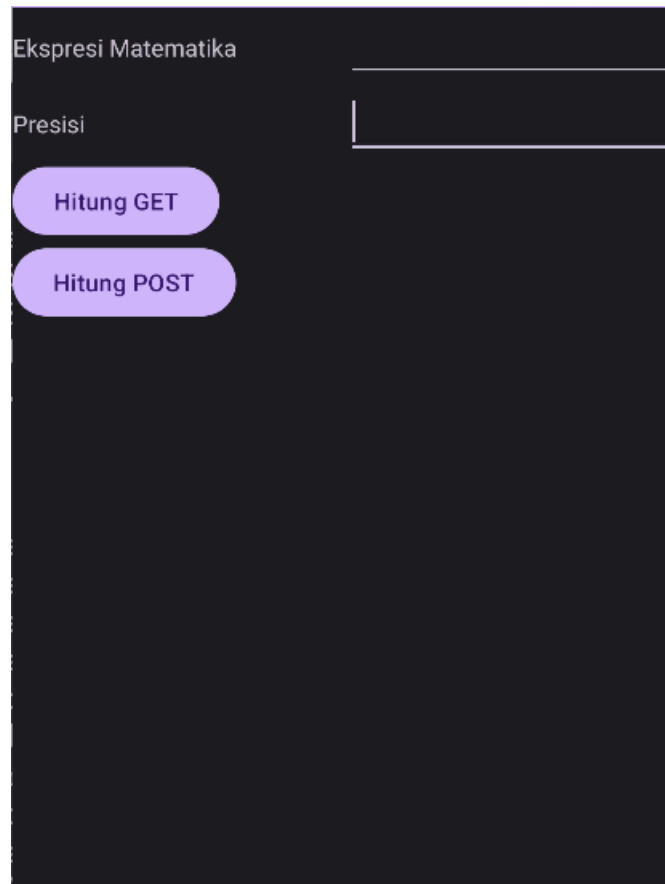


Webservice

Pada modul ini kita akan mempelajari cara membuat koneksi internet dengan menggunakan kelas `URLConnection`. Request ini digunakan untuk mengakses webservice <http://api.mathjs.org/>. User interface dari tugas ini dapat dilihat pada Gambar 1.



Gambar 1 Tampilan T10xyyy

Ketika tombol Hitung ditekan, maka aplikasi ini akan berhubungan dengan internet untuk mengakses web service tersebut untuk mengkalkulasi perhitungan matematika, kemudian menampilkan hasilnya di sebuah `TextView`.

Instruksi pengerjaan adalah sebagai berikut:

1. Pelajari dokumentasi API mathjs.org

Baca dokumentasi mathjs.org yang menjelaskan mengenai cara mengakses API tersebut dengan request GET. Hal-hal yang harus anda perhatikan adalah :

- URL yang perlu diakses
- Parameter-parameter yang perlu ada dalam objek JSON pada body dari request
- Isi dari objek JSON yang dikembalikan

Anda dapat mencoba webservice terlebih dahulu melalui browser.

2. Tampilan UI

Buatlah tampilan sesuai dengan Gambar 1 pada aplikasi anda.

3. Permission

Lakukan penambahan permission pada Android Manifest. Karena API ini menggunakan plain text, lakukan konfigurasi agar program dapat memproses keluaran API tersebut.

4. Akses API

Buatlah kelas baru untuk melakukan akses data pada API lalu menampilkan hasilnya pada textview. Pada bagian ini, gunakan API GET.

5. Buat agar pengguna dapat memasukkan beberapa ekspresi secara sekaligus dalam EditText yang anda gunakan untuk input. Ekspresi-ekspresi ini akan dipisahkan dengan menggunakan karakter ','. Keluaran dari ekspresi-ekspresi ini akan dituliskan dalam TextView yang anda gunakan untuk output dengan hasil dari tiap ekspresi dipisahkan oleh sebuah karakter ','.

Contoh masukan: $2+4,4*5/2$

Contoh keluaran: 6,10

Gradle

Gradle merupakan sebuah build tools yang memudahkan developer untuk menambahkan atau mengurangi third party library yang digunakan oleh suatu aplikasi. Pada android studio, gradle adalah build tools yang wajib digunakan dalam pembangunan aplikasi. Jika kita membutuhkan library tambahan tertentu, maka yang perlu dilakukan adalah menambahkan baris di file build.gradle lalu lakukan "Sync". Ketika melakukan sinkronisasi, gradle akan mencari library yang cocok dan memasukkan ke project kita.

Library yang dapat dimasukkan pun bermacam-macam dari yang hanya berguna sebagai kustomisasi UI (misal: FloatingActionButton), mempermudah pemrosesan fungsi (misal: GSON), maupun development tools yang memudahkan pembuatan perangkat lunak (misal: Dagger, Espresso, etc).

Untuk daftar library yang mumpuni dapat dilihat pada link berikut :

https://github.com/codepath/android_guides/wiki/Must-Have-libraries

Retrofit

Pada bagian ini kita akan mencoba menggunakan library Retrofit (<https://square.github.io/retrofit/>).

Pelajari lah dokumentasi yang tertulis dengan baik. Untuk pengolahan JSON kita akan menggunakan GSON. Pertama-tama, tambahkan library yang dibutuhkan pada build.gradle.

Untuk mencoba kedua library ini, kita akan mencoba kasus yang sama dengan bagian pertama (mathjs) tetapi pada kali ini kita akan menggunakan method POST.

1. Buatlah kelas Input sebagai kelas model yang mempunyai 2 atribut yaitu expr (String[]) dan precision (int)
2. Buatlah kelas Result sebagai kelas model yang mempunyai 2 atribut yaitu result (String[]) dan error (String)
3. Buat Interface dengan nama MathAPI yang berisi endpoint yang akan dipanggil retrofit.
4. Lakukan pemanggilan method POST dengan menggunakan Retrofit
5. Ujilah aplikasi yang dibuat.

Kumpulkan dengan nama T10xyyy.zip