

# Input Widget

Salah satu bentuk interaksi dengan pengguna adalah dengan menerima input. Ada beberapa input widget yang bisa digunakan supaya pengguna bisa berinteraksi dengan aplikasi. Perhatikan bahwa input pengguna ini berkaitan dengan state yang dapat sering berubah. Karena itu umumnya input widget akan ditempatkan di dalam `StatefulWidget`.

## TextField

TextField merupakan sebuah widget yang digunakan untuk menerima input berupa teks yang berasal dari *keyboard*. Terdapat beberapa cara yang bisa Anda gunakan untuk mendapatkan nilai dari TextField. Salah satunya adalah melalui parameter `onChanged`.

```
1. String _name = '';
2.
3. TextField(
4.   onChanged: (String value) {
5.     setState(() {
6.       _name = value;
7.     });
8.   },
9. )
```

Parameter `onChanged` berisi sebuah fungsi yang akan dipanggil setiap terjadi perubahan inputan pada TextField. Pada fungsi ini, kita dapat mengubah nilai variabel state dengan memanggil fungsi `setState()`.

Jika Anda tidak ingin mengambil nilai setiap perubahan, tetapi hanya ketika seluruh input sudah selesai di-*submit*, Anda dapat menggunakan parameter `onSubmitted` seperti berikut:

```
1. String _name = '';
2.
3. TextField(
4.   onSubmitted: (String value) {
5.     setState(() {
6.       _name = value;
7.     });
8.   },
9. )
```

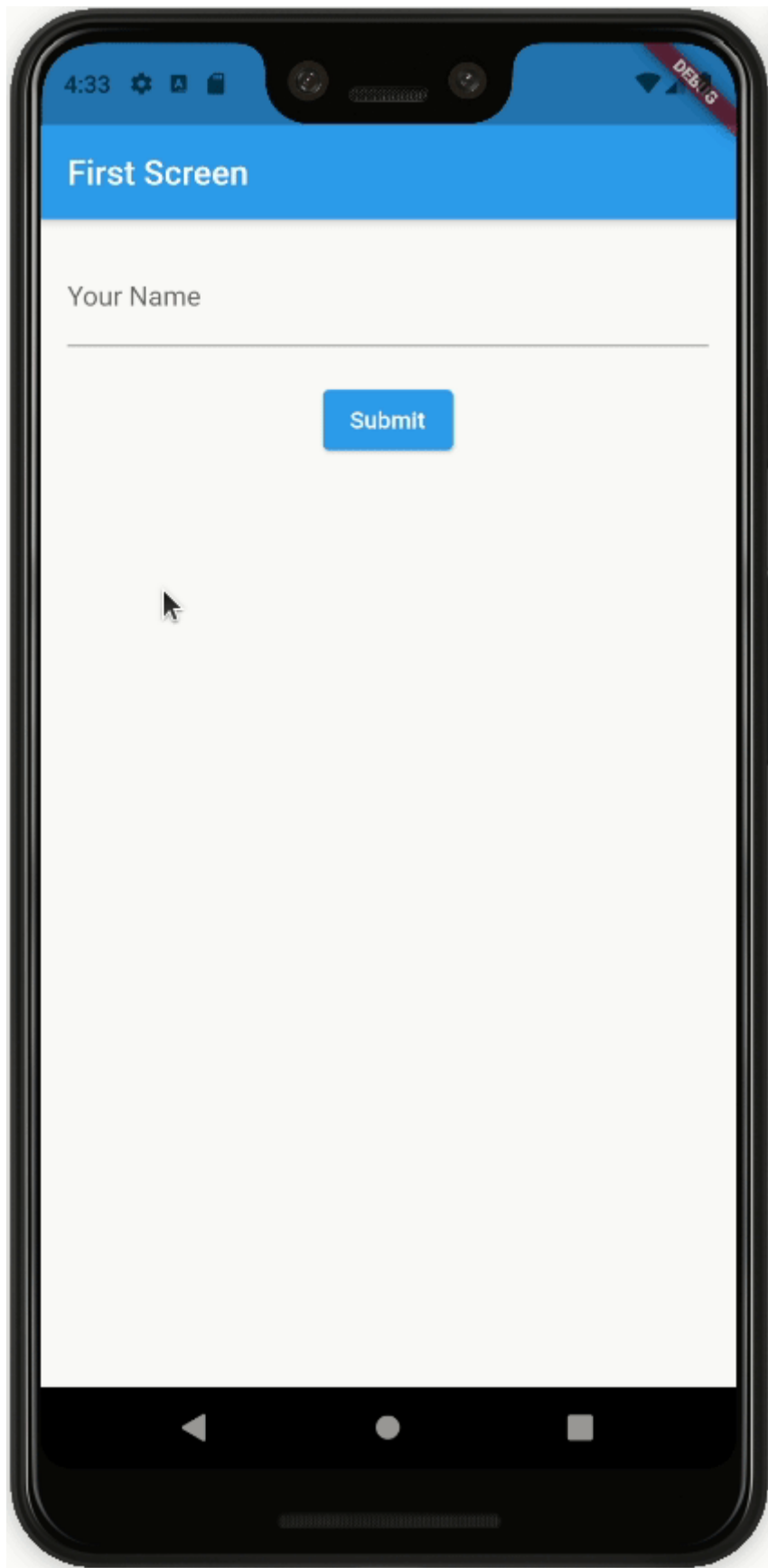
Cara lain yang bisa kita gunakan adalah dengan `TextEditingController`. Dengan controller, kita cukup membuat variabel `TextEditingController` lalu menambahkannya ke widget `TextField`.

```
1. TextEditingController _controller = TextEditingController();  
2.  
3. TextField(  
4.   controller: _controller,  
5. ),
```

Ketika menggunakan controller, pastikan untuk menghapus controller ketika halaman atau widget sudah tidak digunakan. Ini bertujuan supaya tidak menimbulkan kebocoran memori (*memory leak*).

```
1. @override  
2. void dispose() {  
3.   _controller.dispose();  
4.   super.dispose();  
5. }
```

Berikut ini adalah contoh penerapan widget `TextField`:



Untuk membuat TextField seperti di atas, Anda bisa menggunakan kode seperti berikut:

- [onChanged](#)

```
1. class _FirstScreenState extends State<FirstScreen> {
2.   String _name = '';
3.
4.   @override
5.   Widget build(BuildContext context) {
6.     return Scaffold(
7.       appBar: AppBar(
8.         title: Text('First Screen'),
9.       ),
10.      body: Padding(
11.        padding: const EdgeInsets.all(16.0),
12.        child: Column(
13.          children: [
14.            TextField(
15.              decoration: InputDecoration(
16.                hintText: 'Write your name here...',
17.                labelText: 'Your Name',
18.              ),
19.              onChanged: (String value) {
20.                setState(() {
21.                  _name = value;
22.                });
23.              },
24.            ),
25.            SizedBox(height: 20),
26.            ElevatedButton(
27.              onPressed: () {
28.                showDialog(
29.                  context: context,
30.                  builder: (context) {
31.                    return AlertDialog(
32.                      content: Text('Hello, $_name'),
33.                    );
34.                  });
35.              },
36.              child: Text('Submit'),
37.            ),
38.          ],
39.        ),
40.      ),
41.    );
42.  }
43. }
```

### [Controller](#)

```
1. class _FirstScreenState extends State<FirstScreen> {
```

```

2.   TextEditingController _controller = TextEditingController();
3.
4.   @override
5.   Widget build(BuildContext context) {
6.     return Scaffold(
7.       appBar: AppBar(
8.         title: Text('First Screen'),
9.       ),
10.      body: Padding(
11.        padding: const EdgeInsets.all(16.0),
12.        child: Column(
13.          children: [
14.            TextField(
15.              controller: _controller,
16.              decoration: InputDecoration(
17.                hintText: 'Write your name here...',
18.                labelText: 'Your Name',
19.              ),
20.            ),
21.            SizedBox(height: 20),
22.            ElevatedButton(
23.              onPressed: () {
24.                showDialog(
25.                  context: context,
26.                  builder: (context) {
27.                    return AlertDialog(
28.                      content: Text('Hello,
29.                        ${_controller.text}'),
30.                    );
31.                  });
32.                child: Text('Submit'),
33.              )
34.            ],
35.          ),
36.        ),
37.      );
38.    }
39.
40.    @override
41.    void dispose() {
42.      _controller.dispose();
43.      super.dispose();
44.    }
45.  }

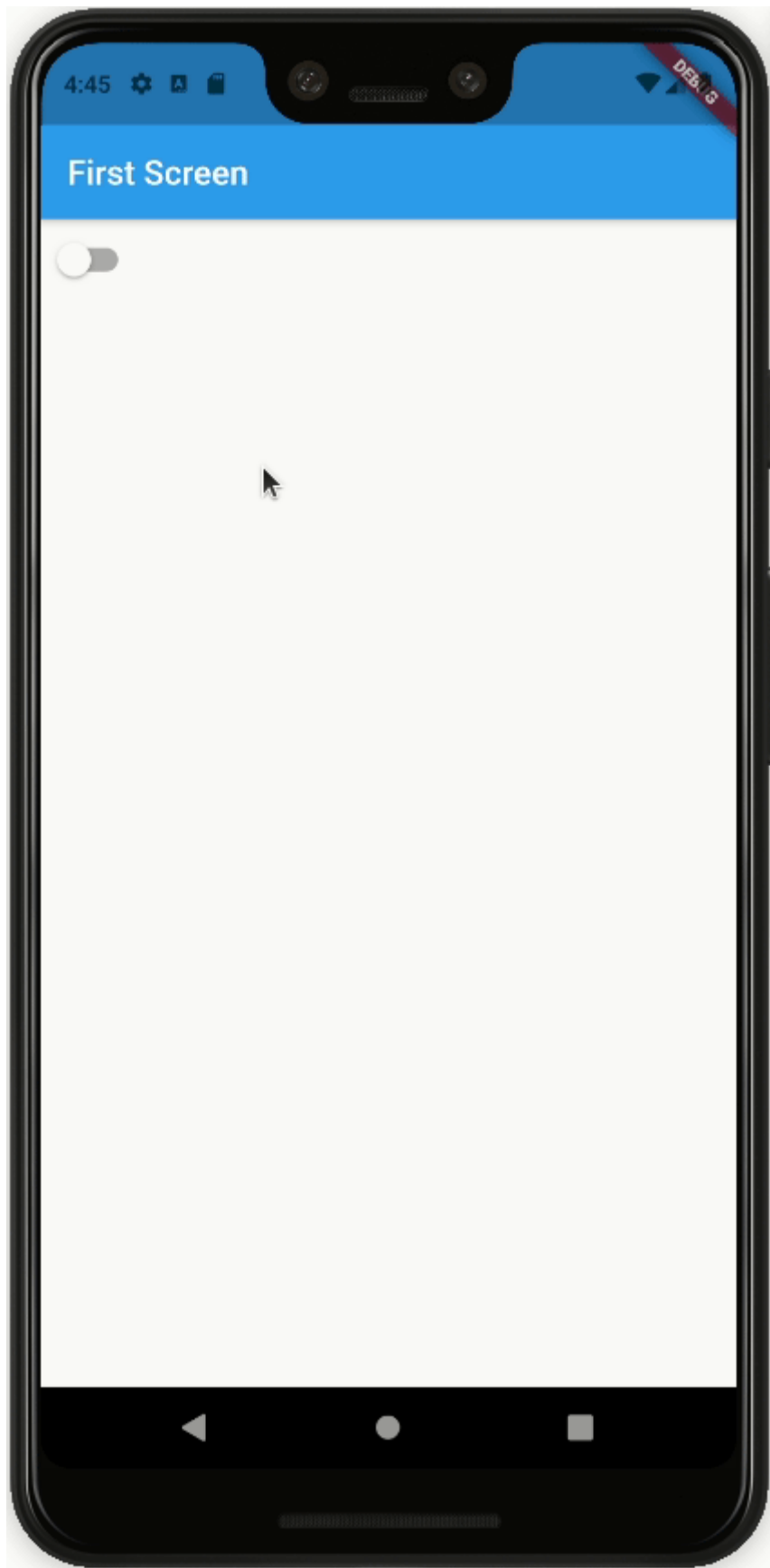
```

## Switch

Switch merupakan inputan yang mengembalikan nilai boolean true atau off. Perhatikan contoh berikut:

```
1. class _FirstScreenState extends State<FirstScreen> {  
2.   bool lightOn = false;  
3.  
4.   @override  
5.   Widget build(BuildContext context) {  
6.     return Scaffold(  
7.       appBar: AppBar(  
8.         title: Text('First Screen'),  
9.       ),  
10.      body: Switch(  
11.        value: lightOn,  
12.        onChanged: (bool value) {  
13.          setState(() {  
14.            lightOn = value;  
15.          });  
16.  
17.          ScaffoldMessenger.of(context).showSnackBar(  
18.            SnackBar(  
19.              content: Text(lightOn ? 'Light On' : 'Light Off'),  
20.              duration: Duration(seconds: 1),  
21.            ),  
22.          );  
23.        },  
24.      ),  
25.    );  
26.  }  
27. }
```

Pada contoh tersebut value dari Switch berupa boolean di mana ketika boolean tersebut false maka Switch akan berada pada posisi nonaktif. Switch umumnya digunakan sebagai konfigurasi on/off pada halaman setting.



## Radio

Radio merupakan inputan yang digunakan untuk memilih salah satu dari beberapa pilihan dalam suatu kelompok. Berikut contohnya:

```

1. class _FirstScreenState extends State<FirstScreen> {
2.   String language;
3.
4.   @override
5.   Widget build(BuildContext context) {
6.     return Scaffold(
7.       appBar: AppBar(
8.         title: Text('First Screen'),
9.       ),
10.      body: Column(
11.        mainAxisAlignment: MainAxisAlignment.min,
12.        children: <Widget>[
13.          ListTile(
14.            leading: Radio<String>(
15.              value: 'Dart',
16.              groupValue: language,
17.              onChanged: (String value) {
18.                setState(() {
19.                  language = value;
20.                  showSnackBar();
21.                });
22.              },
23.            ),
24.            title: Text('Dart'),
25.          ),
26.          ListTile(
27.            leading: Radio<String>(
28.              value: 'Kotlin',
29.              groupValue: language,
30.              onChanged: (String value) {
31.                setState(() {
32.                  language = value;
33.                  showSnackBar();
34.                });
35.              },
36.            ),
37.            title: Text('Kotlin'),
38.          ),
39.          ListTile(
40.            leading: Radio<String>(
41.              value: 'Swift',
42.              groupValue: language,
43.              onChanged: (String value) {
44.                setState(() {
45.                  language = value;
46.                  showSnackBar();
47.                });

```

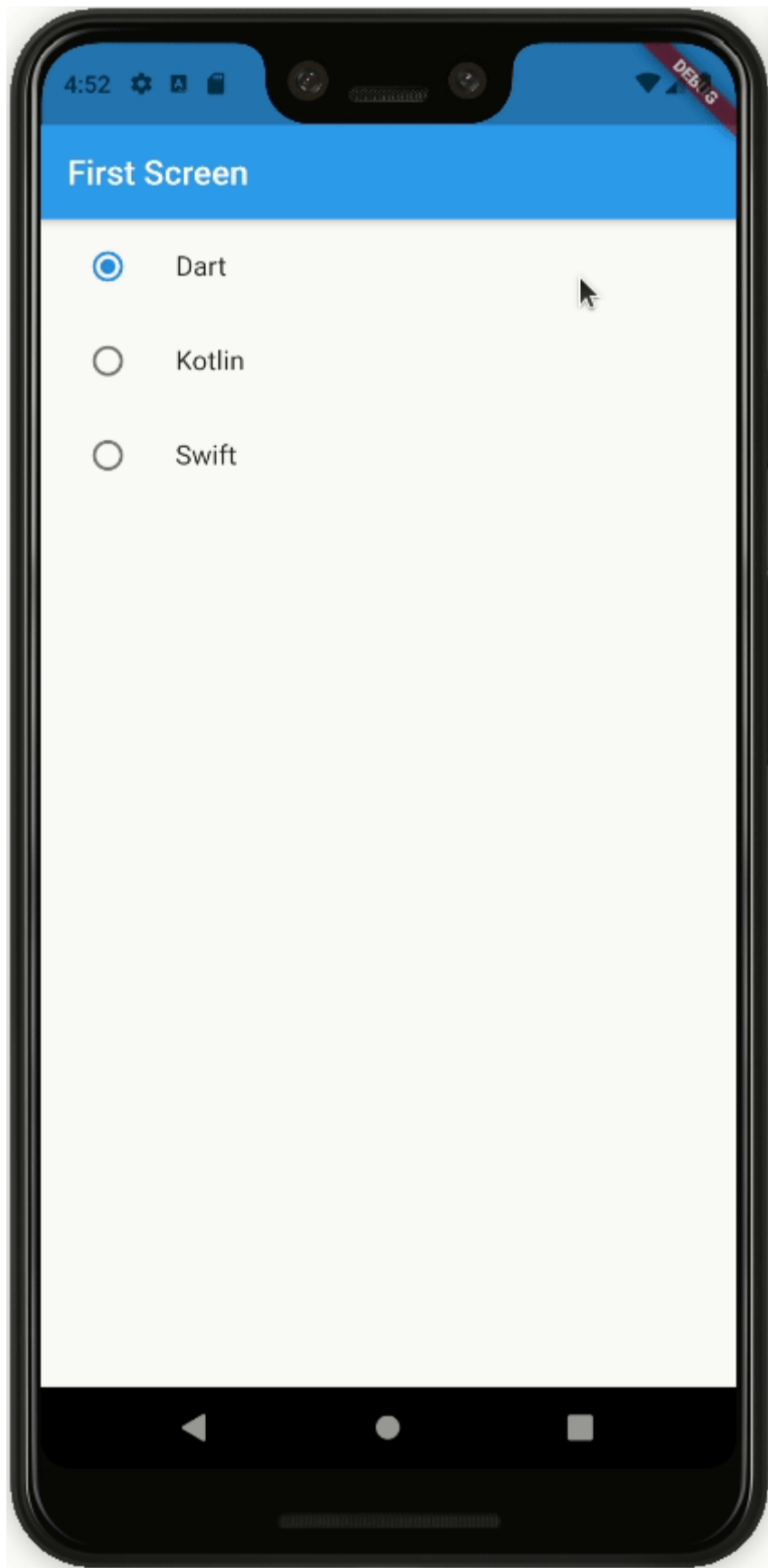


```

48.         },
49.     ),
50.     title: Text('Swift'),
51. ),
52. ],
53. ),
54. );
55. }
56.
57. void showSnackBar() {
58.     ScaffoldMessenger.of(context).showSnackBar(
59.         SnackBar(
60.             content: Text('$language selected'),
61.             duration: Duration(seconds: 1),
62.         ),
63.     );
64. }
65. }

```

Pada contoh tersebut terdapat variable `language` yang digunakan pada `groupValue` tiap Radio. Language inilah yang menyimpan nilai Radio yang dipilih. Nilainya akan berubah ketika fungsi `onChanged` terpanggil.

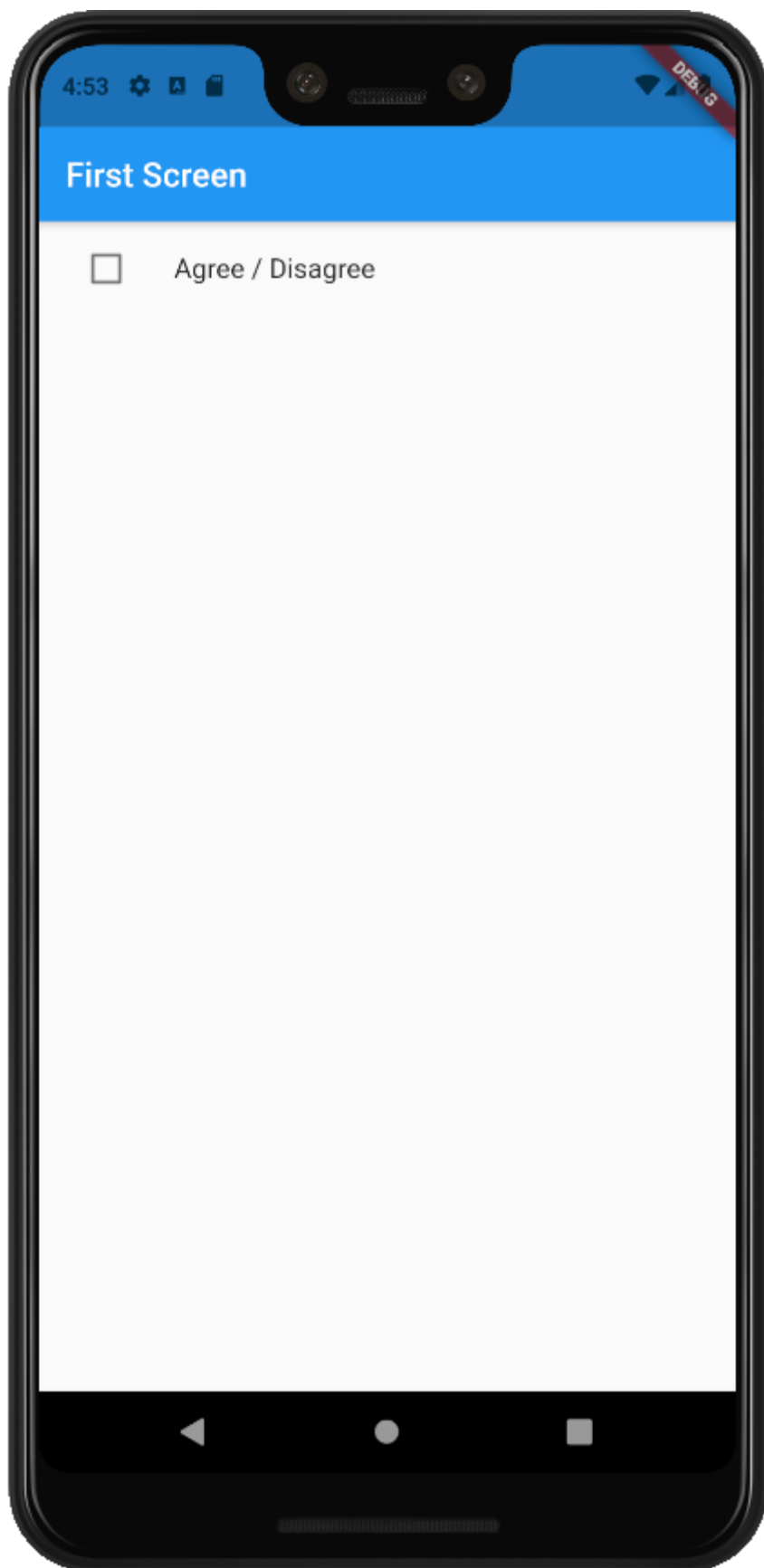


## Checkbox

Checkbox merupakan inputan benar atau salah. Checkbox akan berisi centang jika nilainya adalah benar dan kosong jika salah. Seperti pada contoh berikut:

```
1. class _FirstScreenState extends State<FirstScreen> {  
2.   bool agree = false;  
3.  
4.   @override  
5.   Widget build(BuildContext context) {  
6.     return Scaffold(  
7.       appBar: AppBar(  
8.         title: Text('First Screen'),  
9.       ),  
10.      body: ListTile(  
11.        leading: Checkbox(  
12.          value: agree,  
13.          onChanged: (bool value) {  
14.            setState(() {  
15.              agree = value;  
16.            });  
17.          },  
18.        ),  
19.        title: Text('Agree / Disagree'),  
20.      ),  
21.    );  
22.  }  
23. }
```

Kode di atas jika dijalankan akan tampil seperti berikut:



Ada beberapa tautan yang dapat Anda baca untuk memahami tentang widget-widget input yang ada pada Flutter, antara lain:

- [Input and selections widgets](#)
- [TextField Class](#)
- [Switch Class](#)
- [Radio Class](#)
- [Checkbox Class](#)