## Layout

## [Fuente](https://flutter.io/widgets/layout/)

## MAX

* marzo-2018 [Single-child layout widgets](https://flutter.io/widgets/layout/#Single-child%20layout%20widgets)

#### Container class

Un contenedor primero encierra al hijo con padding, inflado por cualquier border presente en la decoración y luego aplica restricciones al padded.Luego se encierra en espa

cio vacío descrito por el margen

**new** Center(

child: **new** Container(

margin: **const** EdgeInsets.all(10.0),

color: **const** Color(0xFF00FF00),

width: 48.0,

height: 48.0,

),

)

#### Padding class

**new** Padding(

padding: **new** EdgeInsets.all(8.0),

child: **const** Card(child: **const** Text('Hello World!')),

)

La mayoría de los widgets en Flutter son simples composiciones de otros widget , en lugar de herencia

#### Center

#### Align class

#### FittedBox class

Escala y posiciona a su hijo dentro de sí mismo de acuerdo con el ajuste.



#### AspectRatio class

Un widget que intenta ajustar el tamaño del niño a una relación de aspecto específica.

El widget primero prueba el ancho más grande permitido por las restricciones de diseño. La altura del widget se determina aplicando la relación de aspecto dada al ancho, expresada como una relación de ancho a alto.

Por ejemplo, una relación de aspecto anchura: altura de 16: 9 tendría un valor de 16.0 / 9.0. Si el ancho máximo es infinito, el ancho inicial se determina aplicando la relación de aspecto a la altura máxima.

#### ConstrainedBox class}

Un widget que impone restricciones adicionales en su hijo.Por ejemplo, si desea que el niño tenga una altura mínima de 50.0 píxeles lógicos, puede usar const BoxConstraints (minHeight: 50.0) como restricciones.

**new** ConstrainedBox(

constraints: **const** BoxConstraints.expand(),

child: **const** Card(child: **const** Text('Hello World!')),

)

#### Baseline

Un widget que coloca a su hijo de acuerdo con la línea base

#### FractionallySizedBox

Un widget que dimensiona su hijo a una fracción del espacio total disponible

# IntrinsicHeight,IntrinsicWidth

Un widget que dimensiona su hijo a la altura/ancho intrínseca del niño.

Esta clase es útil, por ejemplo, cuando la altura ilimitada está disponible y le gustaría que un niño que de otra manera intentaría expandirse infinitamente se dimensione a una altura más razonable

No usar

#### LimitedBox

Una caja que limita su tamaño solo cuando no está restringida.

Si el ancho máximo de este widget no está restringido, entonces el ancho de su hijo está limitado a maxWidth. Del mismo modo, si la altura máxima de este widget no está restringida, entonces la altura de su hijo está limitada a maxHeight.

Esto tiene el efecto de darle al niño una dimensión natural en entornos ilimitados. Por ejemplo, al proporcionar un maxHeight a un widget que normalmente intenta ser lo más grande posible, el widget normalmente se dimensionará a sí mismo para ajustarse a su elemento principal, pero cuando se lo coloque en una lista vertical, tomará la altura dada.

Esto es útil al componer widgets que normalmente intentan hacer coincidir el tamaño de sus padres, para que se comporten razonablemente en listas (que no tienen límites).

#### Offstage

No entendi

#### OverflowBox

Un widget que impone diferentes restricciones en su hijo de las que obtiene de su padre, posiblemente permitiendo que el hijo desborde al padre.

#### SizedBox