Implementing Form with validation:

- *Form*
- *TextFormField*
- *Validation*

Form- Flutter provides a **Form widget** to create a form. The form widget acts as a container, which allows us to group and validate the multiple form fields. When you create a form, it is necessary to provide the **GlobalKey**. This key uniquely identifies the form and allows you to do any validation in the form fields.

TextFormField- A FormField that contains a TextField. This is **a convenience widget that wraps a TextField widget in a FormField**. A Form ancestor is not required. The Form simply makes it easier to save, reset, or validate multiple fields at once.

Steps to implement Form with validation

We will understand this concept by developing the Simple application in which we will add two numbers.

Process:

- First create a new project in flutter application named ‘add_numbers’.
- Create a code as below in ‘main.dart’ file.

main.dart × addnumber.dart

lib > main.dart > ...

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

Run | Debug | Profile
void main(){runApp(const MyApp());}

class MyApp extends StatelessWidget {
  const MyApp({ Key? key }) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      title: 'Adding Numbers',
      home:AddNumber(),

    ); // MaterialApp
  }
}
```

- Create a new dart file called ‘addnumber.dart’ inside the lib folder which will create a screen (user interface) for adding two numbers.

  Git hub link: https://github.com/samsunk/add_numbers.git

main.dart addnumber.dart M ×

lib > addnumber.dart > _AddNumberState > build

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

class AddNumber extends StatefulWidget {
  const AddNumber({Key? key}) : super(key: key);

  @override
  State<AddNumber> createState() => _AddNumberState();
}

class _AddNumberState extends State<AddNumber> {
  TextEditingController firstnum = TextEditingController();
  TextEditingController secondnum = TextEditingController();
  String result = "0";
//1.  Global key for form.
  final _formkey = GlobalKey<FormState>();
```

```dart
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text('Adding Numbers'),
    ), // AppBar
    body: SingleChildScrollView(
      child: Padding(

        padding: const EdgeInsets.all(30.0),
        child: Form(
          key: _formkey,
          child: Column(
            children: [
              TextFormField(
                //2. textform field with validator logic
                validator: (value) {
                  if (value == null || value.isEmpty) {
                    return "field cannot be empty";
                  }
                  return null;
                },
                controller: firstnum,
                keyboardType: TextInputType.number,
                decoration: const InputDecoration(
                  hintText: 'Enter the first number',
                  labelText: 'First Number',
                ), // InputDecoration
              ), // TextFormField
              const SizedBox(height: 20),
              TextFormField(
                validator: (value) {
                  if (value == null || value.isEmpty) {
                    return "field cannot be empty";
                  }
                  return null;
                },
                controller: secondnum,
                keyboardType: TextInputType.number,
                decoration: const InputDecoration(
                  hintText: 'Enter the second number',
                  labelText: 'Second Number',
                ), // InputDecoration
              ), // TextFormField
              const SizedBox(height: 20),
              ElevatedButton(
                onPressed: () {
                  //3. validate and submit(sum) the form using button.
                  if (_formkey.currentState!.validate()) {
                    setState(() {
                      int sum = int.parse(firstnum.text) +
                          int.parse(secondnum.text);
                      result = sum.toString();
                    });
                  } else {
                    ScaffoldMessenger.of(context).showSnackBar(
                        const SnackBar(content: Text("Somtething wrong")));
                  }
                },
                child: const Text("Sum"),
                style: TextButton.styleFrom(
                  textStyle: const TextStyle(
                    fontSize: 20,
                    fontWeight: FontWeight.bold,
                  ), // TextStyle
                ),
              ), // ElevatedButton
              const SizedBox(height: 40),
```

```
84                  Row(
85                    mainAxisAlignment: MainAxisAlignment.center,
86                    children: [
87                      const Text(
88                        "Result",
89                        style: TextStyle(
90                          fontSize: 30,
91                          fontWeight: FontWeight.bold,
92                        ), // TextStyle
93                      ), // Text
94                      const SizedBox(width: 20),
95                      Text(
96                        result,
97                        style: const TextStyle(
98                          fontSize: 30,
99                          fontWeight: FontWeight.bold,
100                       ), // TextStyle
101                     ), // Text
102                   ],
103                 ), // Row
104               ],
105             ), // Column
106           ), // Form
107         ), // Padding
108       ), // SingleChildScrollView
109     ); // Scaffold
110   }
111 }
```

In this application you have encounter some of the new widgets like Form, TextFormField, SingleChildScrollView, SnackBar.

All of you are requested to go through the attributes of these new introduce widgets yourself.