

# Missão Flutter — Etapas Práticas com Código

Este material reúne as **etapas práticas em Flutter** com seus respectivos **enunciados e soluções completas em código**. Os exemplos utilizam widgets essenciais como `Scaffold`, `TextField`, `ListView`, `Card`, e outros componentes para consolidar o que foi aprendido em Dart.

## ✓ Etapa 6 — Exibir informações fixas na tela

### ◆ Exercício 1: Exibir dados de um produto

**Enunciado:** Crie uma tela que exiba nome, preço, quantidade e disponibilidade de um produto.

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

class ProdutoInfoPage extends StatelessWidget {
  final Map<String, dynamic> produto = {
    'nome': 'Notebook',
    'preco': 3599.99,
    'quantidade': 12,
    'disponivel': true,
  };

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Informações do Produto')),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Text('Nome: \${produto['nome']}', style: TextStyle(fontSize:
18)),
            Text('Preço: R\$ \${produto['preco']}', style:
TextStyle(fontSize: 18)),
            Text('Estoque: \${produto['quantidade']} unidades', style:
TextStyle(fontSize: 18)),
            Text('Status: \${produto['disponivel']} ? 'Disponível' :
'Indisponível'},
              style: TextStyle(fontSize: 18, color:
produto['disponivel'] ? Colors.green : Colors.red)),
          ],
        ),
      ),
    );
  }
}
```

```
}  
}
```

## ♦ Exercício 2: Exibir dados de um aluno com situação

**Enunciado:** Crie uma tela que exiba nome, idade, curso e nota de um aluno. Exiba "Aprovado" ou "Reprovado" com base na nota.

```
import 'package:flutter/material.dart';  
  
class AlunoInfoPage extends StatelessWidget {  
  final Map<String, dynamic> aluno = {  
    'nome': 'Camila',  
    'idade': 20,  
    'curso': 'Engenharia de Software',  
    'nota': 5.8,  
  };  
  
  @override  
  Widget build(BuildContext context) {  
    final aprovado = aluno['nota'] >= 6;  
  
    return Scaffold(  
      appBar: AppBar(title: Text('Informações do Aluno')),  
      body: Padding(  
        padding: const EdgeInsets.all(16.0),  
        child: Column(  
          crossAxisAlignment: CrossAxisAlignment.start,  
          children: [  
            Text('Nome: \${aluno['nome']}', style: TextStyle(fontSize: 18)),  
            Text('Idade: \${aluno['idade']} anos', style:  
TextStyle(fontSize: 18)),  
            Text('Curso: \${aluno['curso']}', style: TextStyle(fontSize:  
18)),  
            Text('Nota final: \${aluno['nota']}', style: TextStyle(fontSize:  
18)),  
            Text(aprovado ? 'Aprovado' : 'Reprovado',  
              style: TextStyle(fontSize: 20, color: aprovado ?  
Colors.green : Colors.red, fontWeight: FontWeight.bold))  
          ],  
        ),  
      ),  
    );  
  }  
}
```

## ✓ Etapa 7 — Entrada de dados com TextFields

### ♦ Exercício 1: Cadastro de aluno e exibição dos dados

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

class CadastroAlunoPage extends StatefulWidget {
  @override
  State<CadastroAlunoPage> createState() => _CadastroAlunoPageState();
}

class _CadastroAlunoPageState extends State<CadastroAlunoPage> {
  final nomeController = TextEditingController();
  final idadeController = TextEditingController();
  final notaController = TextEditingController();
  String resultado = '';

  void exibirResumo() {
    String nome = nomeController.text;
    int idade = int.tryParse(idadeController.text) ?? 0;
    double nota = double.tryParse(notaController.text) ?? 0;

    setState(() {
      resultado = 'Aluno: \$nome\nIdade: \$idade\nNota: \$nota';
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Cadastro de Aluno')),
      body: Padding(
        padding: const EdgeInsets.all(16),
        child: Column(
          children: [
            TextField(controller: nomeController, decoration:
InputDecoration(labelText: 'Nome')),
            TextField(controller: idadeController, decoration:
InputDecoration(labelText: 'Idade'), keyboardType: TextInputType.number),
            TextField(controller: notaController, decoration:
InputDecoration(labelText: 'Nota'), keyboardType: TextInputType.number),
            SizedBox(height: 16),
            ElevatedButton(onPressed: exibirResumo, child: Text('Salvar')),
            SizedBox(height: 16),
            Text(resultado),
          ],
        ),
      ),
    );
  }
}
```

```
}  
}
```

## ♦ Exercício 2: Calculadora de IMC

```
import 'package:flutter/material.dart';  
  
class IMCPage extends StatefulWidget {  
  @override  
  State<IMCPage> createState() => _IMCPageState();  
}  
  
class _IMCPageState extends State<IMCPage> {  
  final pesoController = TextEditingController();  
  final alturaController = TextEditingController();  
  String resultado = '';  
  
  void calcularIMC() {  
    double peso = double.tryParse(pesoController.text) ?? 0;  
    double altura = double.tryParse(alturaController.text) ?? 0;  
    if (peso <= 0 || altura <= 0) return;  
  
    double imc = peso / (altura * altura);  
    String categoria;  
    if (imc < 18.5) {  
      categoria = 'Abaixo do peso';  
    } else if (imc < 24.9) {  
      categoria = 'Peso normal';  
    } else if (imc < 29.9) {  
      categoria = 'Sobrepeso';  
    } else {  
      categoria = 'Obesidade';  
    }  
  
    setState(() {  
      resultado = 'IMC: ${imc.toStringAsFixed(2)} - \">$categoria';  
    });  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(title: Text('Calculadora de IMC')),  
      body: Padding(  
        padding: const EdgeInsets.all(16),  
        child: Column(  
          children: [  
            TextField(controller: pesoController, decoration:  
InputDecoration(labelText: 'Peso (kg)'), keyboardType: TextInputType.number),
```

```

        TextField(controller: alturaController, decoration:
InputDecoration(labelText: 'Altura (m)'), keyboardType:
TextInputType.number),
        SizedBox(height: 16),
        ElevatedButton(onPressed: calcularIMC, child: Text('Calcular')),
        SizedBox(height: 16),
        Text(resultado),
      ],
    ),
  ),
);
}
}

```

## ✓ Etapa 8 — Lista dinâmica de objetos

### ♦ Exercício 1: Cadastro e listagem de produtos

```

import 'package:flutter/material.dart';

class ListaProdutosPage extends StatefulWidget {
  @override
  State<ListaProdutosPage> createState() => _ListaProdutosPageState();
}

class _ListaProdutosPageState extends State<ListaProdutosPage> {
  final nomeController = TextEditingController();
  final precoController = TextEditingController();
  List<Map<String, String>> produtos = [];

  void adicionarProduto() {
    setState(() {
      produtos.add({
        'nome': nomeController.text,
        'preco': precoController.text,
      });
      nomeController.clear();
      precoController.clear();
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Lista de Produtos')),
      body: Padding(
        padding: const EdgeInsets.all(16),
        child: Column(

```

```

        children: [
          TextField(controller: nomeController, decoration:
InputDecoration(labelText: 'Nome do produto')),
          TextField(controller: precoController, decoration:
InputDecoration(labelText: 'Preço'), keyboardType: TextInputType.number),
          SizedBox(height: 16),
          ElevatedButton(onPressed: adicionarProduto, child:
Text('Adicionar')),
          SizedBox(height: 16),
          Expanded(
            child: ListView.builder(
              itemCount: produtos.length,
              itemBuilder: (context, index) {
                final p = produtos[index];
                return ListTile(
                  title: Text(p['nome']!),
                  subtitle: Text('R\$ ${p['preco']}'),
                );
              },
            ),
          ),
        ],
      ),
    ),
  );
}
}

```

## ◆ Exercício 2: Agenda de contatos

```

import 'package:flutter/material.dart';

class AgendaContatosPage extends StatefulWidget {
  @override
  State<AgendaContatosPage> createState() => _AgendaContatosPageState();
}

class _AgendaContatosPageState extends State<AgendaContatosPage> {
  final nomeController = TextEditingController();
  final telefoneController = TextEditingController();
  List<Map<String, String>> contatos = [];

  void adicionarContato() {
    setState(() {
      contatos.add({
        'nome': nomeController.text,
        'telefone': telefoneController.text,
      });
      nomeController.clear();
    });
  }
}

```

```

        telefoneController.clear();
    });
}

@override
Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(title: Text('Agenda de Contatos')),
        body: Padding(
            padding: const EdgeInsets.all(16),
            child: Column(
                children: [
                    TextField(controller: nomeController, decoration:
InputDecoration(labelText: 'Nome')),
                    TextField(controller: telefoneController, decoration:
InputDecoration(labelText: 'Telefone')),
                    SizedBox(height: 16),
                    ElevatedButton(onPressed: adicionarContato, child:
Text('Adicionar')),
                    SizedBox(height: 16),
                    Expanded(
                        child: ListView(
                            children: contatos.map((c) => ListTile(
                                leading: Icon(Icons.phone),
                                title: Text(c['nome']!),
                                subtitle: Text(c['telefone']!),
                            )).toList(),
                        ),
                    ),
                ],
            ),
        ),
    );
}
}

```

## Etapa 9 — Busca e filtragem

### Exercício 1: Buscar produto por nome

```

import 'package:flutter/material.dart';

class BuscaProdutoPage extends StatefulWidget {
    @override
    State<BuscaProdutoPage> createState() => _BuscaProdutoPageState();
}

class _BuscaProdutoPageState extends State<BuscaProdutoPage> {

```

```

final buscaController = TextEditingController();
final produtos = [
  {'nome': 'Notebook'},
  {'nome': 'Smartphone'},
  {'nome': 'Fone'},
  {'nome': 'Carregador'}
];
String resultado = '';

void buscarProduto() {
  final nome = buscaController.text.toLowerCase();
  final encontrado = produtos.any((p) =>
p['nome']!.toLowerCase().contains(nome));
  setState(() {
    resultado = encontrado ? 'Produto encontrado!' : 'Produto não
encontrado.';
  });
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: Text('Buscar Produto')),
    body: Padding(
      padding: const EdgeInsets.all(16),
      child: Column(
        children: [
          TextField(controller: buscaController, decoration:
InputDecoration(labelText: 'Digite o nome')),
          SizedBox(height: 16),
          ElevatedButton(onPressed: buscarProduto, child: Text('Buscar')),
          SizedBox(height: 16),
          Text(resultado),
        ],
      ),
    ),
  );
}

```

## ♦ Exercício 2: Buscar aluno e exibir situação

```

import 'package:flutter/material.dart';

class BuscarAlunoPage extends StatefulWidget {
  @override
  State<BuscarAlunoPage> createState() => _BuscarAlunoPageState();
}

```



```

class _BuscarAlunoPageState extends State<BuscarAlunoPage> {
  final alunos = {
    'Lucas': 7.5,
    'Ana': 5.8,
    'Paulo': 6.9
  };
  final nomeController = TextEditingController();
  String resultado = '';

  void buscarAluno() {
    final nome = nomeController.text;
    final nota = alunos[nome];
    setState(() {
      if (nota != null) {
        resultado = 'Nota: \${nota} - \${nota >= 6 ? 'Aprovado' : 'Reprovado'}';
      } else {
        resultado = 'Aluno não encontrado';
      }
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Buscar Aluno')),
      body: Padding(
        padding: const EdgeInsets.all(16),
        child: Column(
          children: [
            TextField(controller: nomeController, decoration:
InputDecoration(labelText: 'Nome do aluno')),
            SizedBox(height: 16),
            ElevatedButton(onPressed: buscarAluno, child: Text('Buscar')),
            SizedBox(height: 16),
            Text(resultado),
          ],
        ),
      ),
    );
  }
}

```

## Etapa 10 — Exibição de dados em Cards (JSON)

### Exercício 1: Listar usuários simulando JSON

```

import 'package:flutter/material.dart';

```

```

class UsuariosPage extends StatelessWidget {
  final usuarios = [
    {'nome': 'Carlos', 'cidade': 'Londrina', 'idade': 28},
    {'nome': 'Julia', 'cidade': 'Curitiba', 'idade': 22},
    {'nome': 'Renata', 'cidade': 'São Paulo', 'idade': 34},
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Lista de Usuários')),
      body: ListView(
        children: usuarios.map((u) => Card(
          child: ListTile(
            leading: Icon(Icons.person),
            title: Text(u['nome']!),
            subtitle: Text('Cidade: \${u['cidade']} | Idade: \${u['idade']}'),
          ),
        )).toList(),
      ),
    );
  }
}

```

## ◆ Exercício 2: Listar pedidos com valores

```

import 'package:flutter/material.dart';

class PedidosPage extends StatelessWidget {
  final pedidos = [
    {'produto': 'Livro', 'quantidade': 2, 'total': 90.0},
    {'produto': 'Caderno', 'quantidade': 5, 'total': 75.0},
    {'produto': 'Mouse', 'quantidade': 1, 'total': 120.0},
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Pedidos')),
      body: ListView(
        children: pedidos.map((p) => Card(
          child: ListTile(
            leading: Icon(Icons.shopping_cart),
            title: Text(p['produto']!),
            subtitle: Text('Quantidade: \${p['quantidade']} | Total: R\$ \${p['total']}'),
          ),
        )).toList(),
      ),
    );
  }
}

```

```
    ),  
    );  
}  
}
```

---