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

	Posts Larissa 🥀
1	sunt aut facere repellat provident occaecati excepturi optio reprehenderit quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto
2	qui est esse est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui neque nisi nulla
3	ea molestias quasi exercitationem repellat qui ipsa sit aut et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantium quis pariatur molestiae porro elus odio et labore et velit aut
4	eum et est occaecati ullam et saepe reiciendis voluptatem adipisci sit amet autem assumenda provident rerum culpa quis hic commodi nesciunt rem tenetur doloremque ipsam iure quis sunt voluptatem rerum illo velit
5	nesciunt quas odio repudiandae veniam quaerat sunt sed alias aut fugiat sit autem sed est voluptatem omnis possimus esse voluptatibus quis est aut tenetur dolor neque
6	dolorem eum magni eos aperiam quia ut aspernatur corporis harum nihil quis provident sequi mollitta nobis aliquid molestiae perspiciatis et ea nemo ab reprehenderit accusantium quas voluptate dolores velit et doloremque molestiae
7	magnam facilis autem dolore placeat quibusdam ea quo vitae magni quis enim qui quis quo nemo aut saepe quidem repellat excepturi ut quia sunt ut sequi eos ea sed quas

	Posts Larissa
	delectus reiciendis molestiae occaecati non minima eveniet qui voiuptat accusamus in eum beatae sit vel qui neque voluptates ut commodi qui incidunt ut animi commodi
12	in quibusdam tempore odit est dolorem itaque id aut magnam praesentium quia et ea odit et ea voluptas et sapiente quia nihil amet occaecati quia id voluptatem incidunt ea est distinctio odio
13	dolorum ut in voluptas mollitia et saepe quo animi aut dicta possimus sint mollitia voluptas commodi quo doloremque iste corrupti reiciendis voluptatem eius rerum sit cumque quod eligendi laborum minima perferendis recusandae assumenda consectetur porro architecto ipsum ipsam
14	voluptatem eligendi optio fuga et accusamus dolorum perferendis illo voluptas non doloremque neque facere ad qui dolorum molestiae beatae sed aut voluptas totam sit illum
15	eveniet quod temporibus reprehenderit quos placeat velit minima officia dolores impedit repudiandae molestiae nam voluptas recusandae quis delectus officiis harum fugiat vitae
16	sint suscipit perspiciatis velit dolorum rerum ipsa laboriosam odio suscipit nam nisi quo aperiam aut asperiores eos fugit maiores voluptatibus quia voluptatem quis ullam qui in alias quia est consequatur magni mollitia accusamus ea nisi voluptate dicta
17	fugit voluptas sed molestias voluptatem provident eos voluptas et aut odit natus earum aspernatur fuga molestiae ullam deserunt ratione qui eos qui nihil ratione nemo velit ut aut id quo
10	voluptate et itaque vero tempora molestiae

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?

Avaliação 06 Flutter Infinity List Aluna: Larissa dos Santos Holanda

- 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?

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 on Transition na classe Simple Bloc Observer?

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.