```
import 'package:flutter/material.dart'; import
'package:shared_preferences/shared_preferences.dart'; import 'map_screen.dart';
import 'signup.dart'; class LoginPage extends StatefulWidget { const
LoginPage({super.key}); @override State createState() => _LoginPageState(); } class
_LoginPageState extends State { final TextEditingController emailController =
TextEditingController(); final TextEditingController passwordController =
TextEditingController(); bool isLoading = false; bool _obscurePassword = true;
@override void initState() { super.initState(); autoLogin(); } Future autoLogin()
async { SharedPreferences prefs = await SharedPreferences.getInstance(); bool
loggedIn = prefs.getBool("loggedIn") ?? false; if (loggedIn) {
Navigator.pushReplacement( context, MaterialPageRoute(builder: (context) => const
MapScreen()), ); } } Future login() async { String email =
emailController.text.trim(); String password = passwordController.text.trim(); if
(email.isEmpty || password.isEmpty) { ScaffoldMessenger.of(context).showSnackBar(
const SnackBar(content: Text("Please enter email & password")), ); return; }
setState(() => isLoading = true); SharedPreferences prefs = await
SharedPreferences.getInstance(); String? savedEmail = prefs.getString("email");
String? savedPassword = prefs.getString("password"); if (email == savedEmail &&
password == savedPassword) { await prefs.setBool("loggedIn", true);
Navigator.pushReplacement( context, MaterialPageRoute(builder: (context) => const
MapScreen()), ); } else { ScaffoldMessenger.of(context).showSnackBar( const
SnackBar(content: Text("Invalid email or password")), ); } setState(() => isLoading
= false); } @override Widget build(BuildContext context) { return Scaffold( body:
Container( decoration: const BoxDecoration( gradient: LinearGradient( colors:
[Colors.deepPurple, Colors.blue], begin: Alignment.topLeft, end:
Alignment.bottomRight, ), ), child: Center( child: SingleChildScrollView( padding:
const EdgeInsets.symmetric(horizontal: 20), child: Container( padding: const
EdgeInsets.all(25), decoration: BoxDecoration( color: Colors.white.withOpacity(0.1),
borderRadius: BorderRadius.circular(20), border: Border.all(color:
Colors.white.withOpacity(0.3)), boxShadow: [ BoxShadow( color:
Colors.black.withOpacity(0.2), blurRadius: 12, spreadRadius: 2, ), ], ), child:
Column( mainAxisSize: MainAxisSize.min, children: [ const Text( "Crowd Calm", style:
TextStyle( fontSize: 32, fontWeight: FontWeight.bold, color: Colors.white, ), ),
const SizedBox(height: 25), TextField( controller: emailController, decoration:
InputDecoration( hintText: "Email", filled: true, fillColor:
Colors.white.withOpacity(0.8), border: OutlineInputBorder( borderRadius:
BorderRadius.circular(12), borderSide: BorderSide.none, ), ), ), const
SizedBox(height: 15), TextField( controller: passwordController, obscureText:
_obscurePassword, decoration: InputDecoration( hintText: "Password", filled: true,
fillColor: Colors.white.withOpacity(0.8), border: OutlineInputBorder( borderRadius:
BorderRadius.circular(12), borderSide: BorderSide.none, ), suffixIcon: IconButton(
icon: Icon( _obscurePassword ? Icons.visibility_off : Icons.visibility, color:
Colors.deepPurple, ), onPressed: () { setState(() { _obscurePassword =
!_obscurePassword; }); }, ), ), ), const SizedBox(height: 25), ElevatedButton(
onPressed: isLoading ? null : login, style: ElevatedButton.styleFrom(
backgroundColor: Colors.white, foregroundColor: Colors.deepPurple, padding: const
EdgeInsets.symmetric( horizontal: 60, vertical: 15), shape: RoundedRectangleBorder(
borderRadius: BorderRadius.circular(12), ), ), child: isLoading ? const
CircularProgressIndicator( color: Colors.deepPurple, ) : const Text( "Login", style:
TextStyle( fontSize: 18, fontWeight: FontWeight.bold), ), ), const SizedBox(height:
15), TextButton( onPressed: () { Navigator.push( context, MaterialPageRoute(
builder: (context) => const SignupPage()), ); }, child: const Text( "Don't have an
account? Sign up", style: TextStyle(color: Colors.white), ), ), ], ), ), ), ), ), );
} }
```