Experiment 4

Aim: To create an interactive Form using form widget

Theory:

Flutter Forms:

Forms are an integral part of all modern mobile and web applications. It is mainly used to interact with the app as well as gather information from the users. They can perform many tasks, which depend on the nature of your business requirements and logic, such as authentication of the user, adding user, searching, filtering, ordering, booking, etc. A form can contain text fields, buttons, checkboxes, radio buttons, etc.

Some Properties of Form Widget

- **key:** A GlobalKey that uniquely identifies the Form. You can use this key to interact with the form, such as validating, resetting, or saving its state.
- **child:** The child widget that contains the form fields. Typically, this is a Column, ListView, or another widget that allows you to arrange the form fields vertically.
- **autovalidateMode:** An enum that specifies when the form should automatically validate its fields.

Some Methods of Form Widget

- **validate():** This method is used to trigger the validation of all the form fields within the Form. It returns true if all fields are valid, otherwise false. You can use it to check the overall validity of the form before submitting it.
- **save():** This method is used to save the current values of all form fields. It invokes the onSaved callback for each field. Typically, this method is called after validation succeeds.
- **reset():** Resets the form to its initial state, clearing any user-entered data.
- **currentState:** A getter that returns the current FormState associated with the Form.

Creating Form

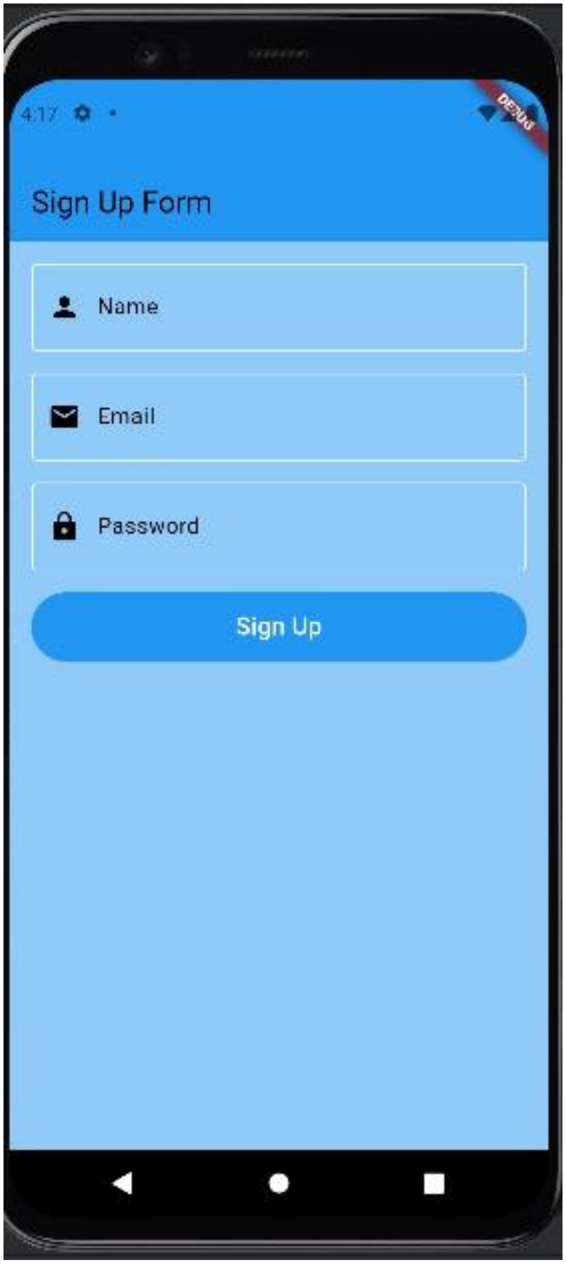
Flutter provides a **Form widget** to create a form. The form widget acts as a container, which allows us to group and validate the multiple form fields. When you create a form, it is necessary to provide the **GlobalKey**. This key uniquely identifies the form and allows you to do any validation in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

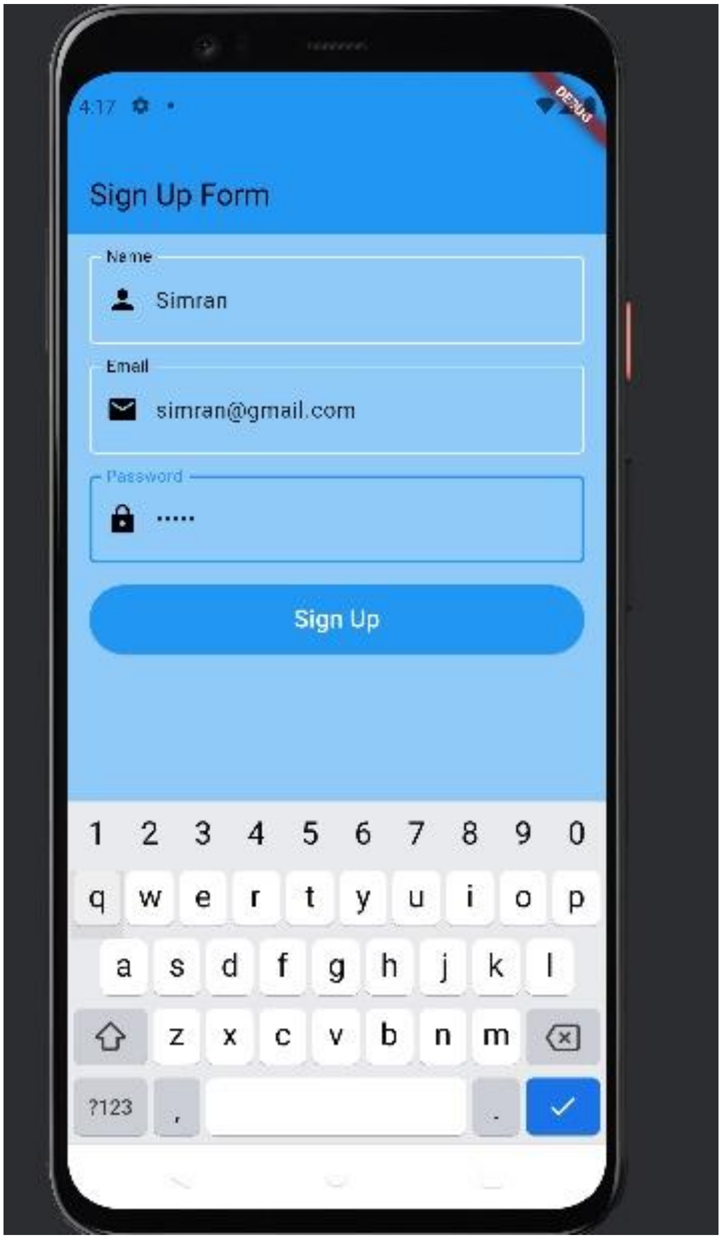
The form widget uses child widget **TextFormField** to provide the users to enter the text field. This widget renders a material design text field and also allows us to display validation errors when they occur.

Code:

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

main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget { @override
  Widget build(BuildContext context) { return
    MaterialApp(
      home: MySignUpForm(),
      theme: ThemeData(
        primaryColor: Colors.teal, // Changed primary color to teal colorScheme:
        ColorScheme.fromSwatch(primarySwatch: Colors.teal), //
Adjusted color scheme
        fontFamily: 'Arial',
      ),
    );
  }
}

class MySignUpForm extends StatefulWidget { @override
  _MySignUpFormState createState() => _MySignUpFormState();
}

class _MySignUpFormState extends State<MySignUpForm> {
  final GlobalKey<FormState> _formKey = GlobalKey<FormState>(); final
  TextEditingController _nameController = TextEditingController();
```

```
String _email = ''; String
_password = '';

String? _validateName(String? value) { if
  (value == null || value.isEmpty) { return
  'Please enter your name';
  }
  return null;
}

String? _validateEmail(String? value) { if
  (value == null || value.isEmpty) { return
  'Please enter your email';
  } else if (!RegExp(r'^[\w-]+(\.[\w-]+)*@([\w-]+\.)+[a-zA-Z]{2,7}$')
      .hasMatch(value)) {
    return 'Please enter a valid email address';
  }
  return null;
}

String? _validatePassword(String? value) { if
  (value == null || value.isEmpty) {  return 'Please
  enter your password';
  } else if (value.length < 6) {
    return 'Password must be at least 6 characters';
  }
  return null;
}

void _submitForm() {
  if (_formKey.currentState?.validate() ?? false) {
    _formKey.currentState?.save();
    _showSignUpCompleteDialog(_nameController.text);
  }
}

void _showSignUpCompleteDialog(String name) { showDialog(
    context: context,
      builder: (BuildContext context) {
        return AlertDialog(
          title: Text('Sign Up Complete'),
          content: Text('Congratulations, $name! You have successfully signed up.'), actions:
          <Widget>[
            TextButton( onPressed:
              () {
                Navigator.of(context).pop();
              },
              child: Text('OK'),
            ),
          ],
        );
      },
    );
  }

  @override
  Widget build(BuildContext context) { return
    Scaffold(
      appBar: AppBar(
        title: Text('Sign Up Form'),
        backgroundColor: Colors.teal, // Changed to teal
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0), child:
        Form(
          key: _formKey, child:
          Column(
            crossAxisAlignment: CrossAxisAlignment.stretch,
            children: [
              TextFormField(
                controller:          _nameController,
                decoration:          InputDecoration(
                labelText: 'Name',
                  hintText: 'Enter your name', border:
                  OutlineInputBorder(), prefixIcon:
                  Icon(Icons.person),




                ),
                style: TextStyle( fontSize:
                  16,
                  color: Colors.black87,
                ),
                validator: _validateName,
                onSaved: (value) {
                  _nameController.text = value ?? '';
                },
              ),
              SizedBox(height: 16), TextFormField(
                decoration: InputDecoration(
                  labelText: 'Email',
                  hintText: 'Enter your email', border:
                  OutlineInputBorder(), prefixIcon:
                  Icon(Icons.email),
                ),
                style: TextStyle( fontSize:
                  16,
                  color: Colors.black87,
                ),
                validator: _validateEmail,
                onSaved: (value) {
                  _email = value ?? '';
                },
              ),
              SizedBox(height: 16),
              TextFormField(
              obscureText: true,
                decoration: InputDecoration(
                  labelText: 'Password',
                  hintText: 'Enter your password',
                  border: OutlineInputBorder(),
                  prefixIcon: Icon(Icons.lock),
                ),
                style: TextStyle( fontSize:
                  16,
                  color: Colors.black87,




                ),
                validator: _validatePassword, onSaved:
                (value) {
                  _password = value ?? '';
                },
              ),
              SizedBox(height: 16),
              ElevatedButton( onPressed:
              _submitForm, child: Text(
                  'Sign Up',
                  style: TextStyle( fontSize:
                    18,
                    color: Colors.white,
                  ),
                ),
                style: ButtonStyle(
                  backgroundColor: MaterialStateProperty.all(Colors.teal), // Changed to
teal
                  padding: MaterialStateProperty.all(
                    EdgeInsets.symmetric(vertical: 12),
                  ),
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```


4:17
DEBUG
Sign Up Form
Name
Email
Password
Sign Up



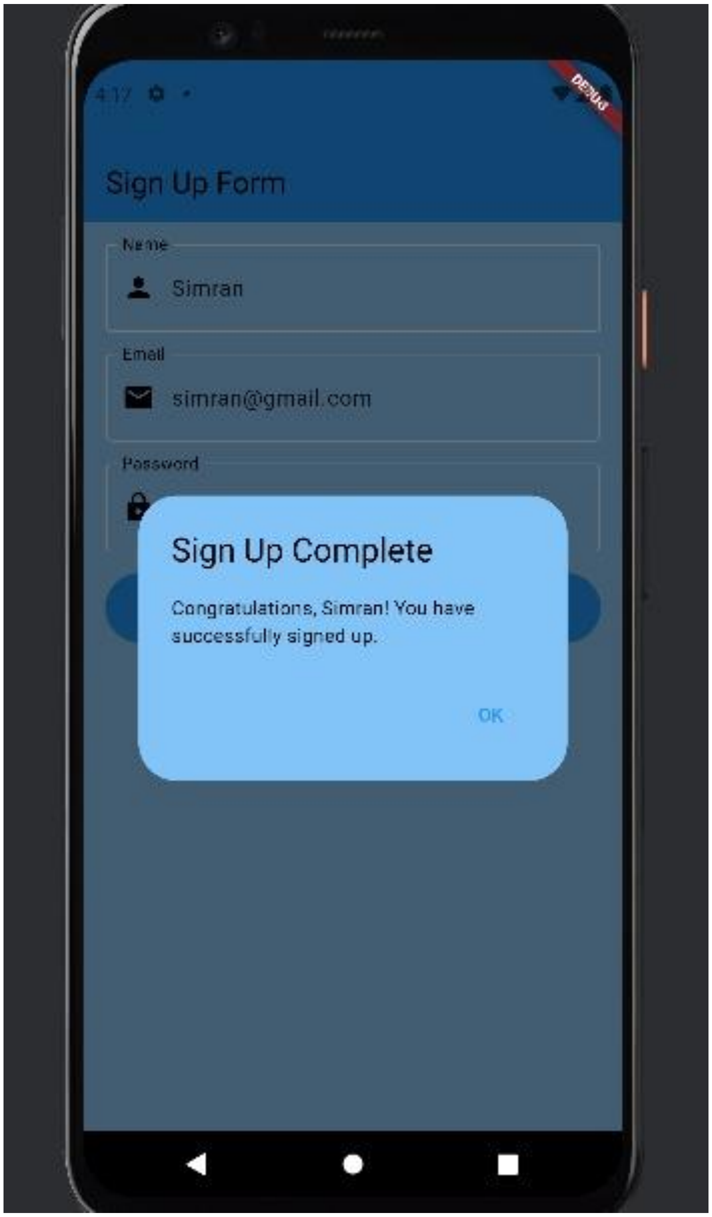
Sign Up Form

Name: Simran

Email: simran@gmail.com

Password: ·····

Sign Up



Sign Up Form

Name: Simran

Email: simran@gmail.com

Password

Sign Up Complete

Congratulations, Simran! You have successfully signed up.

Conclusion:

I understood the form widget in Flutter. Also, learnt to implement form widget to create interactive form for Flutter application.