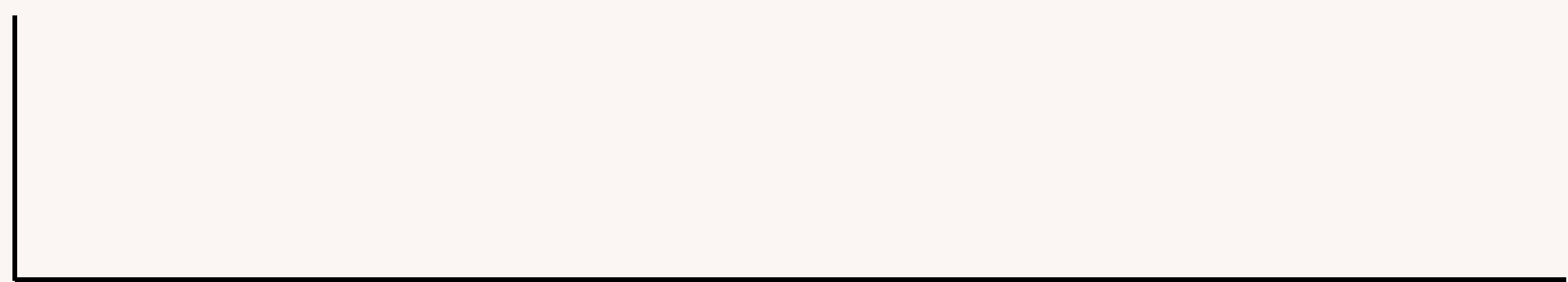


DART FLUTTER

Madu, Thauane.



```
List<String> Mensagens = [];  
  
void adicionarMensagem() {  
    String algumasmensagens = mensagensControlador.text;  
    mensagensControlador.clear();  
    setState(() {  
        Mensagens.add(algumasmensagens); // Adicionando valores  
    });  
}  
  
void limparMensagens() {  
    setState(() {  
        Mensagens.clear();  
    });  
}
```

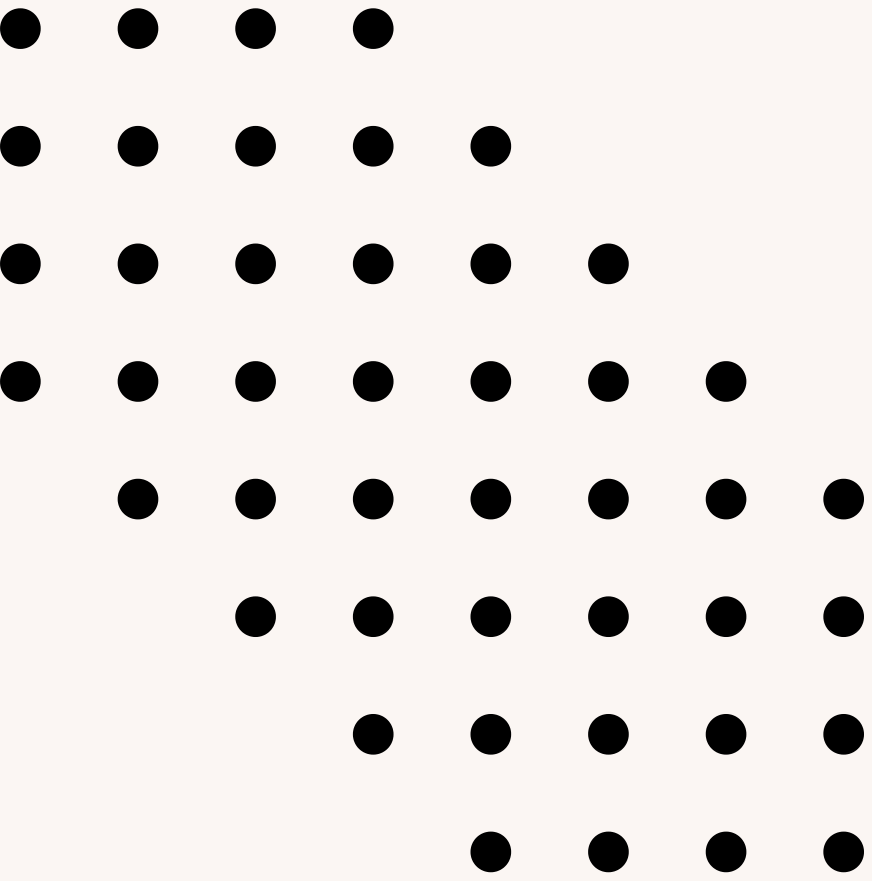
CÓDIGO PARA LIMPAR A LISTA DE TAREFAS

1. O método `limparMensagens()` é uma função que não recebe parâmetros e não retorna nenhum valor (void).
2. O método é chamado quando o usuário clica no botão "Limpar".
3. Dentro do método, temos o `setState(() { ... })`. O `setState()` é usado para notificar o Flutter que o estado do widget mudou, e isso desencadeia uma reconstrução do widget na tela. É importante usar `setState()` sempre que houver mudanças no estado do widget.
4. Dentro do `setState()`, chamamos o método `clear()` da lista Mensagens. O método `clear()` remove todos os elementos da lista, deixando-a vazia.
5. Quando o `setState()` é chamado, a lista é limpa e a interface do usuário é reconstruída, refletindo a lista vazia, ou seja, todas as tarefas são removidas da tela.
6. Dessa forma, quando o botão "Limpar" é pressionado, o método `limparMensagens()` é executado, e a lista de tarefas é limpa, resultando em uma tela sem tarefas pendentes.

CÓDIGO PARA AUMENTAR O NÚMERO DE TAREFAS

1. O método `adicionarMensagem()` é chamado quando o usuário clica no botão "Adicionar".
2. Primeiro, obtemos o texto digitado pelo usuário no `TextField` usando `mensagensControlador.text` e armazenamos em uma variável chamada `algumasmensagens`.
3. Em seguida, limpamos o `TextField` usando `mensagensControlador.clear()`. Isso é feito para limpar o campo de entrada após o usuário adicionar a tarefa.
4. Em seguida, chamamos `setState(() { ... })`. Assim como explicado anteriormente, `setState()` notifica o Flutter sobre a mudança de estado, o que resulta na reconstrução do widget na tela.
5. Dentro do `setState()`, adicionamos o valor da variável `algumasmensagens` à lista `Mensagens` usando o método `add()`. Isso aumenta o tamanho da lista em 1 e adiciona a nova tarefa digitada pelo usuário à lista.
6. Em resumo, ao clicar no botão "Adicionar" e chamar o método `adicionarMensagem()`, o texto digitado pelo usuário é adicionado à lista `Mensagens`, aumentando o número de tarefas na lista, e a interface do usuário é atualizada para exibir a nova tarefa adicionada na lista de tarefas pendentes.

```
Expanded(
  child: TextField(
    controller: mensagensControlador,
    decoration: InputDecoration(
      border: OutlineInputBorder(),
      labelText: "Adicione uma tarefa",
      hintText: "Digite aqui",
    ), // InputDecoration
  ), // TextField
), // Expanded
), // SizedBox
), // ElevatedButton(
  onPressed: adicionarMensagem,
```



fim.

