

# HIVE

---

Julia da Rosa e Leonardo Felix Scandura

# O que é Hive?

---

Hive é um sistema armazenamento de dados de código aberto projetado para ser ágil, eficiente e user-friendly.

Destacando-se como uma escolha proeminente no desenvolvimento de aplicativos Flutter, Hive oferece uma abordagem simplificada para armazenamento local de dados.

Sua popularidade nesse ecossistema decorre não apenas de sua simplicidade, mas também da rapidez e eficiência que proporciona.

# Principais Características

---

- **Leve e Rápido:**
  - Hive é construído para ser extremamente leve e rápido, sendo uma excelente escolha para aplicativos móveis.
- **Suporte a Tipos de Dados:**
  - Hive suporta uma variedade de tipos de dados, incluindo tipos personalizados e coleções.
- **Armazenamento no Dispositivo:**
  - Os dados são armazenados localmente no dispositivo, o que é ideal para aplicativos que precisam de armazenamento offline.
- **Suporte a Transações:**
  - Hive oferece suporte a transações, garantindo consistência nos dados.

BOXES

# Boxes

---

Todos os dados armazenados no Hive são organizados em caixas (Boxes).

Uma caixa pode ser comparada a uma tabela em SQL, mas não possui uma estrutura definida e pode conter qualquer tipo de informação.

Para aplicativos pequenos, uma única caixa pode ser suficiente. Para problemas mais avançados, as caixas são uma excelente maneira de organizar seus dados. As caixas também podem ser criptografadas para armazenar dados sensíveis. [3]

# Características Principais:

---

- **Isolamento Lógico:**
  - Cada Box opera de maneira isolada, o que significa que os dados dentro de um Box são independentes e não compartilham espaço ou estrutura com outros Boxes.
- **Configuração Personalizada:**
  - Os Boxes podem ser configurados com parâmetros específicos, como a definição de tipos de dados ou a inclusão de adaptadores personalizados para objetos complexos.
- **Persistência de Dados:**
  - Os dados armazenados em um Box são persistentes, garantindo que permaneçam disponíveis mesmo quando o aplicativo é fechado ou reiniciado.

EXECUTANDO

# Configuração Inicial

---

- Adicionar Dependência:
  - Inclua a dependência do Hive no seu arquivo *pubspec.yaml*.

dependencies:

hive: any

hive\_flutter: any

dev\_dependencies:

hive\_generator: any

build\_runner: any



# Manipulando boxes

---

Iniciando uma box:

```
await Hive.initFlutter();
```

Abrindo ou criando uma box:

```
var box = await Hive.openBox<E>('testBox');
```

Pegando uma box aberta:

```
var box = Hive.box('myBox');
```

Fechando uma caixa:

```
await box.close();
```

# Create

---

```
// Criar um novo Registro no Hive
ElevatedButton(
  onPressed: () {
    if (_controller.text.isNotEmpty) {
      // Box.add Adiciona o valor na Box
      box.add(_controller.text);

      setState(() {}); // Atualiza a UI
      _controller.clear();
    }
  },
  child: Text('Add Name'),
), // ElevatedButton
```

# Read

---

```
itemBuilder: (context, index) {  
  final name = box.getAt(index) ?? '';  
  return ListTile(  
    title: Text(name),
```

# Update

---

```
onPressed: () {  
  if (_controller.text.isNotEmpty) {  
    box.putAt(index, _controller.text);  
  }  
  Navigator.of(context).pop();  
}
```

# Delete

---

```
// Função para Deletar
void _deleteName(int index) {
    var box = Hive.box<String>('testBox');
    box.deleteAt(index);
    setState(() {}); // Atualiza a UI
}
```

# Outras funções

# HiveType Class

---

Objetos do Hive.

example/lib/main.dart

```
import 'dart:io';
import 'package:hive/hive.dart';
part 'main.g.dart';

@HiveType(typeId: 1)
class Person {
  Person({required this.name, required this.age, required this.friends});

  @HiveField(0)
  String name;

  @HiveField(1)
  int age;

  @HiveField(2)
  List<String> friends;

  @override
  String toString() {
    return '$name: $age';
  }
}

void main() async {
  var path = Directory.current.path;
  Hive
    ..init(path)
    ..registerAdapter(PersonAdapter());

  var box = await Hive.openBox('testBox');

  var person = Person(
    name: 'Dave',
    age: 22,
    friends: ['Linda', 'Marc', 'Anne'],
  );

  await box.put('dave', person);

  print(box.get('dave')); // Dave: 22
}
```

# Referências

---

- [1] hive | Dart Package (Acesso em <https://pub.dev/packages/hive>)
- [2] Quick Start - <https://docs.hivedb.dev/#/README>
- [3] Boxes - <https://docs.hivedb.dev/#/basics/boxes>
- [4] Samples - <https://github.com/hivedb/samples>
- [5] Projeto Teste - <https://github.com/Leoscan/flutter-hive>
- [6] HiveType class - <https://pub.dev/documentation/hive/latest/hive/HiveType-class.html>