#10 Fetch Data from the Supabase Database on the Home Page to Show a List of Courses

Introduction

In this section, we will discuss how to fetch data from the Supabase database and display it on UI our Flutter app. Specifically, we will retrieve and display a list of courses on the home page of an educational application similar to educative.io.

Implementation - Step-by-step guide

Before we define the function to retrieve Courses data from Supabase, firstly we need to insert data into the Supabase database that we already create before. In this step, we will insert data only for the `course` table.

Step 1: Insert data into the `course` table

Please follow the link below to download the course data in a CSV file.

course_rows.csv

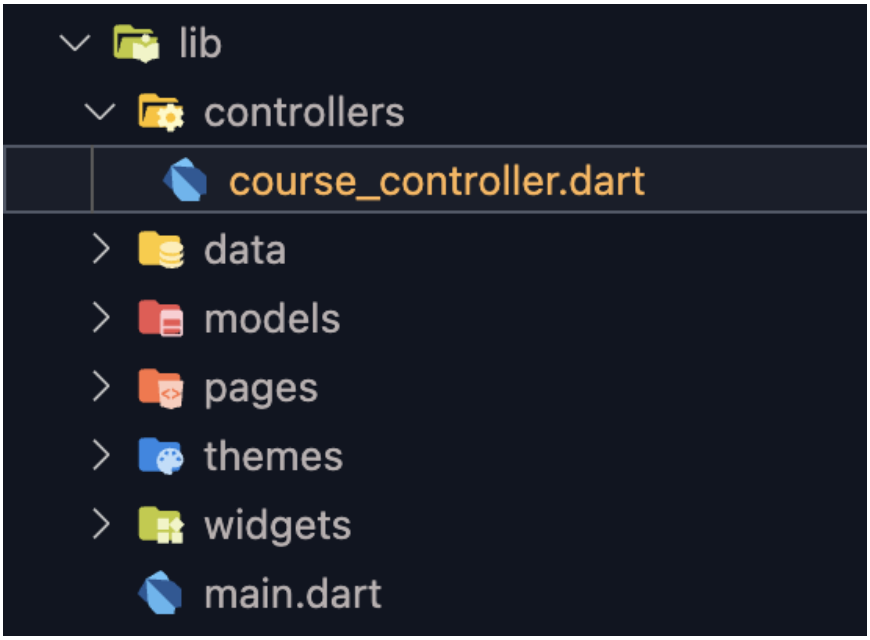
After downloading the file, we need to insert the course data into the `course` table in the Supabase Database. To see how to do this, please watch the video below on inserting data from a CSV file to the `course` table in the Supabase Database.

> https://www.loom.com/share/9426f1dfc60848deaea3267f46bf1143

Okay, next we will define a function that retrieves Courses data from the Database. To achieve this follow the following steps.

Step 2: Create a `course_controller.dart` file

Inside the `lib/controller` folder create a new file called `course_controller.dart`.



Step 3: Retrieve Courses data from the Database

Create a class called `CourseController` inside the `course_controller.dart` file that we created before the step. And inside the `CourseController` class, we define a function that handles retrieving Course data from the Supabase database.

Add the following code snippet below into the `course_controller.dart` file:

```
import 'package:supabase_flutter/supabase_flutter.dart';

import '../models/course.dart';

class CourseController {
  final supabase = Supabase.instance.client;

  Future<List<Course>> getCourses() async {
    final response = await supabase.from('course').select();

    final courses = response
        .map((course) => Course.fromJson(course))
        .toList()
        .cast<Course>();

    print("courses: ${courses.length}");

    return courses;
  }
}
```

The `CourseController` class has a single method called `getCourses()` that returns a `Future` that resolves to a list of `Course` objects retrieved from the `course` table in the database. The method uses the `supabase_flutter` package to interact with the Supabase client and performs a simple select query on the `course` table. The response is then mapped to a list of `Course` objects using the `fromJson()` method, which converts the JSON response to a Dart object. Finally, the method returns the list of courses.

After we define the function to retrieve data from the Supabase database, we next need to define a provider that will use in the UI to show the list of courses.

Here's the code snippet of the provider, add the following code snippet below also into the `course_controller.dart` file:

```
final courseProvider = FutureProvider.autoDispose<List<Course>>((ref) async {
  final courseController = CourseController();
  final courses = await courseController.getCourses();

  return courses;
});
```

The code snippet above defines a provider called `courseProvider` that uses the `FutureProvider` widget from the `flutter_riverpod` package. It retrieves a list of `Course`

objects by calling the `getCourses()` method of the `CourseController` class. The `FutureProvider` widget automatically handles the asynchronous nature of the `getCourses()` method and ensures that the list of courses is available to the UI when it is ready. This provider can be used in the UI to show the list of courses.

And don't forget to import the `FutureProvider` widget from the `flutter_riverpod` package, it should be imported at the top of the file along with the other necessary imports.

```
import 'package:flutter_riverpod/flutter_riverpod.dart';
```

The populate the `course_controller.dart` file will look something like this:

```
import 'package:flutter_riverpod/flutter_riverpod.dart';
import 'package:supabase_flutter/supabase_flutter.dart';

import '../models/course.dart';

final courseProvider = FutureProvider.autoDispose<List<Course>>((ref) async {
  final courseController = CourseController();
  final courses = await courseController.getCourses();

  return courses;
});

class CourseController {
  final supabase = Supabase.instance.client;

  Future<List<Course>> getCourses() async {
    final response = await supabase.from('course').select();

    final courses = response
        .map((course) => Course.fromJson(course))
        .toList()
        .cast<Course>();

    print("courses: ${courses.length}");

    return courses;
  }
}
```

Step 4: Display Courses data on the Home Page

Okay, this is the final step in this section. We will display the list of courses on the home page of our application. To achieve this, we must first define a variable that will listen for

changes to `courseProvider`.

Inside the `home_page.dart` file, add the following code snippet to the `build` method of the `HomePage` widget, specifically before returning the widget.

```
final courseState = ref.watch(courseProvider);
```

And don't forget to import `courseProvider` at the top of the file along with the other necessary imports.

```
import '../controllers/course_controller.dart';
```

Afterward, we can use the `courseState` variable that we defined earlier to build a widget. Update the code within the `body` property of the `build` method of the `HomePage` widget as shown in the following code snippet:

```
      body: courseState.when(
        data: (courses) {
          return ListView.separated(
            itemCount: courses.length,
            padding: const EdgeInsets.only(
              top: 20,
              left: 20,
              right: 20,
              bottom: 30,
            ),
            itemBuilder: (context, index) {
              return CourseCard(
                courses[index],
                onPressed: () {
                  Navigator.push(
                    context,
                    MaterialPageRoute(
                      builder: (context) => CourseDetailPage(
                        course: courses[index],
                      ),
                    ),
                  );
                },
              );
            },
            separatorBuilder: (context, index) {
              return const SizedBox(height: 10);
            },
```

```
          );
        },
        loading: () => const CourseListLoading(),
        error: (error, stack) => Center(
          child: Text(error.toString()),
        ),
      ),
```

The `courseState` variable is a `AsyncValue<List<Course>>` object containing the course list retrieved from the Supabase database. We use the `when()` method to handle the different states of the `courseState` object. If the state is `data`, we build a `ListView.separated` widget to display the list of courses. If the state is `loading`, we display a `CourseListLoading` widget. If the state is `error`, we display an error message.

For the `CourseListLoading` widget, we need to define that widget. Create a file inside the `lib/widgets` directory called `course_list_loading.dart`. And add the following code into `course_list_loading.dart` file:

```
import 'package:flutter/material.dart';
import 'package:shimmer/shimmer.dart';

class CourseListLoading extends StatelessWidget {
  const CourseListLoading({super.key});

  @override
  Widget build(BuildContext context) {
    return Shimmer.fromColors(
      baseColor: Colors.grey[300]!,
      highlightColor: Colors.grey[100]!,
      child: SingleChildScrollView(
        padding: const EdgeInsets.only(
          top: 20,
          left: 24,
          right: 24,
          bottom: 30,
        ),
        child: Column(
          children: [
            for (var i = 0; i < 2; i++)
              Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  // image placeholder
                  Container(
                    width: double.infinity,
                    height: 200.0,
                    margin: const EdgeInsets.symmetric(vertical: 8.0),
                    decoration: BoxDecoration(
```

```
                    color: Colors.white,
                    borderRadius: BorderRadius.circular(10),
                  ),
                ),
                for (var i = 0; i < 2; i++)
                  Container(
                    width: double.infinity,
                    height: 16.0,
                    margin: const EdgeInsets.symmetric(vertical: 8.0),
                    decoration: BoxDecoration(
                      color: Colors.white,
                      borderRadius: BorderRadius.circular(10),
                    ),
                  ),
                Container(
                  width: 100.0,
                  height: 16.0,
                  margin: const EdgeInsets.symmetric(vertical: 8.0),
                  decoration: BoxDecoration(
                    color: Colors.white,
                    borderRadius: BorderRadius.circular(10),
                  ),
                ),
                const SizedBox(height: 20),
              ],
            ),
          ],
        ),
      ),
    );
  }
}
```

The `CourseListLoading` widget is a custom widget that we've created with a shimmer effect. It is used to display a loading animation while the list of courses is being retrieved from the Supabase database. The widget is designed to mimic the appearance of the actual course cards, but with a shimmer effect that adds a subtle animation to the placeholders. This provides a smooth and visually appealing experience for the user while they wait for the data to load.

Inside the `pubspec.yml` file, make sure we have installed the `shimmer` package in the `dependencies` section.

```
dependencies:
  ...
  shimmer: ^2.0.0
  ...
```

And don’t forget to import the `CourseListLoading` widget at the top of the `home_page.dart` file along with the other necessary imports.

```
import '../widgets/course_list_loading.dart';
```

Great job! Here's the full code inside the `home_page.dart` file:

```
import 'package:flutter/material.dart';
import 'package:flutter_riverpod/flutter_riverpod.dart';
import 'package:flutter_svg/flutter_svg.dart';
import 'package:google_fonts/google_fonts.dart';

import '../controllers/course_controller.dart';
import '../themes/colors.dart';
import '../widgets/course_card.dart';
import '../widgets/course_list_loading.dart';
import 'course_detail_page.dart';

class HomePage extends ConsumerWidget {
  const HomePage({super.key});

  @override
  Widget build(BuildContext context, WidgetRef ref) {
    final courseState = ref.watch(courseProvider);

    return Scaffold(
      appBar: AppBar(
        backgroundColor: Colors.white,
        automaticallyImplyLeading: false,
        title: Row(
          children: [
            SvgPicture.asset(
              'assets/logo.svg',
              width: 24,
              color: MyColors.primary,
            ),
            const SizedBox(width: 5),
            Text(
              'educative',
              style: GoogleFonts.nunitoSans(
                color: MyColors.black,
                fontWeight: FontWeight.w500,
              ),
            )
          ],
        ),
        actions: [
          IconButton(
```

```
            icon: const Icon(Icons.search),
            color: Colors.grey,
            onPressed: () {},
          ),
          IconButton(
            padding: const EdgeInsets.only(right: 10),
            icon: const Icon(Icons.account_circle),
            color: Colors.grey,
            iconSize: 30,
            onPressed: () {},
          ),
        ],
      ),
      body: courseState.when(
        data: (courses) {
          return ListView.separated(
            itemCount: courses.length,
            padding: const EdgeInsets.only(
              top: 20,
              left: 20,
              right: 20,
              bottom: 30,
            ),
            itemBuilder: (context, index) {
              return CourseCard(
                courses[index],
                onPressed: () {
                  Navigator.push(
                    context,
                    MaterialPageRoute(
                      builder: (context) => CourseDetailPage(
                        course: courses[index],
                      ),
                    ),
                  );
                },
              );
            },
            separatorBuilder: (context, index) {
              return const SizedBox(height: 10);
            },
          );
        },
        loading: () => const CourseListLoading(),
        error: (error, stack) => Center(
          child: Text(error.toString()),
        ),
      ),
    );
  }
}
```

Testing the App

If you successfully follow step-by-step this guide, now let's test our app. Run the application on an emulator or device using the terminal command. The command `flutter run` will build the app and install it on your device. The result will resemble the video below:

https://www.loom.com/share/5fe4acf3a0c547778db081d0c0ac20ef?sid=8dba4dc4-6e74-4ea7-a0d1-86795a1c8ab7

Conclusion

In this section, we learned how to fetch data from the Supabase database and display it on the home page of our application. We created a `CourseController` class to interact with the Supabase client and defined a provider called `courseProvider` that retrieves a list of `Course` objects from the database. We then used the `courseProvider` in the UI to show the list of courses using a `ListView.separated` widget. Additionally, we created a custom widget called `CourseListLoading` to display a loading animation while the course list is being retrieved from the database. In the next section, we will learn about fetching data from the Supabase database and displaying the details of a selected course.