

Resumen de Widgets en [Flutter](#)

Tipo	Nombre	Subtipos	Sirve para
General	Stateless Widget	stless	Widget sin estado. Contiene MaterialApp
General	Stateful Widget	stless	Widget con estado. Contiene MaterialApp.
		Extiende la clase Estado	Puede ampliar initState y usar SetState
General	MaterialApp	Representa mi aplicación?. Contiene title y home. En home se pone un Scaffold debugShowCheckedModeBanner: false, // para que me quite la etiqueta de desarrollo	
General	Scaffold	Representa el DOM de mi aplicación?	
		appbar: AppBar → title	Barra superior, título y navegación
		body: Container	Estructura principal
		(12) Apartado bottomNavigationBar: con un widget tipo BottomNavigationBar y métodos items: , currentIndex: y widgets BottomNavigationBarItem	
		drawer: Drawer (25)	Menú lateral
Posicionamiento	Container	Objeto general. Se le puede hacer un width, height, margin y padding (EdgeInsets.all (), EdgeInsets.only()...). También color. (26) Cuando se combina con Row, necesita el ancho fijo. Por ejemplo, width: MediaQuery.of(context).size.width*0.5	
Posicionamiento	ListView	Objetos tipo lista. Contiene children: [], array de widgets	
		Video 9. Usar con ListView Builder, y cada item un ListTile	

Resumen de Widgets en [Flutter](#)

Tipo	Nombre	Subtipos	Sirve para
Posicionamiento	ListTile	Vídeo 33. Agregamos al List.Builder que retorne un objeto ListTile. Es una manera de crear listas	
Posicionamiento	Dismissible	Vídeo 34. Wrap de ListTile. permite arrastrar a un lado y realizar una acción. Lo implementamos para borrar datos. Métodos: confirmDismiss, onDismissed, background, direction, key.	
Posicionamiento	Column	Distribución arriba-abajo	
		Propiedades: mainAxisAlignment y mainAxisAlignment, dan la alineación del eje principal (vertical) y el tamaño de la columna. crossAxisAlignment: alineamiento secundario horizontal (26) Modificarse (wraparse) con SingleChildScrollView. Así evitamos overflow vertical.	
Posicionamiento	Row	Distribución izquierda-derecha	
		Propiedades: mainAxisAlignment y mainAxisAlignment, dan la alineación del eje principal (horizontal) y el tamaño de la fila. crossAxisAlignment: alineamiento secundario vertical	
Posicionamiento	GridView (10 y 11)	count: cada cuantos. children: otros widgets	
		En combinación con CARD otro widget	
Posicionamiento	SizedOf	Pequeño separador. Puede ser horizontal (modificar width) o vertical (modificar height)	
		Forma de calcular anchos → width: MediaQuery.of(context).size.width	
Posicionamiento	Expanded	(13)	
Posicionamiento (35)	Divider	pequeño widget que no hace nada. Para reestructurar con column	

Resumen de Widgets en [Flutter](#)

Tipo	Nombre	Subtipos	Sirve para
Posicionamiento (35)	SafeArea	(35) Evita que lo que contiene solape barras de información del propio dispositivo. En el ejemplo envuelve un Scaffold.	
Barra superior (39)	DefaultTabController	Envuelve a un Scaffold. Permite activar la barra superior. Propiedad importante length , con el número de tabs.	
Barra superior (39)	TabBar, Tab	En la propiedad bottom de AppBar. bottom: const TabBar(→ tarBar tabs: [Tab(icon: Icon(Icons.abc)), → Uno, dos, tres iconos Tab(icon: Icon(Icons.abc)), Tab(icon: Icon(Icons.abc)),],)	
Barra superior (39)	TabBarView	Va en el body. Los children son los diferentes cuerpos que se ven según se escogen los tabs.	
Panel deslizante (41)	SlidingUpPanel	body: SlidingUpPanel(color: Colors.transparent, minHeight: 30, maxHeight: MediaQuery.of(context).size.height*0.8,	
Formulario (42)	Form	Trabaja con TextFormField, y dentro de este campo validator. También con variables de estado del formulario final GlobalKey<FormState> _formularioEstado = GlobalKey<FormState>();	
Texto	Text	Contiene un texto.	
Texto	TextStyle	Modificar propiedades del Texto	
Campo de texto	TextField TextFormField	obscureText: → propiedad para convertirlo en contraseña que no se ve. decoration: → forma de decorar el campo. Usar Widget InputDecoration keyboardType: TextInputType.number → cambiar tipo, por ejemplo numérico	
Icono	Icon.network	Propiedad: url → imagen desde internet	
Icono	Icon	Por ejemplo, objetos de la serie Icons: Icons.punch_clock	
Badge	Badge	Envuelve a otro widget, normalmente un icono, y muestra alguna información en un label.	

Resumen de Widgets en [Flutter](#)

Tipo	Nombre	Subtipos	Sirve para
Imagen	NetworkImage	Escribir como string directamente la URL.	
Imagen	Image	.File (Desde fichero)	
Imagen	ImagePicker	De la biblioteca import 'package:image_picker/image_picker.dart'	
Botones	TextButton	Método onPressed: → al presionar foregroundColor, backgroundColor, fixedSize.	
Botones	ElevatedButton		
Botones	IconButton		
Botones	OutlinedButton		
Botones	ButtonStyle	Se usa para modificar el estilo de los botones	
Decoración	BoxDecoration	Dentro de, por ejemplo, un objeto Container. Es la forma de poner una imagen	
		propiedad image: → widget DecorationImage	
Decoración	DecorationImage	image: por ejemplo, NetworkImage fit: → con Widget BoxFit .	
Decoración	InputDecoration	hintText: → pista textual filled y fillColor → van juntas	
Animación (13)	AnimatedContainer	(13) Container que puede animarse. Admite curve: y duration:, métodos de animaciones.	
Animación (13)	InkWell	(13) Puedo añadir método onTap al AnimatedContainer al wraparlo en InkWell	
Animación (14)	CarouselSlider	(14) Animación de imágenes. Instalar en pubspec.yaml carousel_slider: ^4.2.1	
Animación (35)	Hero	Se utiliza normalmente por pares. De imagen a imagen. Hace un efecto curioso. Necesita un tag (que puede ser un uri).	
Alerta (7)	ShowDialog	Suele estar dentro de una función que recibe el contexto. Necesita un builder y dentro el Widget AlertDialog	
Alerta (7)	AlertDialog	Dentro del builder de ShowDialog. La alerta es una ventana, luego vuelve con Navigator.pop	

Resumen de Widgets en [Flutter](#)

Tipo	Nombre	Subtipos	Sirve para
Alerta (36-37)	ScaffoldMessenger	ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content: Text("Imagen subida correctamente"))	
Navegador (6)	Navigator	Nota: crear stateless o stateful widget en otra página con un nombre, por ejemplo Pagina02	
		Navigator.push(context, MaterialPageRoute(builder: (context)=>const Pagina02()),);	
		Navigator.pop(context); → volver	
Future (10 y 11)	FutureBuilder	<p>Recibe unos datos en future: Se construye en builder:</p> <p>Funciona recibiendo datos de internet en una función y asignándoles a una lista. Esta función es asíncrona y tiene variables que deben cargarse de internet. Cargar el componente http en pubspec.yaml</p>	
Persistencia (15)	SharedPreferences	<p>SharedPreferences preferencias = await SharedPreferences.getInstance(); → instancia de SharedPreferences</p> <p>Métodos de cargar preferencias y cambiar preferencias. Guarda información de la aplicación en memoria.</p>	
Datos entre pantallas (16)	TextEditingController	<p>Crear variable tipo TextEditingController → TextEditingController miTexto = TextEditingController(text: "");</p> <p>Conectar esa variable a un TextField, en controller:</p> <p>Pasar esa información con Navigator llamando a otra página.</p> <p>Navigator.push(context, MaterialPageRoute(builder:(context)=> PStless(miTexto.text)),)</p> <p>La recibo como una variable según sea el widget. Stateful y Stateless</p>	

Resumen de Widgets en [Flutter](#)

Tipo	Nombre	Subtipos	Sirve para
Detector de eventos (35)	GestureDetector		Cualquier widget envuelto en este adquiere métodos como onTap, etc. Usado para envolver imágenes, por ejemplo.
Firebase (18)	Firebase		Mejor ver vídeo 18. Conectar a firebase implica una serie de pasos.

Definir	Widget nombreDeMiWidget() {}	return... → retornar un Widget
importar	import 'package:a_primer_programa/ paginas/pagina02.dart';	Primera línea. Forma de importar contenido de otras páginas. Si se inserta un Widget ya escrito en otra página, se escribe automáticamente.
Clase(8)	Class NombreClase {}	Llamada a la clase NombreVariable = new NombreClase(); Ejemplo, Facebook = new Empresa();
Listas (9)	List<Tipo> Nombre = []	final List<Persona> _personas= Persona("Luis","Suárez","003562589"), por ejemplo...
Vídeo 21	SplashScreen	Seguir instrucciones
Vídeo 23	Cambio de logo	La mejor forma, la profesional con https://icon.kitchen
Vídeo 36-37	File, XFile	Clase de objeto tipo fichero