhkan perhatian** pengguna.

# Integrasi dengan actions



**Jangan** terapkan progress indicator ke **setiap tombol**

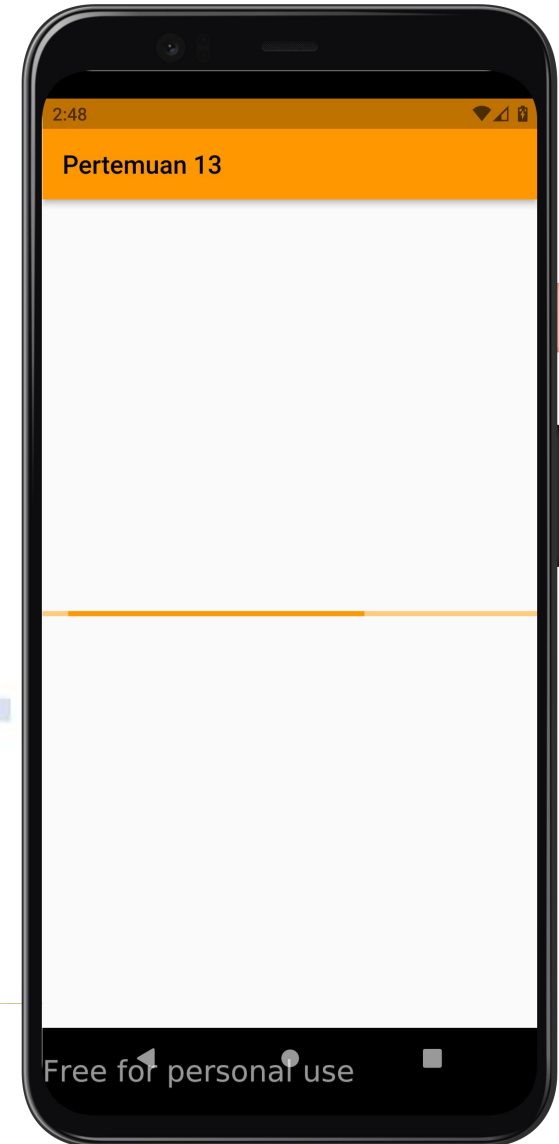
# Penerapan di Flutter



# Linear ProgressIndicator



```
body: Center(  
  child: LinearProgressIndicator(),  
)
```



# Property di Linear ProgressIndicator

Property	Class/DataType	Deskripsi
color	Color?	Mengatur warna ProgressIndicator
backgroundColor	Color?	Mengatur warna track yang terisi
minHeight	double?	Mengatur tinggi minimum
value	double?	Sebagai nilai pada progress

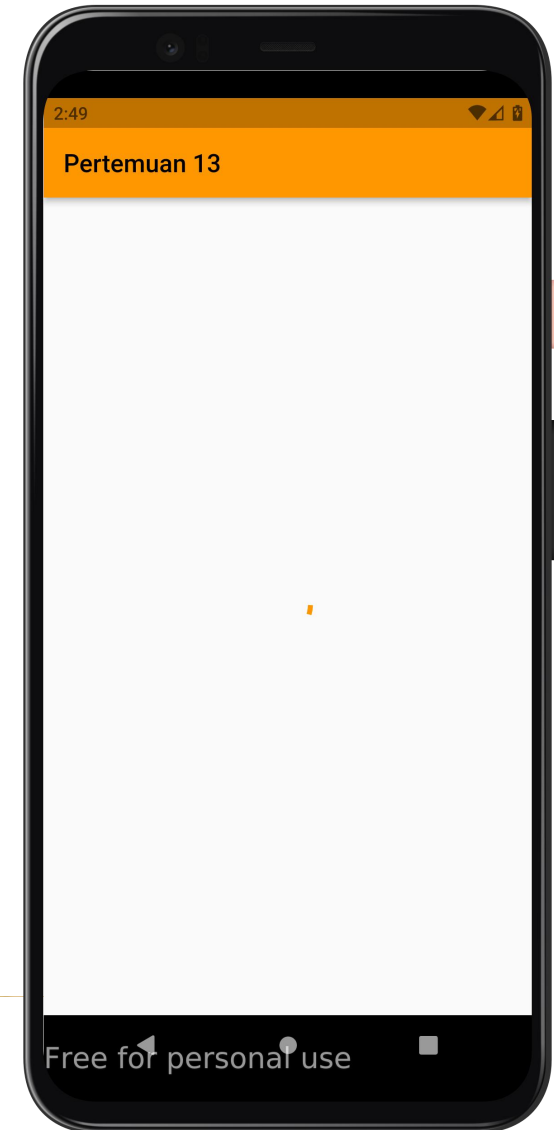
MIKROSKIL

Property lain dapat dilihat [disini](#)

# Circular ProgressIndicator



```
body: Center(  
  child: CircularProgressIndicator(),  
)
```





# Property di Circular ProgressIndicator

Property	Class/DataType	Deskripsi
color	Color?	Mengatur warna ProgressIndicator
backgroundColor	Color?	Mengatur warna track yang terisi
strokeWidth	double	Mengatur lebar garis untuk lingkaran track
value	double?	Sebagai nilai pada progress

MIKROSKIL

Property lain dapat dilihat [disini](#)

# Terima Kasih

