Experiment 08

AIM: Integrate sensors such as accelerometer, gyroscope, or GPS into your mobile application to capture and utilize sensor data.

OBJECTIVE:

1. To implement accelerometer functionality in a Flutter application.
2. To capture real-time acceleration data along the X, Y, and Z axes.
3. To use the sensors_plus package for accessing accelerometer values.
4. To display live accelerometer readings in the user interface.
5. To manage continuous data updates efficiently within the app.

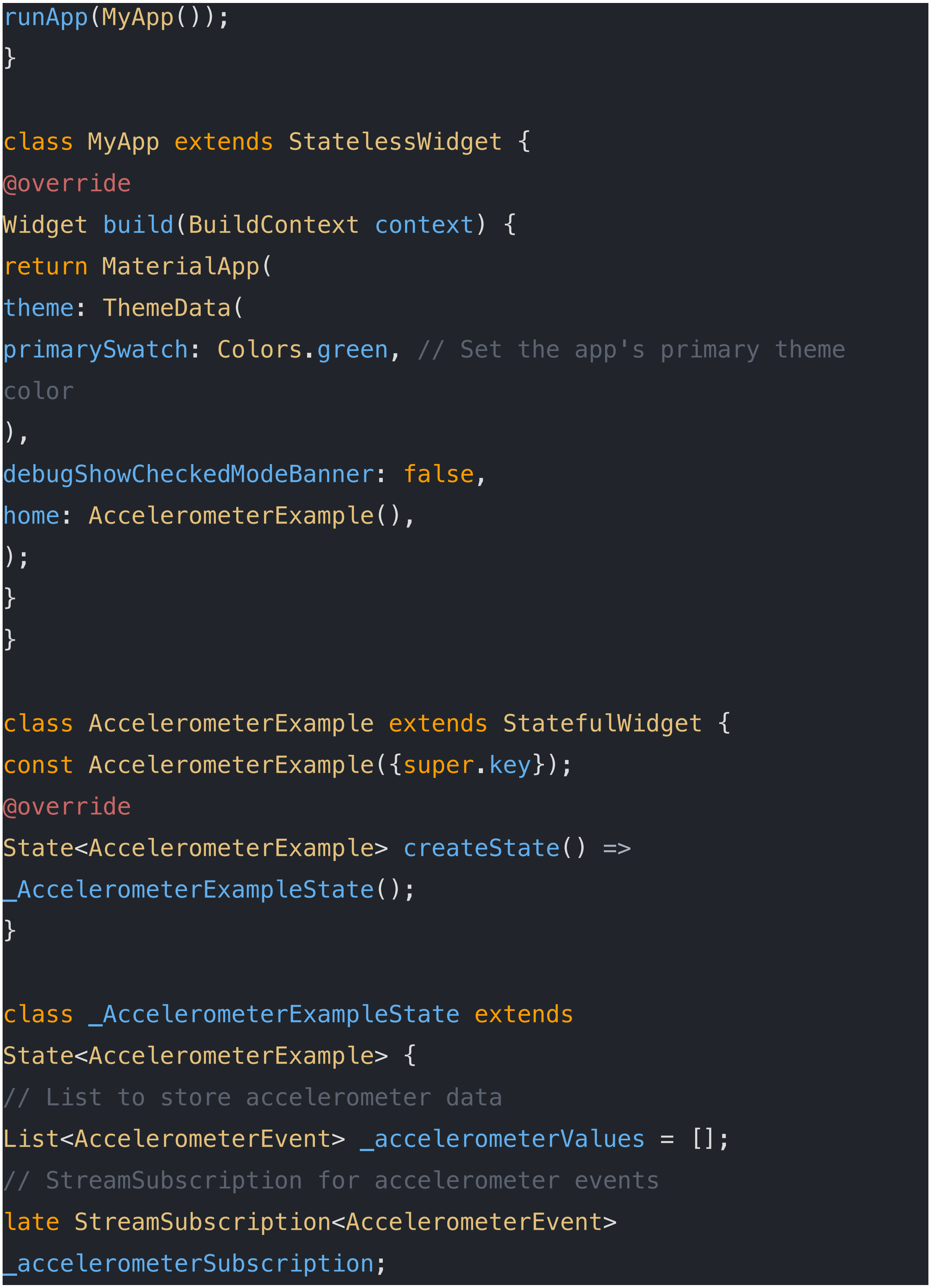
THEORY

This experiment focuses on implementing accelerometer functionality within a Flutter mobile application. The accelerometer detects the acceleration of the device along the X, Y, and Z axes, which can be used to identify motion or orientation changes. Using the sensors_plus package in Flutter, the app subscribes to accelerometer data streams and updates the user interface in real-time.

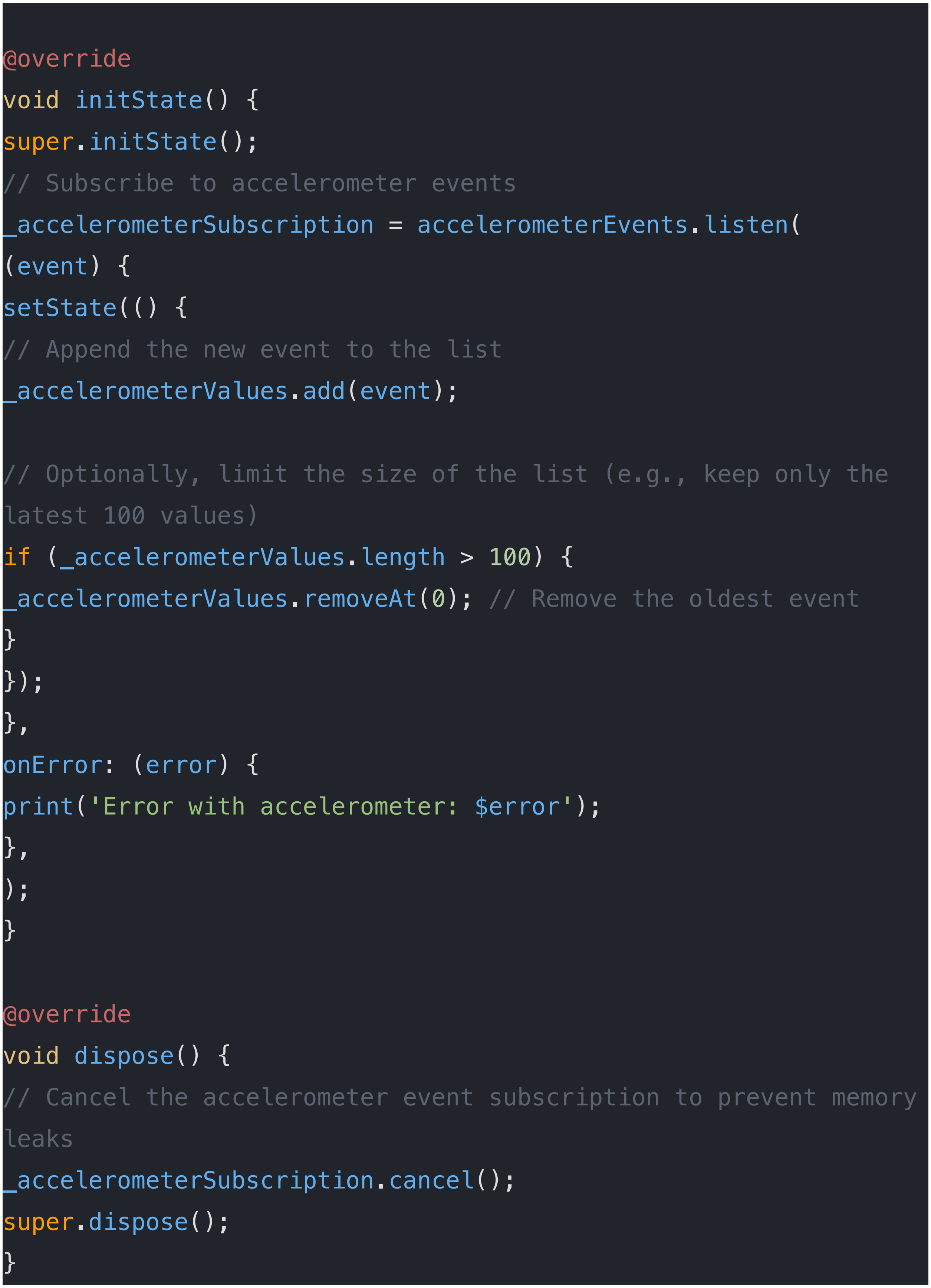
The app is designed to listen for acceleration changes and dynamically display the most recent values on the screen. This involves handling data streams with StreamSubscription and ensuring efficient updates using Flutter's state management. This experiment demonstrates how Flutter apps can interact with device-level features to create responsive, motion-aware applications.

CODE:

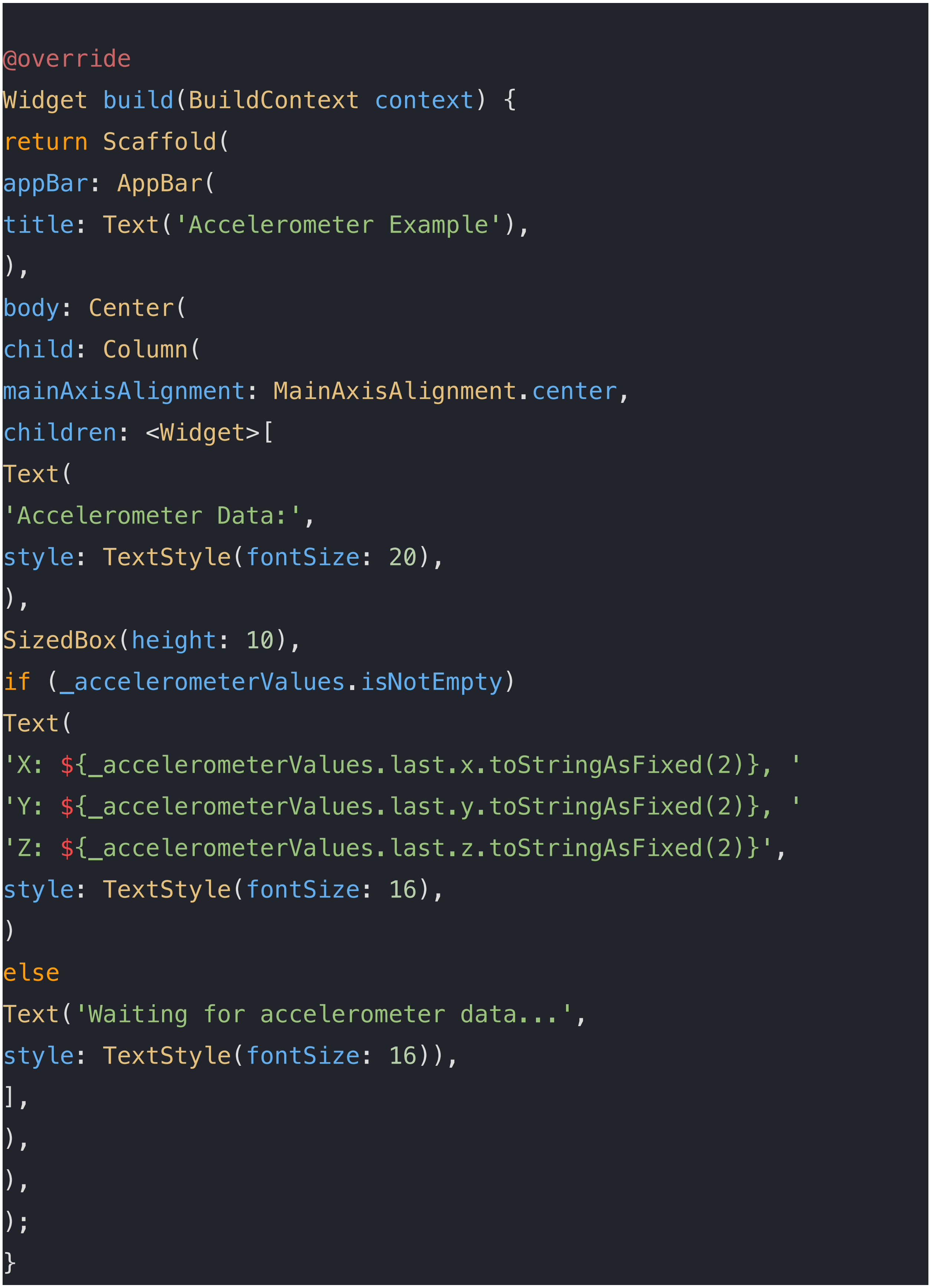
```
import 'dart:async';
import 'package:flutter/material.dart';
import 'package:sensors_plus/sensors_plus.dart';

void main() {
```



```
runApp(MyApp());
}

class MyApp extends StatelessWidget {
@override
Widget build(BuildContext context) {
return MaterialApp(
theme: ThemeData(
primarySwatch: Colors.green, // Set the app's primary theme
color
),
debugShowCheckedModeBanner: false,
home: AccelerometerExample(),
);
}
}

class AccelerometerExample extends StatefulWidget {
const AccelerometerExample({super.key});
@override
State<AccelerometerExample> createState() =>
_AccelerometerExampleState();
}

class _AccelerometerExampleState extends
State<AccelerometerExample> {
// List to store accelerometer data
List<AccelerometerEvent> _accelerometerValues = [];
// StreamSubscription for accelerometer events
late StreamSubscription<AccelerometerEvent>
_accelerometerSubscription;
```



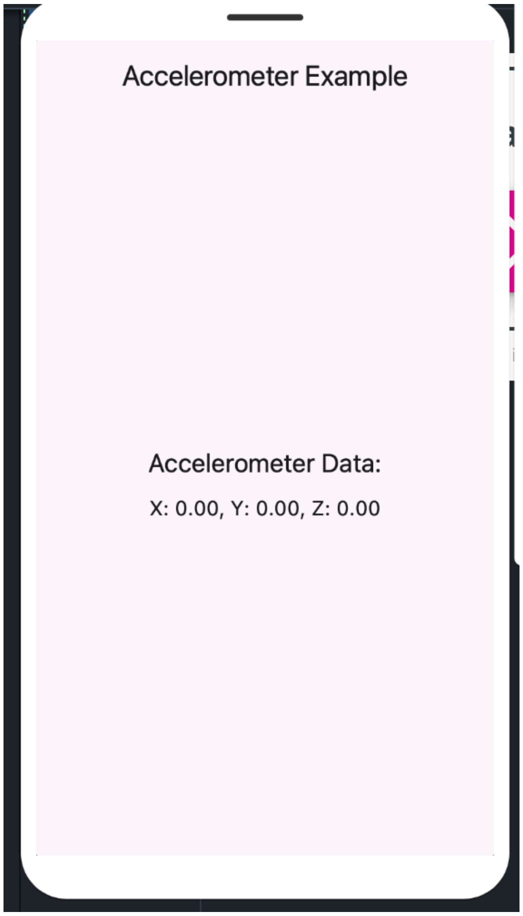
```
@override
void initState() {
super.initState();
// Subscribe to accelerometer events
_accelerometerSubscription = accelerometerEvents.listen(
(event) {
setState(() {
// Append the new event to the list
_accelerometerValues.add(event);

// Optionally, limit the size of the list (e.g., keep only the
latest 100 values)
if (_accelerometerValues.length > 100) {
_accelerometerValues.removeAt(0); // Remove the oldest event
}
});
},
onError: (error) {
print('Error with accelerometer: $error');
},
);
}

@override
void dispose() {
// Cancel the accelerometer event subscription to prevent memory
leaks
_accelerometerSubscription.cancel();
super.dispose();
}
```



```
@override
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar(
title: Text('Accelerometer Example'),
),
body: Center(
child: Column(
mainAxisAlignment: MainAxisAlignment.center,
children: <Widget>[
Text(
'Accelerometer Data:',
style: TextStyle(fontSize: 20),
),
SizedBox(height: 10),
if (_accelerometerValues.isNotEmpty)
Text(
'X: ${_accelerometerValues.last.x.toStringAsFixed(2)}, '
'Y: ${_accelerometerValues.last.y.toStringAsFixed(2)}, '
'Z: ${_accelerometerValues.last.z.toStringAsFixed(2)}',
style: TextStyle(fontSize: 16),
)
else
Text('Waiting for accelerometer data...',
style: TextStyle(fontSize: 16)),
],
),
),
);
}
```

```
}
```

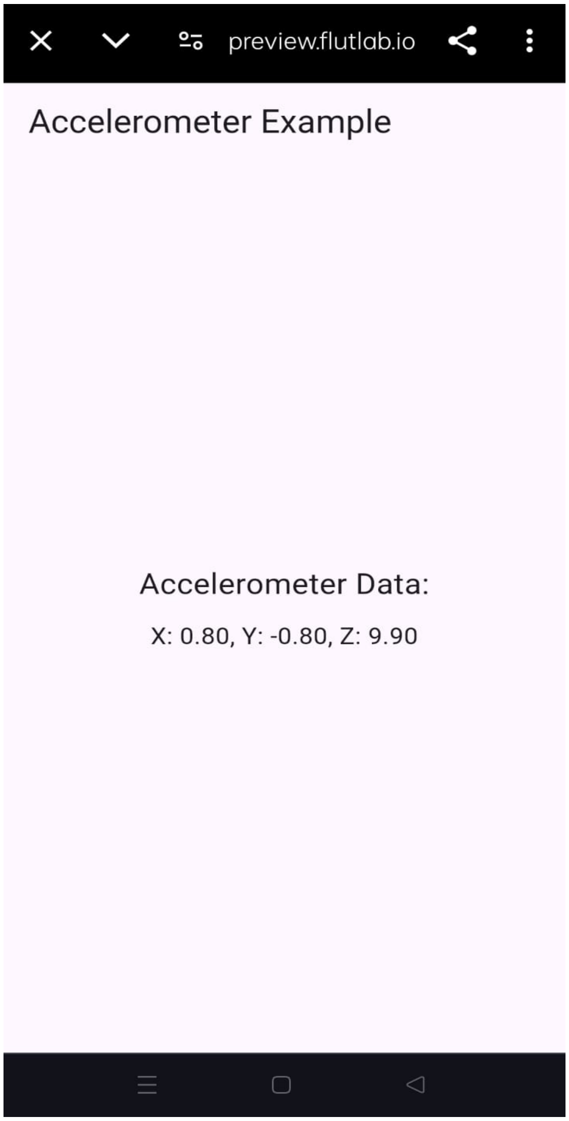
OUTPUT



Accelerometer Example

Accelerometer Data:

X: 0.00, Y: 0.00, Z: 0.00



Accelerometer Example

Accelerometer Data:

X: 0.80, Y: -0.80, Z: 9.90

Learning Outcome :

Experiment 09

AIM : Add multimedia functionality to your mobile application, such as capturing photos/videos, playing audio files, or integrating with social media sharing.

OBJECTIVE:

1. To implement multimedia features in a mobile app using Flutter.
2. To capture photos and videos using the device camera.
3. To play audio files from local or online sources.
4. To integrate social media sharing functionality.
5. To enhance user interaction through multimedia support.

THEORY:

Multimedia in mobile applications refers to the integration of media elements such as audio, video, and images to improve the user experience.

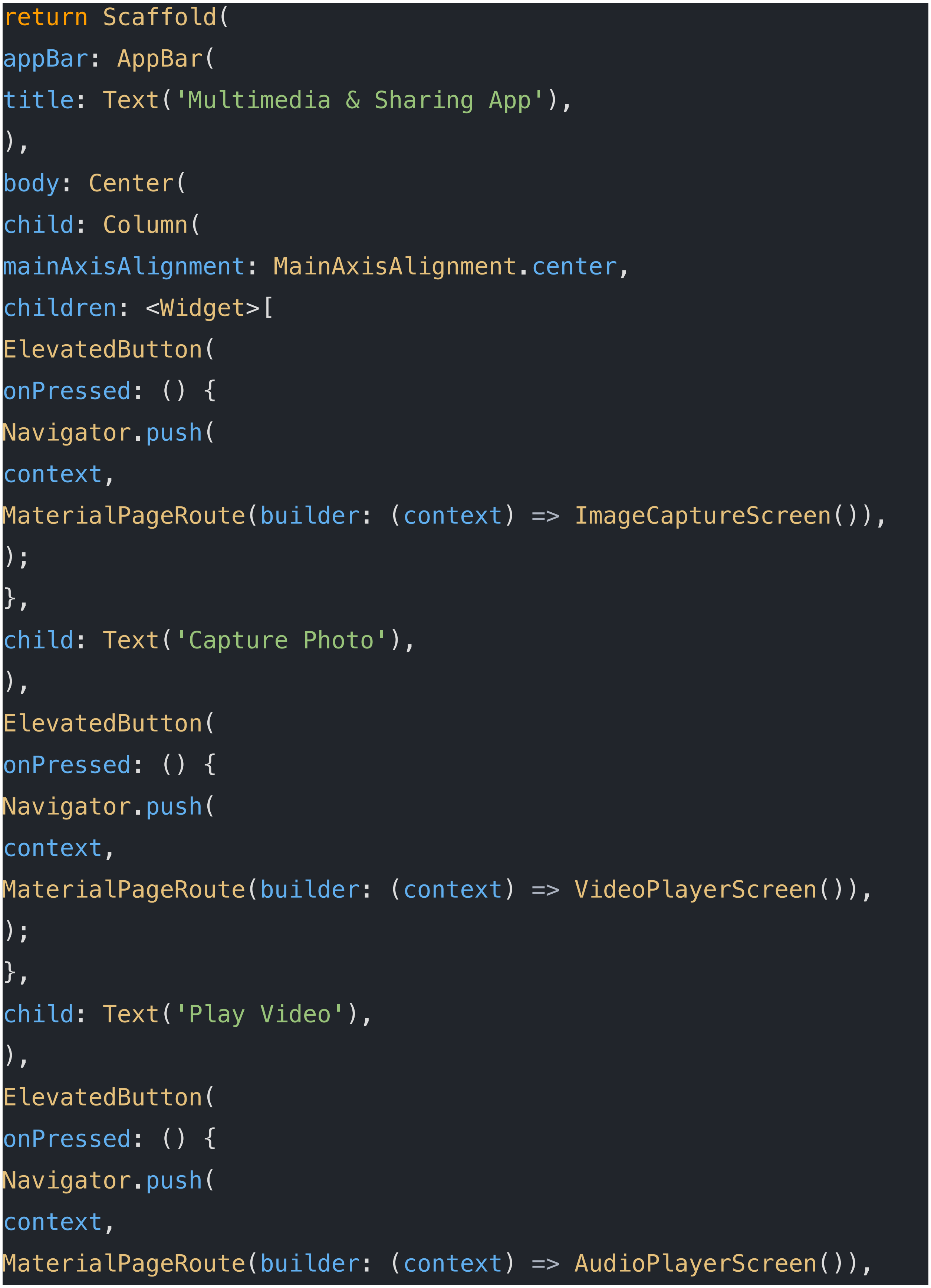
- **Camera Integration** allows users to capture images or videos directly from the app using packages like image_picker or camera.
- **Audio Playback** enhances the app by enabling music or sound effects using packages like audioplayers or just_audio.
- **Social Media Sharing** lets users share content from the app to platforms like WhatsApp, Facebook, etc., using packages like share_plus.

Flutter provides plugins that bridge the gap between Dart code and native platform functionalities, making it easier to integrate multimedia features efficiently across both Android and iOS platforms.

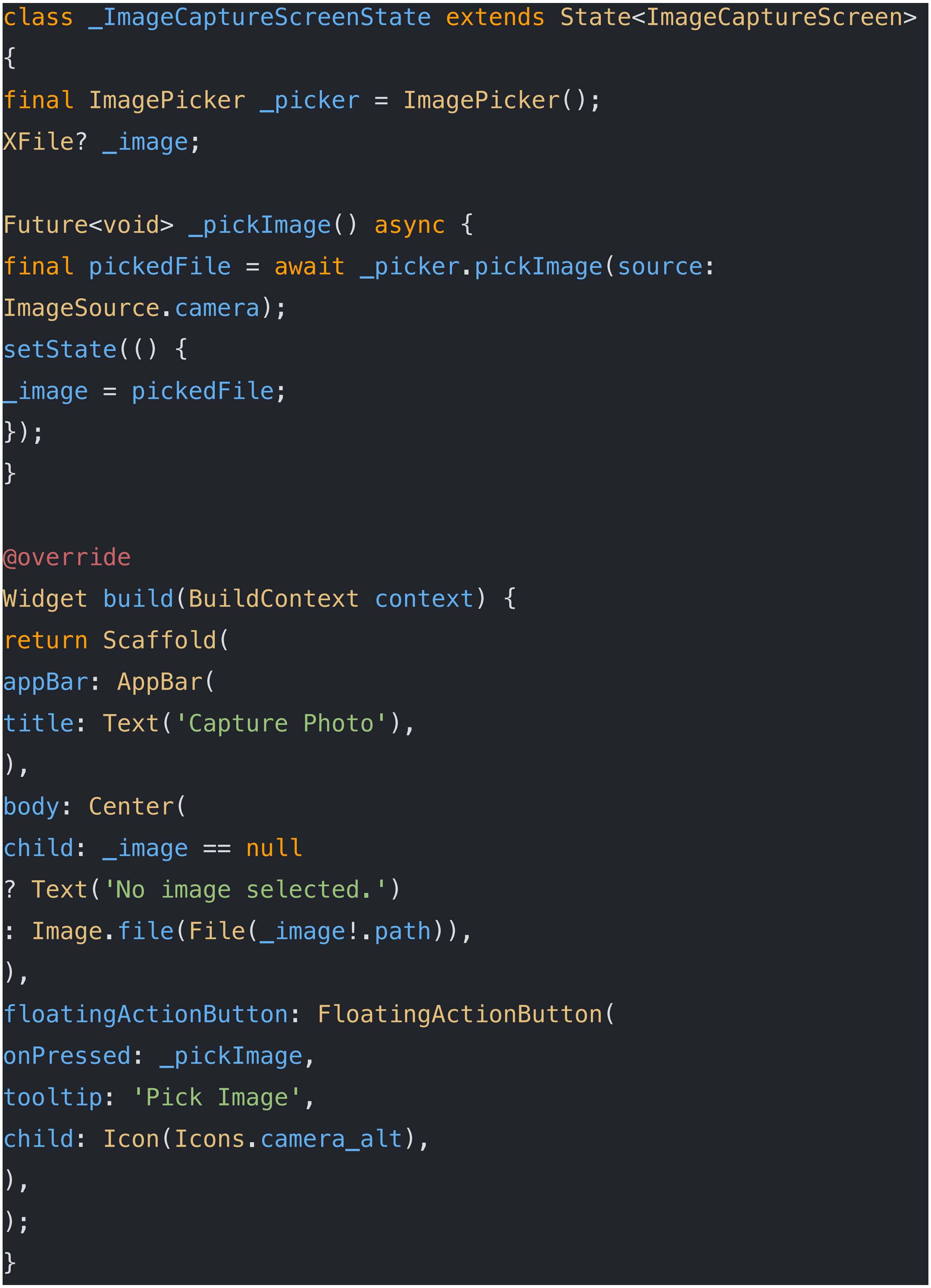
CODE:

main.dart

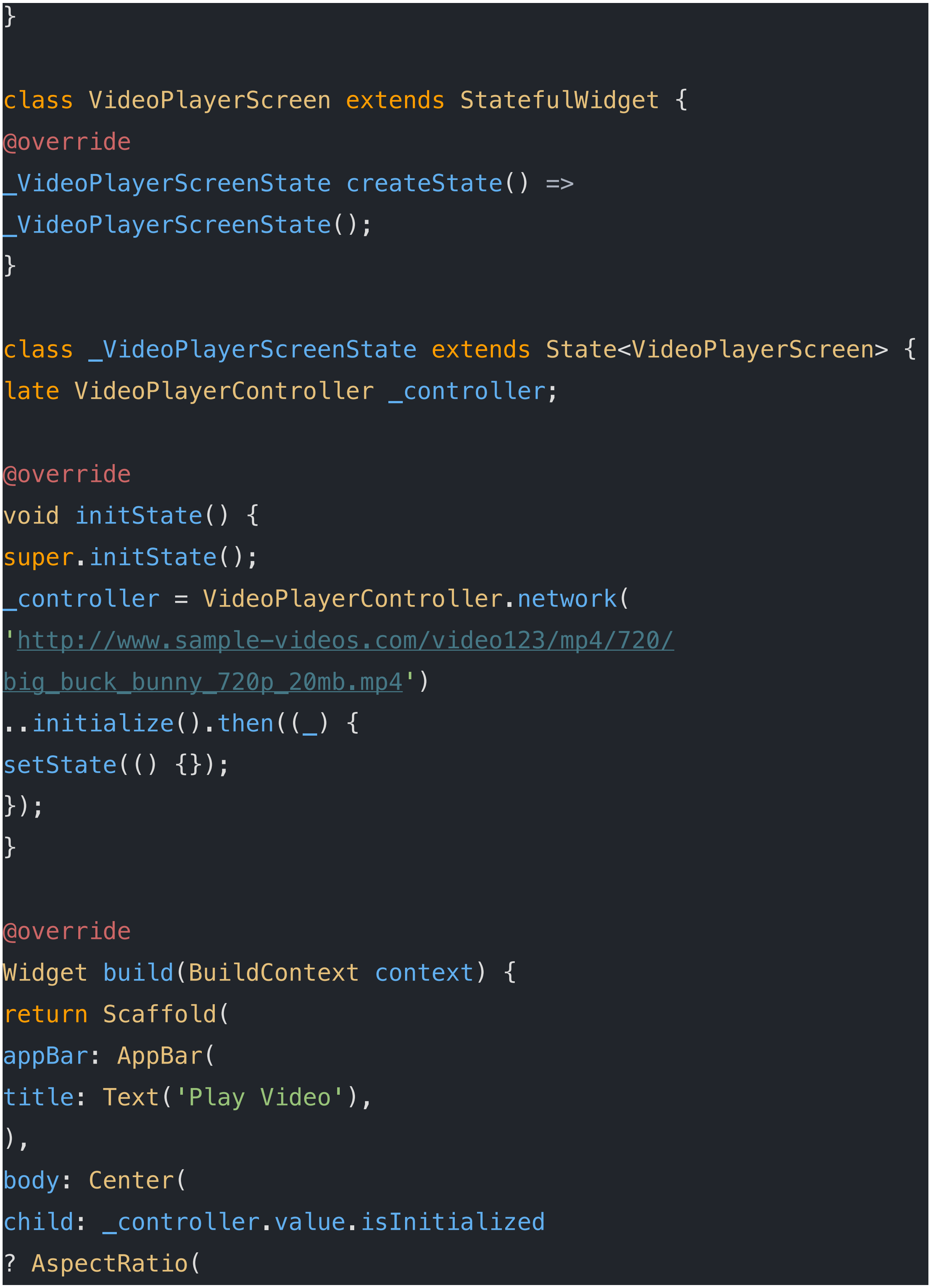
```
import 'dart:io';
import 'package:flutter/material.dart';
import 'package:image_picker/image_picker.dart';
import 'package:video_player/video_player.dart';
import 'package:audioplayers/audioplayers.dart';
import 'package:flutter_share/flutter_share.dart';
import 'package:share_plus/share_plus.dart';

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

class MyApp extends StatelessWidget {
@override
Widget build(BuildContext context) {
return MaterialApp(
title: 'Multimedia & Sharing App',
theme: ThemeData(
primarySwatch: Colors.blue,
),
home: HomeScreen(),
);
}
}

class HomeScreen extends StatelessWidget {
@override
Widget build(BuildContext context) {
```



```
return Scaffold(
appBar: AppBar(
title: Text('Multimedia & Sharing App'),
),
body: Center(
child: Column(
mainAxisAlignment: MainAxisAlignment.center,
children: <Widget>[
ElevatedButton(
onPressed: () {
Navigator.push(
context,
MaterialPageRoute(builder: (context) => ImageCaptureScreen()),
);
},
child: Text('Capture Photo'),
),
ElevatedButton(
onPressed: () {
Navigator.push(
context,
MaterialPageRoute(builder: (context) => VideoPlayerScreen()),
);
},
child: Text('Play Video'),
),
ElevatedButton(
onPressed: () {
Navigator.push(
context,
MaterialPageRoute(builder: (context) => AudioPlayerScreen()),
);
},
child: Text('Play Audio'),
),
ElevatedButton(
onPressed: () {
_shareContent('Check out this amazing app!');
},
child: Text('Share Content'),
),
],
),
),
);
}

void _shareContent(String content) async {
await FlutterShare.share(
title: 'Share via',
text: content,
);
}
}

class ImageCaptureScreen extends StatefulWidget {
@override
_ImageCaptureScreenState createState() =>
_ImageCaptureScreenState();
}
```



```
class _ImageCaptureScreenState extends State<ImageCaptureScreen>
{
final ImagePicker _picker = ImagePicker();
XFile? _image;

Future<void> _pickImage() async {
final pickedFile = await _picker.pickImage(source:
ImageSource.camera);
setState(() {
_image = pickedFile;
});
}

@override
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar(
title: Text('Capture Photo'),
),
body: Center(
child: _image == null
? Text('No image selected.')
: Image.file(File(_image!.path)),
),
floatingActionButton: FloatingActionButton(
onPressed: _pickImage,
tooltip: 'Pick Image',
child: Icon(Icons.camera_alt),
),
);
}
```



```
}

class VideoPlayerScreen extends StatefulWidget {
@override
_VideoPlayerScreenState createState() =>
_VideoPlayerScreenState();
}

class _VideoPlayerScreenState extends State<VideoPlayerScreen> {
late VideoPlayerController _controller;

@override
void initState() {
super.initState();
_controller = VideoPlayerController.network(
'http://www.sample-videos.com/video123/mp4/720/
big_buck_bunny_720p_20mb.mp4')
..initialize().then((_) {
setState(() {});
});
}

@override
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar(
title: Text('Play Video'),
),
body: Center(
child: _controller.value.isInitialized
? AspectRatio(
aspectRatio: _controller.value.aspectRatio,
child: VideoPlayer(_controller),
)
: CircularProgressIndicator(),
),
floatingActionButton: FloatingActionButton(
onPressed: () {
setState(() {
if (_controller.value.isPlaying) {
_controller.pause();
} else {
_controller.play();
}
});
},
child: Icon(
_controller.value.isPlaying ? Icons.pause : Icons.play_arrow,
),
),
);
}

@override
void dispose() {
super.dispose();
_controller.dispose();
}
}

class AudioPlayerScreen extends StatefulWidget {
@override
```

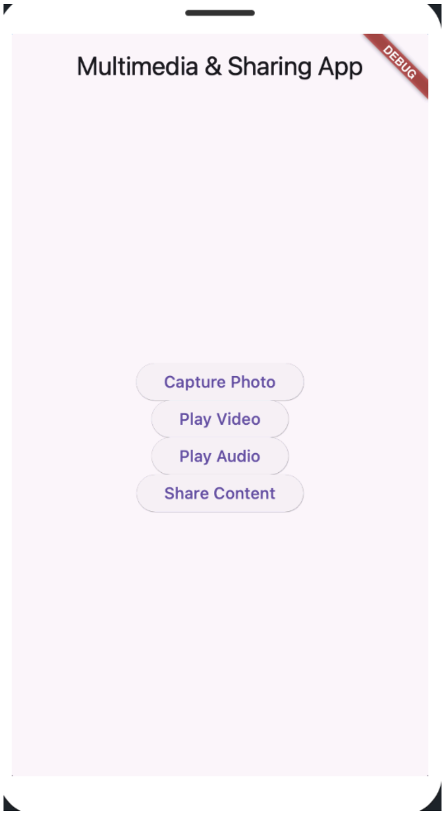
```
_AudioPlayerScreenState createState() =>
_AudioPlayerScreenState();
}

class _AudioPlayerScreenState extends State<AudioPlayerScreen> {
final AudioPlayer _audioPlayer = AudioPlayer();
bool _isPlaying = false;

void _togglePlay() async {
if (_isPlaying) {
await _audioPlayer.pause();
} else {
await _audioPlayer.play(UrlSource(
'https://www.soundhelix.com/examples/mp3/SoundHelix-
Song-1.mp3'));
}
setState(() {
_isPlaying = !_isPlaying;
});
}

@override
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar(
title: Text('Play Audio'),
),
body: Center(
child: IconButton(
icon: Icon(_isPlaying ? Icons.pause : Icons.play_arrow),
onPressed: _togglePlay,
),
),
);
}
}
```

pubspec.yaml

```
name: audiovideo
description: A new Flutter project.
publish_to: 'none'

version: 1.0.0+1

environment:
sdk: ^3.0.0

dependencies:
flutter:
sdk: flutter
image_picker: ^1.0.4
video_player: ^2.7.0
audioplayers: ^5.2.1
flutter_share: ^2.0.0
share_plus: ^7.2.1
# Check for the latest version
```

```
cupertino_icons: ^1.0.2
dev_dependencies:
flutter_test:
sdk: flutter
flutter_lints: ^4.0.0

flutter:
uses-material-design: true
```

OUTPUT:



Learning Outcome :

Experiment 10

AIM: To develop a portfolio mobile application

OBJECTIVES:

- To implement various Flutter widgets for structured content presentation.
- To create a responsive and visually appealing digital resume accessible across multiple devices.

THEORY:

The Portfolio Flutter App is a mobile application developed using the Flutter framework to showcase a professional portfolio. It serves as a digital resume, allowing users to present their skills, education, projects, and other relevant professional information in an interactive and visually appealing manner.

Technology Stack

The app is built using Flutter, an open-source UI software development toolkit created by Google. Flutter uses the Dart programming language and provides a cross-platform development environment, enabling the application to run seamlessly on both Android and iOS devices.

Features

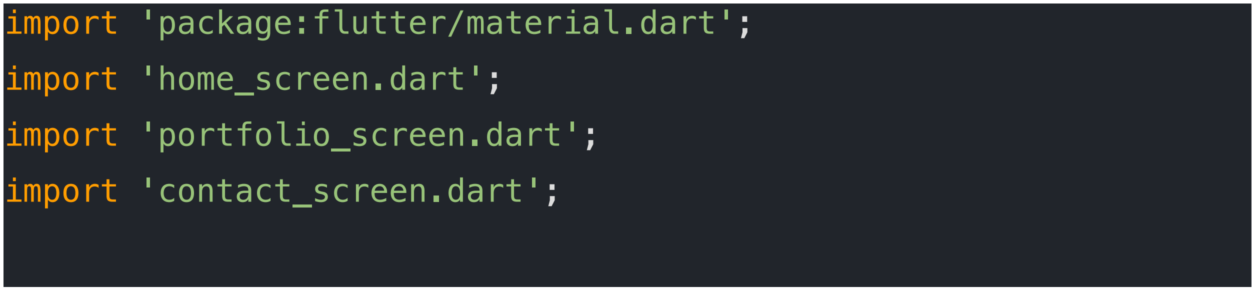
1. **Profile Section**: Displays the user's profile picture, name, and a brief introduction.
2. **Education Section**: Lists academic qualifications, including university details.
3. **Skills Section**: Highlights key technical and soft skills.
4. **Projects Section**: Provides an overview of completed and ongoing projects.
5. **Interactive UI**: A well-structured and visually appealing interface using Material Design principles.
6. **Responsiveness**: Ensures a smooth experience across various screen sizes.
7. **Scalability**: Can be expanded to include additional sections like work experience, certifications, and contact details.

UI/UX Design Principles

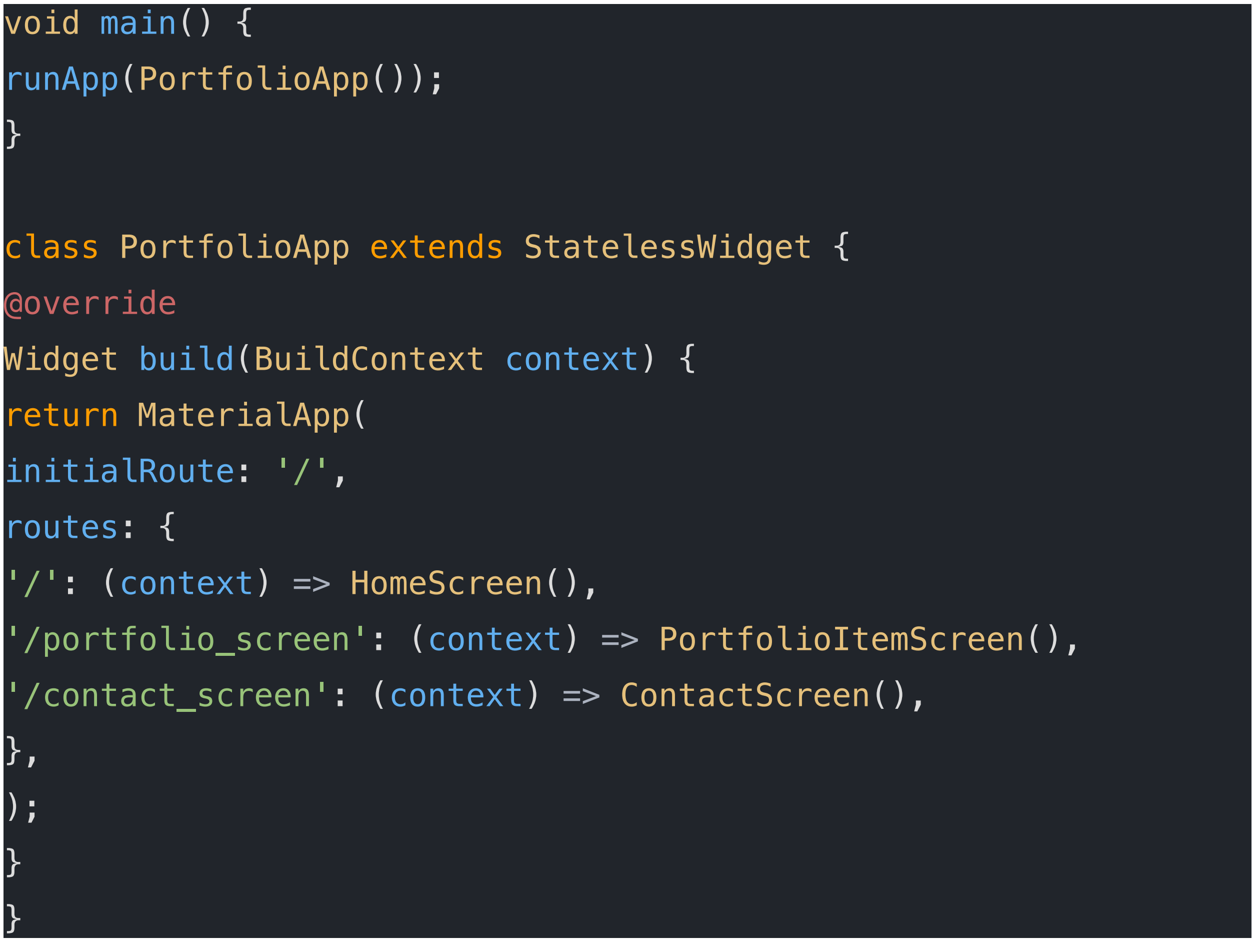
The app follows modern UI/UX design principles with a clean layout, readable typography, and intuitive navigation. The design ensures that users can easily browse through different sections with a seamless experience.

CODE:

main.dart

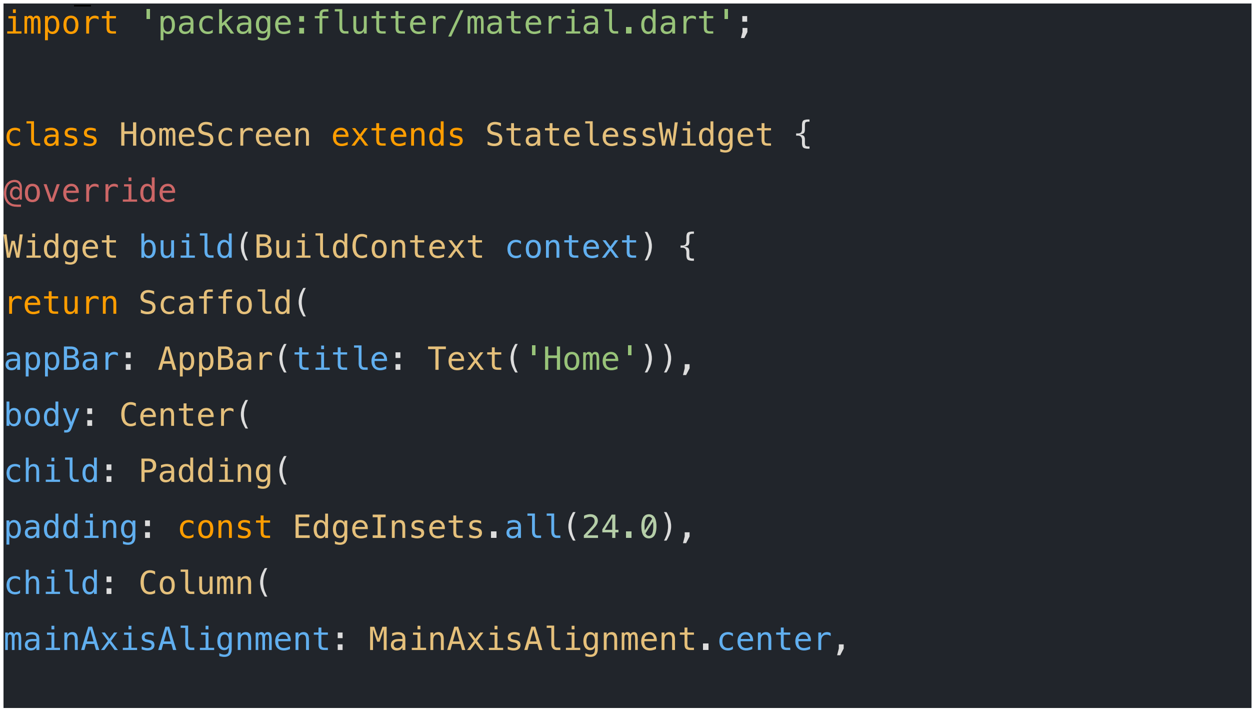


```dart
import 'package:flutter/material.dart';
import 'home_screen.dart';
import 'portfolio_screen.dart';
import 'contact_screen.dart';
```



```
void main() {
runApp(PortfolioApp());
}

class PortfolioApp extends StatelessWidget {
@override
Widget build(BuildContext context) {
return MaterialApp(
initialRoute: '/',
routes: {
'/': (context) => HomeScreen(),
'/portfolio_screen': (context) => PortfolioItemScreen(),
'/contact_screen': (context) => ContactScreen(),
},
);
}
}
```

home_screen.dart



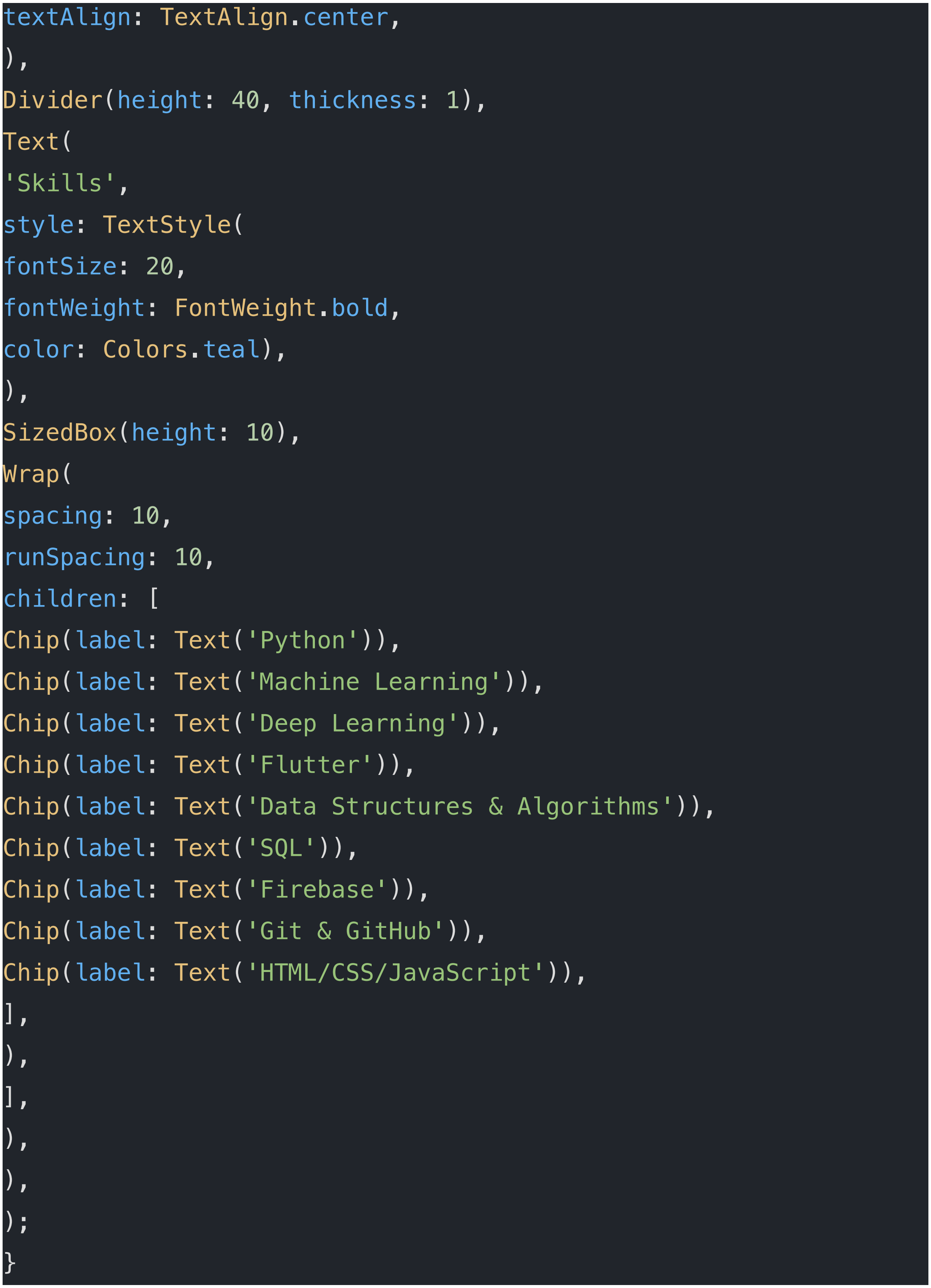
```
import 'package:flutter/material.dart';

class HomeScreen extends StatelessWidget {
@override
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar(title: Text('Home')),
body: Center(
child: Padding(
padding: const EdgeInsets.all(24.0),
child: Column(
mainAxisAlignment: MainAxisAlignment.center,
```

```dart
children: [
Text(
'Welcome to My Portfolio',
style: TextStyle(fontSize: 28, fontWeight: FontWeight.bold),
textAlign: TextAlign.center,
),
SizedBox(height: 40),
ElevatedButton.icon(
icon: Icon(Icons.person),
label: Text('Portfolio'),
style: ElevatedButton.styleFrom(
minimumSize: Size(double.infinity, 50)),
onPressed: () {
Navigator.pushNamed(context, '/portfolio_screen');
},
),
SizedBox(height: 20),
ElevatedButton.icon(
icon: Icon(Icons.contact_mail),
label: Text('Contact'),
style: ElevatedButton.styleFrom(
minimumSize: Size(double.infinity, 50)),
onPressed: () {
Navigator.pushNamed(context, '/contact_screen');
},
),
],
),
),
),
);
```

```
}
}
```

portfolio_screen.dart

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

class PortfolioItemScreen extends StatelessWidget {
@override
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar(title: Text('My Portfolio')),
body: SingleChildScrollView(
padding: EdgeInsets.all(16),
child: Column(
children: [
CircleAvatar(
radius: 60,
backgroundImage: AssetImage('assets/Kinjal.jpeg'),
),
SizedBox(height: 20),
Text(
'Kinjal Bansal',
style: TextStyle(
fontSize: 24,
fontWeight: FontWeight.bold,
color: Colors.teal),
),
SizedBox(height: 10),
Text(
'B.Tech 3rd Year - AI & DS\nVIPS, GGSIPU',
style: TextStyle(fontSize: 16),
```

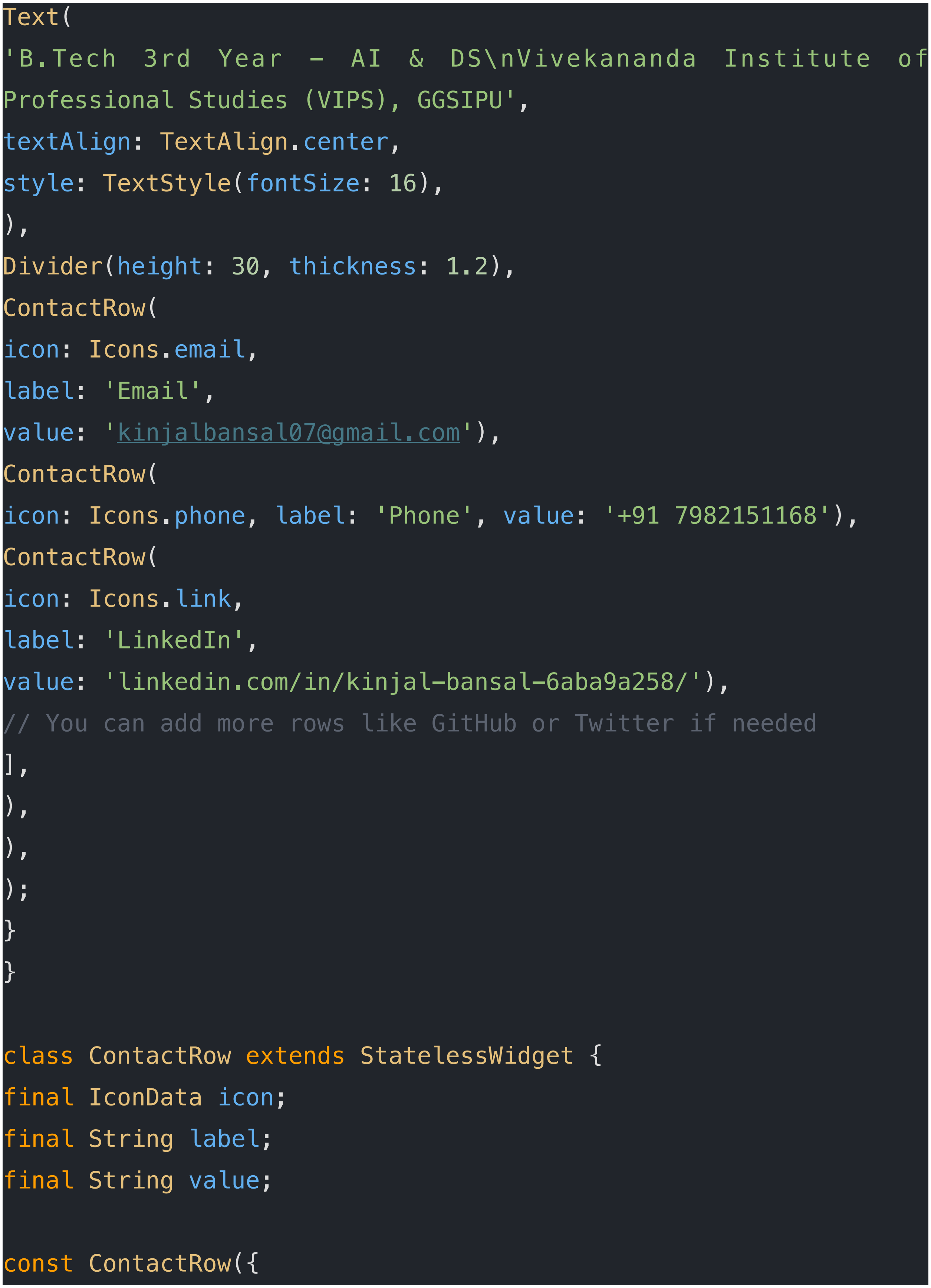


```
textAlign: TextAlign.center,
),
Divider(height: 40, thickness: 1),
Text(
'Skills',
style: TextStyle(
fontSize: 20,
fontWeight: FontWeight.bold,
color: Colors.teal),
),
SizedBox(height: 10),
Wrap(
spacing: 10,
runSpacing: 10,
children: [
Chip(label: Text('Python')),
Chip(label: Text('Machine Learning')),
Chip(label: Text('Deep Learning')),
Chip(label: Text('Flutter')),
Chip(label: Text('Data Structures & Algorithms')),
Chip(label: Text('SQL')),
Chip(label: Text('Firebase')),
Chip(label: Text('Git & GitHub')),
Chip(label: Text('HTML/CSS/JavaScript')),
],
),
],
),
),
);
}
```

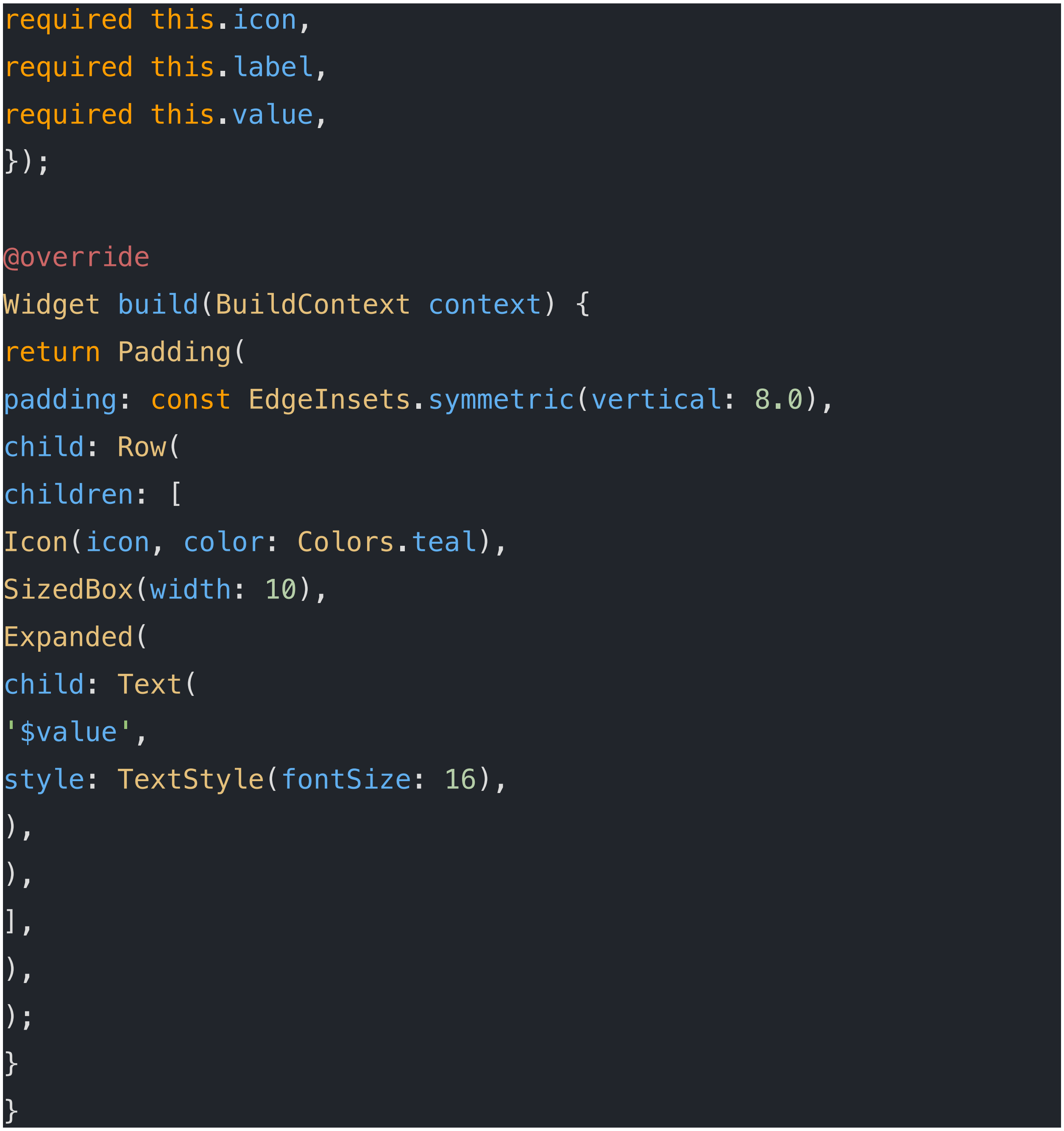
```
}
```

contact_screen.dart

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

class ContactScreen extends StatelessWidget {
@override
Widget build(BuildContext context) {
return Scaffold(
appBar: AppBar(
title: Text('Contact Me'),
),
body: Padding(
padding: const EdgeInsets.all(20.0),
child: Column(
children: [
CircleAvatar(
radius: 60,
backgroundImage: AssetImage(
'assets/Kinjal.jpeg'), // Make sure the image is in assets
),
SizedBox(height: 16),
Text(
'Kinjal Bansal',
style: TextStyle(
fontSize: 24,
fontWeight: FontWeight.bold,
color: Colors.teal),
),
SizedBox(height: 8),
```

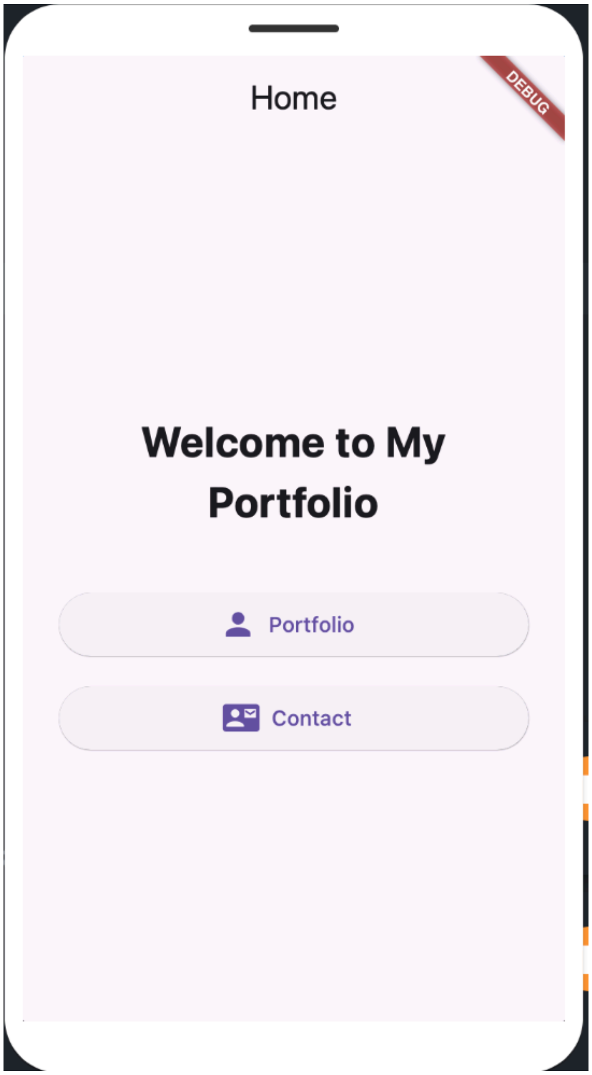


```
Text(
'B.Tech 3rd Year - AI & DS\nVivekananda Institute of Professional Studies (VIPS), GGSIPU',
textAlign: TextAlign.center,
style: TextStyle(fontSize: 16),
),
Divider(height: 30, thickness: 1.2),
ContactRow(
icon: Icons.email,
label: 'Email',
value: 'kinjalbansal07@gmail.com'),
ContactRow(
icon: Icons.phone, label: 'Phone', value: '+91 7982151168'),
ContactRow(
icon: Icons.link,
label: 'LinkedIn',
value: 'linkedin.com/in/kinjal-bansal-6aba9a258/'),
// You can add more rows like GitHub or Twitter if needed
],
),
),
);
}
}

class ContactRow extends StatelessWidget {
final IconData icon;
final String label;
final String value;

const ContactRow({
```



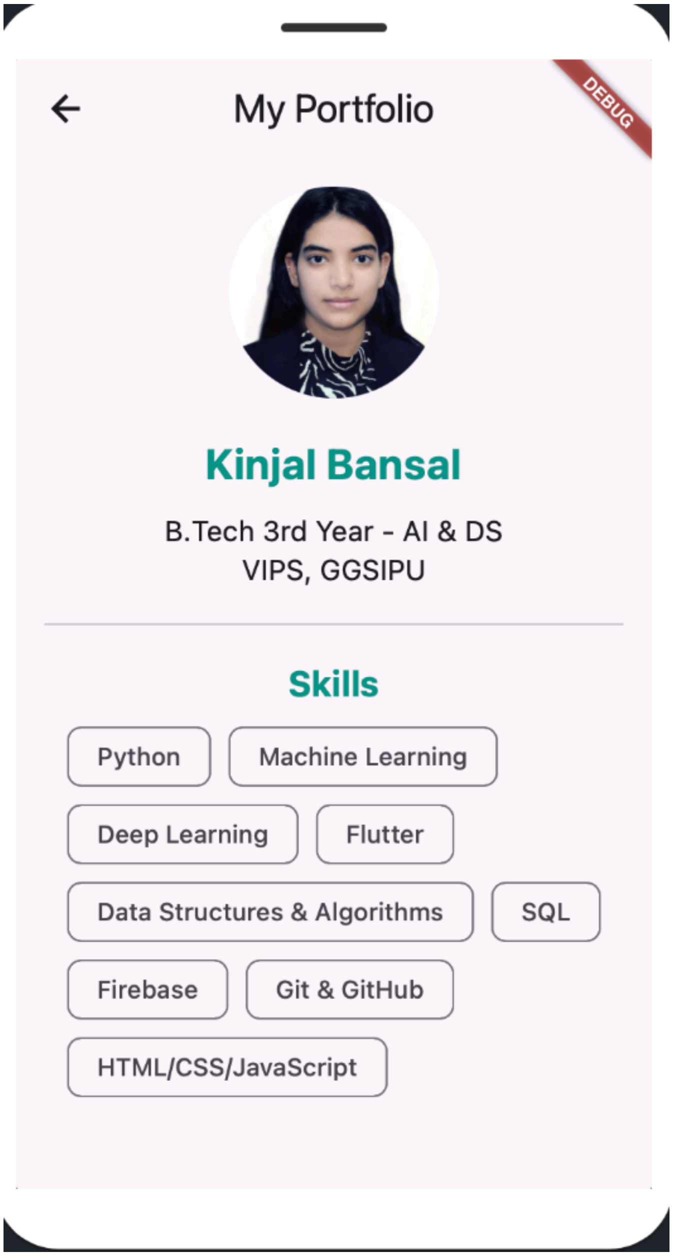
```
required this.icon,
required this.label,
required this.value,
});

@override
Widget build(BuildContext context) {
return Padding(
padding: const EdgeInsets.symmetric(vertical: 8.0),
child: Row(
children: [
Icon(icon, color: Colors.teal),
SizedBox(width: 10),
Expanded(
child: Text(
'$value',
style: TextStyle(fontSize: 16),
),
),
],
),
);
}
}
```

OUTPUT:



Home

Welcome to My Portfolio



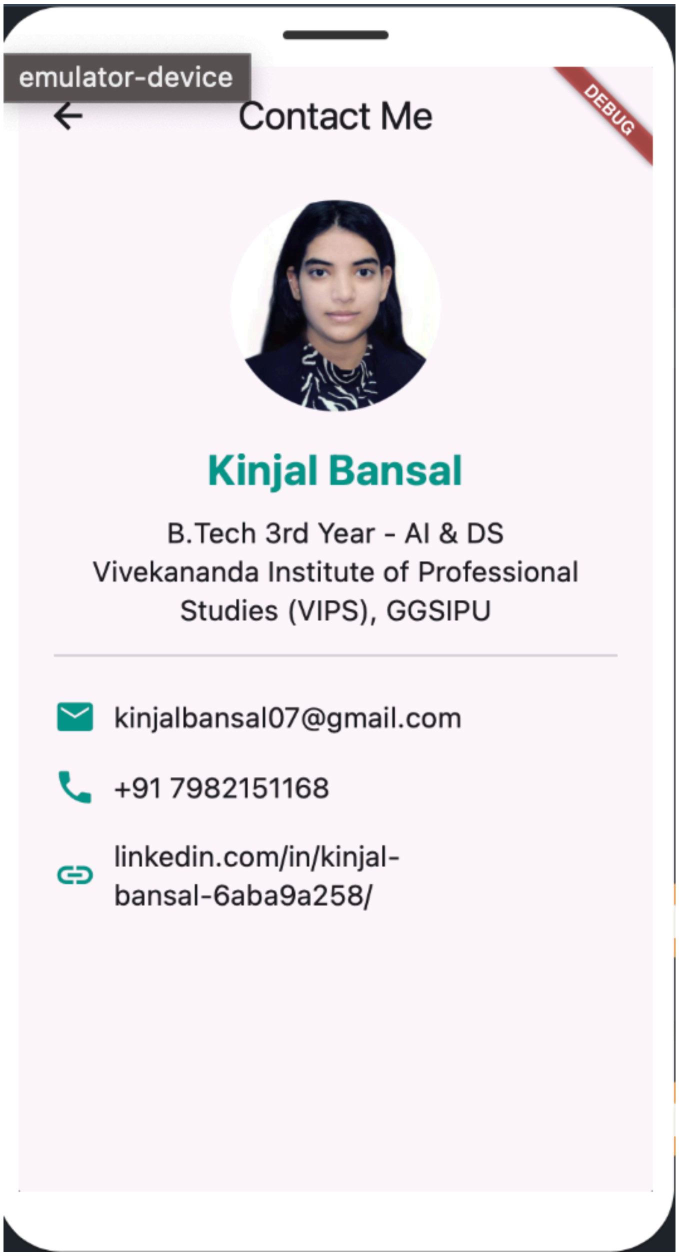
My Portfolio

Kinjal Bansal

B.Tech 3rd Year - AI & DS
VIPS, GGSIPU

Skills

Python
Machine Learning
Deep Learning
Flutter
Data Structures & Algorithms
SQL
Firebase
Git & GitHub
HTML/CSS/JavaScript



Contact Me

Kinjal Bansal

B.Tech 3rd Year - AI & DS
Vivekananda Institute of Professional
Studies (VIPS), GGSIPU

✉ kinjalbansal07@gmail.com

☎ +91 7982151168

🔗 linkedin.com/in/kinjal-bansal-6aba9a258/

Learning Outcome :