**Crud operation**

**Database**

import 'package:sqflite/sqflite.dart';

import 'package:path/path.dart';

class MyDataBase {

Future<Database> initDatabase() async {

return await openDatabase(

join(await getDatabasesPath(), 'MyDatabase.db'),

version: 2,

onCreate: (db, version) {

db.execute(

'CREATE TABLE Tbl\_User(UserId INTEGER PRIMARY KEY AUTOINCREMENT, UserName TEXT)');

},

onUpgrade: (db, oldVersion, newVersion) {

db.execute(

'CREATE TABLE Tbl\_User(UserId INTEGER PRIMARY KEY AUTOINCREMENT, UserName TEXT)');

},

);

}

Future<int> insertUserDetailInTblUser({required userName}) async {

Database db = await initDatabase();

Map<String, dynamic> map = {};

map['UserName'] = userName;

int pk = await db.insert('Tbl\_User', map);

return pk;

}

Future<int> updateUserDetailInTblUser({required userName,required userId}) async {

Database db =await initDatabase();

Map<String,dynamic> map = {};

map['UserName'] = userName;

int pk = await db.update('Tbl\_User', map,where: 'UserId = ?', whereArgs: [userId]);

return pk;

}

Future<List<Map<String, dynamic>>> getUserListFromTblUser() async {

Database db = await initDatabase();

List<Map<String, dynamic>> userList =

await db.rawQuery('SELECT \* FROM Tbl\_User');

return userList;

}

Future<int> deleteUserFromTblUser({required userId}) async {

Database db = await initDatabase();

int pk =

await db.delete('Tbl\_User', where: 'UserId = ?', whereArgs: [userId]);

return pk;

}

}

**Data addd and edit**

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

import 'package:flutter\_4b/my\_database.dart';

import 'package:sqflite/sqflite.dart';

class UserEntryPage extends StatefulWidget {

Map? userDetail;

UserEntryPage({super.key, required this.userDetail});

@override

State<UserEntryPage> createState() => \_UserEntryPageState();

}

class \_UserEntryPageState extends State<UserEntryPage> {

TextEditingController userNameController = TextEditingController();

TextEditingController cityController = TextEditingController();

GlobalKey<FormState> \_validationKey = GlobalKey();

MyDataBase db = MyDataBase();

@override

void initState() {

super.initState();

userNameController.text =

widget.userDetail != null ? widget.userDetail!['UserName'] : '';

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text(

'User Detail',

style: TextStyle(

color: Colors.white,

),

),

backgroundColor: Theme.of(context).primaryColor,

),

body: Form(

key: \_validationKey,

child: Padding(

padding: const EdgeInsets.only(

left: 10,

right: 10,

top: 20,

),

child: Column(

children: [

TextFormField(

decoration: InputDecoration(

border: OutlineInputBorder(

borderRadius: BorderRadius.circular(

20,

),

),

labelText: 'User Name',

hintStyle: TextStyle(

fontSize: 20,

),

hintText: 'Enter User Name'),

validator: (value) {

if (value!.isEmpty) {

return 'Please Enter Name';

}

},

controller: userNameController,

),

const SizedBox(

height: 20,

),

TextFormField(

decoration: InputDecoration(

border: OutlineInputBorder(

borderRadius: BorderRadius.circular(

20,

),

),

hintText: 'Enter City Name',

labelText: 'City'),

validator: (value) {

if (value!.isEmpty) {

return 'Please Enter City Name';

}

},

controller: cityController,

),

const SizedBox(

height: 20,

),

ElevatedButton(

style: ElevatedButton.styleFrom(

backgroundColor: Colors.green,

),

onPressed: () async {

if (\_validationKey.currentState!.validate()) {

if(widget.userDetail==null) {

await db.insertUserDetailInTblUser(

userName: userNameController.text.toString());

}else{

await db.updateUserDetailInTblUser(

userName: userNameController.text.toString(),

userId: widget.userDetail!['UserId'],);

}

Navigator.pop(context, true);

}

},

child: const Text(

'SUBMIT',

style: TextStyle(color: Colors.white, fontSize: 30),

),

),

],

),

),

),

);

}

void showAlert(ctx, title, id) {

showCupertinoDialog(

context: ctx,

builder: (context) {

return CupertinoAlertDialog(

title: Text(title),

actions: [

TextButton(

onPressed: () {

if (id >= 1) {

Navigator.of(ctx).pop();

Navigator.of(ctx).pop(true);

} else {

Navigator.pop(ctx);

}

},

child: Text('Ok'))

],

);

},

);

}

}

**Display user list**

import 'package:flutter/cupertino.dart';

import 'package:flutter/material.dart';

import 'package:flutter/widgets.dart';

import 'package:flutter\_4b/my\_database.dart';

import 'package:flutter\_4b/user\_entery\_page.dart';

import 'package:sqflite/sqflite.dart';

class UserListPage extends StatefulWidget {

@override

State<UserListPage> createState() => \_UserListPageState();

}

class \_UserListPageState extends State<UserListPage> {

TextEditingController filterController = TextEditingController();

MyDataBase db = MyDataBase();

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text(

'User List',

style: TextStyle(

color: Colors.white,

),

),

actions: [

Padding(

padding: const EdgeInsets.all(8.0),

child: InkWell(

child: Icon(

Icons.add,

size: 25,

color: Colors.white,

),

onTap: () {

Navigator.of(context).push(

MaterialPageRoute(

builder: (context) {

return UserEntryPage(

userDetail: null,

);

},

),

).then((value) {

if (value) {

setState(() {});

}

});

},

),

),

],

backgroundColor: Theme.of(context).primaryColor,

),

body: FutureBuilder(

future: db.getUserListFromTblUser(),

builder: (context, snapshot) {

if (snapshot.data != null) {

if (snapshot.data!.isNotEmpty) {

return ListView.builder(

itemBuilder: (context, index) {

return Card(

child: Padding(

padding: const EdgeInsets.all(8.0),

child: Row(

children: [

Expanded(

child: Text(

snapshot.data![index]['UserName'].toString(),

style: TextStyle(

fontSize: 20,

),

),

),

IconButton(

onPressed: () {

Navigator.of(context).push(MaterialPageRoute(

builder: (context) {

return UserEntryPage(

userDetail: snapshot.data![index],

);

},

)).then((value) {

if (value) {

setState(() {});

}

});

},

icon: Icon(Icons.edit),

),

IconButton(

onPressed: () {

showDeleteAlert(

userId: snapshot.data![index]['UserId']);

},

icon: Icon(

Icons.delete,

color: Colors.red,

),

),

],

),

),

);

},

itemCount: snapshot.data!.length,

);

} else {

return Center(

child: Text('No User Found'),

);

}

} else {

return CircularProgressIndicator();

}

},

),

);

}

showDeleteAlert({required userId}) {

showCupertinoDialog(

context: context,

builder: (context) {

return CupertinoAlertDialog(

title: Text('Alert!'),

content: Text('Are you sure want to delete?'),

actions: [

TextButton(

onPressed: () {

db.deleteUserFromTblUser(userId: userId);

Navigator.pop(context);

setState(() {});

},

child: Text('Yes'),

),

TextButton(

onPressed: () {

Navigator.pop(context);

},

child: Text('No'),

),

],

);

},

);

}

}

**Sharedpref**

import 'package:flutter/material.dart';

import 'package:flutter\_4b/my\_database.dart';

import 'package:flutter\_4b/post\_new\_ad\_page.dart';

import 'package:flutter\_4b/pre\_login\_screen.dart';

import 'package:flutter\_4b/user\_list\_page.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import 'bottom\_nav\_page.dart';

void main() {

runApp(MyApp());

}

class MyApp extends StatefulWidget {

MyApp({super.key});

@override

State<MyApp> createState() => \_MyAppState();

}

class \_MyAppState extends State<MyApp> {

@override

void initState() {

super.initState();

}

@override

Widget build(BuildContext context) {

return MaterialApp(

debugShowCheckedModeBanner: false,

title: 'Flutter Demo',

theme: ThemeData(

colorScheme: ColorScheme.fromSeed(seedColor: Colors.blue),

useMaterial3: true,

primaryColor: Colors.blue,

shadowColor: Colors.black45,

),

home: FutureBuilder(

future: MyDataBase().initDatabase(),

builder: (context, snapshot) {

if (snapshot.hasData) {

if (snapshot.data != null) {

return FutureBuilder(

future: SharedPreferences.getInstance(),

builder: (context, snapshot1) {

if (snapshot1.data != null) {

if (snapshot1.data!.getBool('IsUserLogin') != null &&

snapshot1.data!.getBool('IsUserLogin')!) {

return UserListPage();

} else {

return PreLoginScreenPage();

}

} else {

return Center(

child: Text('ERROR'),

);

}

},

);

} else {

return Center(

child: Text(snapshot.error.toString()),

);

}

} else {

return CircularProgressIndicator();

}

},

),

);

}

}

**Set preference**

import 'package:flutter/material.dart';

import 'package:flutter\_4b/user\_list\_page.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

class PreLoginScreenPage extends StatelessWidget {

@override

Widget build(BuildContext context) {

return Scaffold(

body: Column(

children: [

Expanded(

child: Stack(

fit: StackFit.expand,

children: [

Image.asset(

'assets/images/bg\_1.jpg',

fit: BoxFit.cover,

),

Container(color: Colors.black38),

Column(

children: [

SizedBox(

height: 100,

),

Container(

width: 200,

height: 100,

decoration: BoxDecoration(

color: Colors.orange,

borderRadius: BorderRadius.circular(

10,

),

),

),

Text(

'INDIA\'S\nMOST TRUSTED MATRIMONY BRAND',

textAlign: TextAlign.center,

style: TextStyle(

color: Colors.white,

fontSize: 30,

fontWeight: FontWeight.bold,

),

),

Container(

child: Text(

'THIS IS NEW YEAR HAPPY NEW YEAR',

textAlign: TextAlign.center,

style: TextStyle(

color: Colors.white,

fontSize: 15,

fontWeight: FontWeight.bold,

),

),

),

],

)

],

),

),

Row(children: [

Expanded(

child: InkWell(

onTap: () async {

SharedPreferences pref =

await SharedPreferences.getInstance();

await pref.setBool('IsUserLogin', true);

await pref.setString('UserName', 'Mehul');

await pref.setString('Password', 'ABC');

Navigator.of(context).pushReplacement(

MaterialPageRoute(

builder: (context) {

return UserListPage();

},

),

);

},

child: Container(

padding: EdgeInsets.all(30),

color: Colors.green,

child: Center(

child: Text(

'LOGIN',

style: TextStyle(

color: Colors.white,

fontSize: 20,

fontWeight: FontWeight.bold,

),

),

),

),

),

),

Expanded(

child: Container(

padding: EdgeInsets.all(30),

color: Colors.black,

child: Row(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text(

'SIGN UP',

style: TextStyle(

color: Colors.white,

fontSize: 20,

fontWeight: FontWeight.bold,

),

),

SizedBox(

width: 10,

),

Icon(

Icons.arrow\_right\_alt\_sharp,

color: Colors.white,

)

],

),

),

),

]),

],

),

);

}

}

**OR**

ElevatedButton(

onPressed: () async {

String username = \_username.text;

String password = \_password.text;

SharedPreferences preference = await SharedPreferences.getInstance();

await preference.setString('ScreenName', 'PreLoginPage');

Navigator.of(context).push(

MaterialPageRoute(

builder: (context) {

return NavDemo();

},

),

);

if (\_formkey.currentState!.validate()) {

setState(() {

text=\_username.text;

text=\_password.text;

});

print('Login successful');

}

else{

setState(() {

\_loginMessage = 'Login failed';

});

print('Login failed');

}

},

child: Padding(

padding: const EdgeInsets.only(left: 20, right: 20),

child: Container(

height: 60,

width: double.infinity,

decoration: BoxDecoration(

color: Colors.blue,

borderRadius: BorderRadiusDirectional.circular(10),

),

child: Center(

child: Text(

"Sign In",

style: TextStyle(

fontFamily: "lexend-medium",

fontSize: 18,

color: Colors.white,

),

),

),

),

),

),

**NavBar DEMO**

import 'package:demo2/Dats\_Entry.dart';

import 'package:demo2/User\_list.dart';

import 'package:flutter/material.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

class NavDemo extends StatefulWidget {

@override

\_NavDemoState createState() => \_NavDemoState();

}

class \_NavDemoState extends State<NavDemo> {

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(title: Text("NavigationDrawer")),

drawer: Drawer(

backgroundColor: Colors.grey,

child: ListView(

children: [

DrawerHeader(

child: Container(

child: Image.network(

"https://images.unsplash.com/photo-1525609004556-c46c7d6cf023?q=80&w=1000&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxzZWFyY2h8Nnx8Y2Fyc3xlbnwwfHwwfHx8MA%3D%3D"),

)),

ListTile(

title: ElevatedButton(

onPressed: () {

Navigator.of(context).push(

MaterialPageRoute(

builder: (context) {

return User\_list();

},

),

);

},

child: Text("user list")),

trailing: Icon(Icons.home),

),

ListTile(

title: Text("balance"),

trailing: Icon(Icons.account\_balance\_outlined),

),

ListTile(

title: Text("power"),

trailing: Icon(Icons.offline\_bolt),

),

ListTile(

title: Text("Contect"),

trailing: Icon(Icons.call),

),

],

),

),

// floatingActionButton: FloatingActionButton(

// child: Icon(Icons.add),

// backgroundColor: Colors.white70,

// onPressed: () {

// Navigator.of(context).push(

// MaterialPageRoute(

// builder: (context) {

// return DataEntry(null);

// // return EnterDetail(Detail: null,);

// },

// ),

// );

// },

// ),

// floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,

body: FutureBuilder(

future: SharedPreferences.getInstance(),

builder: (context, snapshot) {

if (snapshot.hasData) {

if (snapshot.data != null) {

return Center(

child: Text(

snapshot.data!.getString('ScreenName').toString(),

style: TextStyle(

color: Colors.red,

fontSize: 25,

),

),

);

} else {

return Text('No Data Found');

}

} else {

return CircularProgressIndicator();

}

},

),

);

}

}

**TabView**

// import 'dart:html';

import 'package:flutter/material.dart';

class Tab1 extends StatefulWidget {

const Tab1({super.key});

@override

\_Tab1State createState() => \_Tab1State();

}

class \_Tab1State extends State<Tab1> with TickerProviderStateMixin {

TabController? tabcontroller;

@override

void initState() {

tabcontroller = TabController(length: 3, vsync: this);

}

@override

Widget build(BuildContext context) {

return SafeArea(

child:Scaffold(

drawer: Drawer(

backgroundColor: Colors.grey,

child: ListView(

children: [

DrawerHeader(child: Container(

child: Image.network("https://images.unsplash.com/photo-1525609004556-c46c7d6cf023?q=80&w=1000&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxzZWFyY2h8Nnx8Y2Fyc3xlbnwwfHwwfHx8MA%3D%3D"),

)

),

ListTile(title: Text("Home"),trailing: Icon(Icons.home),),

ListTile(title: Text("balance"),trailing: Icon(Icons.account\_balance\_outlined),),

ListTile(title: Text("power"),trailing: Icon(Icons.offline\_bolt),),

ListTile(title: Text("Contect"),trailing: Icon(Icons.call),),

],

),

),

appBar: AppBar(

title: Title(

color: Colors.blueAccent,

child: Text("data")

),

actions: [

IconButton(onPressed: () {

showDialog(context: context,

builder: (context) {

return AlertDialog(

title: Text("Data not found"),

actions: [

ElevatedButton(onPressed: () {

Navigator.of(context).pop();

}, child: Text("ok")),

ElevatedButton(onPressed: () {

Navigator.of(context).pop();

}, child: Text("cancle")),

],

);

},);

}, icon: Icon(Icons.data\_object)

)

],

backgroundColor: Colors.blueAccent,

bottom: TabBar(

controller:tabcontroller,

tabs:[

Tab(text:"page1"),

Tab(text:"page2"),

Tab(text:"page3")

],

),

),

body: TabBarView(controller: tabcontroller,

children: [

Center(child: Text("1")),

Center(child: Text("2")),

Center(child: Text("3")),

]),

)

);

}

}

**login validation use regex form**

import 'package:flutter/material.dart';

// import 'package:designs/pages/homepage.dart';

class form\_demo1 extends StatefulWidget {

const form\_demo1({Key? key}) : super(key: key);

@override

\_form\_demo1State createState() => \_form\_demo1State();

}

class \_form\_demo1State extends State<form\_demo1> {

bool? check1 = false;

TextEditingController \_username = TextEditingController();

TextEditingController \_password = TextEditingController();

String \_loginMessage = '';

String text ="hello";

final \_formkey = GlobalKey<FormState>();

@override

Widget build(BuildContext context) {

return SafeArea(

child: Scaffold(

body: SingleChildScrollView(

child: Form(

key: \_formkey,

child: Column(

mainAxisAlignment: MainAxisAlignment.spaceBetween,

children: [

Padding(

padding: const EdgeInsets.only(left: 25, top: 20),

child: Text(

"Sign In",

style: TextStyle(

fontFamily: "lexend-bold",

fontSize: 35,

),

),

),

SizedBox(

height: 20,

),

Padding(

padding: const EdgeInsets.only(

top: 15,

left: 20,

right: 20,

bottom: 10,

),

child: Container(

height: 50,

decoration: BoxDecoration(

color: Colors.grey.withOpacity(0.1),

boxShadow: [

BoxShadow(

color: Colors.grey.withOpacity(0.1),

spreadRadius: 0.1,

blurRadius: 15,

),

],

borderRadius: BorderRadius.circular(10.0),

),

child: Padding(

padding: const EdgeInsets.only(left: 20, right: 10),

child: TextFormField(

validator: (value) {

if(value!.isEmpty){

return 'please enter name';

}

String pattern="[a-z][A-Z]";

var emailRegExp=RegExp(pattern);

if(!emailRegExp.hasMatch(value!)){

return "please enter valid name";

}

return null;

},

controller: \_username,

cursorColor: Colors.blue,

style: TextStyle(fontFamily: "lexend-bold"),

decoration: InputDecoration(

border: InputBorder.none,

hintText: 'User Name',

),

),

),

),

),

Padding(

padding: const EdgeInsets.only(

top: 15,

left: 20,

right: 20,

bottom: 10,

),

child: Container(

height: 50,

decoration: BoxDecoration(

color: Colors.grey.withOpacity(0.1),

boxShadow: [

BoxShadow(

color: Colors.grey.withOpacity(0.1),

spreadRadius: 0.1,

blurRadius: 15,

),

],

borderRadius: BorderRadius.circular(10.0),

),

child: Padding(

padding: const EdgeInsets.only(left: 20, right: 10),

child: TextFormField(

validator: (value) {

if(value!.isEmpty){

return 'please enter passward';

}

String pattern="[a-z][A-Z]";

var emailRegExp=RegExp(pattern);

if(!emailRegExp.hasMatch(value!)){

return "please enter valid name";

}

return null;

},

controller: \_password,

cursorColor: Colors.blue,

style: TextStyle(fontFamily: "lexend-bold"),

decoration: InputDecoration(

border: InputBorder.none,

hintText: 'Password',

),

),

),

),

),

Padding(

padding: const EdgeInsets.only(left: 10, right: 10),

child: Row(

mainAxisAlignment: MainAxisAlignment.spaceBetween,

children: [

Row(

children: [

Checkbox(

value: check1,

activeColor: Colors.blue,

onChanged: (bool? value) {

setState(() {

check1 = value;

});

},

),

Text(

"Remember Me",

style: TextStyle(

fontFamily: "lexend-medium",

fontSize: 16,

),

)

],

),

Text(

"Forget Password?",

style: TextStyle(

fontFamily: "lexend-medium",

color: Colors.blue,

fontSize: 16,

),

)

],

),

),

ElevatedButton(

onPressed: () {

String username = \_username.text;

String password = \_password.text;

if (\_formkey.currentState!.validate()) {

setState(() {

text=\_username.text;

text=\_password.text;

});

print('Login successful');

}

else{

setState(() {

\_loginMessage = 'Login failed';

});

print('Login failed');

}

},

child: Padding(

padding: const EdgeInsets.only(left: 20, right: 20),

child: Container(

height: 60,

width: double.infinity,

decoration: BoxDecoration(

color: Colors.blue,

borderRadius: BorderRadiusDirectional.circular(10),

),

child: Center(

child: Text(

"Sign In",

style: TextStyle(

fontFamily: "lexend-medium",

fontSize: 18,

color: Colors.white,

),

),

),

),

),

),

SizedBox(

height: 50,

),

Row(

children: [

Expanded(

child: Divider(

color: Colors.blue,

indent: 15,

endIndent: 15,

thickness: 2,

),

),

Text(

"OR",

style: TextStyle(

fontFamily: "lexend-medium",

color: Colors.blue,

),

),

Expanded(

child: Divider(

color: Colors.blue,

indent: 15,

endIndent: 15,

thickness: 2,

),

),

],

),

SizedBox(

height: 20,

),

Center(

child: Text(

"Sign In With",

style: TextStyle(

fontFamily: "lexend-medium",

fontSize: 18,

),

),

),

Padding(

padding: const EdgeInsets.only(top: 20, bottom: 10, left: 5, right: 5),

child: Row(

mainAxisAlignment: MainAxisAlignment.spaceEvenly,

children: [

Expanded(

child: Padding(

padding: const EdgeInsets.only(left: 15, right: 7),

child: Container(

height: 60,

decoration: BoxDecoration(

borderRadius: BorderRadiusDirectional.circular(10),

color: Colors.grey.withOpacity(0.2),

),

child: Padding(

padding: const EdgeInsets.all(4.0),

child: Image.asset(

"assets/images/facebook.png",

),

),

),

),

),

Expanded(

child: Padding(

padding: const EdgeInsets.only(left: 7, right: 15),

child: Container(

height: 60,

decoration: BoxDecoration(

borderRadius: BorderRadiusDirectional.circular(10),

color: Colors.grey.withOpacity(0.2),

),

child: Image.asset(

"assets/images/google.png",

),

),

),

),

],

),

),

Padding(

padding: const EdgeInsets.all(8.0),

child: Row(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text(

"Don't have an account?",

style: TextStyle(

fontFamily: "lexend-medium",

fontSize: 18,

),

),

Text(

" Sign up",

style: TextStyle(

fontFamily: "lexend-medium",

fontSize: 18,

color: Colors.blue,

),

),

],

),

),

SizedBox(

height: 40,

),

],

),

),

),

),

);

}

}