|  |  |
| --- | --- |
|  | Peer review -Group of 4 |

You need to **evaluate the code** from W6-S1 in team of 4

1 – Each member evaluates the code of another member

2 – You discuss to select the best code

3 – For each exercise, a group is selected to present to the whole class

|  |  |
| --- | --- |
| *YOUR NAME* | Phon Sovatanak |
| *YOUR TEAM MEMBERS* | Ly Mengthong, Thy Sovichet, Orn Chanlyhov, Pay Panha |
| *EVALUATED STUDENT NAME* | Pay Panha |

|  |  |  |  |
| --- | --- | --- | --- |
| **W6-S1 EX 1** | **YES** | **PART** | **NO** |
| The list of colors is displayed using a Using a **direct for loop** |  |  |  |
| The list of colors is displayed using the map method |  |  |  |
| The list of colors is displayed using dedicated function to return a list of widgets |  |  |  |
| **Comments** | | | |
| import 'package:flutter/material.dart';  List<String> colors = ["red", "blue", "green"];  List<Widget> getColors() {  return colors.map((items) => Text(items)).toList();  }  void main() {  runApp(MaterialApp(  debugShowCheckedModeBanner: false,  home: Scaffold(  body: Padding(  padding: const EdgeInsets.all(10),  child: ListView(  children: <Widget>[  Label("Method 1: Loop in Array", bold: true),  for (var i = 0; i < colors.length; i++) Text(colors[i]),  Label("Method 2: Map", bold: true),  ...colors.map((items) => Text(items)),  Label("Method 23: Dedicated Function", bold: true),  ...getColors(),  ],  ),  ),  ),  ));  }  class Label extends StatelessWidget {  const Label(this.text, {super.key, this.bold = false});  final bool bold;  final String text;  @override  Widget build(BuildContext context) {  return Text(  text,  style:  TextStyle(fontWeight: (bold ? FontWeight.bold : FontWeight.normal)),  );  }  } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **W6-S1 EX 2** | **YES** | **PART** | **NO** |
| The ProfileApp widget must take as argument a ProfileData object. |  |  |  |
| Each tile of the given profile data must be rendered in the profile app, as a card. |  |  |  |
| the scrolling is managed, when too many tiles need to be displayed |  |  |  |
| **Comments** | | | |
| import 'package:flutter/material.dart';  import 'package:w3s2\_flutter\_class/W6-S2-Practice/Ex2/model/profile\_tile\_model.dart';  import '../Ex2/data/profile\_data.dart';  void main() {  runApp(MaterialApp(  debugShowCheckedModeBanner: false,  home: ProfileApp(  profileData: myProfile,  ),  ));  }  const Color mainColor = Color(0xff5E9FCD);  class ProfileApp extends StatelessWidget {  final ProfileData profileData;  const ProfileApp({super.key, required this.profileData});  Iterable<ProfileTile> getTiles() {  return profileData.tiles.map((tiles) =>  ProfileTile(icon: tiles.icon, title: tiles.title, data: tiles.value));  }  @override  Widget build(BuildContext context) {  return Scaffold(  backgroundColor: mainColor.withAlpha(100),  appBar: AppBar(  backgroundColor: mainColor,  title: const Text(  'CADT student Profile',  style: TextStyle(color: Colors.white),  ),  centerTitle: true,  ),  body: Center(  child: Column(  mainAxisAlignment: MainAxisAlignment.start,  children: [  SizedBox(height: 40),  CircleAvatar(  radius: 60,  backgroundImage: AssetImage(profileData.avatarUrl),  ),  SizedBox(height: 20),  Text(  profileData.name,  style: TextStyle(  fontSize: 24,  fontWeight: FontWeight.bold,  color: mainColor,  ),  ),  Text(  profileData.position,  style: TextStyle(  fontSize: 16,  color: Colors.grey,  ),  ),  SizedBox(height: 20),  Expanded(  child: ListView(  children: [...getTiles()],  ))  ],  ),  ),  );  }  }  class ProfileTile extends StatelessWidget {  const ProfileTile({  super.key,  required this.icon,  required this.title,  required this.data,  });  final IconData icon;  final String title;  final String data;  @override  Widget build(BuildContext context) {  return Padding(  padding: const EdgeInsets.all(8.0),  child: Card(  margin: const EdgeInsets.symmetric(horizontal: 16, vertical: 4),  child: ListTile(  leading: Icon(icon, color: mainColor),  title: Text(title),  subtitle: Text(data),  ),  ),  );  }  } | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **W6-S1 EX 3** | **YES** | **PART** | **NO** |
| A TemperatureApp stateful widget is created |  |  |  |
| The TemperatureApp manages an active screen state (ex: a Enum) |  |  |  |
| The current screen is provided using a function with a switch case |  |  |  |
| The Welcome widget has a callback attribute and the callback function change the active screen |  |  |  |
| **Comments** | | | |
| import 'package:flutter/material.dart';  import 'package:w3s2\_flutter\_class/W6-S2-Practice/Ex3/screen/temperature.dart';  import 'screen/welcome.dart';  class TemperatureApp extends StatefulWidget {  const TemperatureApp({super.key});  @override  State<TemperatureApp> createState() {  return \_TemperatureAppState();  }  }  class \_TemperatureAppState extends State<TemperatureApp> {  bool isWelcome = true;  Widget get screen => isWelcome  ? Welcome(  onPressed: changeScreen,  )  : Temperature();  void changeScreen() {  setState(() {  isWelcome = !isWelcome;  });  }  @override  Widget build(context) {  return MaterialApp(  debugShowCheckedModeBanner: false,  home: Scaffold(  body: Container(  decoration: const BoxDecoration(  gradient: LinearGradient(  colors: [  Color(0xff16C062),  Color(0xff00BCDC),  ],  begin: Alignment.topLeft,  end: Alignment.bottomRight,  ),  ),  child: screen,  ),  ),  );  }  }  void main() {  runApp(const TemperatureApp());  } | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **W6-S1 EX 4** | | **YES** | **PART** | **NO** |
| Only 1 joke can be selected as favorite | |  |  |  |
| Which **data** stores the favorite jokes? | **Data are stored using the class JokeModel and in the JokeTile. It also stores in separate file.** | | | |
| Which **widget** stores the favorite joke? | **It was supposed to be in the FavoriteCard but he put it in main.** | | | |
| Which widgets are **stateful**?  Which widgets are **stateless**? | **The FavoriteCard should be statelesss.**  **The FavoriteJokeCard should be stateful.** | | | |
| How do widgets interact?  Is there **callback function**? | **The FavoriteCard have the VoidCallBack function of the on\_pressed and other variable that is needed.** | | | |
| **Comments** | | | | |
| import 'package:flutter/material.dart';  import '../Ex4/favorite\_card.dart';  import '../Ex4/data/jokes.dart';  Color appColor = Colors.green[300] as Color;  void main() => runApp(  MaterialApp(debugShowCheckedModeBanner: false, home: FavoriteJokeCard()));  class FavoriteJokeCard extends StatefulWidget {  const FavoriteJokeCard({super.key});  @override  State<FavoriteJokeCard> createState() => \_FavoriteJokeCardState();  }  class \_FavoriteJokeCardState extends State<FavoriteJokeCard> {  int? \_isFavorite;  void \_onFavoriteClick(int index) {  setState(() {  \_isFavorite = \_isFavorite == index ? null : index;  });  }  @override  Widget build(BuildContext context) {  return Scaffold(  backgroundColor: Colors.white,  appBar: AppBar(  backgroundColor: appColor,  title: const Text("Favorite Jokes"),  ),  body: ListView.builder(  itemCount: myJokes.tiles.length,  itemBuilder: (context, index) {  var tile = myJokes.tiles[index];  return FavoriteCard(  jokes: tile.title,  description: tile.description,  isFavorite: \_isFavorite == index,  onFavoriteClick: () => \_onFavoriteClick(index));  },  ),  );  }  } | | | | |