

```

import 'package:flutter/material.dart';
import 'home_screen.dart';
import '../utils/storage.dart';

class LockScreen extends StatefulWidget {
  const LockScreen({Key? key}) : super(key: key);

  @override
  State<LockScreen> createState() => _LockScreenState();
}

class _LockScreenState extends State<LockScreen> {
  final _pinController = TextEditingController();
  String _error = "";
  int _attempts = 0;
  static const maxAttempts = 10;

  Future<void> _checkPin() async {
    String? savedPin = await Storage.getPin();
    if (savedPin == null) {
      // First run: save default PIN 99066
      await Storage.savePin('99066');
      savedPin = '99066';
    }
    if (_pinController.text == savedPin) {
      Navigator.pushReplacement(
        context, MaterialPageRoute(builder: (_) => const HomeScreen()));
    } else {
      setState(() {
        _attempts++;
        _error = 'Incorrect PIN. Attempts: $_attempts';
      });
      if (_attempts >= maxAttempts) {
        // TODO: self-destruct data here
        // For now just show alert
        showDialog(
          context: context,
          builder: (_) => AlertDialog(
            title: const Text('Too many attempts'),
            content:
              const Text('Data will be erased to protect your privacy.'),
            actions: [
              TextButton(
                onPressed: () => Navigator.of(context).pop(),
                child: const Text('OK'))
            ],
          ));
    }
  }
}

```

```

    }
    _pinController.clear();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Darby Lock'),
      ),
      body: Padding(
        padding: const EdgeInsets.all(32.0),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            const Text('Enter your 5-digit PIN',
              style: TextStyle(fontSize: 22)),
            const SizedBox(height: 20),
            TextField(
              controller: _pinController,
              maxLength: 5,
              keyboardType: TextInputType.number,
              obscureText: true,
              decoration: const InputDecoration(
                border: OutlineInputBorder(),
                counterText: "",
                hintText: 'PIN',
              ),
              onSubmitted: (_) => _checkPin(),
            ),
            if (_error.isNotEmpty)
              Padding(
                padding: const EdgeInsets.only(top: 12.0),
                child: Text(_error, style: const TextStyle(color: Colors.red)),
              ),
            const SizedBox(height: 30),
            ElevatedButton(
              onPressed: _checkPin,
              child: const Text('Unlock'),
            ),
          ],
        ),
      ),
    );
  }
}

```