```
import 'dart:math';
import 'package:flutter/material.dart';
enum AuthMode { Signup, Login }
class AuthPage extends StatelessWidget {
  @override
 Widget build(BuildContext context) {
    final deviceSize = MediaQuery.of(context).size;
    return Scaffold(
      body: Stack(
        children: <Widget>[
          Container(
            decoration: BoxDecoration(
              gradient: LinearGradient(
                colors: [
                  const Color.fromRGBO(215, 117, 255, 1).withOpacity(0.5),
                  const Color.fromRGBO(255, 188, 117, 1).withOpacity(0.9),
                ],
                begin: Alignment.topLeft,
                end: Alignment.bottomRight,
                stops: const [0, 1],
              ),
            ),
          ),
          SingleChildScrollView(
            child: SizedBox(
              height: deviceSize.height,
              width: deviceSize.width,
              child: Column(
                mainAxisAlignment: MainAxisAlignment.center,
                crossAxisAlignment: CrossAxisAlignment.center,
                children: <Widget>[
                  Flexible(
                    child: Container(
                      margin: const EdgeInsets.only(bottom: 20.0),
                      padding: const EdgeInsets.symmetric(
                          vertical: 8.0, horizontal: 94.0),
                      transform: Matrix4.rotationZ(-8 * pi / 180)
                        ..translate(-10.0),
                      decoration: BoxDecoration(
                        borderRadius: BorderRadius.circular(20),
                        color: Colors.deepOrange.shade900,
```

```
boxShadow: const [
                           BoxShadow(
                             blurRadius: 8,
                             color: Colors.black26,
                             offset: Offset(0, 2),
                           )
                         ],
                      ),
                      child: const Text(
                         'MyShop',
                         style: TextStyle(
                           fontSize: 50,
                           fontFamily: 'Anton',
                           fontWeight: FontWeight.normal,
                         ),
                      ),
                    ),
                  ),
                  Flexible(
                    flex: deviceSize.width > 600 ? 2 : 1,
                    child: const AuthCard(),
                  ),
                ],
              ),
            ),
          ),
        ],
      ),
    );
  }
}
class AuthCard extends StatefulWidget {
  const AuthCard({
    Key? key,
  }) : super(key: key);
  @override
  _AuthCardState createState() => _AuthCardState();
}
class _AuthCardState extends State<AuthCard> {
  final GlobalKey<FormState> _formKey = GlobalKey();
  AuthMode _authMode = AuthMode.Login;
  bool _isLoading = false;
```

```
final _passwordController = TextEditingController();
void switchAuthMode() {
  if (_authMode == AuthMode.Login) {
    setState(() {
      _authMode = AuthMode.Signup;
    });
  } else {
    setState(() {
      _authMode = AuthMode.Login;
    });
 }
}
@override
Widget build(BuildContext context) {
  final deviceSize = MediaQuery.of(context).size;
  return Card(
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(10.0),
    ),
    elevation: 8.0,
    child: Container(
      height: _authMode == AuthMode.Signup ? 320 : 260,
      constraints:
          BoxConstraints(minHeight: _authMode == AuthMode.Signup ? 320 : 260),
      width: deviceSize.width * 0.75,
      padding: const EdgeInsets.all(16.0),
      child: Form(
        key: formKey,
        child: SingleChildScrollView(
          child: Column(
            children: <Widget>[
              TextFormField(
                decoration: const InputDecoration(labelText: 'E-Mail'),
                keyboardType: TextInputType.emailAddress,
              ),
              TextFormField(
                decoration: const InputDecoration(labelText: 'Password'),
                obscureText: true,
                controller: _passwordController,
              ),
              if ( authMode == AuthMode.Signup)
                TextFormField(
                  enabled: _authMode == AuthMode.Signup,
                  decoration:
```

```
const InputDecoration(labelText: 'Confirm Password'),
                   obscureText: true,
                  ),
                const SizedBox(height: 20),
                if (_isLoading)
                  const CircularProgressIndicator()
                else
                  ElevatedButton(
                   onPressed: () {},
                    child:
                        Text(_authMode == AuthMode.Login ? 'LOGIN' : 'SIGN UP'),
                const SizedBox(height: 2),
                TextButton(
                  onPressed: _switchAuthMode,
                  child: Text(
                      '${_authMode == AuthMode.Login ? 'SIGNUP' : 'LOGIN'}
INSTEAD'),
               ),
             ],
           ),
      ),
     ),
   );
```