Name: Tanvi Saraf **Class: D15B** **Roll No.: 60**

Experiment No. 4

Aim: To create an interactive Form using form widget

Theory:

In Flutter, the Form widget is a crucial component for building interactive user input forms. It facilitates input validation, data submission, and error handling. Here's a brief overview of creating an interactive form using the Form widget in Flutter:

1. Form Widget:

The Form widget in Flutter is a container for form fields and allows you to group multiple form fields together. It helps in managing form state, validation, and submission. When a form is submitted, the Form widget calls a callback function (onFormSubmitted) where you can handle the submitted data.

2. TextFormField:

TextFormField is a widget used to capture textual input from users. It provides options for text input, such as validation, input formatting, and obscure text (password fields). You can use validators like required, minLength, maxLength, email, etc., to validate user input.

3. DropdownButtonFormField:

DropdownButtonFormField is a form field widget that presents a dropdown menu of items for users to select from. It's commonly used when users need to choose from a predefined list of options. You can set an initial value, define the items in the dropdown menu, and handle the selection using onChanged callback.

4. CheckboxListTile and RadioListTile:

These widgets are used for capturing boolean and single-choice input, respectively. CheckboxListTile presents a checkbox that can be checked or unchecked by the user, while RadioListTile presents a group of radio buttons from which the user can select one option.

5. Form Validation:

Form validation ensures that user input meets certain criteria or constraints before it's accepted as valid. In Flutter, you can implement form validation by providing a validator function for each form field. The validator function takes the current value of the field as input and returns an error message if the value is not valid. The form will display error messages if validation fails.

6. Form Submission:

After users have filled out the form, they typically submit it to perform an action, such as saving the data to a database or sending it to a server. You can handle form submission by providing an onSubmit callback function to the Form widget. This function is called when the form is submitted, and you can access the current values of the form fields within this callback to perform the necessary actions.

Code:

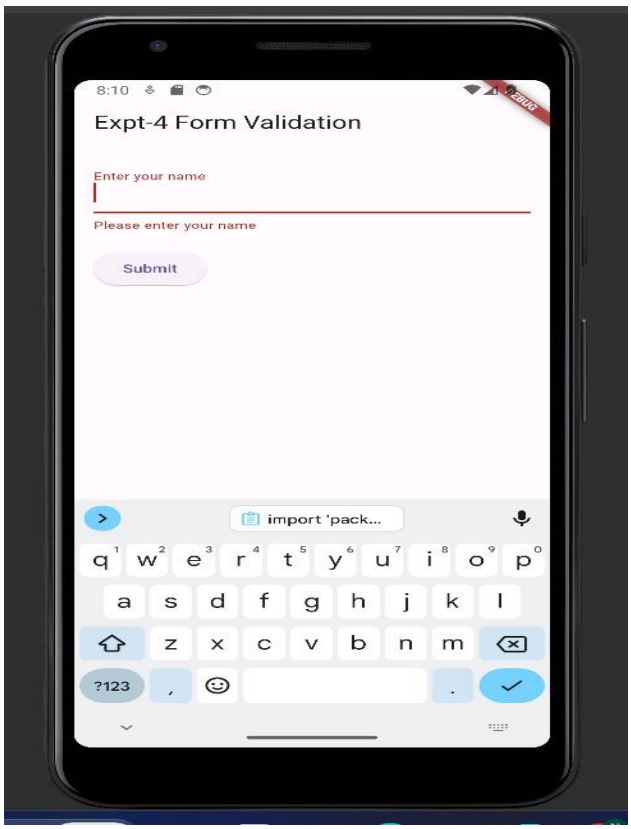
```
import 'package:flutter/material.dart';

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

class MyApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {return
   MaterialApp(
     title: 'Interactive Form Example',theme:
     ThemeData(
       primaryColor: Colors.blue, // Change primary color of the app
       appBarTheme: AppBarTheme(
         titleTextStyle: TextStyle(
           fontSize: 20.0,
           fontWeight: FontWeight.bold,
           color: Colors.white, // Change text color of the AppBar title
         ),
       ),
     ),
     home: MyForm(),
   );
 }
}

class MyForm extends StatefulWidget {
 @override
 _MyFormState createState() => _MyFormState();
}

class _MyFormState extends State<MyForm> {
 final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
 TextEditingController _nameController = TextEditingController();
 TextEditingController _emailController = TextEditingController();
 TextEditingController _mobileController = TextEditingController();
 String? _gender;
```

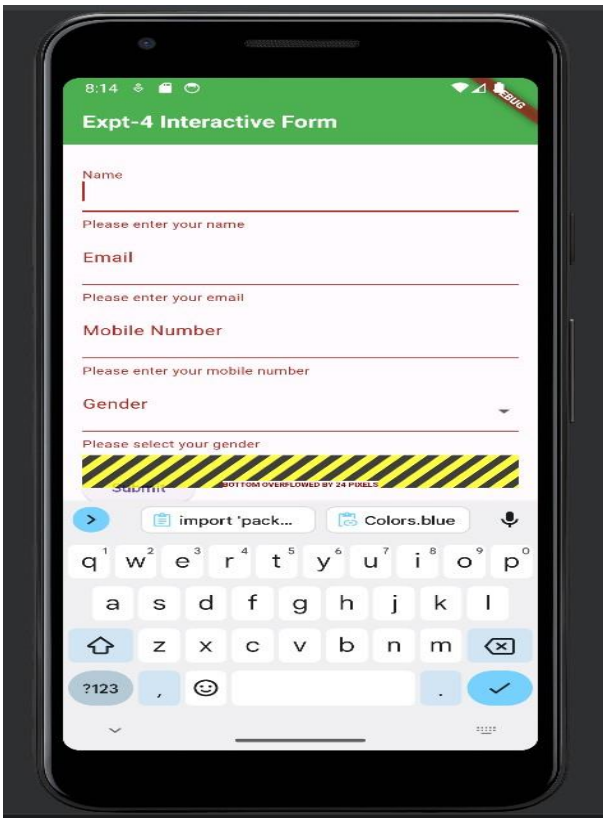
```
@override
void dispose() {
  _nameController.dispose();
  _emailController.dispose();
  _mobileController.dispose();
  super.dispose();
}

void _submitForm() {
  if (_formKey.currentState != null && _formKey.currentState!.validate()) {
    // If the form is valid, perform actions like saving to a database or API
    ScaffoldMessenger.of(context).showSnackBar(
      SnackBar(content: Text('Form is validated and submitted!')),
    );
  }
}

@override
Widget build(BuildContext context) {return
  Scaffold(
    appBar: AppBar(
      title: Text('Expt-4 Interactive Form'),
      // Customize background color of the AppBar
      backgroundColor: Colors.green,
    ),
    body: Padding(
      padding: EdgeInsets.all(16.0),
      child: Form(
        key: _formKey,
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,children:
          <Widget>[
            TextFormField(
              controller: _nameController,
              decoration: InputDecoration(
              labelText: 'Name',
              ),
              validator: (value) {
                if (value == null || value.isEmpty) {
                  return 'Please enter your name';
                }
                return null;
              },
            ),
```

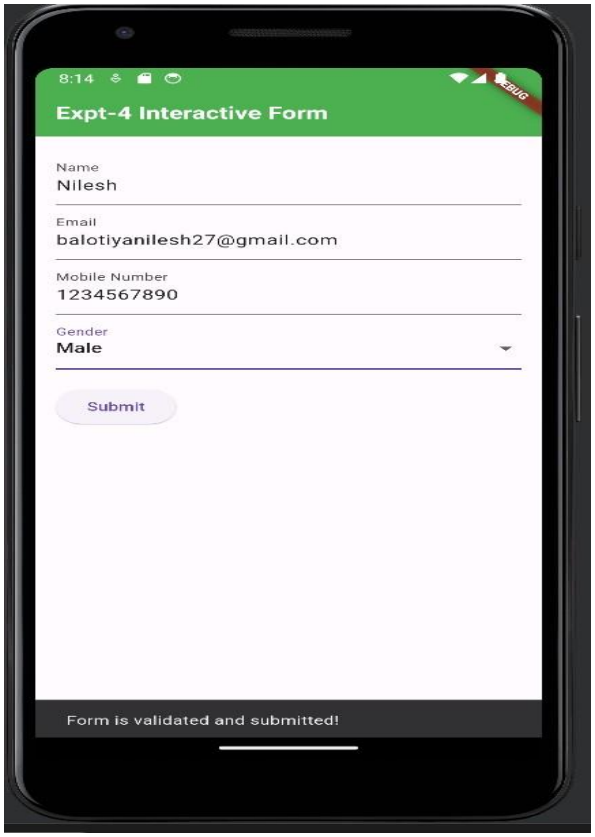
```
TextFormField(
  controller: _emailController,
  decoration: InputDecoration(
  labelText: 'Email',
  ),
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter your email';
    }
    if (!value.contains('@')) {
      return 'Please enter a valid email';
    }
    return null;
  },
),
TextFormField(
  controller:    _mobileController,
  decoration:     InputDecoration(
  labelText: 'Mobile Number',
  ),
  keyboardType: TextInputType.phone,
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter your mobile number';
    }
    if (value.length != 10) {
      return 'Please enter a valid 10-digit mobile number';
    }
    return null;
  },
),
DropdownButtonFormField<String>(
  decoration: InputDecoration( labelText:
  'Gender',
  ),
  value: _gender,
  items: ['Male', 'Female', 'Other']
      .map((gender) => DropdownMenuItem<String>(value:
    gender,
    child: Text(gender),
  ))
      .toList(),
  onChanged: (value) {
  setState(() {
```

```
                _gender = value;
              });
            },
            validator: (value) {
              if (value == null) {
                return 'Please select your gender';
              }
              return null;
            },
          ),
          SizedBox(height: 20),
          ElevatedButton( onPressed:
          _submitForm, child:
          Text('Submit'),
          ),
        ],
      ),
    ),
  ),
 );
 }
}
```

Output:


8:10
Expt-4 Form Validation
Enter your name
Please enter your name
Submit
import 'pack...


8:14
Expt-4 Interactive Form
Name
Please enter your name
Email
Please enter your email
Mobile Number
Please enter your mobile number
Gender
Please select your gender
Submit
import 'pack...
Colors.blue


8:14
Expt-4 Interactive Form
Name
Nilesh
Email
balotiyanilesh27@gmail.com
Mobile Number
1234567890
Gender
Male
Submit
Form is validated and submitted!

2.Login Form:

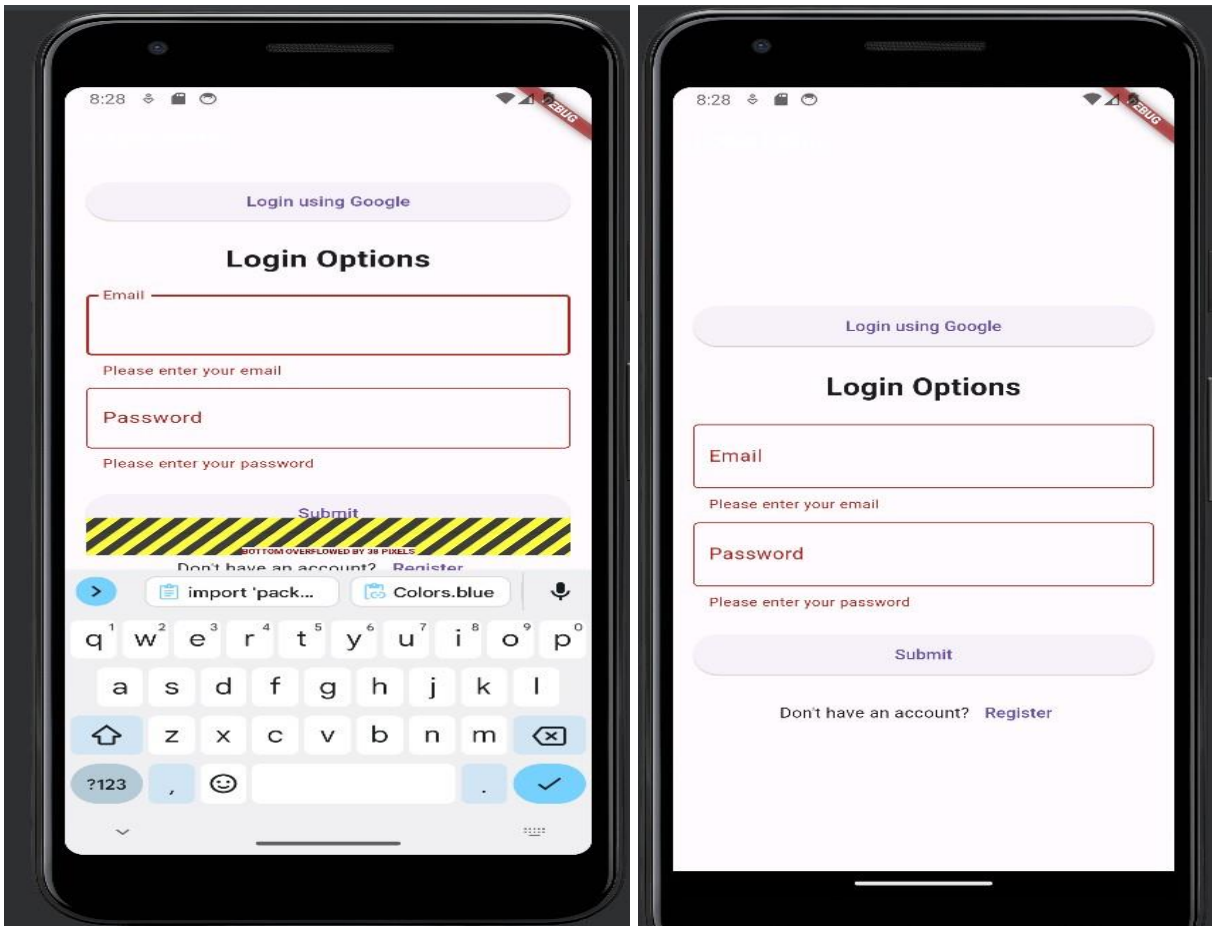
```
import 'package:flutter/material.dart';

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

class MyApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {return
   MaterialApp(
     title: 'Login Form',
     theme: ThemeData(
       primaryColor: Colors.blue, // Change primary color of the app
       appBarTheme: AppBarTheme(
         titleTextStyle: TextStyle(
           fontSize: 20.0,
           fontWeight: FontWeight.bold,
           color: Colors.white, // Change text color of the AppBar title
         ),
       ),
     ),
     home: LoginForm(),
   );
 }
}

class LoginForm extends StatefulWidget {
 @override
 _LoginFormState createState() => _LoginFormState();
}

class _LoginFormState extends State<LoginForm> {
 final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
 TextEditingController _emailController = TextEditingController();
 TextEditingController _passwordController = TextEditingController();

 @override
 void dispose() {
   _emailController.dispose();
   _passwordController.dispose();
   super.dispose();
 }
```

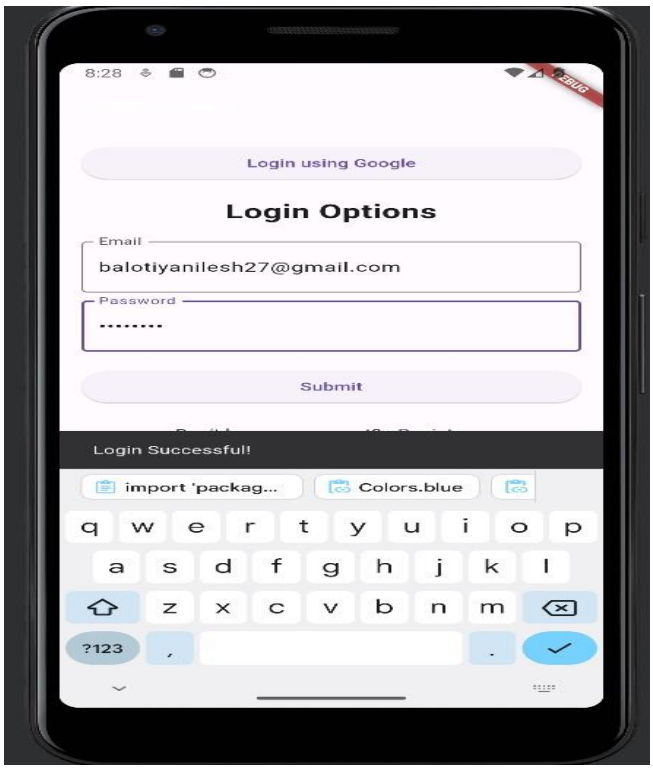
```
void _submitForm() {
  if (_formKey.currentState != null && _formKey.currentState!.validate()) {
    // If the form is valid, perform actions like login authentication
    ScaffoldMessenger.of(context).showSnackBar( SnackBar(content:
    Text('Login Successful!')),
    );
  }
}

@override
Widget build(BuildContext context) {return
  Scaffold(
    appBar: AppBar(
      title: Text('Login Form'),
    ),
    body: Padding(
      padding: EdgeInsets.all(16.0),
      child: Form(
        key: _formKey,
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          crossAxisAlignment: CrossAxisAlignment.stretch,children:
          <Widget>[
            ElevatedButton(
              onPressed: () {
                // Action for Login using Google
              },
              child: Text('Login using Google'),
            ),
            SizedBox(height: 20),
            Text(
              'Login Options',
              style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
              textAlign: TextAlign.center,
            ),
            SizedBox(height: 20),
            TextFormField(
              controller: _emailController,
              decoration: InputDecoration(
              labelText: 'Email',
                border: OutlineInputBorder(),
              ),
              validator: (value) {
                if (value == null || value.isEmpty) {
```

```
      return 'Please enter your email';
    }
    if (!value.contains('@')) {
      return 'Please enter a valid email';
    }
    return null;
  },
),
SizedBox(height: 10),
TextFormField(
  controller: _passwordController,
  obscureText: true,
  decoration: InputDecoration(
    labelText: 'Password', border:
    OutlineInputBorder(),
  ),
  validator: (value) {
    if (value == null || value.isEmpty) { return
      'Please enter your password';
    }
    if (value.length < 6) {
      return 'Password must be at least 6 characters long';
    }
    return null;
  },
),
SizedBox(height: 20),
ElevatedButton( onPressed:
_submitForm, child:
Text('Submit'),
),
SizedBox(height: 10),
Row(
  mainAxisAlignment: MainAxisAlignment.center,
  children: <Widget>[
    Text('Don\'t have an account?'),
    TextButton(
      onPressed: () {
        // Action for Register Button
      },
      child: Text('Register'),
    ),
  ],
),
```

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

Output:


8:28
DEBUG
Login using Google
Login Options
Email
Please enter your email
Password
Please enter your password
Submit
BOTTOM OVERFLOWED BY 38 PIXELS
Don't have an account? Register
import 'pack...
Colors.blue
8:28
DEBUG
Login using Google
Login Options
Email
Please enter your email
Password
Please enter your password
Submit
Don't have an account? Register



Conclusion:

I have successfully created an interactive Form using form widget in Flutter.