

# Widgets de Base en Flutter avec Explications et Exemples

## 1. Container

- Description: Le Container est un widget polyvalent utilisé pour la mise en page. Il peut contenir des marges, des bordures, des arrière-plans et des transformations.
- Propriétés: height, width, padding, margin, color, etc.

Exemple:

```
Container(  
  height: 100,  
  width: 100,  
  color: Colors.blue,  
  child: Center(  
    child: Text('Hello'),  
  ),  
)
```

## 2. Text

- Description: Le widget Text permet d'afficher du texte sur l'interface.
- Propriétés: style, textAlign, overflow, maxLines, etc.

Exemple:

```
Text(  
  'Bonjour tout le monde',  
  style: TextStyle(fontSize: 24, color: Colors.black),  
)
```

### 3. Row

- Description: Le widget Row dispose ses enfants horizontalement en une seule ligne.
- Propriétés: mainAxisAlignment, crossAxisAlignment, children, etc.

Exemple:

```
Row(  
  mainAxisAlignment: MainAxisAlignment.spaceAround,  
  children: [  
    Icon(Icons.star, color: Colors.red),  
    Icon(Icons.star, color: Colors.green),  
    Icon(Icons.star, color: Colors.blue),  
  ],  
)
```

### 4. Column

- Description: Le widget Column dispose ses enfants verticalement en une seule colonne.
- Propriétés: mainAxisAlignment, crossAxisAlignment, children, etc.

Exemple:

```
Column(  
  mainAxisAlignment: MainAxisAlignment.center,  
  children: [  
    Text('Flutter'),  
    Text('est'),  
    Text('Génial!'),  
  ],  
)
```

)

## 5. Image

- Description: Le widget Image permet d'afficher une image à partir d'une URL, d'un fichier local ou d'autres sources.
- Propriétés: width, height, fit, alignment, etc.

Exemple:

```
Image.network(  
  'https://flutter.dev/images/flutter-logo-sharing.png',  
  width: 100,  
  height: 100,  
)
```

## 6. Scaffold

- Description: Le Scaffold est un widget de mise en page qui fournit une structure de base avec une AppBar, un body, un drawer, un bottomNavigationBar, etc.
- Propriétés: appBar, body, floatingActionButton, drawer, etc.

Exemple:

```
Scaffold(  
  appBar: AppBar(  
    title: Text('Mon Application Flutter'),  
  ),  
  body: Center(  
    child: Text('Bienvenue!'),  
  ),  
)
```

```
floatingActionButton: FloatingActionButton(  
  onPressed: () {},  
  child: Icon(Icons.add),  
),  
)
```

## 7. ListView

- Description: Le ListView est un widget qui permet d'afficher une liste défilante d'éléments.
- Propriétés: children, padding, scrollDirection, etc.

Exemple:

```
ListView(  
  children: [  
    ListTile(  
      leading: Icon(Icons.person),  
      title: Text('Alice'),  
    ),  
    ListTile(  
      leading: Icon(Icons.person),  
      title: Text('Bob'),  
    ),  
    ListTile(  
      leading: Icon(Icons.person),  
      title: Text('Charlie'),  
    ),  
  ],  
)
```

## 8. RaisedButton (ou ElevatedButton dans les versions récentes)

- Description: Le bouton ElevatedButton est un bouton surélevé qui réagit aux pressions. Utilisé pour déclencher des actions.
- Propriétés: onPressed, child, style, etc.

Exemple:

```
ElevatedButton(  
  onPressed: () {  
    print('Bouton appuyé');  
  },  
  child: Text('Cliquez ici'),  
)
```

## 9. Padding

- Description: Le widget Padding ajoute un espace autour de son enfant.
- Propriétés: padding, child.

Exemple:

```
Padding(  
  padding: const EdgeInsets.all(16.0),  
  child: Text('Texte avec un padding de 16.0'),  
)
```