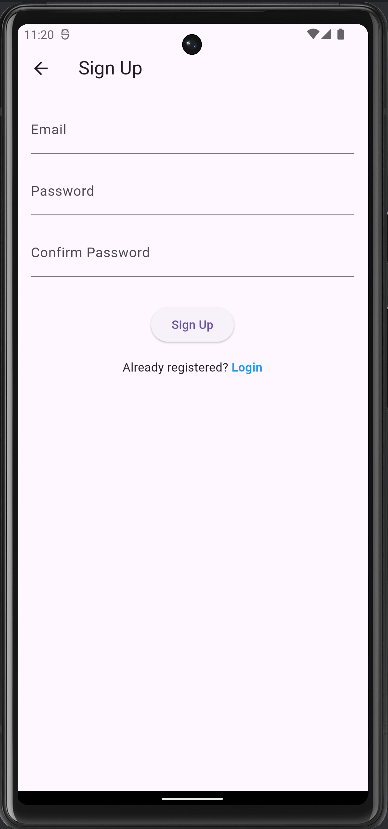
**Exp 4**

**AIM:- To create an interactive Form using form widget**

**Code:-**

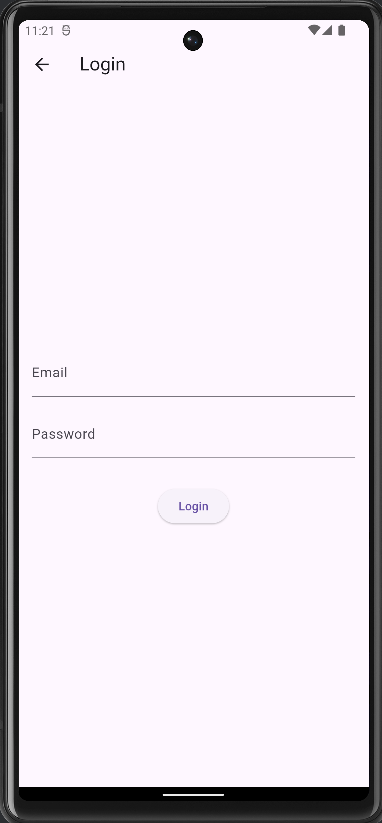
**signup\_screen.dart**

import 'package:flutter/material.dart';  
import 'expenses\_screen.dart';  
  
class SignupScreen extends StatefulWidget {  
 @override  
 \_SignupScreenState createState() => \_SignupScreenState();  
}  
  
class \_SignupScreenState extends State<SignupScreen> {  
 final \_formKey = GlobalKey<FormState>();  
 final \_emailController = TextEditingController();  
 final \_passwordController = TextEditingController();  
 final \_confirmPasswordController = TextEditingController();  
 String \_errorMessage = '';  
  
 void \_signup() {  
 if (\_formKey.currentState!.validate()) {  
 if (\_passwordController.text != \_confirmPasswordController.text) {  
 setState(() {  
 \_errorMessage = 'Passwords do not match';  
 });  
 return;  
 }  
  
 // Simulate successful signup  
 Navigator.pushReplacement(  
 context,  
 MaterialPageRoute(builder: (context) => ExpensesScreen()),  
 );  
 }  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(title: const Text('Sign Up')),  
 body: Padding(  
 padding: const EdgeInsets.all(16.0),  
 child: Form(  
 key: \_formKey,  
 child: Column(  
 children: [  
 TextFormField(  
 controller: \_emailController,  
 decoration: const InputDecoration(labelText: 'Email'),  
 keyboardType: TextInputType.emailAddress,  
 validator: (value) {  
 if (value == null || value.isEmpty) {  
 return 'Enter email';  
 }  
 final emailRegex = RegExp(r'^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$');  
 if (!emailRegex.hasMatch(value)) {  
 return 'Enter a valid email';  
 }  
 return null;  
 },  
 ),  
 const SizedBox(height: 16),  
 TextFormField(  
 controller: \_passwordController,  
 obscureText: true,  
 decoration: const InputDecoration(labelText: 'Password'),  
 validator: (value) {  
 if (value == null || value.isEmpty) {  
 return 'Enter password';  
 }  
 if (value.length < 6) {  
 return 'Password must be at least 6 characters';  
 }  
 final passwordRegex = RegExp(r'^(?=.\*[A-Za-z])(?=.\*\d)[A-Za-z\d]{6,}$');  
 if (!passwordRegex.hasMatch(value)) {  
 return 'Password must have at least 1 letter and 1 number';  
 }  
 return null;  
 },  
 ),  
 const SizedBox(height: 16),  
 TextFormField(  
 controller: \_confirmPasswordController,  
 obscureText: true,  
 decoration: const InputDecoration(labelText: 'Confirm Password'),  
 validator: (value) {  
 if (value == null || value.isEmpty) {  
 return 'Confirm your password';  
 }  
 if (value != \_passwordController.text) {  
 return 'Passwords do not match';  
 }  
 return null;  
 },  
 ),  
 const SizedBox(height: 16),  
 if (\_errorMessage.isNotEmpty)  
 Text(\_errorMessage, style: const TextStyle(color: Colors.red)),  
 const SizedBox(height: 16),  
 ElevatedButton(  
 onPressed: \_signup,  
 child: const Text('Sign Up'),  
 ),  
 ],  
 ),  
 ),  
 ),  
 );  
 }  
}

****

**login\_screen:-**

**// login\_screen.dart  
import 'package:flutter/material.dart';  
import 'expenses\_screen.dart';  
  
class LoginScreen extends StatefulWidget {  
 @override  
 \_LoginScreenState createState() => \_LoginScreenState();  
}  
  
class \_LoginScreenState extends State<LoginScreen> {  
 final \_formKey = GlobalKey<FormState>();  
 final \_emailController = TextEditingController();  
 final \_passwordController = TextEditingController();  
 String \_errorMessage = '';  
  
 void \_login() {  
 if (\_formKey.currentState!.validate()) {  
 // Simulate successful login  
 Navigator.pushReplacement(  
 context,  
 MaterialPageRoute(builder: (context) => ExpensesScreen()),  
 );  
 }  
 }  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(title: const Text('Login')),  
 body: Padding(  
 padding: const EdgeInsets.all(16.0),  
 child: Form(  
 key: \_formKey,  
 child: Column(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: [  
 TextFormField(  
 controller: \_emailController,  
 decoration: const InputDecoration(labelText: 'Email'),  
 keyboardType: TextInputType.emailAddress,  
 validator: (value) {  
 if (value == null || value.isEmpty) {  
 return 'Enter email';  
 }  
 final emailRegex = RegExp(r'^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$');  
 if (!emailRegex.hasMatch(value)) {  
 return 'Enter a valid email';  
 }  
 return null;  
 },  
 ),  
 const SizedBox(height: 16),  
 TextFormField(  
 controller: \_passwordController,  
 obscureText: true,  
 decoration: const InputDecoration(labelText: 'Password'),  
 validator: (value) {  
 if (value == null || value.isEmpty) {  
 return 'Enter password';  
 }  
 if (value.length < 6) {  
 return 'Password must be at least 6 characters';  
 }  
 return null;  
 },  
 ),  
 const SizedBox(height: 16),  
 if (\_errorMessage.isNotEmpty)  
 Text(  
 \_errorMessage,  
 style: const TextStyle(color: Colors.red),  
 ),  
 const SizedBox(height: 16),  
 ElevatedButton(  
 onPressed: \_login,  
 child: const Text('Login'),  
 ),  
 ],  
 ),  
 ),  
 ),  
 );  
 }  
}**

****