

Home screen

Weather - Bengaluru

DEMO



Current Temperature

23.5 °C



2025-08-11

Partly cloudy

Max: 25.2°C

Min: 20.0°C



2025-08-12

Thunderstorm

Max: 26.3°C

Min: 19.6°C



2025-08-13

Thunderstorm

Max: 26.4°C

Min: 19.7°C



2025-08-14

Thunderstorm

Max: 25.1°C

Min: 18.9°C



2025-08-15

Unknown

Max: 25.2°C

Min: 19.4°C



2025-08-16

Unknown

Max: 26.8°C

Min: 19.5°C



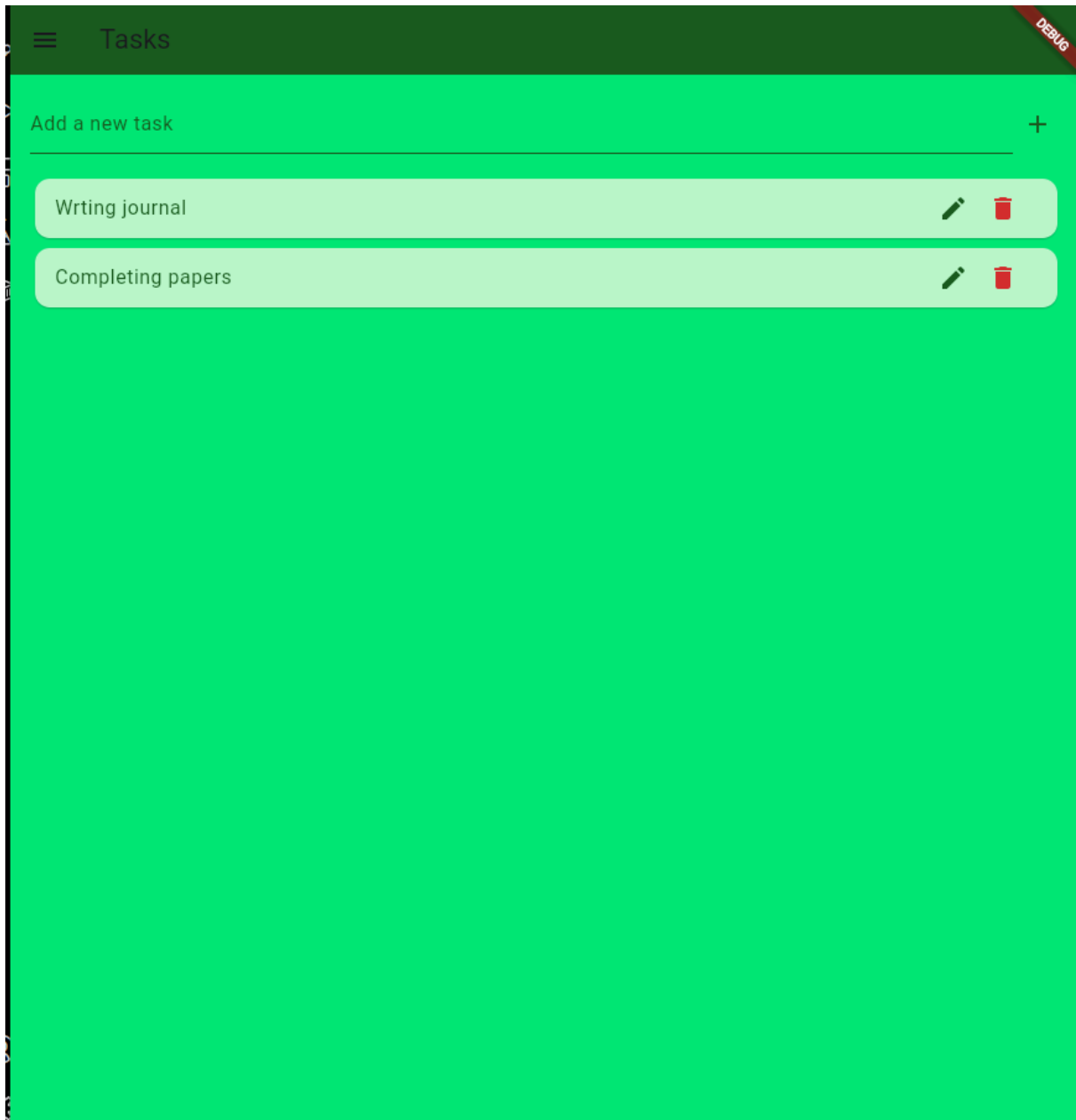
2025-08-17

Partly cloudy

Max: 25.9°C

Min: 19.9°C

Weather screen



Tasks screen

☰

Contact Admin

DEBUG

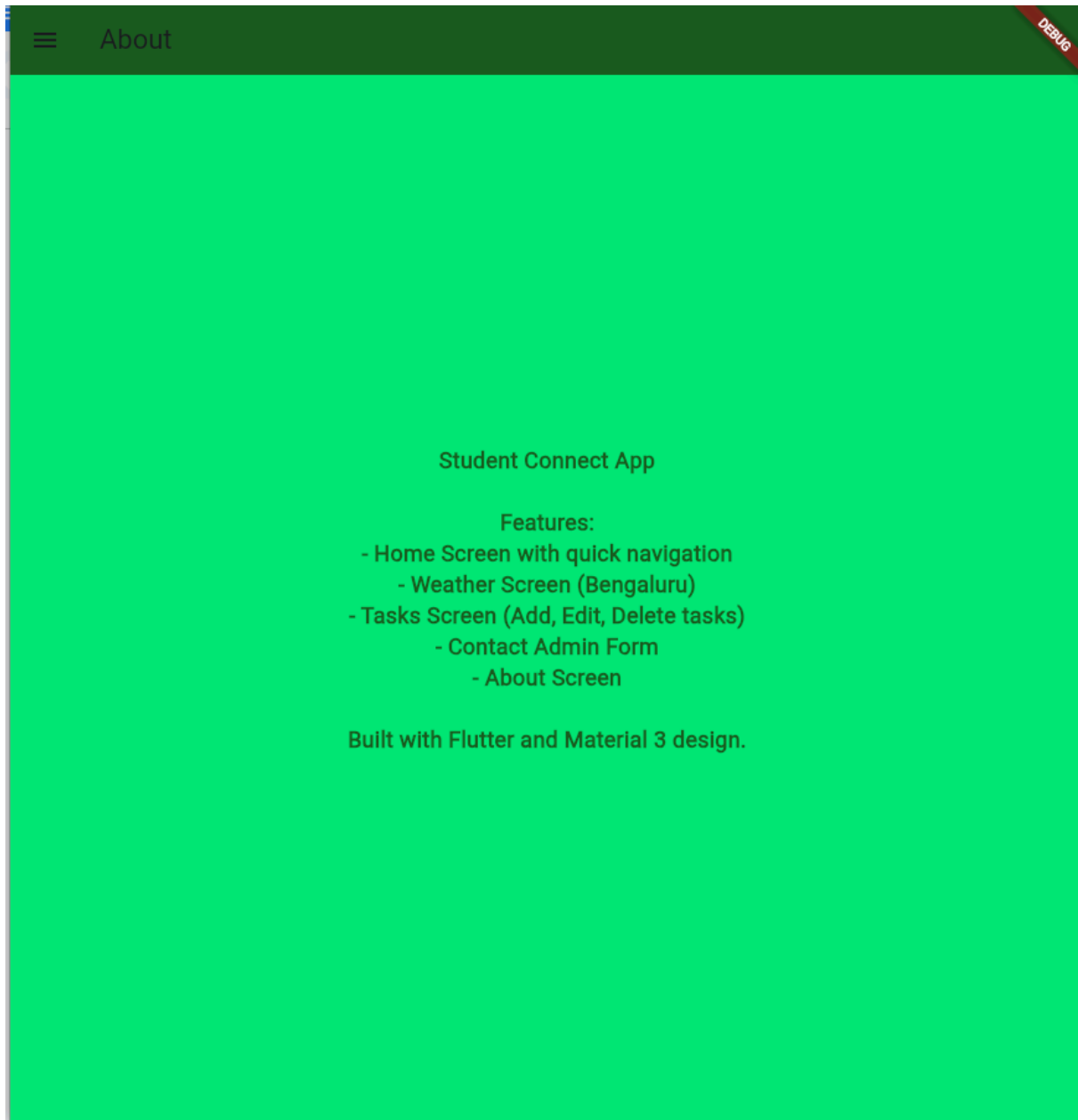
Your Name

Your Email

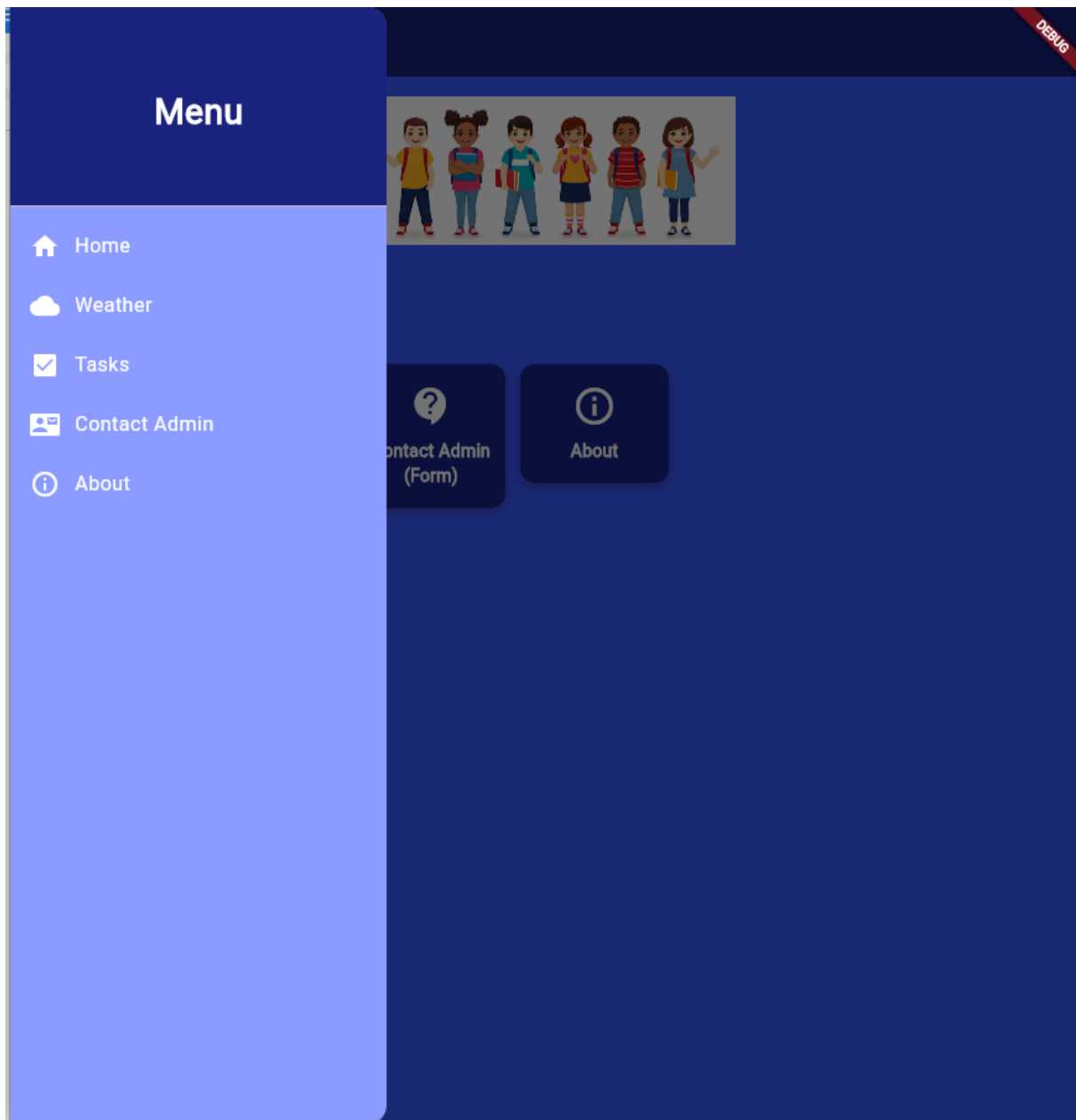
Message

Send Email

Contact admin screen



About Screen



Drawer

Main.dart

```
import 'package:flutter/material.dart';
import 'home_screen.dart';
import 'weather_screen.dart';
import 'tasks_screen.dart';
import 'contact_admin_screen.dart';
import 'about_screen.dart';

void main() {
```

```

runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Student Connect',
      theme: ThemeData(
        useMaterial3: true,
      ),
      initialRoute: '/',
      routes: {
        '/': (context) => const HomeScreen(),
        '/weather': (context) => const WeatherScreen(),
        '/tasks': (context) => const TasksScreen(),
        '/contact': (context) => const ContactAdminScreen(),
        '/about': (context) => const AboutScreen(),
      },
    );
  }
}

```

Home_screen.dart

```

import 'package:flutter/material.dart';

class HomeScreen extends StatelessWidget {
  const HomeScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.indigoAccent.shade400, // neon background
      appBar: AppBar(
        title: const Text('Student Connect'),
        backgroundColor: Colors.indigo.shade900,
      ),
      drawer: Drawer(
        backgroundColor: Colors.indigoAccent.shade100,
        child: ListView(

```

```
padding: EdgeInsets.zero,
children: [
  DrawerHeader(
    decoration: BoxDecoration(color: Colors.indigo.shade900),
    child: const Center(
      child: Text(
        'Menu',
        style: TextStyle(fontSize: 28, color: Colors.white,
fontWeight: FontWeight.bold),
      ),
    ),
  ),
  ListTile(
    leading: const Icon(Icons.home, color: Colors.white),
    title: const Text('Home', style: TextStyle(color:
Colors.white)),
    onTap: () => Navigator.pop(context),
  ),
  ListTile(
    leading: const Icon(Icons.cloud, color: Colors.white),
    title: const Text('Weather', style: TextStyle(color:
Colors.white)),
    onTap: () => Navigator.pushReplacementNamed(context,
'/weather'),
  ),
  ListTile(
    leading: const Icon(Icons.check_box, color: Colors.white),
    title: const Text('Tasks', style: TextStyle(color:
Colors.white)),
    onTap: () => Navigator.pushReplacementNamed(context,
'/tasks'),
  ),
  ListTile(
    leading: const Icon(Icons.contact_mail, color:
Colors.white),
    title: const Text('Contact Admin', style: TextStyle(color:
Colors.white)),
    onTap: () => Navigator.pushReplacementNamed(context,
'/contact'),
  ),
  ListTile(
    leading: const Icon(Icons.info_outline, color:
Colors.white),
```



```

        title: const Text('About', style: TextStyle(color:
Colors.white)),
        onTap: () => Navigator.pushReplacementNamed(context,
'/about'),
      ),
    ],
  ),
),
body: Padding(
  padding: const EdgeInsets.all(16),
  child: Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      // Image below AppBar
      Center(
        child: Image.asset(
          '/home/cselab3/1MS24AD029/Dart
Test/myapp/lib/240_F_293560899_5Cpb9oAwM0cRw8xcaGFIIJCZ265wqyQVu.jpg',
          height: 120,
          fit: BoxFit.contain,
        ),
      ),
      const SizedBox(height: 20),

      const Text(
        'Welcome!',
        style: TextStyle(
          fontSize: 24,
          fontWeight: FontWeight.bold,
          color: Colors.white,
        ),
      ),
      const SizedBox(height: 6),
      const Text(
        'Quick shortcuts to useful features.',
        style: TextStyle(
          fontSize: 14,
          color: Colors.white70,
        ),
      ),
      const SizedBox(height: 16),
      Wrap(
        spacing: 12,

```

```

        runSpacing: 12,
        children: [
          _buildOption(
            context,
            icon: Icons.wb_sunny,
            label: 'Weather\n(Bengaluru)',
            onTap: () => Navigator.pushNamed(context, '/weather'),
          ),
          _buildOption(
            context,
            icon: Icons.checklist,
            label: 'Tasks (CRUD)',
            onTap: () => Navigator.pushNamed(context, '/tasks'),
          ),
          _buildOption(
            context,
            icon: Icons.contact_support,
            label: 'Contact Admin\n(Form)',
            onTap: () => Navigator.pushNamed(context, '/contact'),
          ),
          _buildOption(
            context,
            icon: Icons.info_outline,
            label: 'About',
            onTap: () => Navigator.pushNamed(context, '/about'),
          ),
        ],
      ),
    ],
  ),
);
}

```

```

Widget _buildOption(BuildContext context,
  {required IconData icon, required String label, required
VoidCallback onTap}) {
  return GestureDetector(
    onTap: onTap,
    child: Container(
      width: 120,
      padding: const EdgeInsets.symmetric(vertical: 16),
      decoration: BoxDecoration(

```

```

        color: Colors.indigo.shade900.withOpacity(0.85),
        borderRadius: BorderRadius.circular(12),
        boxShadow: [
          BoxShadow(
            color: Colors.indigo.shade700.withOpacity(0.8),
            blurRadius: 6,
            offset: const Offset(0, 3),
          ),
        ],
      ),
      child: Column(
        mainAxisAlignment: MainAxisAlignment.min,
        children: [
          Icon(icon, size: 36, color: Colors.white),
          const SizedBox(height: 8),
          Text(
            label,
            textAlign: TextAlign.center,
            style: const TextStyle(color: Colors.white, fontWeight:
FontWeight.w600),
          ),
        ],
      ),
    ),
  );
}
}

```

About_screen.dart

```

import 'package:flutter/material.dart';

class AboutScreen extends StatelessWidget {
  const AboutScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.greenAccent.shade400,
      appBar: AppBar(
        title: const Text('About'),
        backgroundColor: Colors.green.shade900,
      ),
    ),
  ),
}

```

```

drawer: Drawer(
  backgroundColor: Colors.greenAccent.shade100,
  child: ListView(
    padding: EdgeInsets.zero,
    children: [
      DrawerHeader(
        decoration: BoxDecoration(color: Colors.green.shade700),
        child: Center(
          child: Text(
            'Menu',
            style: TextStyle(
              fontSize: 28,
              color: Colors.green.shade900,
              fontWeight: FontWeight.bold,
            ),
          ),
        ),
      ListTile(
        leading: Icon(Icons.home, color: Colors.green.shade900),
        title: const Text('Home'),
        onTap: () => Navigator.pushReplacementNamed(context, '/'),
      ),
      ListTile(
        leading: Icon(Icons.cloud, color: Colors.green.shade900),
        title: const Text('Weather'),
        onTap: () => Navigator.pushReplacementNamed(context,
'/weather'),
      ),
      ListTile(
        leading: Icon(Icons.check_box, color:
Colors.green.shade900),
        title: const Text('Tasks'),
        onTap: () => Navigator.pushReplacementNamed(context,
'/tasks'),
      ),
      ListTile(
        leading: Icon(Icons.contact_mail, color:
Colors.green.shade900),
        title: const Text('Contact Admin'),
        onTap: () => Navigator.pushReplacementNamed(context,
'/contact'),
      ),
    ],
  ),
)

```

```

        ListTile(
          leading: Icon(Icons.info_outline, color:
Colors.green.shade900),
          title: const Text('About'),
          onTap: () => Navigator.pop(context),
        ),
      ],
    ),
  ),
  body: Padding(
    padding: const EdgeInsets.all(16),
    child: Center(
      child: Text(
        'Student Connect App\n\n'
        'Features:\n'
        '- Home Screen with quick navigation\n'
        '- Weather Screen (Bengaluru)\n'
        '- Tasks Screen (Add, Edit, Delete tasks)\n'
        '- Contact Admin Form\n'
        '- About Screen\n\n'
        'Built with Flutter and Material 3 design.',
        style: TextStyle(
          fontSize: 18,
          color: Colors.green.shade900,
          fontWeight: FontWeight.w600,
          height: 1.4,
        ),
        textAlign: TextAlign.center,
      ),
    ),
  ),
),
);
}
}

```

Weather_screen.dart

```

import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
import 'dart:convert';

class WeatherScreen extends StatefulWidget {
  const WeatherScreen({super.key});

```

```

@override
State<WeatherScreen> createState() => _WeatherScreenState();
}

class _WeatherScreenState extends State<WeatherScreen> {
  Map<String, dynamic>? currentWeather;
  List<dynamic>? dailyTime;
  List<dynamic>? dailyWeatherCode;
  List<dynamic>? dailyTempMax;
  List<dynamic>? dailyTempMin;

  bool isLoading = true;
  String error = '';

  Future<void> fetchWeather() async {
    setState(() {
      isLoading = true;
      error = '';
    });

    try {
      final url = Uri.parse(
'https://api.open-meteo.com/v1/forecast?latitude=12.9716&longitude=77.5
946&current_weather=true&daily=weathercode,temperature_2m_max,temperatu
re_2m_min&timezone=Asia/Kolkata');
      final response = await http.get(url);

      if (response.statusCode == 200) {
        final data = json.decode(response.body);

        setState(() {
          currentWeather = data['current_weather'];
          dailyTime = data['daily']['time'];
          dailyWeatherCode = data['daily']['weathercode'];
          dailyTempMax = data['daily']['temperature_2m_max'];
          dailyTempMin = data['daily']['temperature_2m_min'];
          isLoading = false;
        });
      } else {
        setState(() {
          error = 'Failed to load weather data.';

```

```

        isLoading = false;
    });
}
} catch (e) {
    setState(() {
        error = 'Error: $e';
        isLoading = false;
    });
}
}

IconData getWeatherIcon(int code) {
    if ([0].contains(code)) return Icons.wb_sunny; // Clear sky
    if ([1, 2, 3].contains(code)) return Icons.cloud; // Partly cloudy
    if ([45, 48].contains(code)) return Icons.grain; // Fog
    if ([51, 53, 55, 56, 57].contains(code)) return Icons.grain; //
Drizzle
    if ([61, 63, 65, 66, 67].contains(code)) return Icons.umbrella; //
Rain
    if ([71, 73, 75, 77, 85, 86].contains(code)) return Icons.ac_unit;
// Snow
    if ([95, 96, 99].contains(code)) return Icons.flash_on; //
Thunderstorm
    return Icons.cloud_queue; // Default
}

String getWeatherDescription(int code) {
    switch (code) {
        case 0:
            return 'Clear sky';
        case 1:
        case 2:
        case 3:
            return 'Partly cloudy';
        case 45:
        case 48:
            return 'Fog';
        case 51:
        case 53:
        case 55:
        case 56:
        case 57:
            return 'Drizzle';
    }
}

```

```

        case 61:
        case 63:
        case 65:
        case 66:
        case 67:
            return 'Rain';
        case 71:
        case 73:
        case 75:
        case 77:
        case 85:
        case 86:
            return 'Snow';
        case 95:
        case 96:
        case 99:
            return 'Thunderstorm';
        default:
            return 'Unknown';
    }
}

@override
void initState() {
    super.initState();
    fetchWeather();
}

@override
Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(
            title: const Text('Weather - Bengaluru'),
            actions: [
                IconButton(
                    icon: const Icon(Icons.refresh),
                    onPressed: fetchWeather,
                )
            ],
        ),
        body: isLoading
            ? const Center(child: CircularProgressIndicator())
            : error.isNotEmpty

```



```

        ? Center(
          child: Text(
            error,
            style: const TextStyle(color: Colors.red),
          ),
        )
      : Padding(
        padding: const EdgeInsets.all(12),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.center,
          children: [
            // Current Weather
            Card(
              color: Colors.indigo.shade50,
              child: ListTile(
                leading: Icon(
getWeatherIcon(currentWeather!['weathercode']),
                  size: 48,
                  color: Colors.indigo,
                ),
                title: Text(
                  'Current Temperature',
                  style: TextStyle(
                    fontSize: 20,
                    fontWeight: FontWeight.bold,
                    color: Colors.indigo.shade900),
                ),
                subtitle: Text(
                  '${currentWeather!['temperature'].toStringAsFixed(1)} °C',
                  style: TextStyle(
                    fontSize: 18, color:
Colors.indigo.shade700),
                ),
              ),
            ),
            const SizedBox(height: 16),

            // Forecast List
            Expanded(
              child: ListView.builder(
                itemCount: dailyTime?.length ?? 0,

```

```

        itemBuilder: (context, index) {
            final date = dailyTime![index];
            final code = dailyWeatherCode![index];
            final maxTemp = dailyTempMax![index];
            final minTemp = dailyTempMin![index];

            return Card(
                margin:
                    const EdgeInsets.symmetric(vertical:
6),

                color: Colors.indigo.shade50,
                shape: RoundedRectangleBorder(
                    borderRadius:
BorderRadius.circular(12)),

                child: ListTile(
                    leading: Icon(
                        getWeatherIcon(code),
                        size: 36,
                        color: Colors.indigo,
                    ),
                    title: Text(
                        date,
                        style: TextStyle(
                            fontWeight: FontWeight.bold,
                            color: Colors.indigo.shade900),
                    ),
                    subtitle: Text(
                        getWeatherDescription(code),
                        style:
                            TextStyle(color:
Colors.indigo.shade600),
                    ),
                    trailing: Text(
                        'Max:
${maxTemp.toStringAsFixed(1)}°C\nMin: ${minTemp.toStringAsFixed(1)}°C',
                        style: TextStyle(color:
Colors.indigo.shade700),
                        textAlign: TextAlign.right,
                    ),
                ),
            );
        },
    ),

```

```

        ),
      ],
    ),
  ),
);
}
}

```

Contact_admin_screen.dart

```

import 'package:flutter/material.dart';

class ContactAdminScreen extends StatefulWidget {
  const ContactAdminScreen({super.key});

  @override
  State<ContactAdminScreen> createState() => _ContactAdminScreenState();
}

class _ContactAdminScreenState extends State<ContactAdminScreen> {
  final _formKey = GlobalKey<FormState>();
  final TextEditingController _nameController = TextEditingController();
  final TextEditingController _emailController =
TextEditingController();
  final TextEditingController _messageController =
TextEditingController();

  void _sendEmail() {
    if (!_formKey.currentState!.validate()) return;

    final emailContent = '''
Name: ${_nameController.text}
Email: ${_emailController.text}

Message:
${_messageController.text}
''';

    showDialog(
      context: context,
      builder: (_) => AlertDialog(
        title: const Text('Confirm Email Content'),
        content: SingleChildScrollView(child: Text(emailContent)),

```

```

actions: [
  TextButton(
    onPressed: () => Navigator.pop(context),
    child: const Text('Cancel'),
  ),
  ElevatedButton(
    onPressed: () {
      Navigator.pop(context);
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(content: Text('Pretending to send
email...'))),
      );
      // Here you can reset form if you want
      _formKey.currentState!.reset();
      _nameController.clear();
      _emailController.clear();
      _messageController.clear();
    },
    child: const Text('Send'),
  ),
],
),
);
}

Widget buildDrawer(BuildContext context) {
  return Drawer(
    child: ListView(
      padding: EdgeInsets.zero,
      children: [
        DrawerHeader(
          decoration: BoxDecoration(color: Colors.indigo.shade400),
          child: Center(
            child: Text(
              'Menu',
              style: TextStyle(
                fontSize: 28,
                color: Colors.indigo.shade900,
                fontWeight: FontWeight.bold,
              ),
            ),
          ),
        ),
      ],
    ),
  ),
);
}

```

```

        ListTile(
          leading: const Icon(Icons.home),
          title: const Text('Home'),
          onTap: () =>
Navigator.of(context).pushReplacementNamed('/'),
        ),
        ListTile(
          leading: const Icon(Icons.cloud),
          title: const Text('Weather'),
          onTap: () =>
Navigator.of(context).pushReplacementNamed('/weather'),
        ),
        ListTile(
          leading: const Icon(Icons.check_box),
          title: const Text('Tasks'),
          onTap: () =>
Navigator.of(context).pushReplacementNamed('/tasks'),
        ),
        ListTile(
          leading: const Icon(Icons.contact_mail),
          title: const Text('Contact Admin'),
          onTap: () =>
Navigator.of(context).pushReplacementNamed('/contact'),
        ),
        ListTile(
          leading: const Icon(Icons.info),
          title: const Text('About'),
          onTap: () =>
Navigator.of(context).pushReplacementNamed('/about'),
        ),
      ],
    ),
  );
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    drawer: buildDrawer(context),
    appBar: AppBar(title: const Text('Contact Admin')),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Form(

```

```

    key: _formKey,
    child: ListView(
      children: [
        TextFormField(
          controller: _nameController,
          decoration: const InputDecoration(
            labelText: 'Your Name',
            border: OutlineInputBorder(),
          ),
          validator: (value) => value == null || value.isEmpty ?
'Please enter your name' : null,
        ),
        const SizedBox(height: 16),
        TextFormField(
          controller: _emailController,
          decoration: const InputDecoration(
            labelText: 'Your Email',
            border: OutlineInputBorder(),
          ),
          keyboardType: TextInputType.emailAddress,
          validator: (value) {
            if (value == null || value.isEmpty) return 'Please
enter your email';
            final emailRegex = RegExp(r'^^[^@]+@[^@]+\.[^@]+' );
            if (!emailRegex.hasMatch(value)) return 'Please enter
a valid email';
            return null;
          },
        ),
        const SizedBox(height: 16),
        TextFormField(
          controller: _messageController,
          decoration: const InputDecoration(
            labelText: 'Message',
            border: OutlineInputBorder(),
          ),
          maxLines: 5,
          validator: (value) => value == null || value.isEmpty ?
'Please enter your message' : null,
        ),
        const SizedBox(height: 20),
        ElevatedButton(
          onPressed: _sendEmail,

```

```

        child: const Text('Send Email'),
      ),
    ],
  ),
),
);
}
}

```

Tasks_screen.dart

```

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

class TasksScreen extends StatefulWidget {
  const TasksScreen({super.key});

  @override
  State<TasksScreen> createState() => _TasksScreenState();
}

class _TasksScreenState extends State<TasksScreen> {
  final List<String> _tasks = [];
  final TextEditingController _controller = TextEditingController();

  @override
  void initState() {
    super.initState();
    _loadTasks();
  }

  Future<void> _loadTasks() async {
    final prefs = await SharedPreferences.getInstance();
    final savedTasks = prefs.getStringList('tasks') ?? [];
    setState(() {
      _tasks.addAll(savedTasks);
    });
  }

  Future<void> _saveTasks() async {
    final prefs = await SharedPreferences.getInstance();
    await prefs.setStringList('tasks', _tasks);
  }
}

```

```

}

void _addTask() {
  final text = _controller.text.trim();
  if (text.isNotEmpty) {
    setState(() {
      _tasks.add(text);
      _controller.clear();
    });
    _saveTasks();
  }
}

void _editTask(int index) {
  _controller.text = _tasks[index];
  showDialog(
    context: context,
    builder: (context) => AlertDialog(
      backgroundColor: Colors.greenAccent.shade100,
      title: Text(
        'Edit Task',
        style: TextStyle(color: Colors.green.shade900),
      ),
      content: TextField(
        controller: _controller,
        autofocus: true,
        style: TextStyle(color: Colors.green.shade900),
        decoration: InputDecoration(
          enabledBorder: UnderlineInputBorder(
            borderSide: BorderSide(color: Colors.green.shade900),
          ),
          focusedBorder: UnderlineInputBorder(
            borderSide: BorderSide(color: Colors.green.shade900),
          ),
        ),
      ),
      actions: [
        TextButton(
          onPressed: () {
            Navigator.pop(context);
            _controller.clear();
          },
        ),
      ],
    ),
  );
}

```



```

        child: Text('Cancel', style: TextStyle(color:
Colors.green.shade900)),
      ),
      TextButton(
        onPressed: () {
          final newText = _controller.text.trim();
          if (newText.isNotEmpty) {
            setState(() {
              _tasks[index] = newText;
            });
            _saveTasks();
            Navigator.pop(context);
            _controller.clear();
          }
        },
        child: Text('Save', style: TextStyle(color:
Colors.green.shade900)),
      ),
    ],
  ),
);
}

void _deleteTask(int index) {
  setState(() {
    _tasks.removeAt(index);
  });
  _saveTasks();
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Colors.greenAccent.shade400,
    appBar: AppBar(
      title: const Text('Tasks'),
      backgroundColor: Colors.green.shade900,
    ),
    drawer: Drawer(
      backgroundColor: Colors.greenAccent.shade100,
      child: ListView(
        padding: EdgeInsets.zero,
        children: [

```

```

DrawerHeader(
  decoration: BoxDecoration(color: Colors.green.shade700),
  child: Center(
    child: Text(
      'Menu',
      style: TextStyle(
        fontSize: 28,
        color: Colors.green.shade900,
        fontWeight: FontWeight.bold,
      ),
    ),
  ),
),
),
),
),
ListTile(
  leading: Icon(Icons.home, color: Colors.green.shade900),
  title: const Text('Home'),
  onTap: () => Navigator.pushReplacementNamed(context, '/'),
),
ListTile(
  leading: Icon(Icons.cloud, color: Colors.green.shade900),
  title: const Text('Weather'),
  onTap: () => Navigator.pushReplacementNamed(context,
'/weather'),
),
ListTile(
  leading: Icon(Icons.check_box, color:
Colors.green.shade900),
  title: const Text('Tasks'),
  onTap: () => Navigator.pop(context),
),
ListTile(
  leading: Icon(Icons.contact_mail, color:
Colors.green.shade900),
  title: const Text('Contact Admin'),
  onTap: () => Navigator.pushReplacementNamed(context,
'/contact'),
),
ListTile(
  leading: Icon(Icons.info_outline, color:
Colors.green.shade900),
  title: const Text('About'),
  onTap: () => Navigator.pushReplacementNamed(context,
'/about'),

```

```

    ),
  ],
),
),
body: Padding(
  padding: const EdgeInsets.all(16),
  child: Column(
    children: [
      Row(
        children: [
          Expanded(
            child: TextField(
              controller: _controller,
              style: TextStyle(color: Colors.green.shade900),
              decoration: InputDecoration(
                labelText: 'Add a new task',
                labelStyle: TextStyle(color:
Colors.green.shade900),
                enabledBorder: UnderlineInputBorder(
                  borderSide: BorderSide(color:
Colors.green.shade900),
                ),
                focusedBorder: UnderlineInputBorder(
                  borderSide: BorderSide(color:
Colors.green.shade900),
                ),
              ),
            ),
          ),
        ],
      ),
      IconButton(
        icon: Icon(Icons.add, color: Colors.green.shade900),
        onPressed: _addTask,
      ),
    ],
  ),
  const SizedBox(height: 16),
  Expanded(
    child: _tasks.isEmpty
      ? Center(
          child: Text(
            'No tasks yet.\nAdd some!',
            style: TextStyle(
              fontSize: 18,

```

```

        color: Colors.green.shade900,
        fontWeight: FontWeight.bold,
      ),
      textAlign: TextAlign.center,
    ),
  ),
: ListView.builder(
  itemCount: _tasks.length,
  itemBuilder: (context, index) {
    final task = _tasks[index];
    return Card(
      color: Colors.greenAccent.shade100,
      child: ListTile(
        title: Text(
          task,
          style: TextStyle(color:
Colors.green.shade900),
        ),
        trailing: Row(
          mainAxisAlignment: MainAxisAlignment.min,
          children: [
            IconButton(
              icon: Icon(Icons.edit, color:
Colors.green.shade900),
              onPressed: () => _editTask(index),
            ),
            IconButton(
              icon: Icon(Icons.delete, color:
Colors.red.shade700),
              onPressed: () => _deleteTask(index),
            ),
          ],
        ),
      ),
    );
  },
),
),
],
),
),
);
}
),
),
);
}

```

}