**Container**

Parameter of Container:

height & width:

height and width define the dimensions of a widget, specifying its size along the vertical

and horizontal axes, respectively,

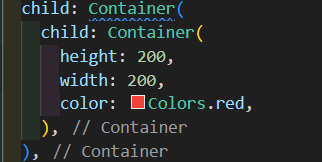
The Container widget allows you to set properties such as height and width to control

the size of the container.

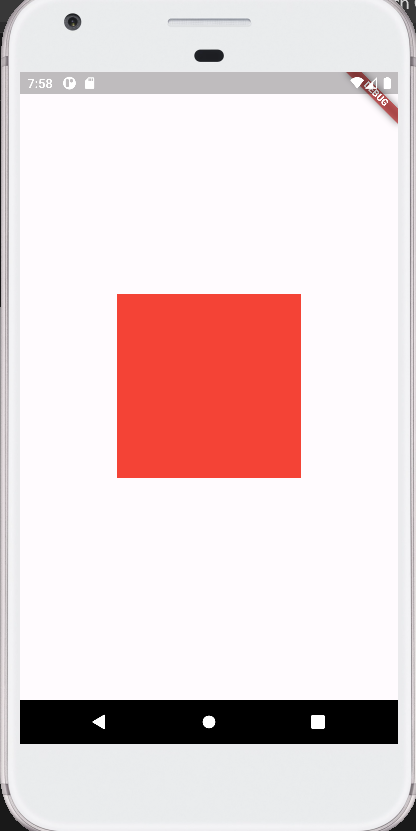
color:

Color defines the background color of the container.

Adjust values and color according to your layout and design preferences



OUTPUT:



padding:-

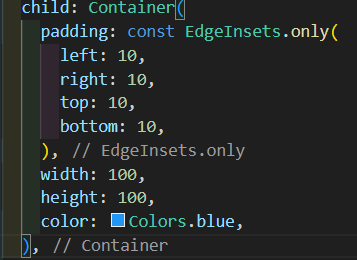
The padding adds space around the child. It is used to create space or margins around

other widgets.

We can add padding in horizontal and vertical directions using EdgeInsets.symmetric().

To add padding to a specific direction use EdgeInsets.only(). Similarly to add

padding to all the directions use EdgeInsets.all().



OUTPUT:



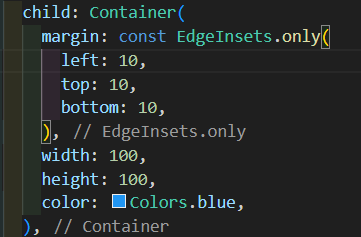
margin:-

margin is a property used to add space outside a widget's border. It represents the space

between the widget's border and the surrounding widgets or the edge of the screen.

Similar to padding we have multiple options to give margin to Container i.e

EdgeInsets.only(), EdgeInsets.all(), EdgeInsets.symmetric()



OUTPUT:

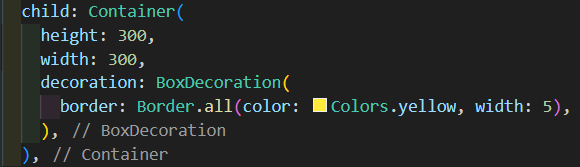


decoration:-

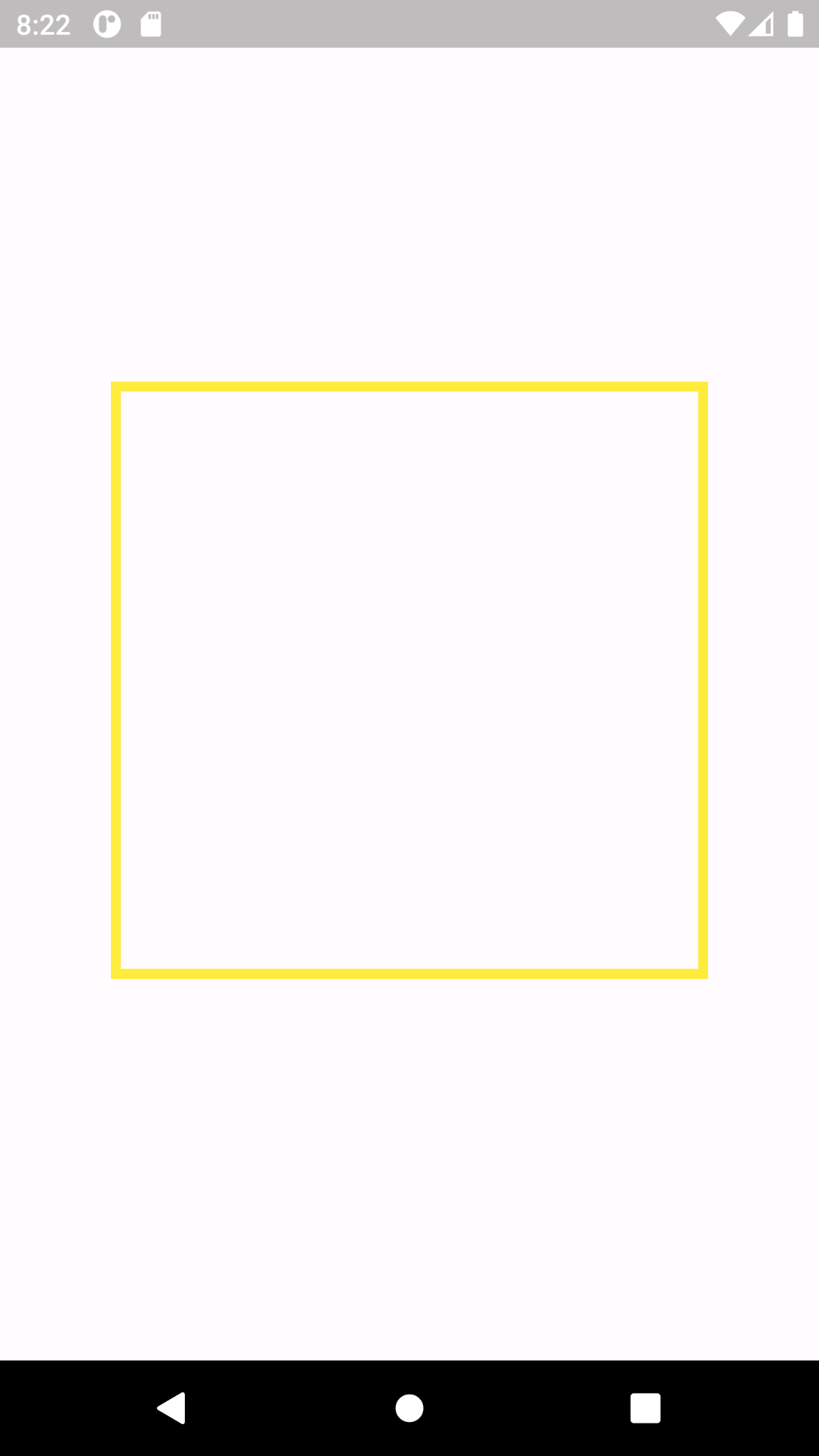
decoration property is used to apply visual styling to a widget, typically a Container. It

takes a BoxDecoration object that defines various visual elements such as color, border

Border:



OUTPUT:

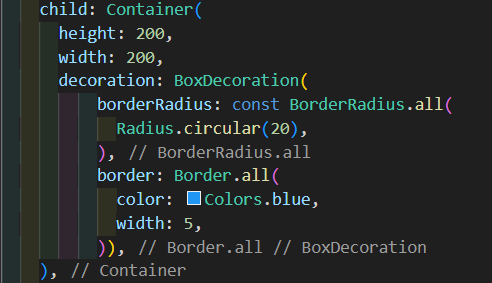


borderRadius:

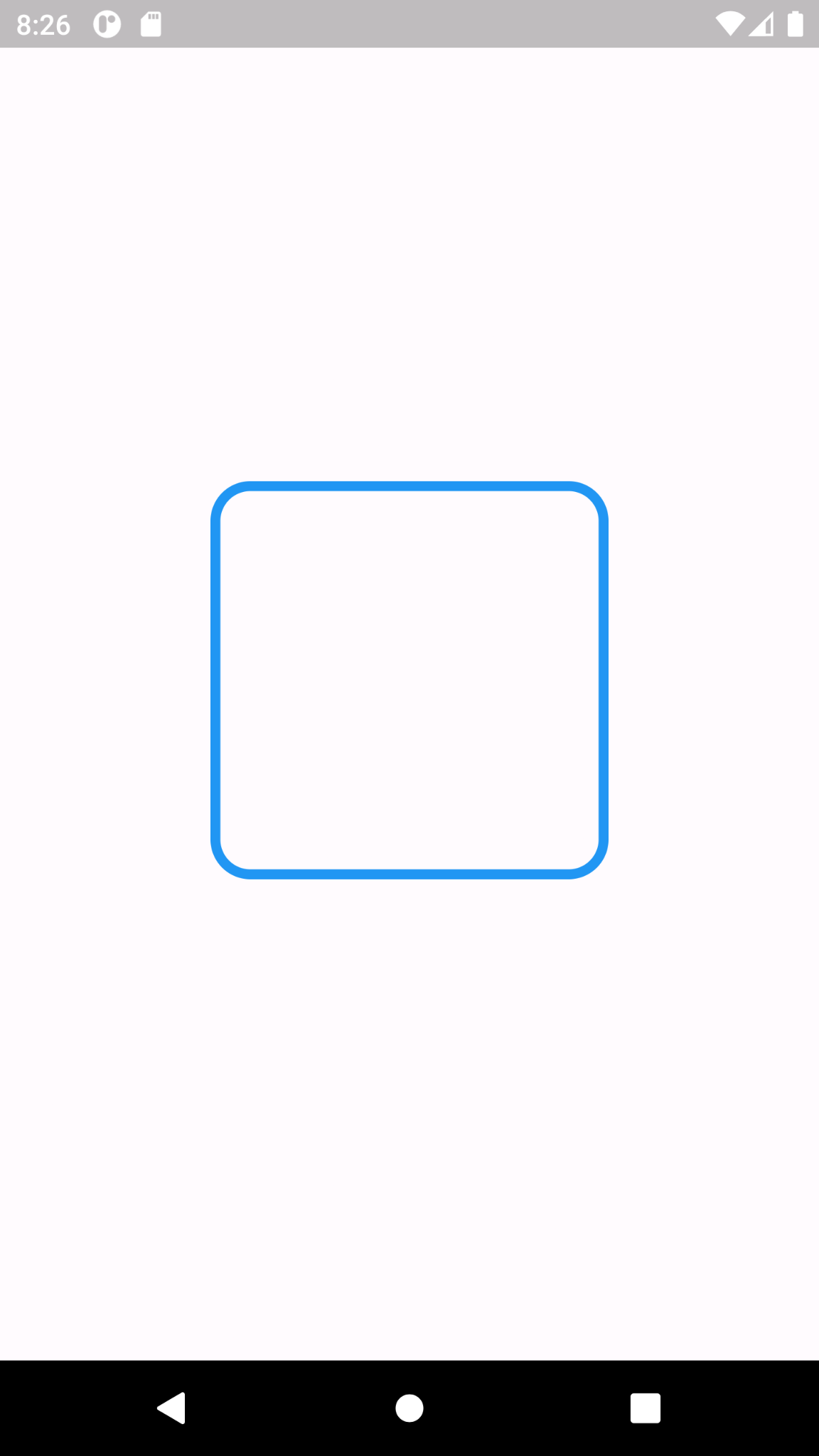
To set the border to the Container we can use Border.all() , Border.symmetric() ,

Border(). We can set the border color, and border width using the Border class. To

give the border to a particular side we can use the Border().

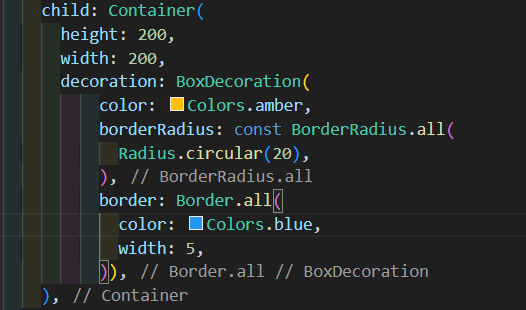


OUTPUT:

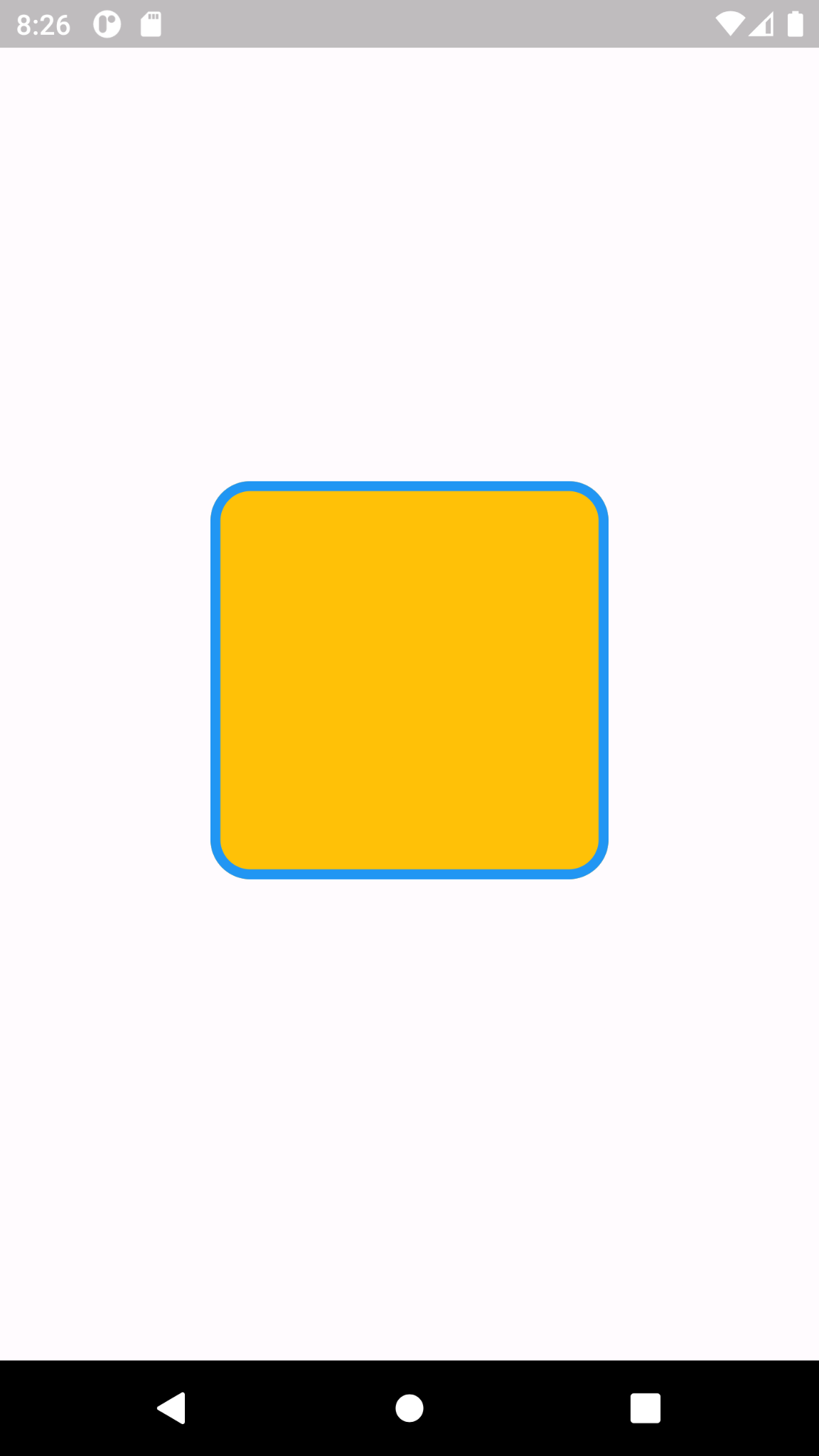


color:

This parameter is used to color the Container inside the decoration .



OUTPUT:



boxShadow:

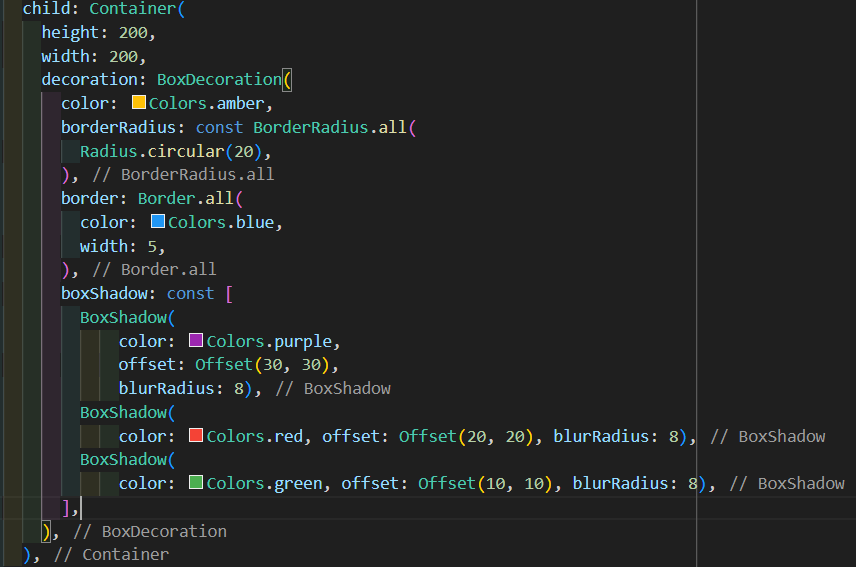
It takes a List of BoxShadow. Using this we can display the shadow behind the

container. We can specify the color, offset(Position) and blurRadius in the BoxShadow.

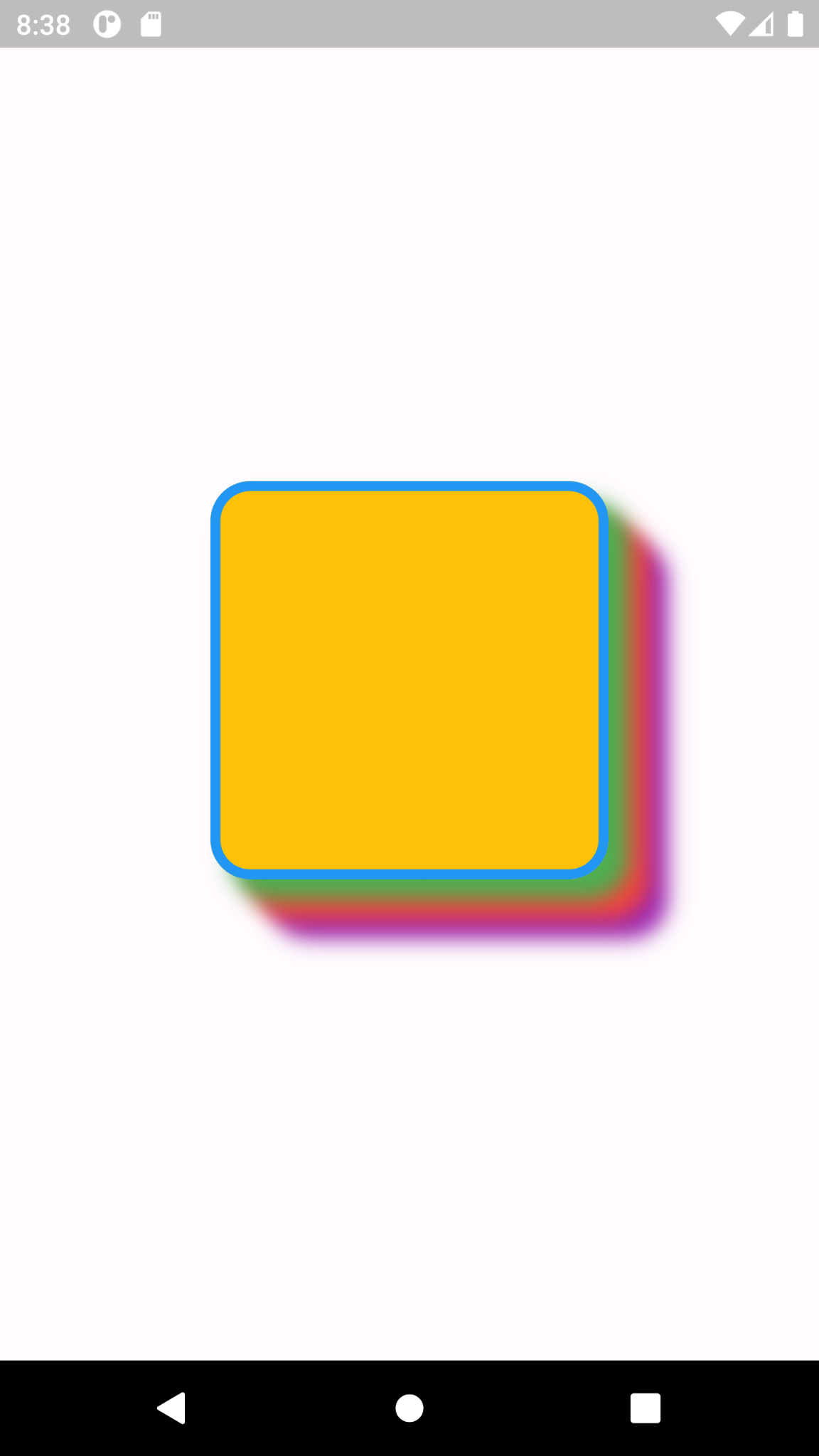
Positive x/y offsets will shift the shadow to the right and down, while

negative offsets shift the shadow to the left and up.

The first item in the list will represent the last layer of shadow.



OUTPUT:



gradient:

Gradient refers to the transition between two or more colors.

We can give a gradient to the Container using LinearGradient(), SweepGradient(),

RadialGradient

In the Linear Gradient, we need to specify the List of colors to be used to display the

gradient.

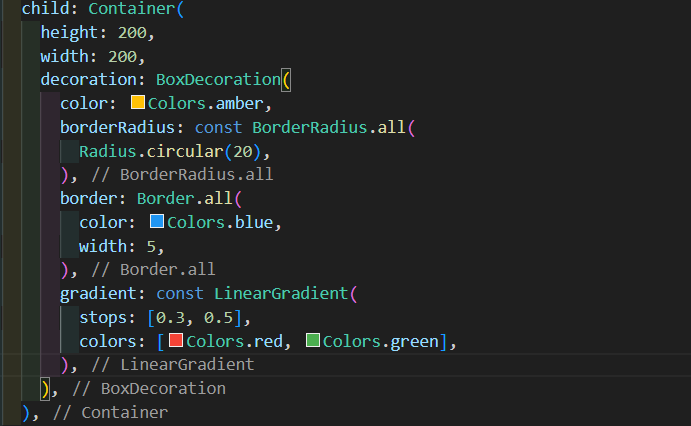
We can also specify the position to place the color using the stops parameter.

The values in the stop parameter range between 0.0 to 1.0.

The values in the stops list must be in ascending order.

If a value in the [stops] list is less than an earlier value in the list, then its value

is assumed to equal the previous value.



OUTPUT:

