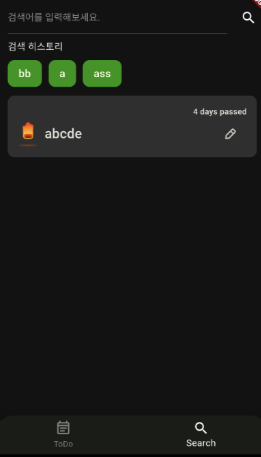
**파이널 프로젝트**

**프로젝트 명 : Todo앱 검색 화면 구현하기**  
  
**프로젝트 설명**  
Todo앱 로컬 저장 챕터 강의에서 완성하지 못했던 검색 탭 화면을 직접 구현해봅니다.  
※ 프로젝트 진행을 위해서는 선택강의 Ch 06. Todo 앱 - 로컬 저장 강의까지 꼭 수강하셔야 합니다.  
  
**프로젝트 과제 상세**  
1. 다음의 깃 주소에서 기본 소스를 클론합니다.  
 [**깃주소 링크(클릭)**](https://github.com/fastcampus-flutter/part3_todo)  
2. f\_search.dart 파일안에 SearchFragment를 StatefulWidget으로 바꿔봅니다.  
3. SearchFragment에 with 키워드를 사용하여 TodoDataProvider(Mixin)을 결합합니다.  
4. 검색 기능을 다음과 같은 위젯들을 이용하여 구현합니다.  
 a. 위젯들의 사용방법을 모를땐, 직접 코드 내부에 들어가서 탐험  
 b. 구글/유튜브 검색  
  ⅰ. TextField (텍스트 입력화면, hintText에 ‘검색을 해보세요' 텍스트 추가)  
  ⅱ. Tap 위젯 혹은 IconButton (TextField 우측에 돋보기 버튼)  
   https://storage.googleapis.com/lms-admin/local/exam/7304/question/20240402T073928_%ED%94%84%EB%A1%9C%EC%A0%9D%ED%8A%B8_%EC%B0%B8%EA%B3%A0%EC%9D%B4%EB%AF%B8%EC%A7%80_%EC%95%B1%EA%B0%9C%EB%B0%9C.png  
  ⅲ. 검색을 눌렀을때 TodoDbModel의 title값을 검색하여 해당 string을 contain하고 있는 todo 리스트를 출력  
  ⅳ. local\_db.dart 파일 안에 \_isar.todoDbModels.filter().titleContains 함수를 이용하는 searchTodo 함수를 만든다.  
   1) 함수 타입 Future<simpleresult<list, LocalDBError>> searchTodo(String query) async {})  
   2)todo\_repository.dart 파일 안에 Future<simpleresult<list, Error>> searchTodo(String query); 추가  
   3)todo\_api.dart 파일은 사용하지 않으므로 제거하셔도 됩니다.  
  ⅴ.todo\_data.dart 안에 final RxList searchList = [].obs;를 선언하고 위에서 만든 searchTodo 함수를 한번 wrapping하는 함수를 만듭니다.  
5. 윗 단계까지 구현을 마무리 했다면 Ttoss앱 주식 검색하기에서 배웠던 최근 히스토리 검색 리스트를 구현해보세요. (참고: 주식 검색화면 구조 만들기 - 8분20초 부터)  
 a. prefs.dart에 아랫줄을 추가하여 preference값을 활용하여 구현할 것  
  static final searchHistoryList = NullablePreferenceItem<list>('searchHistoryList');  
 b. 검색 히스토리 아이템을 클릭했을때 해당 텍스트의 검색 결과가 나오도록 구현  
    
6. 검색 탭에서 할일 상태를 변경했을때, Todo 리스트 화면에서도 동시에 갱신이 되도록 todo\_data.dart 내부 로직을 완성해보세요.