



TRABALHANDO COM DADOS

INTRODUÇÃO AO
ARMAZENAMENTO DE DADOS
EM FLUTTER



TIPOS DE ARMAZENAMENTO

Cache

É usado para armazenar temporariamente dados que são frequentemente acessados e não precisam ser persistentes entre as sessões do aplicativo. O Flutter oferece suporte ao armazenamento em cache por meio de bibliotecas como o `shared_preferences`, permitindo que você armazene dados simples como pares de chave-valor.

Local Database

Para aplicativos que precisam de um armazenamento mais robusto e estruturado, é possível utilizar um banco de dados local. O Flutter possui bibliotecas populares, como o `sqflite` e o `moor`, que permitem a criação e a manipulação de bancos de dados SQLite no dispositivo.

TIPOS DE ARMAZENAMENTO

File Storage

Quando se trata de armazenar arquivos, como imagens, áudios ou documentos, o Flutter oferece facilidades para manipulação de sistemas de arquivos locais. A biblioteca `path_provider` é frequentemente utilizada para obter o caminho do diretório de armazenamento do dispositivo e a biblioteca `dart:io` oferece recursos para trabalhar com arquivos e diretórios.

Outros

Além dessas opções, também é possível integrar o Flutter com serviços de armazenamento em nuvem, como Firebase Storage, Amazon S3 ou Google Cloud Storage, para armazenar dados de forma remota e acessível em vários dispositivos.

COMO ESCOLHER

- Ao escolher o método de armazenamento de dados mais adequado para o seu aplicativo Flutter, é importante considerar fatores como a complexidade dos dados, a quantidade de dados a serem armazenados e a necessidade de persistência entre as sessões do aplicativo. Com a variedade de opções disponíveis, os desenvolvedores têm flexibilidade para escolher a solução que melhor se adapte às necessidades do projeto.



BANCO DE DADOS LOCAL

USO DO BANCO DE DADOS LOCAL
PARA ARMAZENAR DADOS



INTRODUÇÃO

- No desenvolvimento de aplicativos Flutter, é comum a necessidade de armazenar dados localmente.
- Uma das formas de fazer isso é utilizando um banco de dados local para armazenar e recuperar informações no dispositivo do usuário.
- Existem várias opções disponíveis para realizar essa tarefa em Flutter, sendo as mais populares o SQLite e o Moor.

SQLITE

- Um banco de dados relacional embutido, que oferece uma solução leve e eficiente para armazenamento local
- Amplamente utilizado em aplicativos móveis e possui uma implementação nativa em Flutter por meio do pacote "sqflite".
- É possível criar tabelas, executar consultas SQL e realizar operações de inserção, atualização e exclusão de dados

MOOR

- Um gerador de código que facilita a interação com bancos de dados SQLite em Flutter
- Ele permite definir as tabelas e consultas através de classes Dart, gerando o código necessário para acessar o banco de dados
- Com o Moor, você pode escrever consultas em uma sintaxe mais amigável, além de aproveitar recursos como verificação de tipos em tempo de compilação

CONCLUSÃO

- NO GERAL, O ARMAZENAMENTO DE DADOS EM FLUTTER É UMA PARTE ESSENCIAL DO PROCESSO DE DESENVOLVIMENTO DE APLICATIVOS, PERMITINDO QUE OS DESENVOLVEDORES CRIEM APLICATIVOS EFICIENTES, INTERATIVOS E COM RECURSOS RICOS, OFERECENDO AOS USUÁRIOS UMA EXPERIÊNCIA FLUIDA E PERSONALIZADA.
- O USO DE UM BANCO DE DADOS LOCAL EM FLUTTER PERMITE ARMAZENAR DADOS NO DISPOSITIVO DO USUÁRIO. ÚTIL QUANDO SE DESEJA OFERECER RECURSOS OFFLINE, ARMAZENAR PREFERÊNCIAS DO USUÁRIO OU GERENCIAR DADOS LOCAIS ESPECÍFICOS DO APLICATIVO.