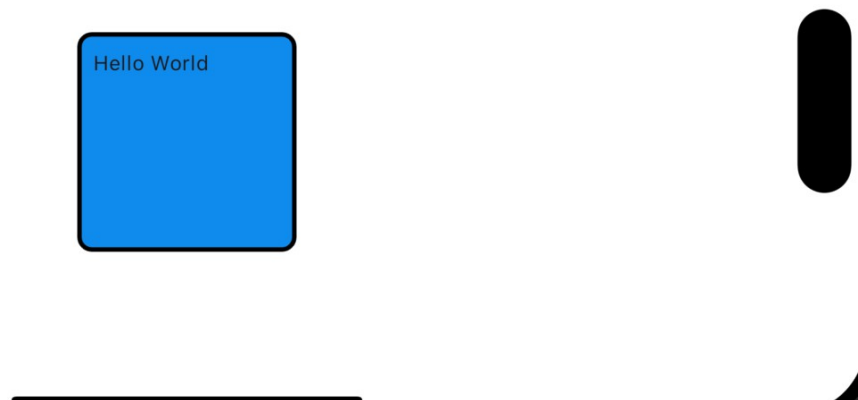


Chapitre 3 – Affichages élémentaires

Table des matières

Centrer un élément	1
Présentation des « card » :.....	3
Les lignes et les colonnes :.....	5
Les lignes :.....	5
Les colonnes :.....	11
Affichage d'une image :.....	15
Afficher du texte :.....	18
Exercice :.....	21

Centrer un élément



```
import 'package:flutter/material.dart';

void main() {
  runApp(const MainApp());
}

class MainApp extends StatelessWidget {
```

```
const MainApp({super.key});
@override
Widget build(BuildContext context) {
  return MaterialApp(
    title: 'Flutter Demo',
    home: const Home(),
    theme: ThemeData(
      appBarTheme: const AppBarTheme(
        backgroundColor: Colors.indigo,
      ),
    ),
    debugShowCheckedModeBanner: false,
  );
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
    return _Home();
  }
}

class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
        centerTitle: true,
      ),
      backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
      body: Center(
        child: Container(
          width: 150.0,
          height: 150.0,
          padding: const EdgeInsets.all(8.0),
          margin: const EdgeInsets.symmetric(horizontal: 10.0),
          decoration: BoxDecoration(
            color: Colors.blue,
```

```
border: Border.all(color: Colors.black, width: 3),
borderRadius: BorderRadius.circular(10)),
child: const Text('Hello World'),
),
),
);
}
}
```

Commentaires :

center: utilisé pour centrer son enfant (le container) dans l'espace disponible.

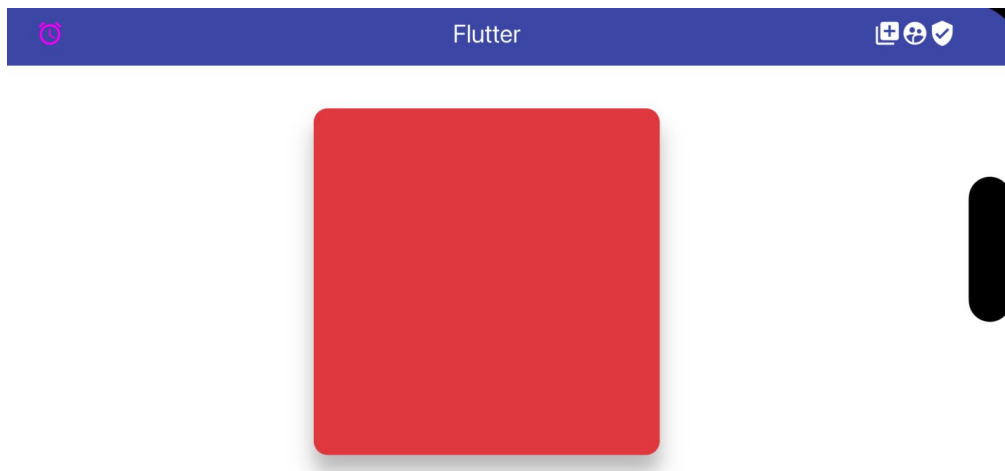
decoration: pour définir le style visuel du Container.

border: Une bordure est appliquée autour du Container avec une épaisseur de 3 pixels et de couleur noire.

borderRadius: Les coins du Container sont arrondis à 10 pixels de rayon.

Présentation des « card » :

Un « Container » permet un contrôle total sur la mise en forme et l'apparence, tandis que le « Card » est utilisé pour les éléments qui doivent se conformer à un style « Material Design » avec une apparence relevée. Vous pouvez utiliser un « Container » à l'intérieur d'une « Card » pour bénéficier à la fois de l'effet visuel de la carte et de la personnalisation du Container.



```
import 'package:flutter/material.dart';

void main() {
  runApp(const MainApp());
}

class MainApp extends StatelessWidget {
  const MainApp({super.key});
```

```
@override
Widget build(BuildContext context) {
  return MaterialApp(
    title: 'Flutter Demo',
    home: const Home(),
    theme: ThemeData(
      appBarTheme: const AppBarTheme(
        backgroundColor: Colors.indigo,
      ),
    ),
    debugShowCheckedModeBanner: false,
  );
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
    return _Home();
  }
}

class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
        centerTitle: true,
      ),
      backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
      body: const Center(
        child: Card(
          elevation: 15.0,
          color: Colors.red,
          child: SizedBox(
            width: 300.0,
            height: 300.0,
          ),
        ),
      ),
    );
  }
}
```

```
    ),  
  );  
}  
}
```

Commentaires :

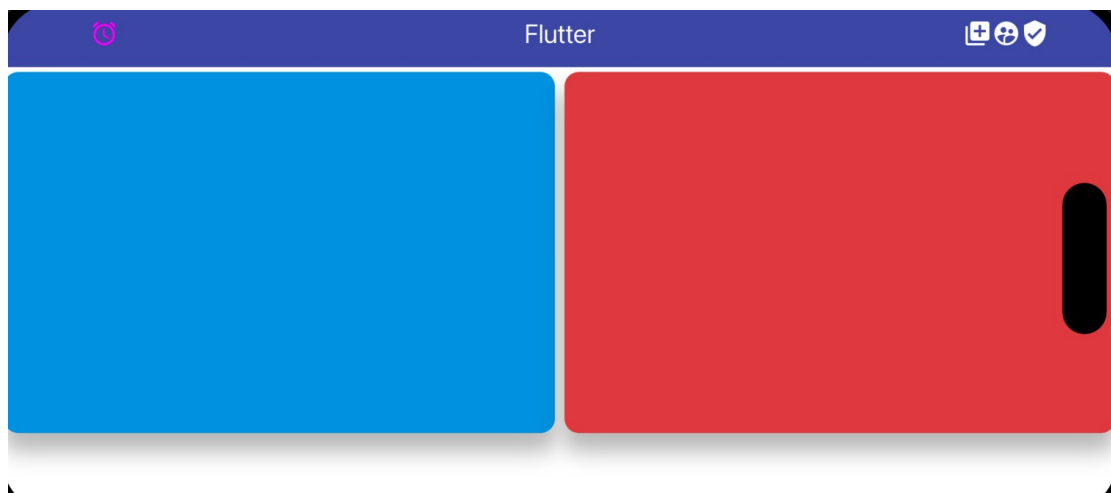
elevation : La propriété elevation de 15.0 définit la taille de l'ombre.

PreferredSize: impose des contraintes de taille spécifiques sur son contenu. Dans ce cas, il n'a pas de contenu mais il est dimensionné avec une largeur et une hauteur de 300.0 pixels.

Les lignes et les colonnes :

Les lignes :

Row: aligne ses enfants horizontalement.



```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(const MainApp());  
}  
  
class MainApp extends StatelessWidget {  
  const MainApp({super.key});  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Demo',  
      home: const Home(),  
      theme: ThemeData(  
        appBarTheme: const AppBarTheme(  
          backgroundColor: Colors.indigo,  
        ),  
      ),  
    );  
  }  
}
```

```

    debugShowCheckedModeBanner: false,
  );
}
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
    return _Home();
  }
}

class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
        centerTitle: true,
      ),
      backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
      body: const Row(children: [
        Expanded(
          child: Card(
            elevation: 15.0,
            color: Colors.lightBlue,
            child: SizedBox(
              width: 300.0,
              height: 300.0,
            ),
          ),
        ),
        Expanded(
          child: Card(
            elevation: 15.0,
            color: Colors.red,
            child: SizedBox(
              width: 300.0,
              height: 300.0,
            ),
          ),
        ),
      ]),
    );
  }
}

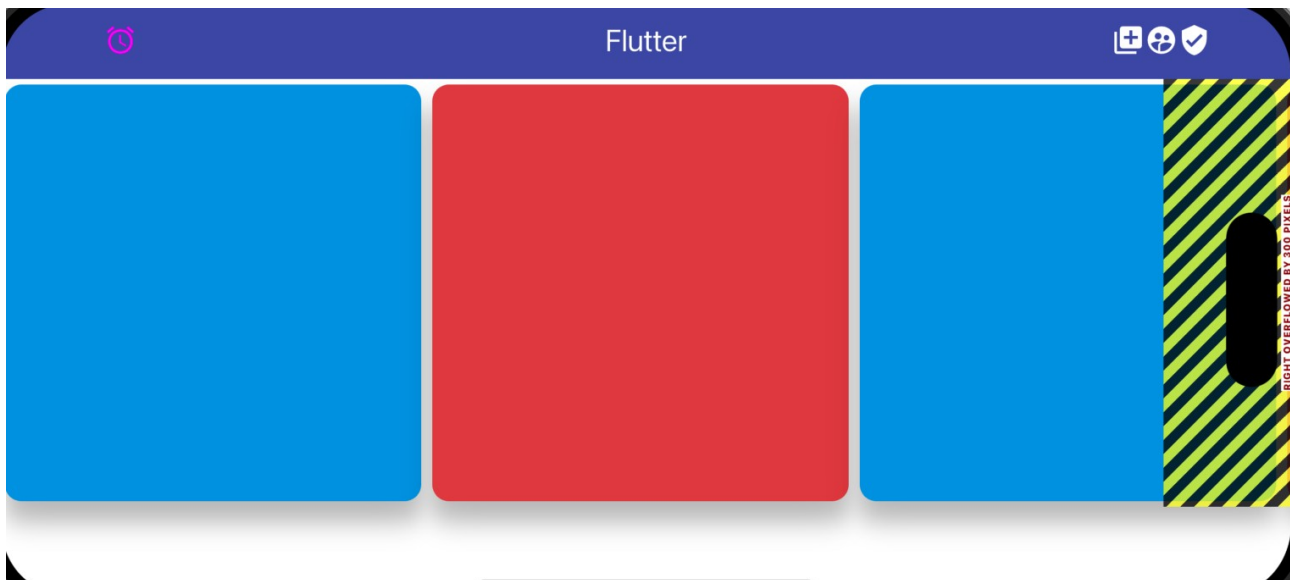
```

```
)),  
  ],  
  );  
}  
}
```

Commentaires :

Expanded: Ce widget est utilisé pour forcer les enfants d'un « Row » (ou Column) à remplir l'espace supplémentaire disponible dans le layout principal. Dans ce cas, chaque Card occupe la moitié de l'espace disponible, car il y a deux widgets Expanded de poids égal.

SizeBox: Bien que « SizeBox » ait des dimensions spécifiques (300.0 x 300.0), ces dimensions ne jouent aucun rôle, car « Expanded » permet à la Card de s'ajuster à la taille disponible (en largeur). La hauteur déclarée n'est pas affectée par « Expanded », car il agit uniquement sur la dimension correspondant à l'axe du « Row » ou « Column ».



```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(const MainApp());  
}  
  
class MainApp extends StatelessWidget {  
  const MainApp({super.key});  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Demo',  
      home: const Home(),  
      theme: ThemeData(  

```

```

        appBarTheme: const AppBarTheme(
          backgroundColor: Colors.indigo,
        ),
      ),
      debugShowCheckedModeBanner: false,
    );
  }
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
    return _Home();
  }
}

class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
        centerTitle: true,
      ),
      backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
      body: const Row(children: [
        Card(
          elevation: 15.0,
          color: Colors.lightBlue,
          child: SizedBox(
            width: 300.0,
            height: 300.0,
          ),
        ),
        Card(
          elevation: 15.0,
          color: Colors.red,
          child: SizedBox(
            width: 300.0,
            height: 300.0,

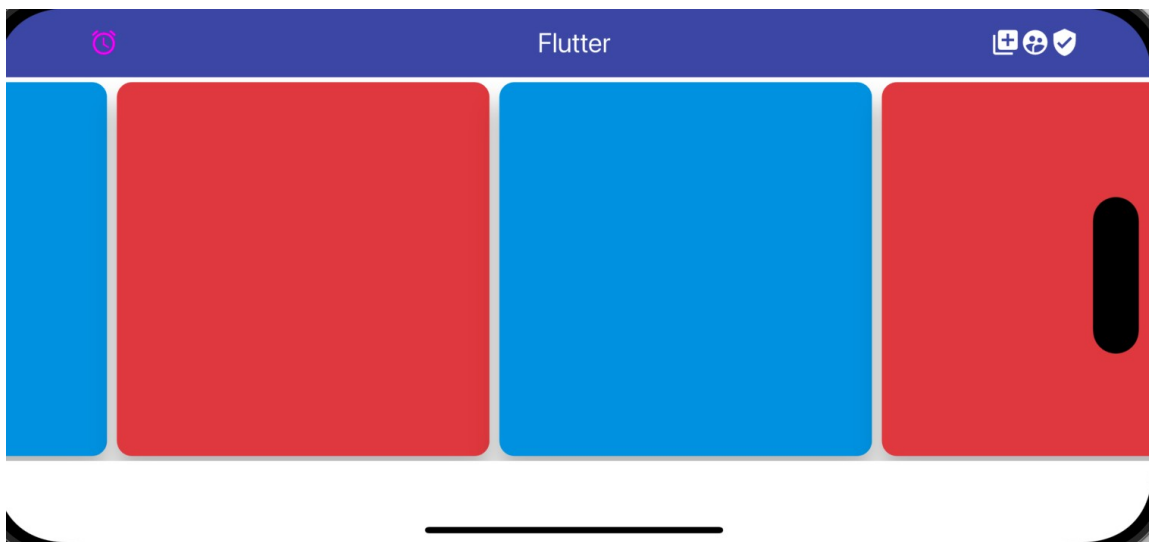
```



```
    ),  
  ),  
  Card(  
    elevation: 15.0,  
    color: Colors.lightBlue,  
    child: SizedBox(  
      width: 300.0,  
      height: 300.0,  
    ),  
  ),  
  Card(  
    elevation: 15.0,  
    color: Colors.red,  
    child: SizedBox(  
      width: 300.0,  
      height: 300.0,  
    ),  
  ),  
],  
);  
}
```

Commentaires :

En retirant les « expanded » et ajoutant d’autres « Card », il y a un problème de débordement. Voici la solution pour éviter ce problème :



```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(const MainApp());  
}
```

```
class MainApp extends StatelessWidget {
  const MainApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      home: const Home(),
      theme: ThemeData(
        appBarTheme: const AppBarTheme(
          backgroundColor: Colors.indigo,
        ),
      ),
      debugShowCheckedModeBanner: false,
    );
  }
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
    return _Home();
  }
}

class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
        centerTitle: true,
      ),
      backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
      body: const SingleChildScrollView(
        scrollDirection: Axis.horizontal,
        child: Row(children: [
          Card(
            elevation: 15.0,
            color: Colors.lightBlue,
```

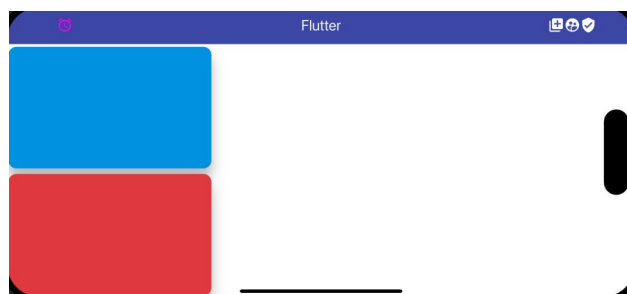
```
        child: SizedBox(
          width: 300.0,
          height: 300.0,
        ),
      ),
      Card(
        elevation: 15.0,
        color: Colors.red,
        child: SizedBox(
          width: 300.0,
          height: 300.0,
        ),
      ),
      Card(
        elevation: 15.0,
        color: Colors.lightBlue,
        child: SizedBox(
          width: 300.0,
          height: 300.0,
        ),
      ),
      Card(
        elevation: 15.0,
        color: Colors.red,
        child: SizedBox(
          width: 300.0,
          height: 300.0,
        ),
      ),
    ],
  ),
);
}
```

Commentaires :

« `scrollDirection` » permet de faire défiler son contenu dans une direction donnée. En utilisant `scrollDirection: Axis.horizontal`, vous indiquez que le défilement doit être horizontal.

Les colonnes :

`Column` : Aligne ses enfants verticalement.



```
import 'package:flutter/material.dart';

void main() {
  runApp(const MainApp());
}

class MainApp extends StatelessWidget {
  const MainApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      home: const Home(),
      theme: ThemeData(
        appBarTheme: const AppBarTheme(
          backgroundColor: Colors.indigo,
        ),
      ),
      debugShowCheckedModeBanner: false,
    );
  }
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
    return _Home();
  }
}

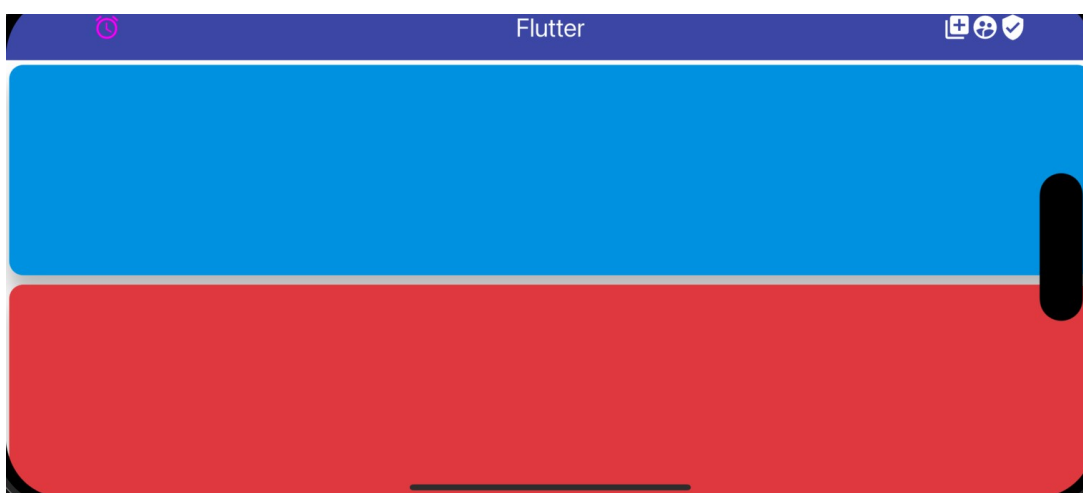
class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
        centerTitle: true,
      ),
```

```
backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
body: const Column(children: [
  Expanded(
    child: Card(
      elevation: 15.0,
      color: Colors.lightBlue,
      child: SizedBox(
        width: 300.0,
        height: 300.0,
      ),
    ),
  ),
  Expanded(
    child: Card(
      elevation: 15.0,
      color: Colors.red,
      child: SizedBox(
        width: 300.0,
        height: 300.0,
      ),
    ),
  ),
]),
);
}
```

Commentaires :

L'espace est partagé sur la hauteur. En revanche, horizontalement, la taille est limitée à 300 pixels.

Etirer le contenu sur l'ensemble de l'espace disponible



```
import 'package:flutter/material.dart';
```

```
void main() {
  runApp(const MainApp());
}

class MainApp extends StatelessWidget {
  const MainApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      home: const Home(),
      theme: ThemeData(
        appBarTheme: const AppBarTheme(
          backgroundColor: Colors.indigo,
        ),
      ),
      debugShowCheckedModeBanner: false,
    );
  }
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
    return _Home();
  }
}

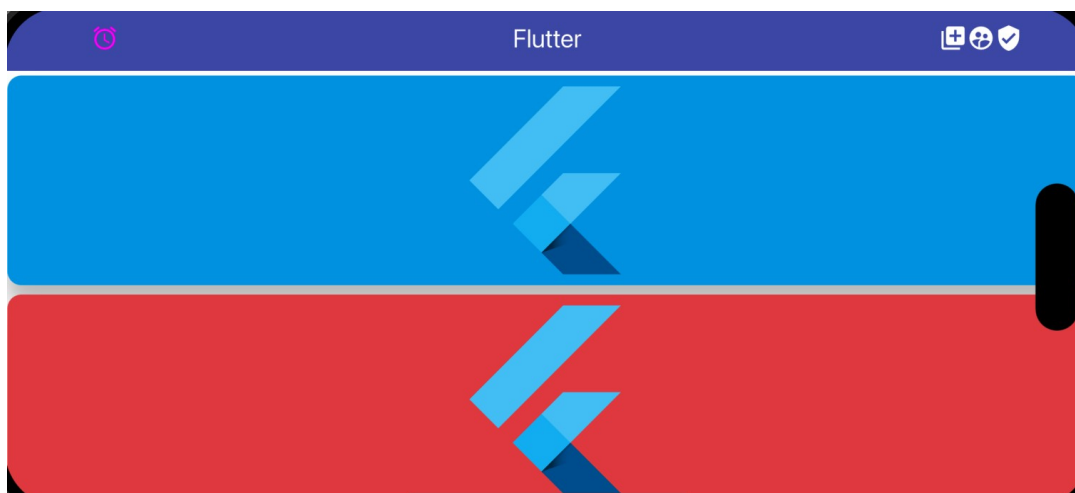
class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
        centerTitle: true,
      ),
      backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
      body: const Column(
        crossAxisAlignment: CrossAxisAlignment.stretch,
        children: [
```

```
Expanded(  
  child: Card(  
    elevation: 15.0,  
    color: Colors.lightBlue,  
    child: SizedBox(  
      width: 300.0,  
      height: 300.0,  
    ),  
  ),  
),  
Expanded(  
  child: Card(  
    elevation: 15.0,  
    color: Colors.red,  
    child: SizedBox(  
      width: 300.0,  
      height: 300.0,  
    ),  
  ),  
)),  
]),  
);  
}  
}
```

Commentaire :

`crossAxisAlignment: CrossAxisAlignment.stretch`, // permet d'étirer la colonne sur la totalité de l'écran.

Affichage d'une image :



Pour afficher une image stockée en local, créer un répertoire dans la racine de votre projet. Par exemple « assets » puis « images ». Mettez-y une image. Dans le répertoire de ce cours, vous trouverez une image.

Dans votre projet, allez dans le fichier pubspec.yaml

```
flutter:  
  uses-material-design: true  
  assets:  
    - assets/images/flutter.png
```

Utilisez Image.network pour afficher une image située sur le réseau.

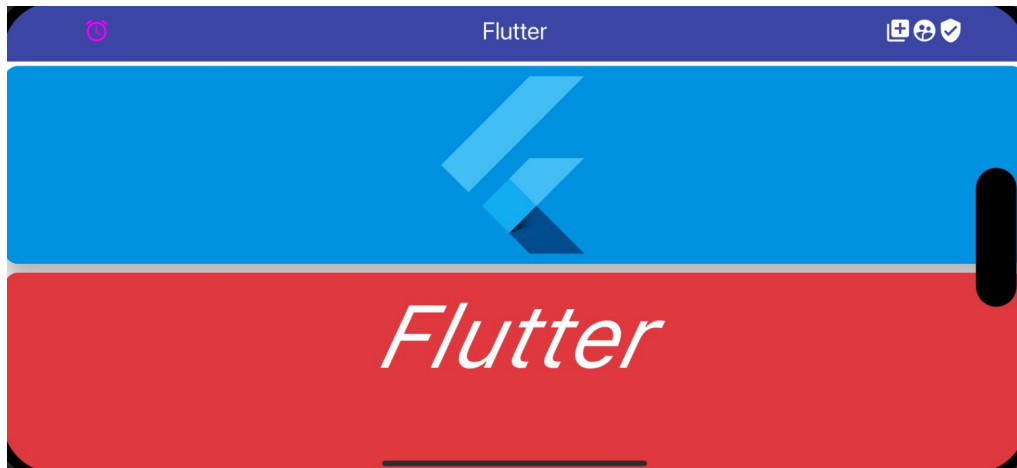
Exemple :

```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(const MainApp());  
}  
  
class MainApp extends StatelessWidget {  
  const MainApp({super.key});  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Demo',  
      home: const Home(),  
      theme: ThemeData(  
        appBarTheme: const AppBarTheme(  
          backgroundColor: Colors.indigo,  
        ),  
      ),  
      debugShowCheckedModeBanner: false,  
    );  
  }  
}  
  
class Home extends StatefulWidget {  
  const Home({super.key});  
  @override  
  State<StatefulWidget> createState() {  
    return _Home();  
  }  
}  
  
class _Home extends State<Home> {  
  @override
```



```
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('Flutter', style: TextStyle(color: Colors.white)),
      leading: const Icon(Icons.access_alarm, color: Color(0xFFFF00FF)),
      actions: const <Widget>[
        Icon(Icons.add_to_photos, color: Colors.white),
        Icon(Icons.supervised_user_circle, color: Colors.white),
        Icon(Icons.verified_user, color: Colors.white),
      ],
      elevation: 10.0,
      centerTitle: true,
    ),
    backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
    body: Column(crossAxisAlignment: CrossAxisAlignment.stretch, children: [
      Expanded(
        child: Card(
          elevation: 15.0,
          color: Colors.lightBlue,
          child: SizedBox(
            width: 300.0,
            height: 300.0,
            child: Image.network(
              "https://s3-4987.nuage-peda.fr/juniexp/DBE302/images/flutter.png"),
          ),
        ),
      ),
      Expanded(
        child: Card(
          elevation: 15.0,
          color: Colors.red,
          child: SizedBox(
            width: 300.0,
            height: 300.0,
            child: Image.asset("assets/images/flutter.png"),
          ),
        ),
      ),
    ]),
  );
}
```

Afficher du texte :



```
import 'package:flutter/material.dart';

void main() {
  runApp(const MainApp());
}

class MainApp extends StatelessWidget {
  const MainApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      home: const Home(),
      theme: ThemeData(
        appBarTheme: const AppBarTheme(
          backgroundColor: Colors.indigo,
        ),
      ),
      debugShowCheckedModeBanner: false,
    );
  }
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
```

```

    return _Home();
  }
}

class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
        centerTitle: true,
      ),
      backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
      body: Column(crossAxisAlignment: CrossAxisAlignment.stretch, children: [
        Expanded(
          child: Card(
            elevation: 15.0,
            color: Colors.lightBlue,
            child: SizedBox(
              child: Image.network(
                "https://s3-4987.nuage-peda.fr/juniexp/DBE302/images/flutter.png"),
            ),
          ),
        ),
        const Expanded(
          child: Card(
            elevation: 15.0,
            color: Colors.red,
            child: SizedBox(
              child: Text(
                "Flutter",
                textAlign: TextAlign.center,
                style: TextStyle(
                  color: Colors.white,
                  fontSize: 80.0,
                  fontStyle: FontStyle.italic,
                ),
                maxLines: 2,
              ),
            ),
          ),
        ),
      ],
    ),
  ),
}

```

```
    )),
  ],
);
}
}import 'package:flutter/material.dart';

void main() {
  runApp(const MainApp());
}

class MainApp extends StatelessWidget {
  const MainApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      home: const Home(),
      theme: ThemeData(
        appBarTheme: const AppBarTheme(
          backgroundColor: Colors.indigo,
        ),
      ),
      debugShowCheckedModeBanner: false,
    );
  }
}

class Home extends StatefulWidget {
  const Home({super.key});
  @override
  State<StatefulWidget> createState() {
    return _Home();
  }
}

class _Home extends State<Home> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Flutter', style: TextStyle(color: Colors.white)),
        leading: const Icon(Icons.access_alarm, color: Color(0xFFFFF00FF)),
        actions: const <Widget>[
          Icon(Icons.add_to_photos, color: Colors.white),
          Icon(Icons.supervised_user_circle, color: Colors.white),
          Icon(Icons.verified_user, color: Colors.white),
        ],
        elevation: 10.0,
```

```
        centerTitle: true,
      ),
      backgroundColor: const Color.fromRGBO(255, 255, 255, 1),
      body: Column(crossAxisAlignment: CrossAxisAlignment.stretch, children: [
        Expanded(
          child: Card(
            elevation: 15.0,
            color: Colors.lightBlue,
            child: Image.network(
              "https://s3-4987.nuage-peda.fr/juniexp/DBE302/images/flutter.png"),
          ),
        ),
        const Expanded(
          child: Card(
            elevation: 15.0,
            color: Colors.red,
            child: Text(
              "Flutter",
              textAlign: TextAlign.center,
              style: TextStyle(
                color: Colors.white,
                fontSize: 80.0,
                fontStyle: FontStyle.italic,
              ),
            ),
            maxLines: 2,
          ),
        ),
      ]),
    );
  }
}
```

Commentaire :

`maxLines: 2` : contrôle le nombre maximum de lignes que le texte peut occuper. Ici, il est défini sur 2, ce qui signifie que si le texte "Flutter" devait déborder, il pourrait s'étendre sur deux lignes avant de tronquer le texte.

Exercice :

En vous appuyant sur cette page, <https://docs.flutter.dev/ui/widgets/basics>, notamment sur le « Text », obtenez le résultat suivant :

