ELSEWEDY UNIVERSITY OF TECHNOLOGY
- POLYTECHNIC OF EGYPT -

CET215: Mobile Application Development

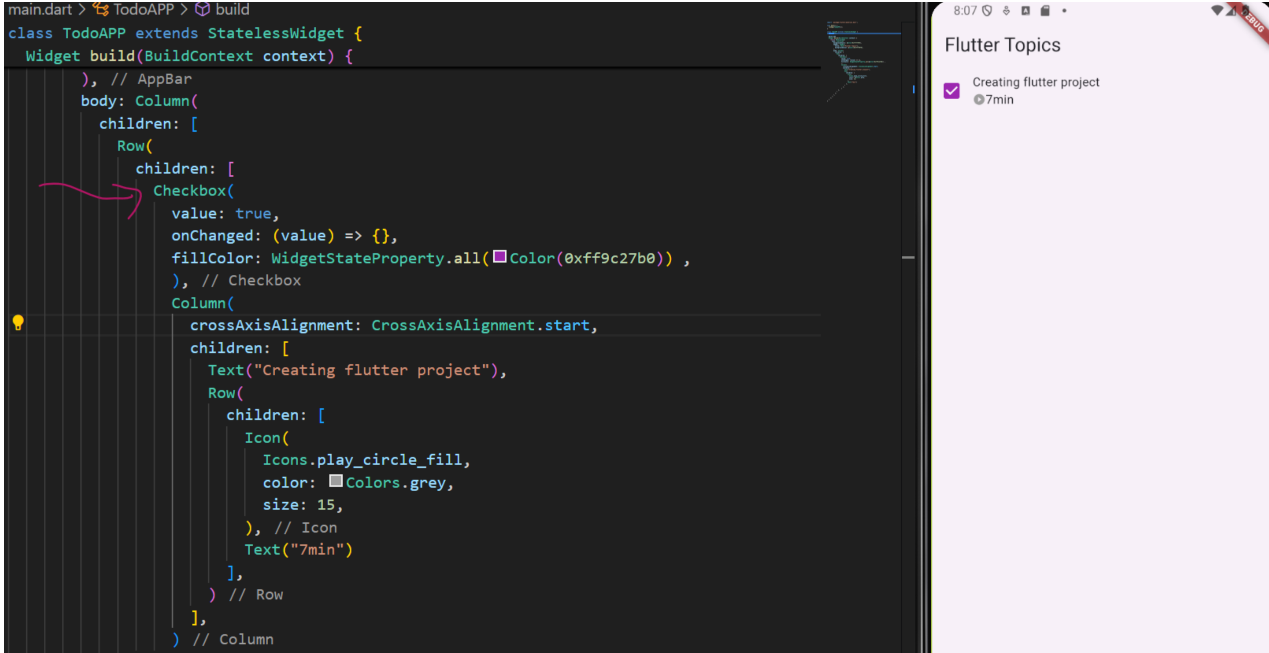
Lecture 4: Tasks APP & Birthday Card APP

© Spring 2025– **Dr. Ahmed Elrefaiy**

Ahmed.elrefai@sut.edu.eg

Checkbox Widget

- supports **custom colors**, **states**, and **interactions**.
- <u>Value:</u> true or false (checked/not checked)
- <u>onChange:</u> callback function executes when user tap checkbox
- <u>fillColor:</u> Apply diff colors with diff scenarios.
- <u>WidgetStateProperty:</u> is a Flutter class that allows **different UI properties** (like color, shape, padding) to **change dynamically based on widget states**.



main.dart > TodoAPP > build

```
class TodoAPP extends StatelessWidget {
  Widget build(BuildContext context) {
    ), // AppBar
    body: Column(
     children: [
      Row(
       children: [
        Checkbox(
         value: true,
         onChanged: (value) => {},
         fillColor: WidgetStateProperty.all(■Color(0xff9c27b0)) ,
         ), // Checkbox
         Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
           Text("Creating flutter project"),
           Row(
            children: [
             Icon(
              Icons.play_circle_fill,
              color: ■Colors.grey,
              size: 15,
             ), // Icon
             Text("7min")
            ],
           ) // Row
          ],
         ) // Column
```

Flutter Topics

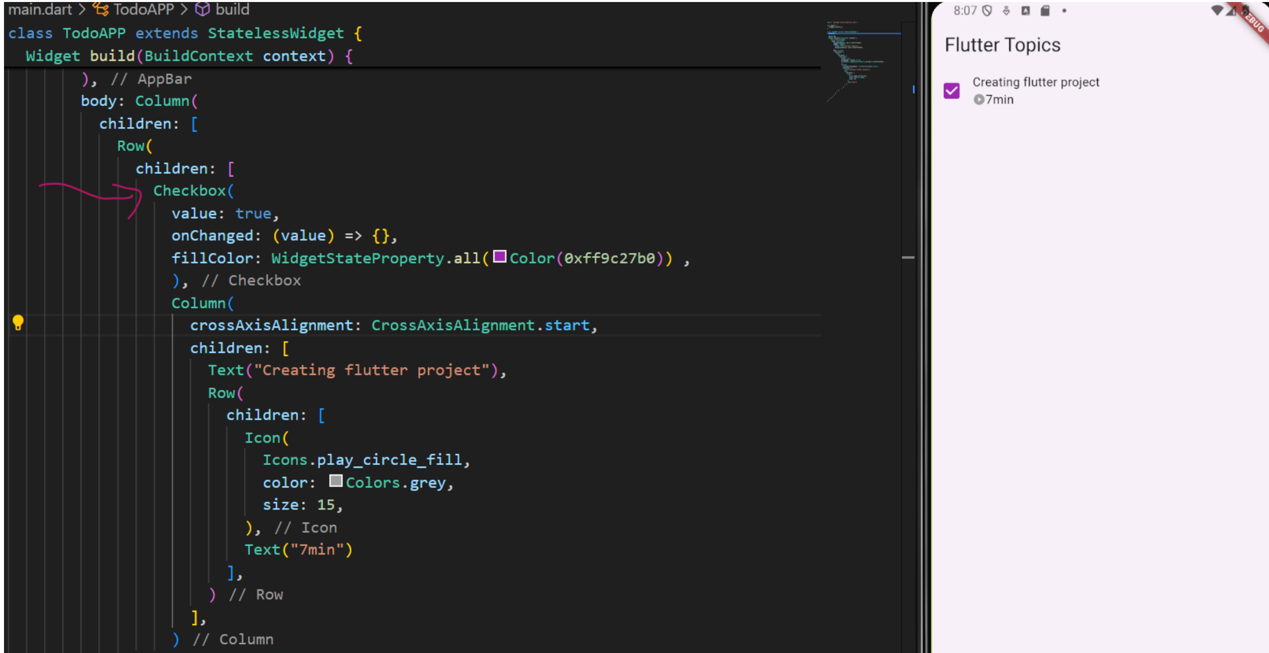
☑ Creating flutter project
7min

WidgetStateProperty

- WidgetStateProperty: is a Flutter class that allows **different UI properties** (like color, shape, padding) to **change dynamically based on widget states**.

- `WidgetStateProperty.all(Color(0xff9c27b0))` → all used to make static color over all states

- **States →** mean states after user interact with app (selected, not selected, hovered, pressed, Focused)



main.dart > TodoAPP > build

```
class TodoAPP extends StatelessWidget {
  Widget build(BuildContext context) {
        ), // AppBar
        body: Column(
          children: [
            Row(
              children: [
                Checkbox(
                  value: true,
                  onChanged: (value) => {},
                  fillColor: WidgetStateProperty.all(Color(0xff9c27b0)) ,
                ), // Checkbox
                Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  children: [
                    Text("Creating flutter project"),
                    Row(
                      children: [
                        Icon(
                          Icons.play_circle_fill,
                          color: Colors.grey,
                          size: 15,
                        ), // Icon
                        Text("7min")
                      ],
                    ) // Row
                  ],
                ) // Column
```

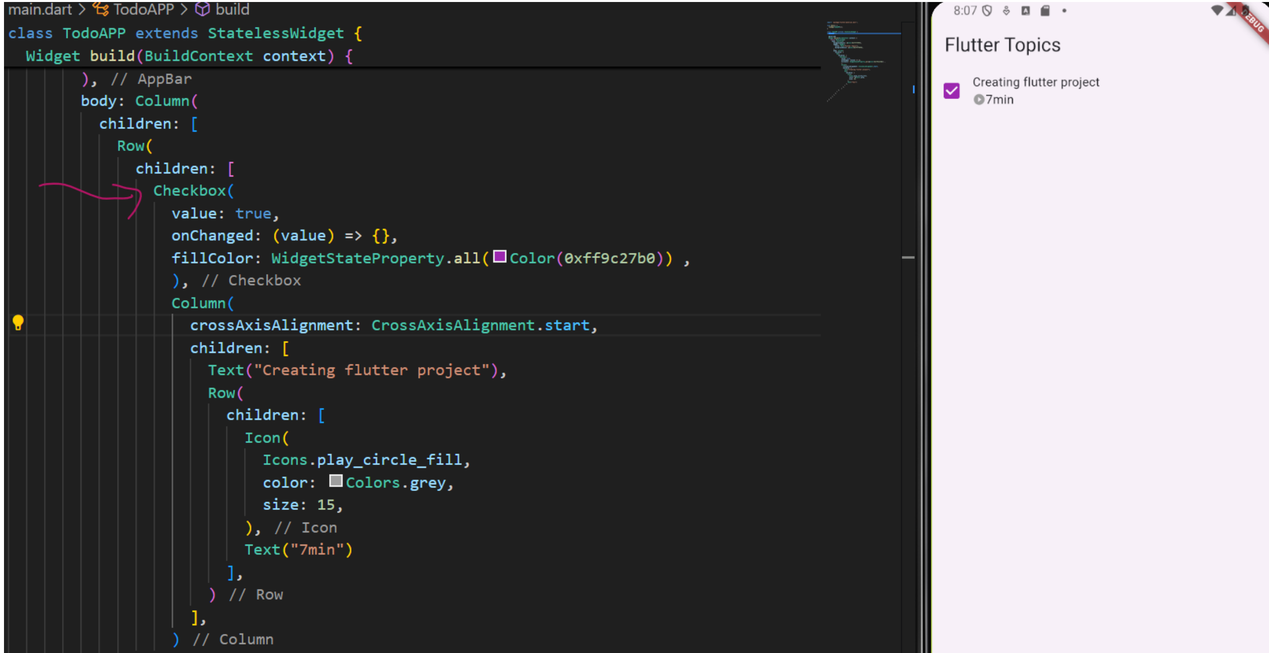
8:07

Flutter Topics

☑ Creating flutter project
7min

Color

- Can be from enumeration like → Colors.red
- Can be from hexa colors like → Color(0xffRRGGBB)



main.dart > TodoAPP > build

```
class TodoAPP extends StatelessWidget {
  Widget build(BuildContext context) {
        ), // AppBar
        body: Column(
          children: [
            Row(
              children: [
                Checkbox(
                  value: true,
                  onChanged: (value) => {},
                  fillColor: WidgetStateProperty.all(Color(0xff9c27b0)) ,
                ), // Checkbox
                Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  children: [
                    Text("Creating flutter project"),
                    Row(
                      children: [
                        Icon(
                          Icons.play_circle_fill,
                          color: Colors.grey,
                          size: 15,
                        ), // Icon
                        Text("7min")
                      ],
                    ) // Row
                  ],
                ) // Column
```

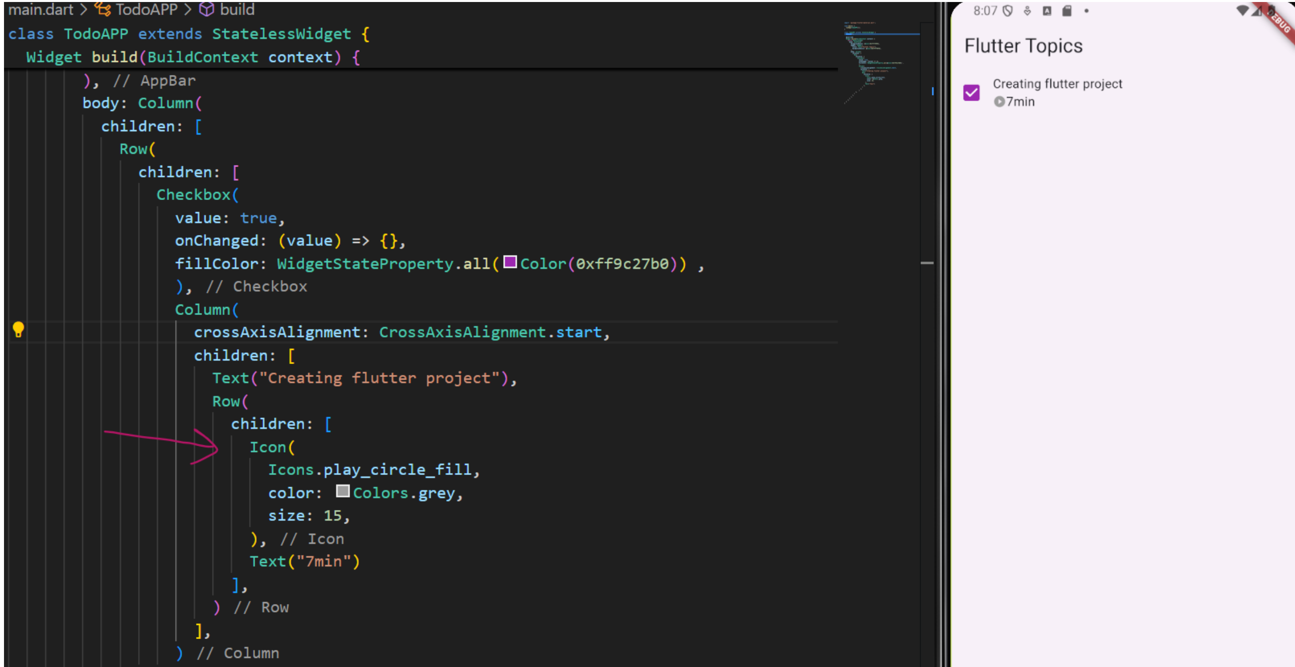
Flutter Topics

☑ Creating flutter project
7min

Icon Widget

- Icon Widget have enumeration Icons to choose one from.

- Followed by other parameters such as → Color, size



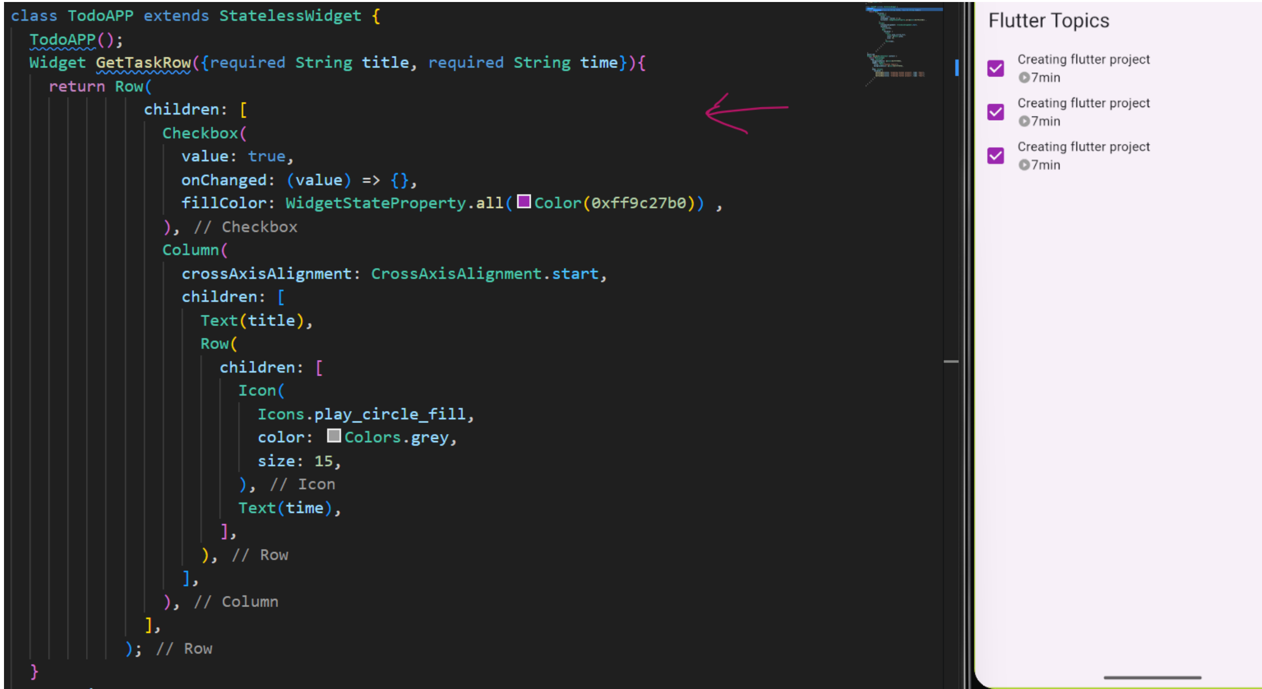
main.dart > TodoAPP > build

```
class TodoAPP extends StatelessWidget {
  Widget build(BuildContext context) {
        ), // AppBar
        body: Column(
          children: [
            Row(
              children: [
                Checkbox(
                  value: true,
                  onChanged: (value) => {},
                  fillColor: WidgetStateProperty.all(Color(0xff9c27b0)) ,
                  ), // Checkbox
                  Column(
                    crossAxisAlignment: CrossAxisAlignment.start,
                    children: [
                      Text("Creating flutter project"),
                      Row(
                        children: [
                          Icon(
                            Icons.play_circle_fill,
                            color: Colors.grey,
                            size: 15,
                          ), // Icon
                          Text("7min")
                        ],
                      ) // Row
                    ],
                  ) // Column
```

How to handle many rows of tasks?

- Instead of repeat each row many time, we create function that return row as widget.

- So, we can call it many times below in an organized manner.
  See the next slide



```
class TodoAPP extends StatelessWidget {
 TodoAPP();
 Widget GetTaskRow({required String title, required String time}){
  return Row(
       children: [
        Checkbox(
         value: true,
         onChanged: (value) => {},
         fillColor: WidgetStateProperty.all(■Color(0xff9c27b0)) ,
        ), // Checkbox
        Column(
         crossAxisAlignment: CrossAxisAlignment.start,
         children: [
          Text(title),
          Row(
           children: [
            Icon(
             Icons.play_circle_fill,
             color: ■Colors.grey,
             size: 15,
            ), // Icon
            Text(time),
           ],
          ), // Row
         ],
        ), // Column
       ],
      ); // Row
 }
```

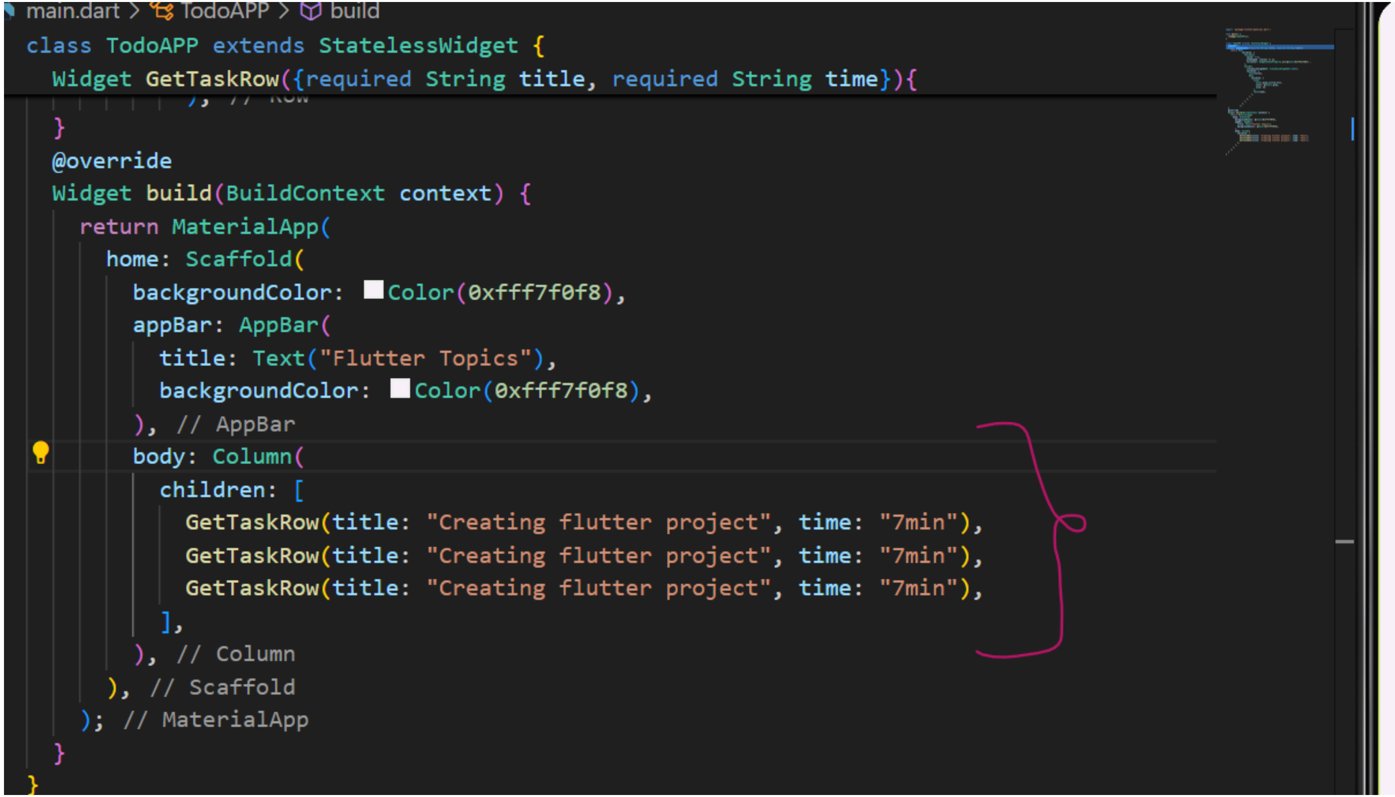
Flutter Topics

☑ Creating flutter project
▶7min

☑ Creating flutter project
▶7min

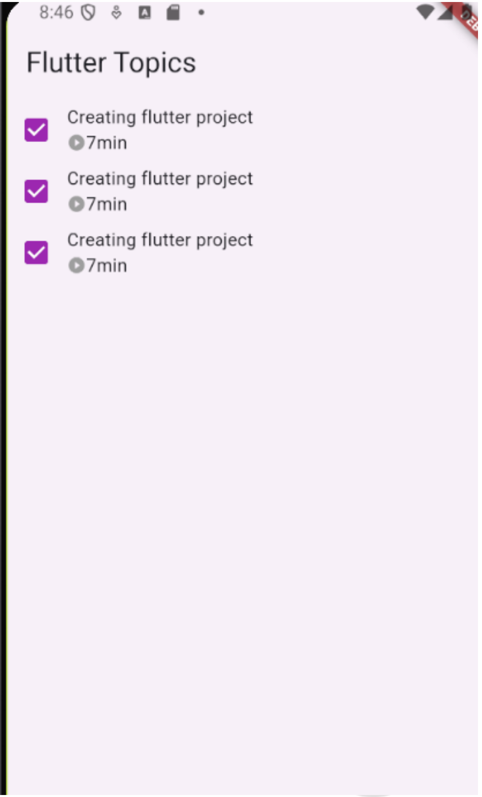
☑ Creating flutter project
▶7min

How to handle many rows of tasks?



main.dart > TodoAPP > build

```
class TodoAPP extends StatelessWidget {
  Widget GetTaskRow({required String title, required String time}){
          ); // Row
  }
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        backgroundColor: Color(0xfff7f0f8),
        appBar: AppBar(
          title: Text("Flutter Topics"),
          backgroundColor: Color(0xfff7f0f8),
        ), // AppBar
        body: Column(
          children: [
            GetTaskRow(title: "Creating flutter project", time: "7min"),
            GetTaskRow(title: "Creating flutter project", time: "7min"),
            GetTaskRow(title: "Creating flutter project", time: "7min"),
          ],
        ), // Column
      ), // Scaffold
    ); // MaterialApp
  }
}
```



Flutter Topics

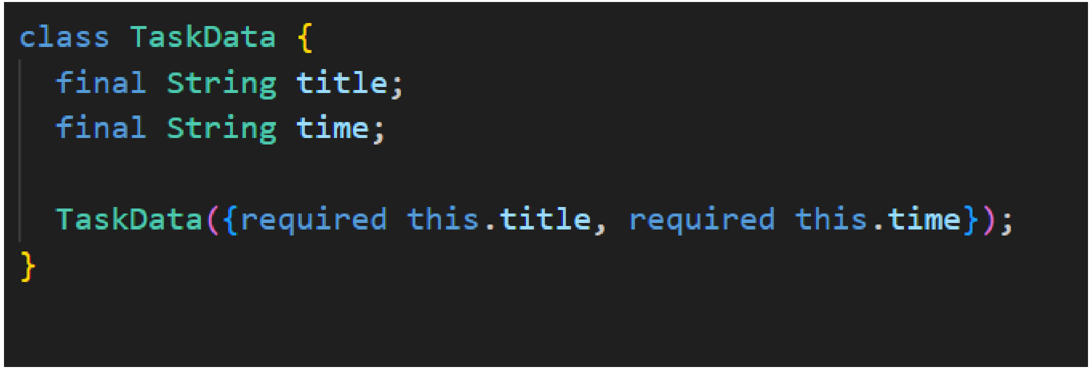
☑ Creating flutter project
7min

☑ Creating flutter project
7min

☑ Creating flutter project
7min

More Organization of code

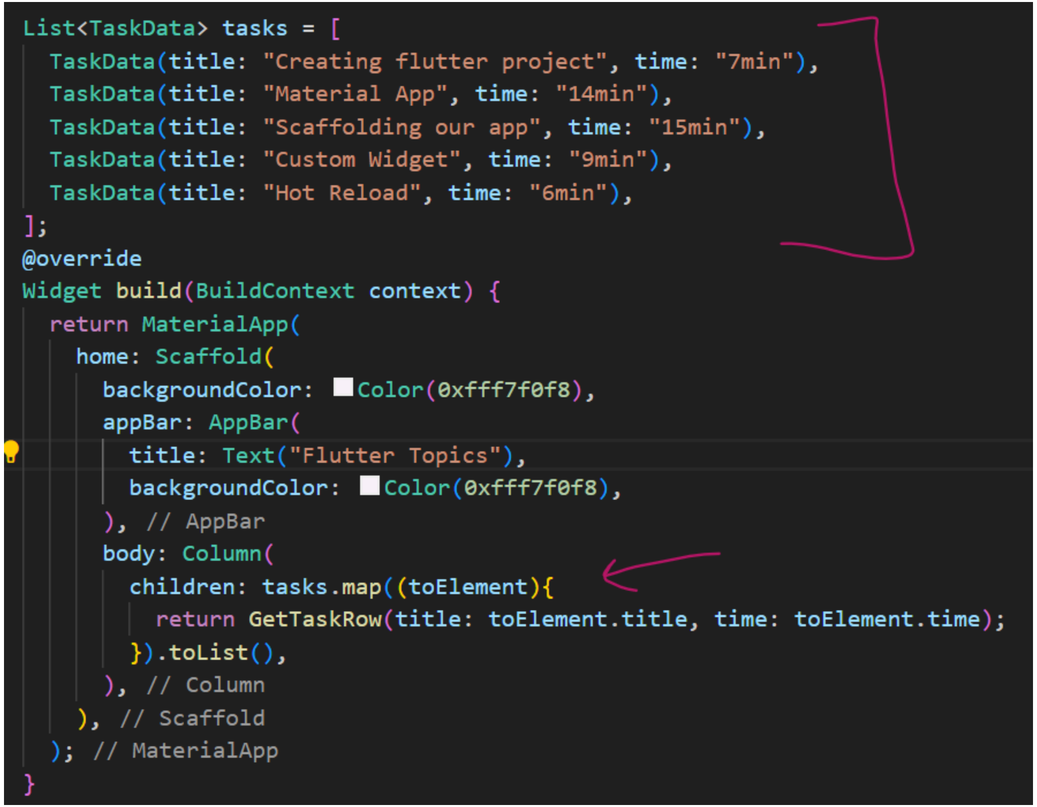
- Create TaskData class to hold all tasks as list.
- final → make values cannot be changed after the object is created, called (immutable).
- Assignment done in Shorthand version of constructor.



```
class TaskData {
  final String title;
  final String time;

  TaskData({required this.title, required this.time});
}
```

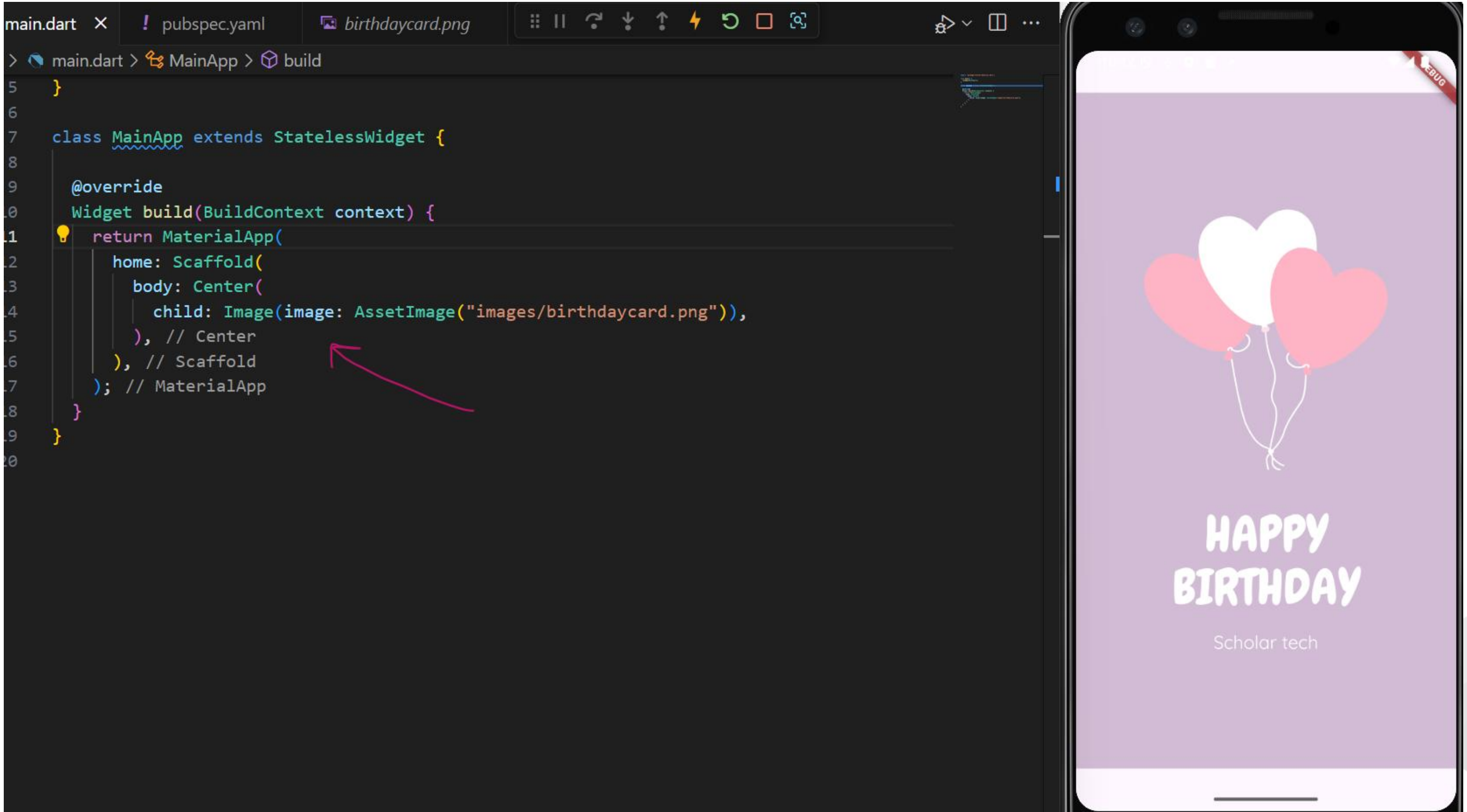
More Organization of code

- Create List of TaskData to hold all tasks
- Pass it to **children** using **map**



```
List<TaskData> tasks = [
  TaskData(title: "Creating flutter project", time: "7min"),
  TaskData(title: "Material App", time: "14min"),
  TaskData(title: "Scaffolding our app", time: "15min"),
  TaskData(title: "Custom Widget", time: "9min"),
  TaskData(title: "Hot Reload", time: "6min"),
];
@override
Widget build(BuildContext context) {
  return MaterialApp(
    home: Scaffold(
      backgroundColor: Color(0xfff7f0f8),
      appBar: AppBar(
        title: Text("Flutter Topics"),
        backgroundColor: Color(0xfff7f0f8),
      ), // AppBar
      body: Column(
        children: tasks.map((toElement){
          return GetTaskRow(title: toElement.title, time: toElement.time);
        }).toList(),
      ), // Column
    ), // Scaffold
  ); // MaterialApp
}
```

Birthday Card APP

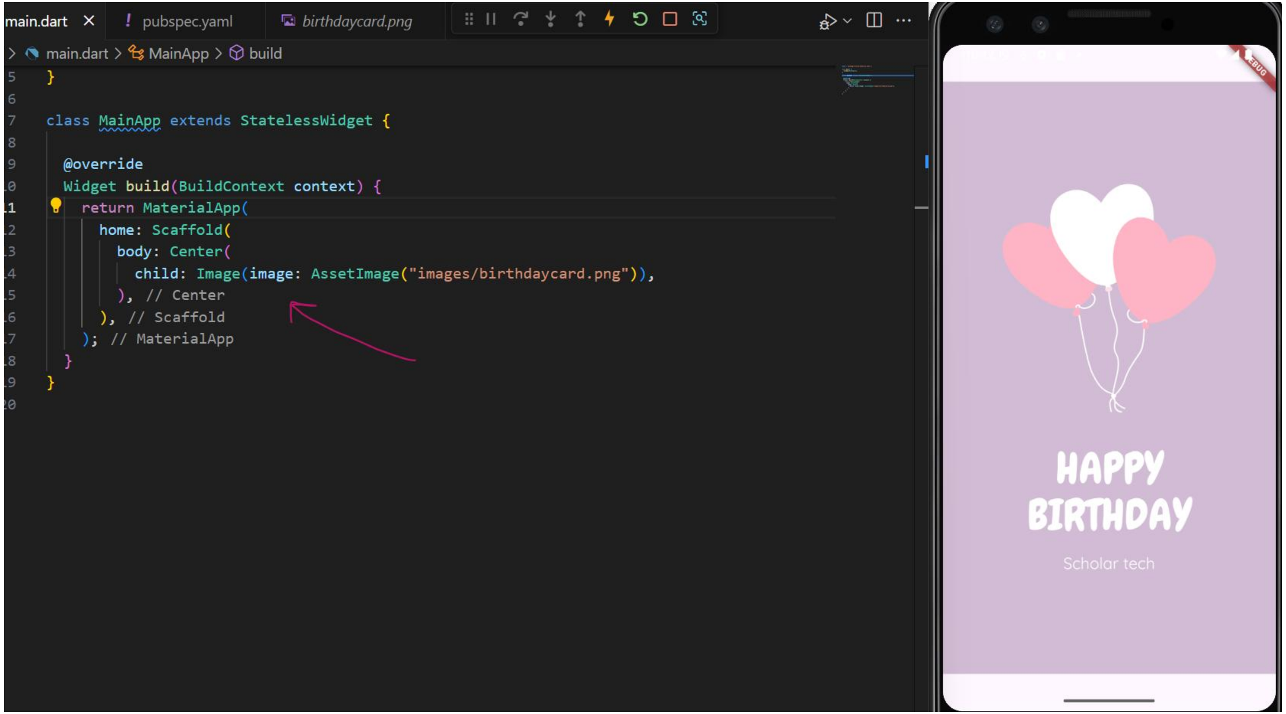


```
}

class MainApp extends StatelessWidget {

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        body: Center(
          child: Image(image: AssetImage("images/birthdaycard.png")),
        ), // Center
      ), // Scaffold
    ); // MaterialApp
  }
}
```

Image Widget

- Use **Image Widget** → it take image provider which is **AssetImage** (image from asset folder), or **NetworkImage** (image from network)

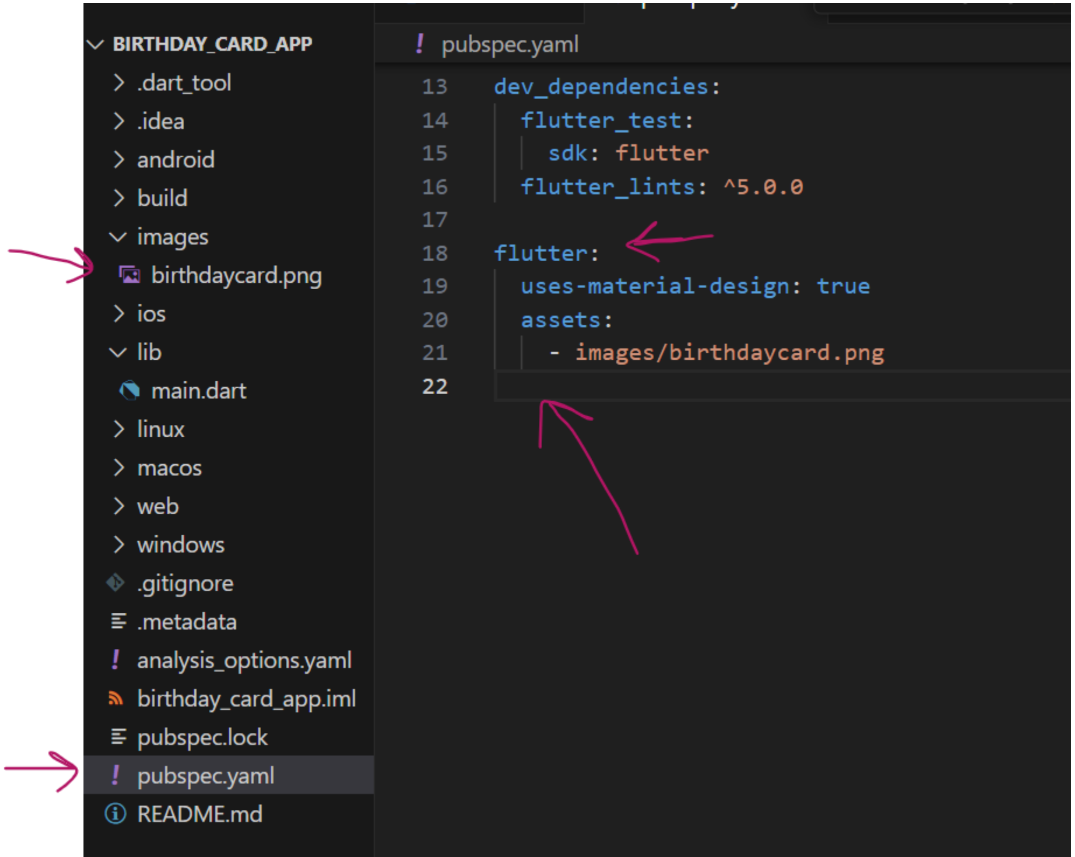
- You must save the image in the project files, and configure it in **pubspec.yaml** file



```dart
}

class MainApp extends StatelessWidget {

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        body: Center(
          child: Image(image: AssetImage("images/birthdaycard.png")),
        ), // Center
      ), // Scaffold
    ); // MaterialApp
  }
}
```

Image Widget

- Put your image inside new folder, here we create **images folder.**
- Configure assets inside **flutter** section with your image.
- **Pubspec file** use indentation to write these configuration



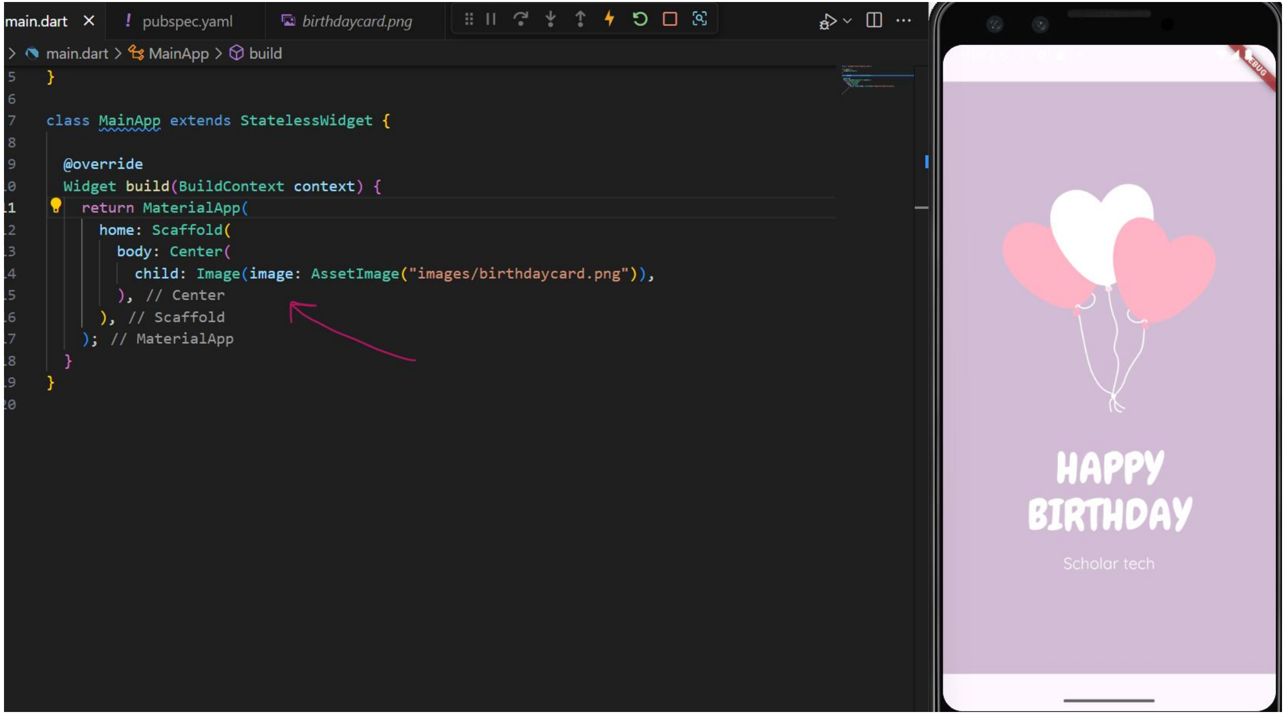
```yaml
dev_dependencies:
  flutter_test:
    sdk: flutter
  flutter_lints: ^5.0.0

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

```

Image Widget

- Now, image appears.

- But, how to color the whole screen?

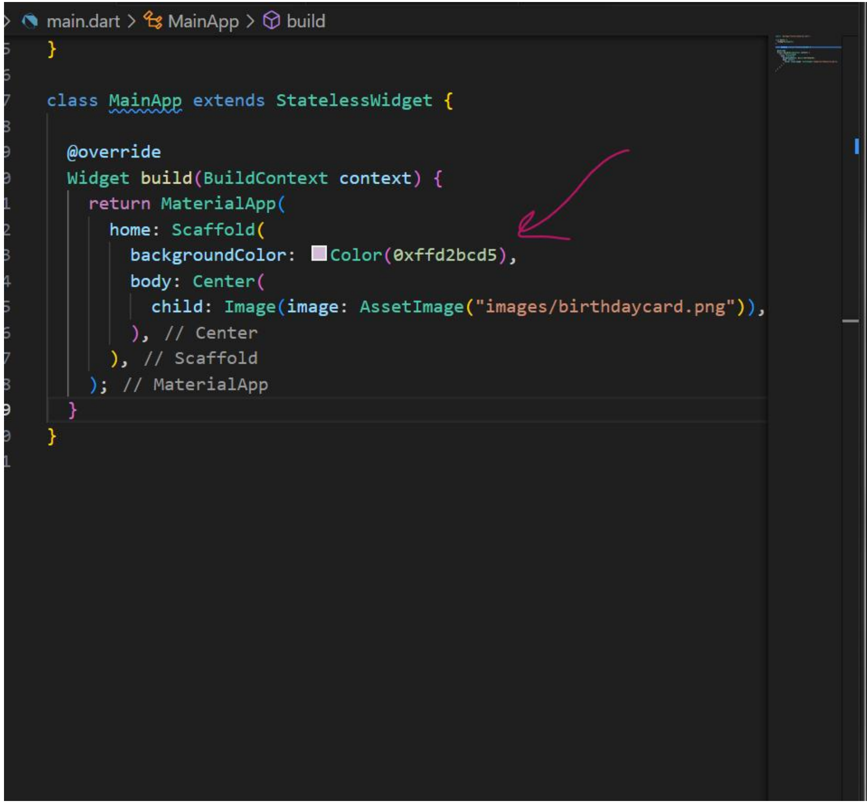
- use **Just color picker** to pick color, and set it for the scaffold directly.



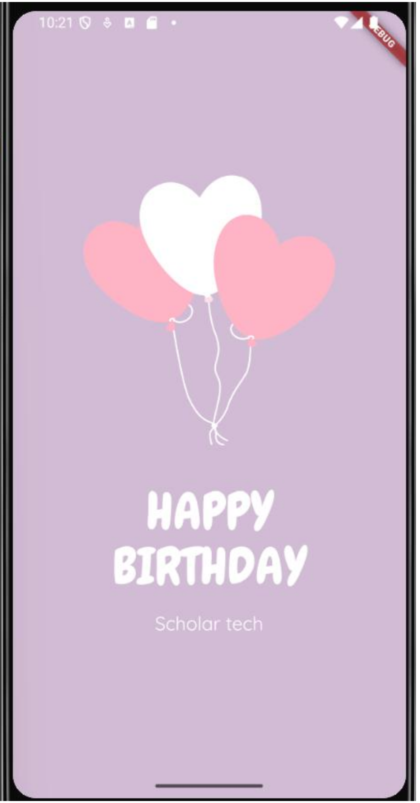
```
}

class MainApp extends StatelessWidget {

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        body: Center(
          child: Image(image: AssetImage("images/birthdaycard.png")),
        ), // Center
      ), // Scaffold
    ); // MaterialApp
  }
}
```

Image Widget

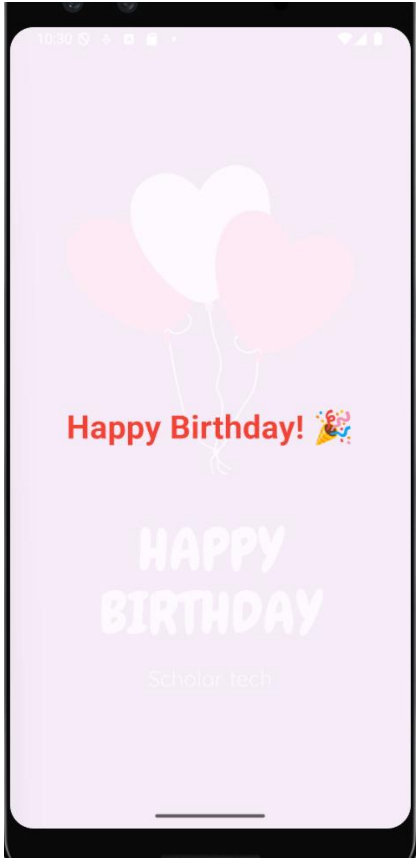
- Now, image appears.
- But, how to color the whole screen?
- use **Just color picker** to pick color, and set it for the scaffold directly.



```
}

class MainApp extends StatelessWidget {

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        backgroundColor: Color(0xffd2bcd5),
        body: Center(
          child: Image(image: AssetImage("images/birthdaycard.png")),
        ), // Center
      ), // Scaffold
    ); // MaterialApp
  }
}
```



Practice1: make background image with opacity and text above it



Happy Birthday! 🎉

Practice2: Show image in the center, and make it circle 200*200