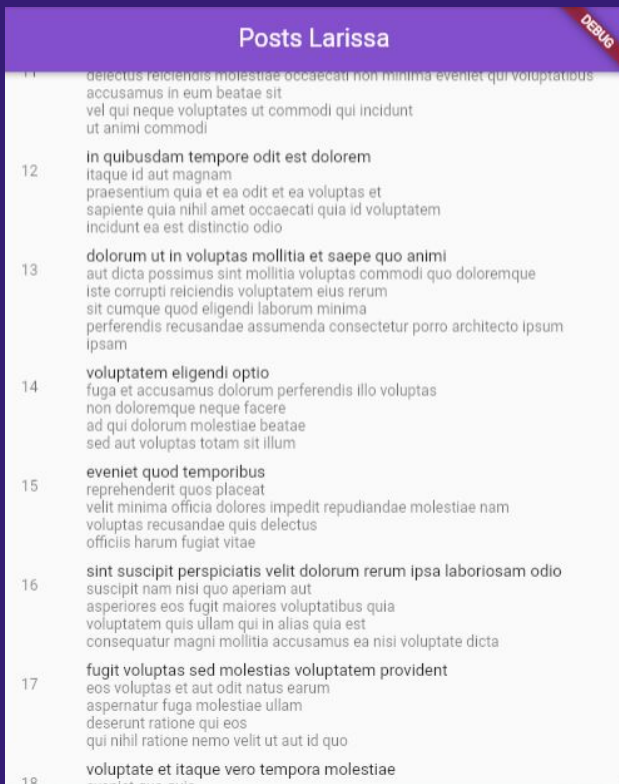


1) Faça o código dessa app (<https://bloclibrary.dev/#/flutterinfinitelistsutorial>) funcionar, print a execução para evidenciar o funcionamento, modifique o Título da aplicação de Posts para Posts (<nome-do-aluno>);



Avaliação 06
Flutter Infinity List
Aluna: Larissa dos Santos Holanda

- 2) Crie um questionário abordando os conceitos envolvidos na aplicação, como:
 - 2.1 Bloc como está implementado nessa app. Identificar os componentes da arquitetura Bloc dentro do código;
 - 2.2 A estrutura de pastas da app para atender questões como modularização, componentes e funcionalidades;
 - 2.3 Que dados são tratados pelo Gerenciamento de Estado.

- 1) Qual é o papel do SimpleBlocObserver neste código e como ele influencia o comportamento do Bloc?
- 2) O que o método main faz nesta aplicação Flutter e por que é necessário chamar runApp(const App())?
- 3) Qual é o propósito do método onTransition na classe SimpleBlocObserver?
- 4) Como o método onError na classe SimpleBlocObserver contribui para o tratamento de erros nos blocos?

Avaliação 06

Flutter Infinity List

Aluna: Larissa dos Santos Holanda

Qual é o papel do SimpleBlocObserver neste código e como ele influencia o comportamento do Bloc?

Resposta: O SimpleBlocObserver é utilizado para observar e imprimir mudanças de estado nos blocos da aplicação. Ao atribuir uma instância de SimpleBlocObserver a Bloc.observer, a aplicação registra todas as transições de estado e eventos, oferecendo uma maneira de monitorar e depurar o comportamento dos blocos.

O que o método main faz nesta aplicação Flutter e por que é necessário chamar runApp(const App())?

Resposta: O método main é o ponto de entrada da aplicação Flutter. Neste código, ele configura o observador de blocos usando Bloc.observer = const SimpleBlocObserver() para observar e imprimir mudanças de estado nos blocos. Em seguida, chama runApp(const App()) para iniciar a execução da aplicação, passando uma instância de App como o widget principal. Isso inicia a árvore de widgets e, por conseguinte, a aplicação Flutter, que será construída com base na lógica definida no widget App.

Qual é o propósito do método onTransition na classe SimpleBlocObserver?

Resposta: O método onTransition é parte da classe BlocObserver e é chamado sempre que ocorre uma transição de estado em qualquer bloco da aplicação. No código fornecido, ele é utilizado para imprimir informações sobre a transição, como o estado anterior, o evento que desencadeou a transição e o novo estado resultante. Isso é útil para rastrear e depurar o comportamento dos blocos durante a execução da aplicação.

Como o método onError na classe SimpleBlocObserver contribui para o tratamento de erros nos blocos?

Resposta: O método onError é chamado quando ocorre um erro durante a execução de um bloco. Neste código, ele imprime o erro no console usando print(error) e repassa a chamada para o método onError da superclasse BlocObserver. Isso permite que desenvolvedores identifiquem e compreendam erros que ocorrem nos blocos da aplicação, facilitando o processo de depuração e manutenção do código.