**Fetch Test: API Data Retriever**

**Overview**

Fetch Test is an Android native application developed in Java that retrieves data from a specified API and displays the contents sorted by list Id and name. This project utilizes GitHub for source control and backup.

**Features**

* Fetch data from a specified API endpoint.
* Display retrieved data in a list, sorted by list ID.
* Display retrieved data in a list, sorted by name.
* User-friendly interface with a simple and clean design.
* Error handling for network requests and data parsing.

**Prerequisites**

Before you begin, ensure you have met the following requirements:

* Android Studio installed on your development machine.
* Java Development Kit (JDK) version 8 or higher.
* Git installed on your development machine.

**Installation**

1. **Clone the Repository**

Clone the project repository from GitHub:

bash

Copy code

git clone https://github.com/Adperry32/FetchChallenge-.git

cd myapp

1. **Open the Project in Android Studio**
   * Launch Android Studio.
   * Select "Open an existing Android Studio project".
   * Navigate to the cloned repository folder and open it.
2. **Configure API Endpoint**

In the app/src/main/java/com/yourpackage/myapp/network/ApiClient.java file, configure your API endpoint:

java

Copy code

public class ApiClient {

private static final String BASE\_URL = "https://api.yourservice.com/";

// ... rest of the code

}

1. **Build the Project**
   * Click on "Build" in the top menu.
   * Select "Make Project" to build the project.

**Usage**

1. **Run the Application**
   * Connect your Android device or start an emulator.
   * Click on the "Run" button in Android Studio.
2. **Fetch and Display Data**
   * The app will fetch data from the configured API endpoint upon launch.
   * The data will be displayed in a list, sorted by user ID.

**Project Structure**

* app/src/main/java/com/yourpackage/myapp/ - Main source code directory.
  + activities/ - Contains the main activity for the app.
  + adapters/ - Contains the adapter for displaying data in a RecyclerView.
  + models/ - Contains the data model classes.
  + network/ - Contains classes for API communication.
  + utils/ - Contains utility classes and methods.

**Development**

**Adding New Features**

1. **Create a New Branch**

bash

Copy code

git checkout -b feature/your-feature-name

1. **Implement the Feature**

Add your code and make necessary changes.

1. **Commit and Push**

bash

Copy code

git add .

git commit -m "Add new feature: your feature name"

git push origin feature/your-feature-name

1. **Create a Pull Request**
   * Go to the GitHub repository.
   * Create a pull request to merge your feature branch into the main branch.

**Bug Fixes**

1. **Create a New Branch**

bash

Copy code

git checkout -b bugfix/your-bugfix-name

1. **Fix the Bug**

Make the necessary changes to fix the bug.

1. **Commit and Push**

bash

Copy code

git add .

git commit -m "Fix bug: your bugfix name"

git push origin bugfix/your-bugfix-name

1. **Create a Pull Request**
   * Go to the GitHub repository.
   * Create a pull request to merge your bugfix branch into the main branch.

**Contributing**

Contributions are always welcome! To contribute:

1. Fork the repository.
2. Create a new branch (git checkout -b feature/your-feature-name).
3. Make your changes.
4. Commit and push (git push origin feature/your-feature-name).
5. Create a pull request.

**License**

This project is licensed under the GPL-3.0 License - see the LICENSE file for details.

**Contact**

If you have any questions, feel free to reach out to the project maintainer:

* Your Name: Arrington.perry@outlook.com

Happy coding!