

FLUTTER MINI ASSIGNMENT

Task 1: Contact Manager App (App-4):

Code:// main.dart

```
import 'package:flutter/material.dart';
import 'screens/contact_list_page.dart';

void main() {                                     //starting point of the app.
    runApp(ContactApp());
}

class ContactApp extends StatelessWidget {          //Root widget of app
    @override
    Widget build(BuildContext context) {
        return MaterialApp(
            debugShowCheckedModeBanner: false,
            home: ContactListPage(),                  //First Screen
        );
    }
}
```

//  screens/contact_list_page.dart

```
import 'package:flutter/material.dart';
import 'add_contact_page.dart';                  // navigation to that page.

class ContactListPage extends StatefulWidget {       //Stateful contact list
    @override
    State<ContactListPage> createState() => _ContactListPageState();
}

class _ContactListPageState extends State<ContactListPage> { //controls UI updates
    List<Map<String, String>> contacts = [           //List of contacts.
}
```

FLUTTER MINI ASSIGNMENT

```
{"name": "Keerthi", "phone": "9782145678"},  
 {"name": "Tarun", "phone": "9123456780"},  
 {"name": "Vikram", "phone": "9988776655"},  
 {"name": "Priya", "phone": "9012345678"},  
 {"name": "Arjun", "phone": "9090909090"},  
];  
  
void addContact(String name, String phone) {  
    setState(() {  
        contacts.add({"name": name, "phone": phone});  
    });  
}  
  
@override  
  
Widget build(BuildContext context) {  
    return Scaffold(                                // Basic page structure.  
        appBar: AppBar(title: Text("Contact Manager")), //Top bar  
        body: ListView.builder(                        //List view  
            itemCount: contacts.length,             // Number of contacts.  
            itemBuilder: (context, index) {  
                return ListTile(  
                    leading: Icon(Icons.person),          // Person icon  
                    title: Text(contacts[index]["name"]!), // contact name  
                    subtitle: Text(contacts[index]["phone"]!), // contact phone number  
                    onTap: () {  
                        showDialog(  
                            context: context,  
                            builder: (context) => AlertDialog(      //alert dialog  
                                title: Text("Contact Details"),  
                                content: Text(  
                                    "Name: ${contacts[index]["name"]}\nPhone: ${contacts[index]["phone"]}",  
                                ),  
                            ),  
                        );  
                    },  
                );  
            },  
        ),  
    );  
}
```

FLUTTER MINI ASSIGNMENT

```
        ),  
        actions: [  
            TextButton(  
                onPressed: () {  
                    Navigator.pop(context);  
                },  
                child: Text("OK"),  
            ),  
        ],  
    ),  
);  
},  
);  
},  
),  
floatingActionButton: FloatingActionButton(  
    child: Icon(Icons.add),  
    onPressed: () async {  
        final newContact = await Navigator.push(  
            context,  
            MaterialPageRoute(builder: (context) => AddContactPage()),  
            // Creates route -  
            AddContactPage.  
        );  
        if (newContact != null) {  
            addContact(newContact["name"], newContact["phone"]);  
        }  
    },  
),  
);  
}  
}
```

FLUTTER MINI ASSIGNMENT

//  screens/ add_contact_page.dart

```
import 'package:flutter/material.dart';

class AddContactPage extends StatefulWidget {

  @override

  State<AddContactPage> createState() => _AddContactPageState();
}

class _AddContactPageState extends State<AddContactPage> {

  TextEditingController nameController = TextEditingController();

  TextEditingController phoneController = TextEditingController();

  @override

  Widget build(BuildContext context) { // Builds UI of this screen.

    return Scaffold( // Basic page layout.

      appBar: AppBar(title: Text("Add Contact")),

      body: Padding( // Adds spacing around content.

        padding: EdgeInsets.all(16), // Adds spacing around content.

        child: Column( // Arranges widgets vertically.

          children: [
            TextField( // TextField connected to controller.

              controller: nameController, // Store what user types in TextField

              decoration: InputDecoration(
                labelText: "Enter Name",
                border: OutlineInputBorder(),
              ),
            ),
            SizedBox(height: 20),
            TextField(
              controller: phoneController,
              decoration: InputDecoration(

```

FLUTTER MINI ASSIGNMENT

```
labelText: "Enter Phone number",
border: OutlineInputBorder(),
),
),
SizedBox(height: 20),
ElevatedButton(
onPressed: () {
if (nameController.text.isNotEmpty &&           // Prevents empty contact.
    phoneController.text.isNotEmpty) {
ScaffoldMessenger.of(context).showSnackBar(
    SnackBar(content: Text("Contact Added Successfully")),
);
Future.delayed(Duration(seconds: 1), () {
// Wait 1 second so user sees snackbar.
Navigator.pop(context, {
    "name": nameController.text,
    "phone": phoneController.text,
});
});
}
},
child: Text("Save"),
),
],
),
);
}
}
```

FLUTTER MINI ASSIGNMENT

This screenshot shows the 'contact_list_page.dart' file in the code editor. The code defines a list of contacts and a method to add new ones. The contacts are represented as a map where keys are names and values are phone numbers. The 'addContact' method adds a new contact to this list. The code editor also shows a terminal output indicating the application was restarted in 1,734ms.

```
lib > screens > contact_list_page.dart > _ContactlistPageState > build
4   class ContactListPage extends StatefulWidget {
5     State<ContactListPage> createState() => _ContactlistPageState();
6   }
7
8
9   class _ContactlistPageState extends State<ContactListPage> {
10    List<Map<String, String>> contacts = [
11      {"name": "Keerthi", "phone": "9782145678"},  
      {"name": "Tarun", "phone": "9123456780"},  
      {"name": "Vikram", "phone": "9988776655"},  
      {"name": "Priya", "phone": "9812345678"},  
      {"name": "Arjun", "phone": "9090909090"},  
    ];
18
19   void addContact(String name, String phone) {  
20     setState(() {  
21       contacts.add({"name": name, "phone": phone});  
22     });
23
24   @override  
25   Widget build(BuildContext context) {  
26     return Scaffold(  
27       body:  
28         ListView.builder(  
29           itemCount: contacts.length,  
30           itemBuilder: (context, index) {  
31             Map<String, String> contact = contacts[index];  
32             return ListTile(  
33               title: Text(contact["name"]!),  
34               subtitle: Text(contact["phone"]!),  
35             );
36           },
37         ),
38       );
39     );
40   }
41
42   ElevatedButton(  
43     onPressed: () {  
44       if (nameController.text.isNotEmpty &&  
45         phoneController.text.isNotEmpty) {  
46         ScaffoldMessenger.of(context).showSnackBar(  
47           SnackBar(content: Text("Contact Added Successfully")),  
48         );
49       }
50     },
51     child: Text("Add Contact"),
52   );
53
54   Future.delayed(Duration(seconds: 1), () {  
55     Navigator.pop(context, {  
56       "name": nameController.text,  
57       "phone": phoneController.text,  
58     });
59   });
60 }
61
62 
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS OUTPUT Filter (e.g. text, exclude, \es...)

Restarted application in 1,734ms.
I/contact_manager(5119): AssetManager2(0x77e1b9920d98) locale list changing from [-US]
W/windowOnBackDispatcher(5119): sendCancelIfRunning: isInProgress=false callback=r.embedding.android.FlutterActivity\$1@18a1da8

Ln 25, Col 39 Spaces: 2 UTF-8 CRLF { } Dart ⚡ ⓘ Go Live Pixel 6 (android-x64 emulator) ⓘ Prettier

This screenshot shows the 'add_contact_page.dart' file in the code editor. The code defines a form for adding a new contact with fields for name and phone number. It includes logic to check if both fields are filled and to show a success message via a snack bar. The code editor also shows a terminal output indicating the application was restarted in 1,734ms.

```
lib > screens > add_contact_page.dart > _AddContactPageState > build
8   class _AddContactPageState extends State<AddContactPage> {
9     Widget build(BuildContext context) {
10
11       final TextEditingController nameController = TextEditingController();
12       final TextEditingController phoneController = TextEditingController();
13
14       return Form(
15         child: Column(
16           children: [
17             TextFormField(
18               controller: nameController,
19               decoration: InputDecoration(
20                 labelText: "Enter Name",
21                 border: OutlineInputBorder(),
22               ),
23             ),
24             SizedBox(height: 20),
25             TextFormField(
26               controller: phoneController,
27               decoration: InputDecoration(
28                 labelText: "Enter Phone number",
29                 border: OutlineInputBorder(),
30               ),
31             ),
32             ElevatedButton(
33               onPressed: () {
34                 if (nameController.text.isNotEmpty &&
35                   phoneController.text.isNotEmpty) {
36                   ScaffoldMessenger.of(context).showSnackBar(  
37                     SnackBar(content: Text("Contact Added Successfully")),
38                   );
39                 }
40               },
41               child: Text("Save"),
42             ),
43             Future.delayed(Duration(seconds: 1), () {
44               Navigator.pop(context, {
45                 "name": nameController.text,
46                 "phone": phoneController.text,
47               });
48             });
49           ],
50         ),
51       );
52     }
53   }
54
55 
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS OUTPUT Filter (e.g. text, exclude, \es...)

W/FrameTracker(5119): Missed SF frame@ANK_COMPOSER, 32001, 13700666, CU=] <IME_IN_E_ANIMATION::1@1@com.example.contact_manager>
V/PerfettoTrigger(5119): triggering /system/bin/trigger_perfetto com.android.telemetry-jank-monitor-81
W/windowOnBackDispatcher(5119): sendCancelIfRunning: isInProgress=false callback=r.embedding.android.FlutterActivity\$1@18a1da8
Reloaded 1 of 742 libraries in 623ms (compile: 22 ms, reload: 256 ms, reassemble: 1 ms)

Ln 34, Col 47 Spaces: 2 UTF-8 CRLF { } Dart ⚡ ⓘ Go Live Pixel 6 (android-x64 emulator) ⓘ Prettier

This screenshot shows the 'add_contact_page.dart' file in the code editor with input data entered into the 'Enter Name' and 'Enter Phone number' fields. The 'Save' button is visible below the fields. The code editor also shows a terminal output indicating the application was restarted in 1,734ms.

```
lib > screens > add_contact_page.dart > _AddContactPageState > build
8   class _AddContactPageState extends State<AddContactPage> {
9     Widget build(BuildContext context) {
10
11       final TextEditingController nameController = TextEditingController();
12       final TextEditingController phoneController = TextEditingController();
13
14       return Form(
15         child: Column(
16           children: [
17             TextFormField(
18               controller: nameController,
19               decoration: InputDecoration(
20                 labelText: "Enter Name",
21                 border: OutlineInputBorder(),
22               ),
23             ),
24             SizedBox(height: 20),
25             TextFormField(
26               controller: phoneController,
27               decoration: InputDecoration(
28                 labelText: "Enter Phone number",
29                 border: OutlineInputBorder(),
30               ),
31             ),
32             ElevatedButton(
33               onPressed: () {
34                 if (nameController.text.isNotEmpty &&
35                   phoneController.text.isNotEmpty) {
36                   ScaffoldMessenger.of(context).showSnackBar(  
37                     SnackBar(content: Text("Contact Added Successfully")),
38                   );
39                 }
40               },
41               child: Text("Save"),
42             ),
43             Future.delayed(Duration(seconds: 1), () {
44               Navigator.pop(context, {
45                 "name": nameController.text,
46                 "phone": phoneController.text,
47               });
48             });
49           ],
50         ),
51       );
52     }
53   }
54
55 
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS OUTPUT Filter (e.g. text, exclude, \es...)

D/InputConnectionAdaptor(5119): The input method toggled cursor monitoring on
D/InsetsController(5119): showime(), fromime=false
I/ImeTracker(5119): com.example.contact_manager:548bfc84: onCancelled at PHASE_CLIP_Y_ANIMATION
I/AssistStructure(5119): Flattened final assist data: 436 bytes, containing 1 windows
D/InputConnectionAdaptor(5119): The input method toggled cursor monitoring on
Ln 40, Col 1 Spaces: 2 UTF-8 CRLF { } Dart ⚡ ⓘ Go Live Pixel 6 (android-x64 emulator) ⓘ Prettier

30°C Sunny ENG IN 80% 22-02-2026 14:36 ⌂

FLUTTER MINI ASSIGNMENT

The screenshot shows the VS Code interface with the following details:

- Explorer View:** Shows the project structure with files like `add_contact_page.dart`, `main.dart`, and `contact_manager.iml`.
- Code Editor:** Displays the Dart code for `add_contact_page.dart`. The code includes a `TextField` for entering a phone number and an `ElevatedButton` to add the contact. A `Future.delayed` call ensures the page navigates back after a successful addition.
- Terminal:** Shows log output related to frame tracking and animation issues.
- Output:** Shows the result of the `flutter build` command.
- Emulator:** An Android emulator running on a Pixel 6 device (Android x64 emulator) displays the "Contact Manager" application. It lists contacts: Keerthi, Tarun, Vikram, Priya, Arjun, and Manjunath V P. A success message "Contact Added Successfully" is visible at the bottom of the list.
- Bottom Bar:** Includes icons for search, file operations, and system controls.

This screenshot is nearly identical to the one above, showing the same code editor, terminal output, and running Android emulator. The emulator displays the "Contact Manager" app with the same list of contacts and the "Contact Added Successfully" message.

FLUTTER MINI ASSIGNMENT

Task 2 : Photo Gallery App (App-5) :

Code :- // main.dart

```
import 'package:flutter/material.dart';

import 'screens/gallery_page.dart';

void main() {
    runApp(MyApp());
}

class MyApp extends StatelessWidget {           // Root widget of the application
    @override
    Widget build(BuildContext context) {
        return MaterialApp(debugShowCheckedModeBanner: false,
            home: GalleryPage());           // This will open GalleryPage from gallery_page.dart
    }
}
```

//  screens/gallery_page.dart

```
import 'package:flutter/material.dart';

import 'full_view_page.dart';           // import second screen for full photo view

class GalleryPage extends StatelessWidget {
    // This page shows grid of photos (colors acting as photos)

    final List<Color> colors = [           // List of colors acting like photos
        Colors.red,
        Colors.blue,
        Colors.green,
        Colors.orange,
```

FLUTTER MINI ASSIGNMENT

```
Colors.purple,  
Colors.pink,  
];  
  
// Function to show bottom sheet options when a photo is tapped  
void showOptions(BuildContext context, int index, Color color) {  
  // Shows bottom sheet from bottom of screen  
  showModalBottomSheet(  
    context: context,  
    builder: (context) {  
      return Column(  
        mainAxisSize: MainAxisSize.min, // takes only needed height  
        children: [  
          ListTile(  
            leading: Icon(Icons.fullscreen),  
            title: Text("View Full"),  
            onTap: () {  
              Navigator.pop(context); // close bottom sheet  
              Navigator.push( // Navigate to FullViewPager  
                context,  
                MaterialPageRoute(  
                  builder: (context) =>  
                    FullViewPager(color: color, photoNumber: index + 1),  
                    // send color , photo number  
                ),  
              );  
            },  
            ListTile(  
              leading: Icon(Icons.share),  
            ),  
          ),  
        ],  
      );  
    },  
  );  
}
```

FLUTTER MINI ASSIGNMENT

```
title: Text("Share"),  
onTap: () {  
    Navigator.pop(context);  
    ScaffoldMessenger.of(context).showSnackBar(  
        SnackBar(content: Text("Shared Photo ${index + 1}")),  
        );  
},  
),  
ListTile(  
    leading: Icon(Icons.delete),  
    title: Text("Delete"),  
    onTap: () {  
        Navigator.pop(context);  
        ScaffoldMessenger.of(context).showSnackBar(  
            SnackBar(content: Text("Deleted Photo ${index + 1}")),  
            );  
},  
),  
],  
);  
},  
);  
}  
@override  
// UI of page  
Widget build(BuildContext context) {  
    return Scaffold(  
        appBar: AppBar(title: Text("Photo Gallery")),  
        body: GridView.count(  
            crossAxisCount: 2,  
            // 2 photos per row
```

FLUTTER MINI ASSIGNMENT

```
children: List.generate(6, (index) { // generates 6 grid items
    return GestureDetector(
        onTap: () {
            showOptions(context, index, colors[index]);
        },
        child: Stack(
            children: [ //photo color box
                Container(margin: EdgeInsets.all(8), color: colors[index]),
                Positioned( // photo number text on top
                    bottom: 10,
                    left: 10,
                    child: Text(
                        "Photo ${index + 1}",
                        style: TextStyle(
                            color: Colors.white,
                            fontWeight: FontWeight.bold,
                        ),
                    ),
                ),
            ],
        ),
    );
}),
});
```

},
);
};
}
}

FLUTTER MINI ASSIGNMENT

//  screens/full_view_page.dart

```
import 'package:flutter/material.dart';

class FullViewPager extends StatelessWidget {      // shows a single photo in full screen

final Color color;                                // values are received from GalleryPage

final int photoNumber;

FullViewPager({required this.color, required this.photoNumber});

@Override

Widget build(BuildContext context) {

return Scaffold(

    appBar: AppBar(title: Text("Photo $photoNumber")),

    body: Container(                                // Full screen container showing the color

        color: color,                            // Set background color

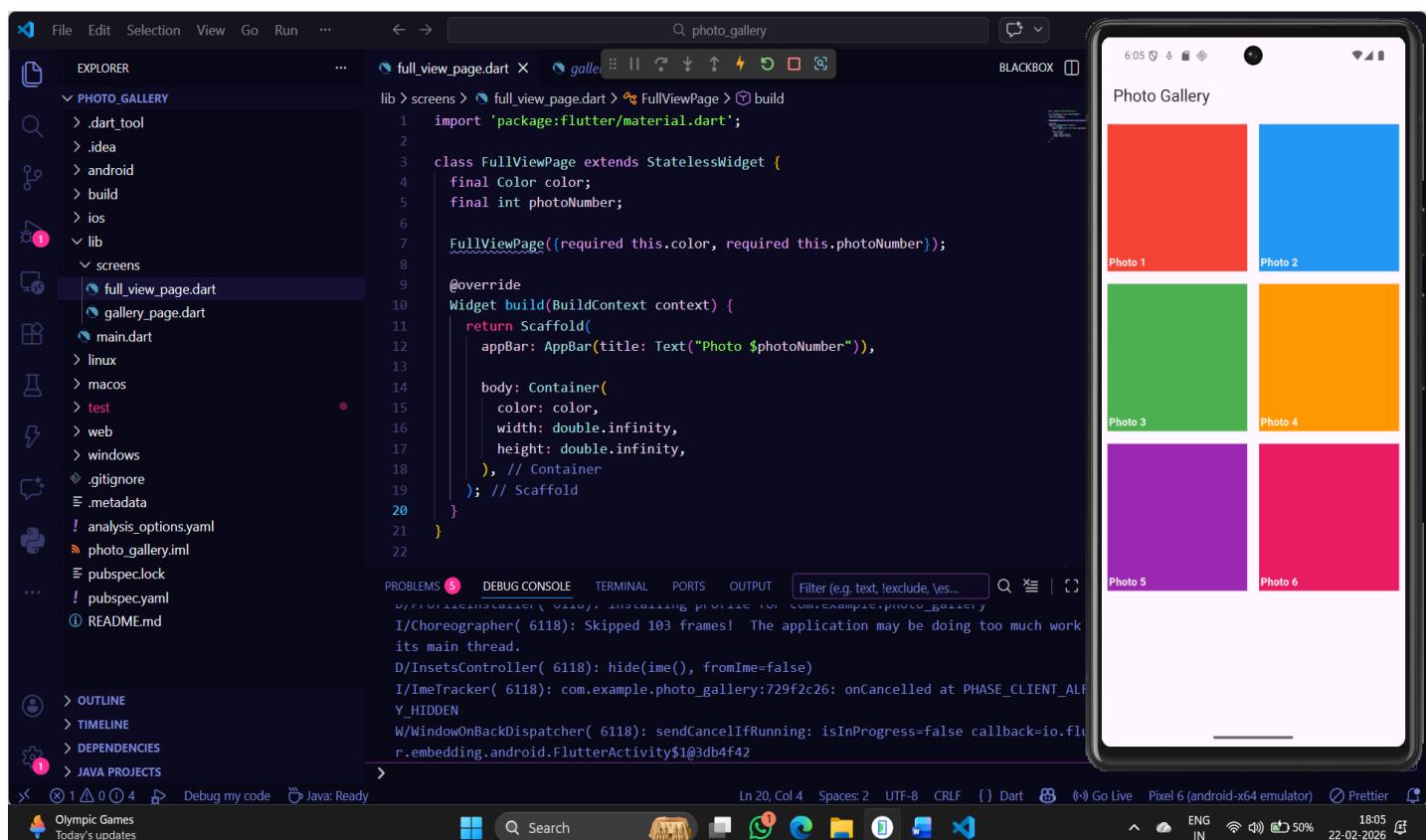
        width: double.infinity,                  // Take full screen width

        height: double.infinity,                 // Take full screen height

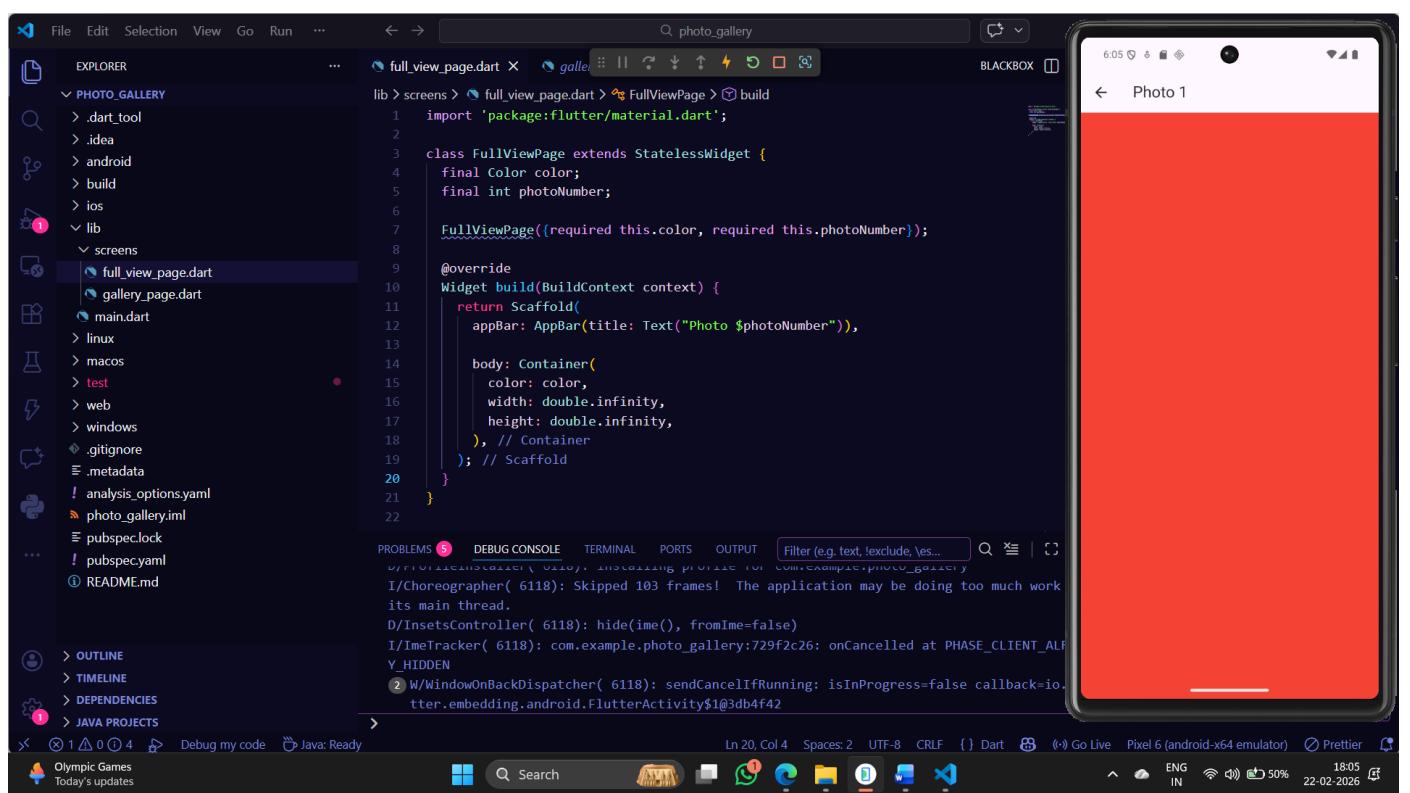
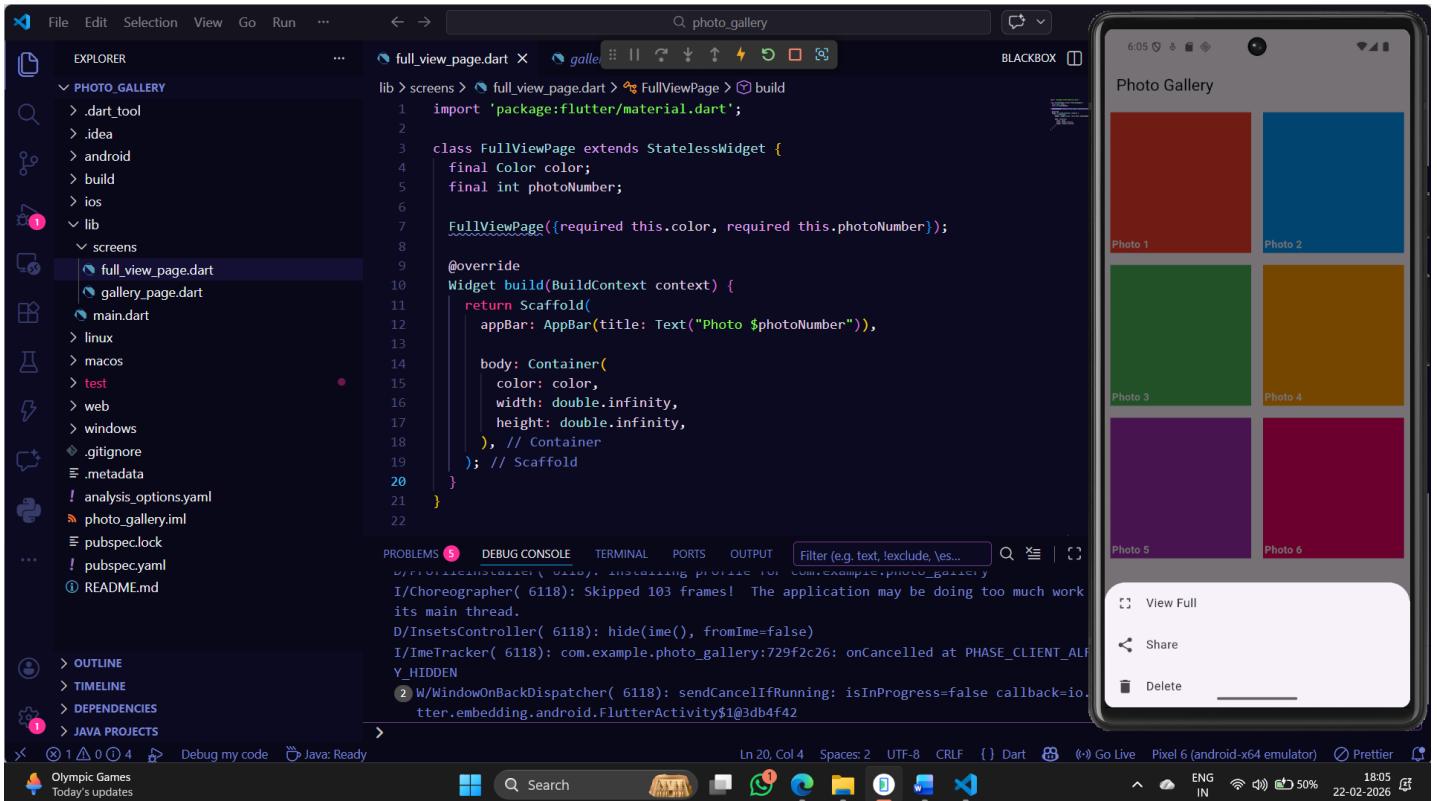
    ),

);

}}
```



FLUTTER MINI ASSIGNMENT



FLUTTER MINI ASSIGNMENT

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "PHOTO_GALLERY".
- Code Editor:** Displays the Dart code for "full_view_page.dart". The code defines a "FullViewPager" StatelessWidget that creates a 2x3 grid of colored photo cards. The cards are labeled "Photo 1" through "Photo 6".
- Terminal:** Shows log output from the Android emulator, indicating frame skipping and other system activity.
- Output:** Shows the current build profile as "BLACKBOX".
- Emulator:** A Pixel 6 (Android-x64 emulator) displays the "Photo Gallery" screen with the 6 photo cards.
- Bottom Bar:** Includes icons for search, file operations, and extensions like Prettier.

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "PHOTO_GALLERY".
- Code Editor:** Displays the Dart code for "full_view_page.dart". The code defines a "FullViewPager" StatelessWidget that creates a 2x3 grid of colored photo cards. The cards are labeled "Photo 1" through "Photo 6".
- Terminal:** Shows log output from the Android emulator, indicating frame skipping and other system activity.
- Output:** Shows the current build profile as "BLACKBOX".
- Emulator:** A Pixel 6 (Android-x64 emulator) displays the "Photo Gallery" screen with the 6 photo cards. The card "Photo 2" is highlighted with a red border, indicating it is being shared.
- Bottom Bar:** Includes icons for search, file operations, and extensions like Prettier.