

```
b) Write a simple dart program to understand the language basics.
import 'package:flutter/material.dart';
 void main() {
  runApp(Abc());
class Abc extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
   return MaterialApp(
    home: Def(),
   );
class Def extends StatelessWidget {
  const Def({super.key});
@override
  Widget build(BuildContext context) {
   return Scaffold(
    appBar: AppBar(
     title: Text("Welcome"),
     backgroundColor: Colors.purple,
    ),
    body: Column(
     children: [
      //Widgets
     ],
OUTPUT:
```



1.c) Write a Dart console program that prints your name, checks age with conditionals, uses a loop to count from 1 to 5, and defines a function to return the sum of two numbers.

PROGRAM:

```
int addNumbers(int a, int b) {
 return a + b;
void main() {
 // 1. Print your name
 String name = "Chandini"; // you can replace with your own name
 print("My name is $name");
 // 2. Check age with conditionals
 int age = 22; // change value to test
 if (age >= 18) {
  print("You are an adult.");
 } else {
  print("You are a minor.");
 // 3. Loop to count from 1 to 5
 print("Counting from 1 to 5:");
 for (int i = 1; i \le 5; i++) {
  print(i);
 // 4. Use the sum function
 int x = 10, y = 20;
 int result = addNumbers(x, y);
 print("The sum of $x and $y is $result");
```



7 b) Implement form validation and error handling. PROGRAM:

```
import 'package:flutter/material.dart';
void main() {
 runApp(MyApp());
}
class MyApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
  return MaterialApp(
   title: 'Form Example',
   home: Scaffold(
    appBar: AppBar(
      title: Text('Form Example'),
    ),
    body: SingleChildScrollView(
      padding: EdgeInsets.all(16),
      child: FormWidget(),
    ),
   ),
class FormWidget extends StatefulWidget {
 @override
 FormWidgetState createState() => FormWidgetState();
class FormWidgetState extends State<FormWidget> {
 final formKey = GlobalKey<FormState>();
 String? name;
 String? email;
 String? password;
 String? phone;
```

Aditya College of Engineering and Technology (A)

Reg No:

2 3 P 3 1 A 4 2



```
String? _address;
@override
Widget build(BuildContext context) {
 return Form(
  key: formKey,
  child: Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: <Widget>[
     // Name field
     TextFormField(
      decoration: InputDecoration(labelText: 'Name'),
      validator: (value) {
       if (value == null || value.isEmpty) {
        return 'Please enter your name';
       return null;
      },
      onSaved: (value) => name = value,
     SizedBox(height: 16),
     // Email field
     TextFormField(
      decoration: InputDecoration(labelText: 'Email'),
      keyboardType: TextInputType.emailAddress,
      validator: (value) {
       if (value == null || value.isEmpty) {
         return 'Please enter your email';
       if (!RegExp(r'\S+@\S+\.\S+').hasMatch(value)) {
        return 'Please enter a valid email';
       return null;
      onSaved: (value) => email = value,
     SizedBox(height: 16),
```



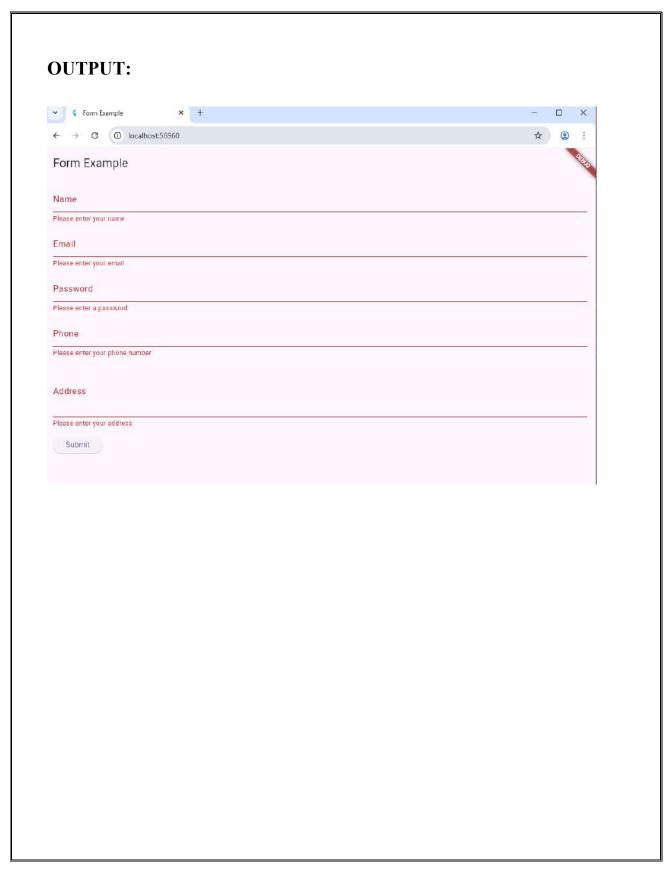
```
// Password field
 TextFormField(
  decoration: InputDecoration(labelText: 'Password'),
  obscureText: true,
  validator: (value) {
   if (value == null || value.isEmpty) {
     return 'Please enter a password';
   if (value.length < 6) {
     return 'Password must be at least 6 characters';
   return null;
  },
  onSaved: (value) => _password = value,
 SizedBox(height: 16),
 // Phone field
 TextFormField(
  decoration: InputDecoration(labelText: 'Phone'),
  keyboardType: TextInputType.phone,
  validator: (value) {
   if (value == null || value.isEmpty) {
     return 'Please enter your phone number';
   return null;
  onSaved: (value) => phone = value,
 SizedBox(height: 16),
 // Address field
 TextFormField(
  decoration: InputDecoration(labelText: 'Address'),
  maxLines: 3,
  validator: (value) {
   if (value == null || value.isEmpty) {
     return 'Please enter your address';
```



```
return null;
      onSaved: (value) => address = value,
    SizedBox(height: 16),
    // Submit button
    ElevatedButton(
     onPressed: submitForm,
     child: Text('Submit'),
void submitForm() {
if ( formKey.currentState!.validate()) {
  formKey.currentState!.save();
  // Print form data
  print('Form submitted:');
  print('Name: $ name');
  print('Email: $ email');
  print('Password: $ password');
  print('Phone: $_phone');
  print('Address: $ address');
  // Optional: Show confirmation on screen
  ScaffoldMessenger.of(context).showSnackBar(
   SnackBar(content: Text('Form submitted successfully!')),
  );
```

EXP NO: DATE:







9. a) Fetch data from REST API

```
dependancy in pubspec.yaml:
 dependencies:
  flutter:
   sdk: flutter
  http: ^1.1.0
In your terminal, run:
flutter pub get
 PROGRAM:
 import 'dart:convert';
import 'package:flutter/material.dart';
 import 'package:http/http.dart' as http;
 void main() {
  runApp(MyApp());
 }
class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
   return MaterialApp(
    title: 'API Data Example',
    home: HomePage(),
   );
class HomePage extends StatefulWidget {
  @override
  HomePageState createState() => HomePageState();
 class HomePageState extends State<HomePage> {
 List<dynamic> data = [];
  bool isLoading = true;
```



```
String? _error;
@override
void initState() {
 super.initState();
  _fetchDataFromApi();
Future<void> fetchDataFromApi() async {
 try {
   final response = await http
     .get(Uri.parse('https://jsonplaceholder.typicode.com/posts'));
  if (response.statusCode == 200) {
    setState(() {
     data = json.decode(response.body);
     isLoading = false;
    });
   } else {
    setState(() {
     error = 'Failed to load data';
     _isLoading = false;
    });
  } catch (e) {
   setState(() {
    _error = e.toString();
    _isLoading = false;
  });
@override
Widget build(BuildContext context) {
 return Scaffold(
   appBar: AppBar(
    title: Text('API Data Example'),
  ),
```

Aditya College of Engineering and Technology (A)

Reg No:



EXP NO: DATE:



```
body: isLoading
           ? Center(child: CircularProgressIndicator())
           : error != null
           ? Center(child: Text('Error: $ error'))
           : ListView.builder(
        itemCount: data.length,
        itemBuilder: (context, index) {
           return ListTile(
             title: Text(_data[index]['title'] ?? "),
             subtitle: Text( data[index]['body'] ?? "),
           );
OUTPUT:
✓ 《 API Data Example
                                                                                                                         @ :

← → ♂ ⊙ localhost:58229

 API Data Example
 sunt aut facere repellat provident occaecati excepturi optio reprehenderit
 quia et suscipit
 suscipit recusandae consequuntur expedita et cum
 reprehenderit molestiae ut ut quas totam
 nostrum rerum est autem sunt rem eveniet architecto
 qui est esse
 est rerum tempore vitae
 segui sint nihil reprehenderit dolor beatae ea dolores neque
 fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis
 qui aperiam non debitis possimus qui neque nisi nulla
 ea molestias quasi exercitationem repellat qui ipsa sit aut
 et iusto sed quo iure
 voluptatem occaecati omnis eligendi aut ad
voluptatem doloribus vel accusantium quis pariatur
 molestiae porro eius odio et labore et velit aut
 eum et est occaecati
 ullam et saepe reiciendis voluptatem adipisci
 sit amet autem assumenda provident rerum culpa
 quis hic commodi nesciunt rem tenetur doloremque ipsam lure quis sunt voluptatem rerum illo velit
 nesciunt quas odio
 repudiandae veniam quaerat sunt sed
 alias aut fugiat sit autem sed est
 voluptatem omnis possimus esse voluptatibus quis
 est aut tenetur dolor neque
 dolorem eum magni eos aperiam quia
 ut aspernatur corporis harum nihil quis provident sequi
 mollitia nobis aliquid molestiae
perspiciatis et ea nemo ab reprehenderit accusantium quas
```