

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

```
void main() => runApp(const MyApp());
```

```
class Produto {
```

```
  String nome;
```

```
  double precoCompra;
```

```
  double precoVenda;
```

```
  int quantidade;
```

```
  String descricao;
```

```
  String categoria;
```

```
  String imageUrl;
```

```
  bool ativo;
```

```
  bool promocao;
```

```
  double desconto;
```

```
  Produto({
```

```
    required this.nome,
```

```
    required this.precoCompra,
```

```
    required this.precoVenda,
```

```
    required this.quantidade,
```

```
    required this.descricao,
```

```
    required this.categoria,
```

```
    required this.imageUrl,
```

```
    required this.ativo,
```

```
    required this.promocao,
```

```
    required this.desconto,
```

```
  });
```

```
}
```

```
class MyApp extends StatelessWidget {
```

```
  const MyApp({super.key});
```

```
  @override
```

```
  Widget build(BuildContext context) {
```

```
    return MaterialApp(
```

```
      title: 'Cadastro de Produtos',
```

```
      theme: ThemeData(primarySwatch: Colors.blue),
```

```
      home: const CadastroEListaProdutos(),
```

```
    );
```

```
}  
}
```

```
class CadastroEListaProdutos extends StatefulWidget {  
  const CadastroEListaProdutos({super.key});  
  
  @override  
  State<CadastroEListaProdutos> createState() =>  
    _CadastroEListaProdutosState();  
}
```

```
class _CadastroEListaProdutosState extends State<CadastroEListaProdutos> {  
  final _formKey = GlobalKey<FormState>();  
  final List<Produto> _produtos = [];
```

```
  String _nome = "";  
  double _precoCompra = 0;  
  double _precoVenda = 0;  
  int _quantidade = 1;  
  String _descricao = "";  
  String _categoria = 'Alimentos';  
  String _imagemUrl = "";  
  bool _ativo = true;  
  bool _promocao = false;  
  double _desconto = 0;
```

```
  void _adicionarProduto() {  
    if (_formKey.currentState!.validate()) {  
      _formKey.currentState!.save();  
      setState(() {  
        _produtos.add(Produto(  
          nome: _nome,  
          precoCompra: _precoCompra,  
          precoVenda: _precoVenda,  
          quantidade: _quantidade,  
          descricao: _descricao,  
          categoria: _categoria,  
          imagemUrl: _imagemUrl,  
          ativo: _ativo,  
          promocao: _promocao,
```

```

        desconto: _desconto,
    ));
});
_formKey.currentState!.reset();
_ativo = true;
_promocao = false;
_desconto = 0;
_categoria = 'Alimentos';
}
}

```

@override

```

Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: const Text('Cadastro e Lista de Produtos')),
    body: Padding(
      padding: const EdgeInsets.all(12.0),
      child: Column(
        children: [
          Expanded(
            child: Form(
              key: _formKey,
              child: ListView(
                children: [
                  TextFormField(
                    decoration: const InputDecoration(labelText: 'Nome'),
                    validator: (value) => value!.isEmpty ? 'Informe o nome' : null,
                    onSave: (value) => _nome = value!,
                  ),
                  TextFormField(
                    decoration: const InputDecoration(labelText: 'Preço de Compra'),
                    keyboardType: TextInputType.number,
                    validator: (value) => value!.isEmpty ? 'Informe o preço de compra' :
null,
                    onSave: (value) => _precoCompra = double.parse(value!),
                  ),
                  TextFormField(
                    decoration: const InputDecoration(labelText: 'Preço de Venda'),
                    keyboardType: TextInputType.number,
                    validator: (value) => value!.isEmpty ? 'Informe o preço de venda' : null,

```

```
    onSave: (value) => _precoVenda = double.parse(value!),
  ),
  TextFormField(
    decoration: const InputDecoration(labelText: 'Quantidade'),
    keyboardType: TextInputType.number,
    validator: (value) => value!.isEmpty ? 'Informe a quantidade' : null,
    onSave: (value) => _quantidade = int.parse(value!),
  ),
  TextFormField(
    decoration: const InputDecoration(labelText: 'Descrição'),
    validator: (value) => value!.isEmpty ? 'Informe a descrição' : null,
    onSave: (value) => _descricao = value!,
  ),
  const SizedBox(height: 10),
  const Text('Categoria:'),
  RadioListTile(
    title: const Text('Alimentos'),
    value: 'Alimentos',
    groupValue: _categoria,
    onChanged: (value) => setState(() => _categoria = value!),
  ),
  RadioListTile(
    title: const Text('Eletrônicos'),
    value: 'Eletrônicos',
    groupValue: _categoria,
    onChanged: (value) => setState(() => _categoria = value!),
  ),
  TextFormField(
    decoration: const InputDecoration(labelText: 'URL da Imagem'),
    onSave: (value) => _imagemUrl = value ?? "",
  ),
  SwitchListTile(
    title: const Text('Produto Ativo'),
    value: _ativo,
    onChanged: (value) => setState(() => _ativo = value),
  ),
  CheckboxListTile(
    title: const Text('Em Promoção'),
    value: _promocao,
    onChanged: (value) => setState(() => _promocao = value!),
```

```

    ),
    const Text('Desconto:'),
    Slider(
      value: _desconto,
      onChanged: (value) => setState(() => _desconto = value),
      min: 0,
      max: 100,
      divisions: 20,
      label: '${_desconto.round()}%',
    ),
    const SizedBox(height: 10),
    ElevatedButton(
      onPressed: _adicionarProduto,
      child: const Text('Cadastrar Produto'),
    ),
    const Divider(),
    const Text('Produtos Cadastrados:', style: TextStyle(fontSize: 18)),
  ],
),
),
),
Expanded(
  child: ListView.builder(
    itemCount: _produtos.length,
    itemBuilder: (context, index) {
      final produto = _produtos[index];
      return Card(
        child: ListTile(
          leading: Image.network(
            produto.imagemUrl,
            width: 50,
            height: 50,
            errorBuilder: (context, error, stackTrace) => const
Icon(Icons.broken_image),
          ),
          title: Text(produto.nome),
          subtitle: Text('R\$ ${produto.precoVenda.toStringAsFixed(2)} - $
{produto.categoria}'),
        ),
      );

```

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