**Assignment**

By

**Suleman Anwar FA19-BCS-061**

Course: Mobile Application Development

Topic: Xylophone App

**Submitted to: Sir Abdullah**

Date: 29-03-2022

Department of Computer Science

COMSATS University Islamabad, Vehari campusLogo, company name

Description automatically generated

**Main** (Code)

import "package:flutter/material.dart";  
import 'package:audioplayers/audioplayers.dart';  
import 'package:flutter/rendering.dart';  
import 'package:splash\_screen\_view/SplashScreenView.dart';  
import 'package:flutter\_colorpicker/flutter\_colorpicker.dart';  
  
void main() {  
 runApp(  
 MaterialApp(  
 debugShowCheckedModeBanner: false,  
 home: myapp(),  
 ),  
 );  
}  
  
class myapp extends StatelessWidget {  
  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 debugShowCheckedModeBanner: false,  
 home:Scaffold(  
 body:  
 SplashScreenView(  
 navigateRoute: colorpicker(),  
 duration: 6000,  
 imageSize: 200,  
 imageSrc: "images/icon.png",  
 backgroundColor: Colors.cyan,  
 text: 'Welcome',textStyle:TextStyle(fontSize: 30,color: Colors.amberAccent),  
 ),  
 )  
 );  
 }  
}  
  
  
  
  
class colorpicker extends StatefulWidget {  
 @override  
 State<colorpicker> createState() => \_colorpickerState();  
}  
  
class \_colorpickerState extends State<colorpicker> {  
 Color n1 = Colors.redAccent;  
 Color n2 = Colors.blue;  
 Color n3 = Colors.green;  
 Color n4 = Colors.yellow;  
 Color n5 = Colors.purple;  
 Color n6 = Colors.deepOrange;  
 Color n7 = Colors.brown;  
  
 List<int> items =[1, 2, 3, 4, 5,6,7];  
 int a1 =1;  
 int a2=2;  
 int a3=3;  
 int a4=4;  
 int a5=5;  
 int a6=6;  
 int a7=7;  
  
  
 void changeColor1(Color color) {  
 setState(() {  
 n1=color;  
 });  
 }  
 void changeColor2(Color color) {  
 setState(() {  
 n2=color;  
 });  
 }  
 void changeColor3(Color color) {  
 setState(() {  
 n3=color;  
 });  
 }  
 void changeColor4(Color color) {  
 setState(() {  
 n4=color;  
 });  
 }  
 void changeColor5(Color color) {  
 setState(() {  
 n5=color;  
 });  
 }  
 void changeColor6(Color color) {  
 setState(() {  
 n6=color;  
 });  
 }  
 void changeColor7(Color color) {  
 setState(() {  
 n7=color;  
 });  
 }  
  
  
 @override  
 Widget build(BuildContext context) {  
 return MaterialApp(  
 home: SafeArea(  
 child: Scaffold(  
 body: Column(  
 children: [  
 Expanded(child:  
 Container(  
 color: Colors.cyan.shade100,  
  
 child: Column(  
 children: [  
 SizedBox(  
 height: 60,  
 ),  
  
 Row(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: [  
 Text('Here, You can change Xialophone\ncolors & Audio notes sequence',style: TextStyle(color: Colors.brown,fontFamily: 'Bold',fontSize: 20),)  
 ],  
 ),  
 SizedBox(  
 height: 20,  
 ),  
 Container(  
 decoration: BoxDecoration(  
 border: Border.all(color:Colors.blueGrey)  
 ),  
 child: Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
  
 Text("Change Color",style: TextStyle(fontSize: 20,fontFamily: 'Fonts Pacifico',color: Colors.blueGrey.shade900)),  
  
 Column(  
 children: [  
 Text("Change Audio",style: TextStyle(fontSize: 20,fontFamily: 'Fonts Pacifico',color: Colors.blueGrey.shade900)),  
 ]  
 ),  
 ],  
 ),  
 ),  
 SizedBox(  
 height: 20,  
 ),  
 Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
  
 ElevatedButton(  
 style: ButtonStyle(  
 backgroundColor: MaterialStateProperty.all(n1),  
 shape:MaterialStateProperty.all(  
 RoundedRectangleBorder(  
 borderRadius: BorderRadius.circular(18.0),  
 side: BorderSide(color:Colors.deepPurple),  
  
 )  
 )  
 ),  
 onPressed: () {  
 showDialog(  
 context: context,  
 builder: (context) => AlertDialog(  
 content: SingleChildScrollView(  
 child: ColorPicker(  
 onColorChanged: changeColor1,  
 pickerColor: n1,  
  
 ),  
 ),  
 ),  
 );  
 },  
 child: Text("Change Color")),  
  
 Column(  
 children: [  
 SizedBox(  
 height: 60,  
 width:80,  
 child: DropdownButtonFormField<int>(  
 decoration: InputDecoration(  
 enabledBorder: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(12),  
 borderSide: BorderSide(width:3,color:n1)  
 ),  
 ),  
 value: a1,  
 items: items.map((item)=> DropdownMenuItem<int>(  
 value: item,  
 child: Text('$item',style: TextStyle(fontSize: 12,fontFamily: 'Bold'),),  
 ))  
 .toList(),  
  
 onChanged:(item)=> setState(()=> a1=item!)  
 ),  
 ),  
 ]  
 ),  
 ],  
 ),  
  
  
  
 SizedBox(  
 height: 20,  
 ),  
 Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
  
  
 ElevatedButton(  
 style: ButtonStyle(  
 backgroundColor: MaterialStateProperty.all(n2),  
 shape:MaterialStateProperty.all(  
 RoundedRectangleBorder(  
 borderRadius: BorderRadius.circular(18.0),  
 side: BorderSide(color:Colors.deepPurple),  
  
 )  
 )  
 ),  
 onPressed: () {  
 showDialog(  
 context: context,  
 builder: (context) => AlertDialog(  
 content: SingleChildScrollView(  
 child: ColorPicker(  
 onColorChanged: changeColor2,  
 pickerColor: n2,  
  
 ),  
 ),  
 ),  
 );  
 },  
 child: Text("Change Color")),  
  
 Column(  
 children: [  
 SizedBox(  
 height:60,  
 width:80,  
 child: DropdownButtonFormField<int>(  
 decoration: InputDecoration(  
 enabledBorder: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(12),  
 borderSide: BorderSide(width:3,color:n2)  
 ),  
 ),  
 value: a2,  
 items: items.map((item)=> DropdownMenuItem<int>(  
 value: item,  
 child: Text('$item',style: TextStyle(fontSize: 12,fontFamily: 'Bold'),),  
 ))  
 .toList(),  
  
 onChanged:(item)=> setState(()=> a2=item!)  
 ),  
 ),  
 ]  
 ),  
  
 ],  
  
 ),  
  
 SizedBox(  
 height: 20,  
 ),  
 Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
  
 ElevatedButton(  
 style: ButtonStyle(  
 backgroundColor: MaterialStateProperty.all(n3),  
 shape:MaterialStateProperty.all(  
 RoundedRectangleBorder(  
 borderRadius: BorderRadius.circular(18.0),  
 side: BorderSide(color:Colors.deepPurple),  
  
 )  
 )  
 ),  
 onPressed: () {  
 showDialog(  
 context: context,  
 builder: (context) => AlertDialog(  
 content: SingleChildScrollView(  
 child: ColorPicker(  
 onColorChanged: changeColor3,  
 pickerColor: n3,  
  
 ),  
 ),  
 ),  
 );  
 },  
 child: Text("Change Color")),  
 Column(  
 children: [  
 SizedBox(  
 height:60,  
 width:80,  
 child: DropdownButtonFormField<int>(  
 decoration: InputDecoration(  
 enabledBorder: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(12),  
 borderSide: BorderSide(width:3,color:n3)  
 ),  
 ),  
 value: a3,  
 items: items.map((item)=> DropdownMenuItem<int>(  
 value: item,  
 child: Text('$item',style: TextStyle(fontSize: 12,fontFamily: 'Bold'),),  
 ))  
 .toList(),  
  
 onChanged:(item)=> setState(()=> a3=item!)  
 ),  
 ),  
 ]  
 ),  
 ],  
 ),  
  
 SizedBox(  
 height: 20,  
 ),  
 Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
  
 ElevatedButton(  
 style: ButtonStyle(  
 backgroundColor: MaterialStateProperty.all(n4),  
 shape:MaterialStateProperty.all(  
 RoundedRectangleBorder(  
 borderRadius: BorderRadius.circular(18.0),  
 side: BorderSide(color:Colors.deepPurple),  
  
 )  
 )  
 ),  
 onPressed: () {  
 showDialog(  
 context: context,  
 builder: (context) => AlertDialog(  
 content: SingleChildScrollView(  
 child: ColorPicker(  
 onColorChanged: changeColor4,  
 pickerColor: n4,  
  
 ),  
 ),  
 ),  
 );  
 },  
 child: Text("Change Color")),  
 Column(  
 children: [  
 SizedBox(  
 height:60,  
 width:80,  
 child: DropdownButtonFormField<int>(  
 decoration: InputDecoration(  
 enabledBorder: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(12),  
 borderSide: BorderSide(width:3,color:n4)  
 ),  
 ),  
 value: a4,  
 items: items.map((item)=> DropdownMenuItem<int>(  
 value: item,  
 child: Text('$item',style: TextStyle(fontSize: 12,fontFamily: 'Bold'),),  
 ))  
 .toList(),  
  
 onChanged:(item)=> setState(()=> a4=item!)  
 ),  
 ),  
 ]  
 ),  
 ],  
 ),  
  
 SizedBox(  
 height: 20,  
 ),  
 Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
  
 ElevatedButton(  
 style: ButtonStyle(  
 backgroundColor: MaterialStateProperty.all(n5),  
 shape:MaterialStateProperty.all(  
 RoundedRectangleBorder(  
 borderRadius: BorderRadius.circular(18.0),  
 side: BorderSide(color:Colors.deepPurple),  
  
 )  
 )  
 ),  
 onPressed: () {  
 showDialog(  
 context: context,  
 builder: (context) => AlertDialog(  
 content: SingleChildScrollView(  
 child: ColorPicker(  
 onColorChanged: changeColor5,  
 pickerColor: n5,  
  
 ),  
 ),  
 ),  
 );  
 },  
 child: Text("Change Color")),  
 Column(  
 children: [  
 SizedBox(  
 height:60,  
 width:80,  
 child: DropdownButtonFormField<int>(  
 decoration: InputDecoration(  
 enabledBorder: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(12),  
 borderSide: BorderSide(width:3,color:n5)  
 ),  
 ),  
 value: a5,  
 items: items.map((item)=> DropdownMenuItem<int>(  
 value: item,  
 child: Text('$item',style: TextStyle(fontSize: 12,fontFamily: 'Bold'),),  
 ))  
 .toList(),  
  
 onChanged:(item)=> setState(()=> a5=item!)  
 ),  
 ),  
 ]  
 ),  
 ],  
 ),  
  
 SizedBox(  
 height: 20,  
 ),  
 Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
  
 ElevatedButton(  
 style: ButtonStyle(  
 backgroundColor: MaterialStateProperty.all(n6),  
 shape:MaterialStateProperty.all(  
 RoundedRectangleBorder(  
 borderRadius: BorderRadius.circular(18.0),  
 side: BorderSide(color:Colors.deepPurple),  
  
 )  
 )  
 ),  
 onPressed: () {  
 showDialog(  
 context: context,  
 builder: (context) => AlertDialog(  
 content: SingleChildScrollView(  
 child: ColorPicker(  
 onColorChanged: changeColor6,  
 pickerColor: n6,  
  
 ),  
 ),  
 ),  
 );  
 },  
 child: Text("Change Color")),  
 Column(  
 children: [  
 SizedBox(  
 height: 60,  
 width:80,  
 child: DropdownButtonFormField<int>(  
 decoration: InputDecoration(  
 enabledBorder: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(12),  
 borderSide: BorderSide(width:3,color:n6)  
 ),  
 ),  
 value: a6,  
 items: items.map((item)=> DropdownMenuItem<int>(  
 value: item,  
 child: Text('$item',style: TextStyle(fontSize: 12,fontFamily: 'Bold'),),  
 ))  
 .toList(),  
  
 onChanged:(item)=> setState(()=> a6=item!)  
 ),  
 ),  
 ]  
 ),  
 ],  
 ),  
  
 SizedBox(  
 height: 20,  
 ),  
 Row(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: [  
  
 ElevatedButton(  
 style: ButtonStyle(  
 backgroundColor: MaterialStateProperty.all(n7),  
 shape:MaterialStateProperty.all(  
 RoundedRectangleBorder(  
 borderRadius: BorderRadius.circular(18.0),  
 side: BorderSide(color:Colors.deepPurple),  
  
 )  
 )  
 ),  
 onPressed: () {  
 showDialog(  
 context: context,  
 builder: (context) => AlertDialog(  
 content: SingleChildScrollView(  
 child: ColorPicker(  
 onColorChanged: changeColor7,  
 pickerColor: n7,  
  
 ),  
 ),  
 ),  
 );  
 },  
 child: Text("Change Color")),  
 Column(  
 children: [  
 SizedBox(  
 height: 60,  
 width:80,  
 child: DropdownButtonFormField<int>(  
 decoration: InputDecoration(  
 enabledBorder: OutlineInputBorder(  
 borderRadius: BorderRadius.circular(12),  
 borderSide: BorderSide(width:3,color:n7)  
 ),  
 ),  
 value: a7,  
 items: items.map((item)=> DropdownMenuItem<int>(  
 value: item,  
 child: Text('$item',style: TextStyle(fontSize: 12,fontFamily: 'Bold'),),  
 ))  
 .toList(),  
  
 onChanged:(item)=> setState(()=> a7=item!)  
 ),  
 ),  
 ]  
 ),  
 ],  
 ),  
  
 SizedBox(  
 height: 10,  
 ),  
 Row(  
 mainAxisAlignment: MainAxisAlignment.center,  
 children: [  
 ElevatedButton(onPressed: (){  
 Navigator.push(  
 context,  
 MaterialPageRoute(builder: (context) =>  
 Xylo(color1:n1,color2:n2,color3:n3,color4:n4,color5:n5,color6:n6,color7:n7,  
 au1:a1, au2: a2,au3: a3,au4: a4,au5: a5,au6: a6,au7: a7,)));  
  
  
 }, child: Text("Continue"))  
 ],  
  
 ),  
  
 ],  
 ),  
 ),  
 )  
 ],  
 ),  
 ),  
  
 ),  
 );  
  
 }  
}  
  
class Xylo extends StatelessWidget {  
 Xylo({required this.color1,required this.color2,required this.color3, required this.color4,  
 required this.color5, required this.color6,required this.color7, required this.au1, required this.au2,  
 required this.au3,required this.au4,required this.au5,required this.au6,required this.au7});  
  
 final Color color1;  
 final Color color2;  
 final Color color3;  
 final Color color4;  
 final Color color5;  
 final Color color6;  
 final Color color7;  
 final int au1,au2,au3,au4,au5,au6,au7;  
  
  
  
 void playsound(int value) {  
 final player = AudioCache();  
 player.play('note$value.wav');  
 }  
  
  
 Expanded button(int num, Color color)  
 {  
 return Expanded(  
 child: Padding(  
 padding: const EdgeInsets.all(8.0),  
 child: TextButton(onPressed: (){  
 playsound(num);  
 }, child: Center(child: Text("$num"),),  
 style: TextButton.styleFrom(  
 primary: Colors.white,  
 backgroundColor: color,  
 textStyle: const TextStyle(  
 fontWeight: FontWeight.bold,  
 fontSize: 30.0,  
  
 ),  
 shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(30),)  
 ),  
 ),  
 ),  
 );  
 }  
  
  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(  
 backgroundColor: Colors.blueGrey,  
 title: Text("Xylophone"),  
 centerTitle: true,  
 actions:<Widget>[CircleAvatar(  
 backgroundColor: Colors.white,  
 radius: 20.0,  
 backgroundImage: AssetImage('images/me.png'),  
  
 ),],  
 ),  
  
 body: Container(  
 color: Colors.black,  
 padding: EdgeInsets.fromLTRB(20.0, 10.0, 20.0, 10.0),  
 child: Column(  
 crossAxisAlignment: CrossAxisAlignment.stretch,  
 children: [  
 button(au1, color1),  
 SizedBox(  
 height: 10.0,  
 ),  
 button(au2, color2),  
 SizedBox(  
 height: 10.0,  
 ),  
 button(au3, color3),  
 SizedBox(  
 height: 10.0,  
 ),  
 button(au4, color4),  
 SizedBox(  
 height: 10.0,  
 ),  
 button(au5, color5),  
 SizedBox(  
 height: 10.0,  
 ),  
 button(au6, color6),  
 SizedBox(  
 height: 10.0,  
 ),  
 button(au7, color7),  
 ],  
 ),  
 ),  
 );  
 }  
}

**Screenshots**

Graphical user interface, application

Description automatically generatedA screenshot of a cell phone

Description automatically generated with medium confidence

A picture containing graphical user interface

Description automatically generatedGraphical user interface, application

Description automatically generated

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generated

Graphical user interface

Description automatically generated with low confidenceA screenshot of a cell phone

Description automatically generated with medium confidence