

LAPORAN PRAKTIKUM



NIM : 2003073

Nama : Ica Natasya

Kelas : D3TI.2C

Mata Kuliah : **Pemrograman Perangkat Bergerak (TIU3403)**

Praktikum ke / Judul : 12 / SHARED PREFERENCES

Tanggal Praktikum : 16 MEI 2022

Dosen Pengampu : Fachrul Pralienka Bani Muhamad, S.ST., M.Kom.

**PROGRAM STUDI D3 TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI INDRAMAYU
2022**



PRAKTIKUM 12

SHARED PREFERENCES

A. TUJUAN PRAKTIKUM

Tujuan Umum

Mahasiswa dapat mengimplementasikan shared preferences pada flutter untuk menyimpan state tertentu

Tujuan Khusus

Mahasiswa dapat

1. Menjelaskan pengertian shared preferences
2. Melakukan pengaturan lingkungan (dependency) shared preferences
3. Menyimpan state suatu nilai dan tema dengan shared preferences
4. Menyimpan state login dengan shared preferences

B. TEORI SINGKAT

SharedPreferences Pada Android

Jika anda membutuhkan sekumpulan data dengan tipe key-value yang ingin anda simpan dalam aplikasi mobile, anda dapat menggunakan SharedPreferences API. Data dalam SharedPreferences akan disimpan dalam sebuah file yang mengandung key-value secara berpasangan. Setiap file ini dikelola oleh Android framework yang aksesnya dapat diatur secara private atau dibagikan.

Untuk menyimpan data ke dalam SharedPreferences dibutuhkan class Editor yang bisa didapatkan dari method edit() dari instansiasi variable SharedPreferences.

cara untuk mendapatkan akses SharedPreferences dengan cara getSharedPreferences()

Jika anda membutuhkan lebih dari satu file yang diidentifikasi berdasarkan nama, dimana diatur pada parameter pertama.

NSUserDefaults pada iOS

Dengan kelas NSUserDefaults dapat menyimpan pengaturan dan properti yang terkait dengan aplikasi atau data pengguna. Misalnya, Anda bisa menyimpan gambar profil yang ditetapkan oleh pengguna atau skema warna default untuk aplikasi. Objek akan disimpan dalam apa yang dikenal sebagai "sistem default" iOS. Sistem iOS default tersedia di seluruh kode di aplikasi Anda, dan semua data yang disimpan ke sistem default akan bertahan melalui sesi aplikasi.

shared_preferences pada Flutter

Shared Preferences adalah salah satu opsi penyimpanan data dalam aplikasi android yang mudah digunakan dimana penggunaannya sangat cocok untuk menyimpan data sederhana seperti String, Int dan lain-lain. Penggunaannya biasa saya gunakan untuk menyimpan data login atau pengaturan pada tiap-tiap pengguna.

Penggunaan dalam flutter yaitu kita install terlebih dahulu dan tambahkan dalam pubspec.yaml. kemudian import kedalam file dart, penyimpanan, pemanggilan.



C. PELAKSANAAN PRAKTIKUM

Langkah-langkah praktikum Menyimpan Nilai dan Tema dengan Shared Preferences

<https://www.youtube.com/watch?v=j7J9Kq0sgQI&t=15s>

<https://github.com/icntsy/Praktikum-12/tree/main/prak12>

1. Install shared_preferences

```
29 dependencies:
30   flutter:
31     sdk: flutter
32   shared_preferences: ^2.0.15
33
```

2. Buat kode program didalam file menu.dart seperti dibawah ini.

```
prak12 - main.dart

1 import 'dart:convert';
2 import 'package:flutter/material.dart';
3 import 'package:shared_preferences/shared_preferences.dart';
4
5 void main() {
6   runApp(MyApp());
7 }
8
9 class MyApp extends StatefulWidget {
10   @override
11   _MyAppState createState() => _MyAppState();
12 }
13
14 class _MyAppState extends State<MyApp> {
15   int counter = 0;
16   bool isDark = false;
17
18   Future<void> setPreference() async {
19     final jembatan = await SharedPreferences.getInstance();
20
21     if (jembatan.containsKey('myData')) {
22       jembatan.clear();
23     }
24
25     final myData = json.encode({
26       'counter': counter.toString(),
27       'isDark': isDark.toString(),
28     });
29
30     jembatan.setString('myData', myData);
31     setState(() {});
32   }
33
34   Future<void> getPreference() async {
35     final jembatan = await SharedPreferences.getInstance();
36
37     if (jembatan.containsKey('myData')) {
38       final myData =
39         json.decode(jembatan.getString('myData')) as Map<String, dynamic>;
40
41       counter = int.parse(myData['counter']);
42       isDark = myData['isDark'] == "true" ? true : false;
43     }
44   }
45
46   void minus() {
47     counter = counter - 1;
48
49     setPreference();
50   }
51 }
```



```
prak12 - main.dart

52 void add() {
53   counter = counter + 1;
54   setPreference();
55 }
56
57 void changeTheme() {
58   isDark = !isDark;
59   setPreference();
60 }
61
62 void refresh() {
63   counter = 0;
64   isDark = false;
65   setPreference();
66 }
67
68 ThemeData dark = ThemeData(
69   appBarTheme: AppBarTheme(color: Colors.amber),
70   brightness: Brightness.dark,
71   primaryColor: Colors.amber,
72   outlinedButtonTheme: OutlinedButtonThemeData(
73     style: OutlinedButton.styleFrom(
74       primary: Colors.white,
75       side: BorderSide(
76         color: Colors.white,
77         width: 2,
78       ),
79       shape: RoundedRectangleBorder(
80         borderRadius: BorderRadius.circular(50),
81       ),
82     ),
83   ),
84   colorScheme: ColorScheme.fromSwatch(
85     primarySwatch: Colors.amber, brightness: Brightness.dark)
86     .copyWith(secondary: Colors.amber),
87 );
88
89 ThemeData light = ThemeData(
90   brightness: Brightness.light,
91   primaryColor: Colors.amber,
92   primarySwatch: Colors.amber,
93   accentColor: Colors.amber,
94   outlinedButtonTheme: OutlinedButtonThemeData(
95     style: OutlinedButton.styleFrom(
96       primary: Colors.black,
97       side: BorderSide(
98         color: Colors.black,
99         width: 2,
100     ),
```



```
prak12 - main.dart

101 shape: RoundedRectangleBorder(
102     borderRadius: BorderRadius.circular(50),
103 ),
104 ),
105 ),
106 );
107
108 @override
109 Widget build(BuildContext context) {
110     print('build');
111     return FutureBuilder(
112         future: getPreference(),
113         builder: (context, _) => MaterialApp(
114             debugShowCheckedModeBanner: false,
115             theme: isDark ? dark : light,
116             home: Scaffold(
117                 appBar: AppBar(
118                     title: Center(child: Text("Counter Apps")),
119                     actions: [
120                         IconButton(icon: Icon(Icons.refresh), onPressed: refresh),
121                     ],
122                 ),
123                 body: Center(
124                     child: Column(
125                         mainAxisAlignment: MainAxisAlignment.center,
126                         children: [
127                             Text(
128                                 "Angka saat ini adalah : $counter",
129                                 style: TextStyle(
130                                     fontSize: 25,
131                                 ),
132                             ),
133                             SizedBox(
134                                 height: 50,
135                             ),
136                             Row(
137                                 mainAxisAlignment: MainAxisAlignment.spaceEvenly,
138                                 children: [
139                                     OutlinedButton(
140                                         onPressed: minus,
141                                         child: Icon(Icons.remove),
142                                     ),
143                                     OutlinedButton(
144                                         onPressed: add,
145                                         child: Icon(Icons.add),
146                                     ),
147                                 ],
148                             ),
149                         ],
150                     ),
151                 ),
152                 floatingActionButton: FloatingActionButton(
153                     onPressed: changeTheme,
154                     child: Icon(Icons.color_lens),
155                 ),
156             ),
157         ),
158     );
159 }
160 }
161
```



3. Running program yang telah kita buat, hasilnya yaitu ada dua tema yang pertama tema terang dan yang kedua tema gelap.





4. Tampilan ini sudah menggunakan shared preference yaitu jika kita menginputkan data 3 misalkan maka setelah aplikasi di close maka data yang kita inputkan tidak akan hilang masih tetap data yang telah kita inputkan sebelumnya. Dibawah ini merupakan tampilan dengan tema gelap.





D. LATIHAN

Latihan Implementasi Shared Preferences pada Studi kasus Login

Buatlah implementasi penggunaan shared preferences baik dengan login yang tidak menggunakan basis data maupun menggunakan basis data (pilih salah satu)

<https://github.com/icntsy/Praktikum-12/tree/main/latihan>

1. Install shared_preferences
2. Buat kode program main.dart seperti dibawah ini

```
latihan - main.dart

1 import 'package:flutter/material.dart';
2 import 'package:shared_preferences/shared_preferences.dart';
3 import 'mainpage.dart';
4
5 void main() => runApp(MyApp());
6
7
8 class MyApp extends StatelessWidget {
9
10  // This widget is the root of your application.
11  @override
12  Widget build(BuildContext context) {
13    return MaterialApp(
14      title: 'Flutter Shared Preference',
15      theme: ThemeData(
16        primarySwatch: Colors.green,
17      ),
18      home: MyLoginPage(),
19      debugShowCheckedModeBanner: false,
20    );
21  }
22 }
23
24 class MyLoginPage extends StatefulWidget {
25   @override
26   _MyLoginPageState createState() => _MyLoginPageState();
27 }
28
29 class _MyLoginPageState extends State<MyLoginPage> {
30
31   final username_controller = TextEditingController();
32   final password_controller = TextEditingController();
33
34   late SharedPreferences logindata;
35   late bool newuser;
36
37   @override
38   void initState() {
39     super.initState();
40     check_if_already_login();
41   }
42
43
44   @override
45   Widget build(BuildContext context) {
46     return Scaffold(
47       appBar: AppBar(
48         title: Text(" Shared Preference"),
49       ),
```




```
latihan - mandiri

50 body: Center(
51   child: Column(
52     crossAxisAlignment: CrossAxisAlignment.center,
53     children: <Widget>[
54       Text(
55         "Login ",
56         style: TextStyle(fontSize: 30, fontWeight: FontWeight.bold),
57       ),
58       Padding(
59         padding: const EdgeInsets.all(15.0),
60         child: TextField(
61           controller: username_controller,
62           decoration: InputDecoration(
63             border: OutlineInputBorder(),
64             labelText: 'username',
65           ),
66         ),
67       ),
68       Padding(
69         padding: const EdgeInsets.all(15.0),
70         child: TextField(
71           controller: password_controller,
72           decoration: InputDecoration(
73             border: OutlineInputBorder(),
74             labelText: 'password',
75           ),
76         ),
77       ),
78       RaisedButton(
79         textColor: Colors.white,
80         color: Colors.green,
81         onPressed: () {
82           String username = username_controller.text;
83           String password = password_controller.text;
84
85           if (username != '' && password != '') {
86             print('Successfull');
87             logindata.setBool('login', false);
88
89             logindata.setString('username', username);
90             Navigator.pushReplacement(context,
91               MaterialPageRoute(builder: (context) => MyDashboard()));
92           }
93         },
94         child: Text("Log-in"),
95       ),
96     ],
97   ),
98 );
99
100
101 void check_if_already_login() async {
102   logindata = await SharedPreferences.getInstance();
103   newuser = (logindata.getBool('login') ?? true);
104
105   print(newuser);
106   if (newuser == false) {
107     Navigator.pushReplacement(
108       context, new MaterialPageRoute(builder: (context) => MyDashboard()));
109   }
110 }
111
112 @override
113 void dispose() {
114   username_controller.dispose();
115   password_controller.dispose();
116   super.dispose();
117 }
118 }
119
```



3. Buat kode program mainpage.dart seperti dibawah ini

```
luthan - mainpage.dart
1
2 import 'package:flutter/cupertino.dart';
3 import 'package:flutter/material.dart';
4 import 'main.dart';
5 import 'package:shared_preferences/shared_preferences.dart';
6
7 class MainPage extends StatelessWidget {
8   @override
9   Widget build(BuildContext context) {
10     return MaterialApp(
11       title: 'Flutter Demo',
12       theme: ThemeData(
13         primarySwatch: Colors.green,
14       ),
15       home: MyDashboard(),
16       debugShowCheckedModeBanner: false,
17     );
18   }
19 }
20
21 class MyDashboard extends StatefulWidget {
22   @override
23   _MyDashboardState createState() => _MyDashboardState();
24 }
25
26 class _MyDashboardState extends State<MyDashboard> {
27   late SharedPreferences logindata;
28   late String username;
29
30   @override
31   void initState() {
32     super.initState();
33     initial();
34   }
35
36   void initial() async {
37     logindata = await SharedPreferences.getInstance();
38     setState(() {
39       username = logindata.getString('username')!;
40     });
41   }
42
43   @override
44   Widget build(BuildContext context) {
45     return Scaffold(
46       appBar: AppBar(
47         title: Text("My Dashboard"),
48       ),
49       body: Padding(
50         padding: const EdgeInsets.all(26.0),
51         child: Column(
52           mainAxisAlignment: MainAxisAlignment.center,
53           children: <Widget>{
54             Center(
55               child: Text(
56                 'WELCOME TO MYAPP',
57                 style: TextStyle(fontSize: 25, fontWeight: FontWeight.bold),
58               ),
59             ),
60             Center(
61               child: Text(
62                 'Username : $username',
63                 style: TextStyle(fontSize: 25, fontWeight: FontWeight.bold),
64               ),
65             ),
66             RaisedButton(
67               onPressed: () {
68                 logindata.setBool('login', true);
69                 Navigator.pushReplacement(context,
70                   MaterialPageRoute(builder: (context) => MyLoginPage()));
71               },
72               child: Text('LogOut'),
73             ),
74           ],
75         ),
76       );
77   }
78 }
79
80
```



4. Hasil running dari kedua program, kita inputkan username dan password kemudian pilih Log-In.



5. Setelah login maka kita diarahkan ke menu dashboard, untuk tampilan sudah menggunakan shared preferences sehingga ketika kita meng close aplikasi dan membuka kembali aplikasi maka tampilan masih tetap di halaman dashboard karena belum logout.



6. Jika kita sudah logout maka akan diarahkan ke halaman login kembali.



E. TUGAS INDIVIDU

<https://github.com/aldinurhanudin/Proyek3-VmartMobile-/commit/b4c86c49aff40a30bc4244a41b385c071035331d>

1. Kode program tugas shared preferences pada halaman alamat baru proyek.

```
login - alamat_baru.dart

1 import 'package:flutter/material.dart';
2 import 'package:shared_preferences/shared_preferences.dart';
3
4 class AlamatBaru extends StatefulWidget {
5   // shared preference alamat baru
6   static String tag = 'AlamatBaru';
7   @override
8   _AlamatBaruState createState() => _AlamatBaruState();
9 }
10
11 class _AlamatBaruState extends State<AlamatBaru> {
12
13   final _namaLengkap = new TextEditingController();
14   final _alamat = new TextEditingController();
15   final _noHp = new TextEditingController();
16
17   static const String keyNamaLengkap = "prefKeyNama";
18   static const String keyAlamat = "prefKeyAlamat";
19   static const String keyNoHp = "prefKeyNoHp";
20
21   simpanData() async {
22     SharedPreferences pref = await SharedPreferences.getInstance();
23     pref.setString(keyNamaLengkap, _namaLengkap.text.toString());
24     pref.setString(keyAlamat, _alamat.text.toString());
25     pref.setString(keyNoHp, _noHp.text.toString());
26
27     print("Data Sudah Disimpan");
28   }
29
30   lihatData() async {
31     SharedPreferences pref = await SharedPreferences.getInstance();
32     String? namaLengkap = pref.getString(keyNamaLengkap);
33     String? alamat = pref.getString(keyAlamat);
34     String? noHp = pref.getString(keyNoHp);
35
36
37     print("nama lengkap : $namaLengkap alamat : $alamat No Hp : $noHp");
38   }
39
40   @override
41   Widget build(BuildContext context) {
42     return Scaffold(
43       appBar: AppBar(
44         title: Text("Tambah Alamat"),
45         backgroundColor: Colors.green,
46       ),
47       body: Container(
48         width: MediaQuery.of(context).size.width,
49         child: Container(
50           padding: const EdgeInsets.all(10),
```



```
logix - alamat_baru.dart

31 child Column(
32   children: [
33     SizedBox(height: 20,),
34     TextFormField(
35       controller: _namaLengkap,
36       decoration: InputDecoration(
37         hintText: "Masukkan Nama Lengkap",
38         labelText: "Nama Lengkap"
39       ),
40     ),
41     SizedBox(height: 20,),
42     TextFormField(
43       controller: _alamat,
44       decoration: InputDecoration(
45         hintText: "Masukkan Alamat",
46         labelText: "Alamat"
47       ),
48     ),
49     SizedBox(height: 20,),
50     TextFormField(
51       controller: _noHp,
52       decoration: InputDecoration(
53         hintText: "Masukkan NoHp",
54         labelText: "NoHp"
55       ),
56     ),
57     SizedBox(height: 10,),
58     Row(
59       children: [
60         OutlineButton(child: Text("Simpan"), onPressed: () => simpanData(),),
61         OutlineButton(child: Text("Lihat"), onPressed: () => lihatData(),),
62       ],
63     ),
64   ],
65 ),
66 ],
67 );
68 }
```

2. Tampilan program tugas shared preferences pada halaman alamat baru proyek. Kita inputkan nama lengkap, alamat dan noHp kemudian klik simpan.





- Setelah klik simpan maka Proses shared preferences akan berjalan jika berhasil maka akan ada informasi bahwa Data Sudah Disimpan

- Dan untuk melihat Data yang telah kita simpan maka klik tombol lihat setelah itu terdapat data yaitu nama lengkap Natasya dengan alamat di Bandung dengan noHp 0895778.

F. REFERENSI

- Alberto Miola. "Flutter Complete Reference Create Beautiful, Fast and Native Apps for Any Device".Independently Published. 2020.
- Simone Alessandria, Brian Kayfirz. "Flutter Cookbook: Over 100 proven techniques and solutions for app development with Flutter 2.2 and Dart". Packt Publishing. Birmingham - Mumbai. 2021.
- Dieter Meiller. "Modern App Development with Dart and Flutter 2: A Comprehensive Introduction to Flutter". Walter de Gruyter GmbH. Berlin - Boston. 2021.
- Priyanka Tyagi. "Pragmatic Flutter: Building Cross-Platform Mobile Apps for Android, iOS, Web & Desktop". CRC Press Taylor & Francis Group, LLC. London - New York. 2022.