**Домашнее задание к занятию «1.2. Сетевые запросы: Main Thread & Background»**

**Задача №1. Likes**

**Легенда**

В проекте, который мы рассматривали на лекции, не реализованы лайки:

class PostRepositoryImpl: PostRepository {

override fun likeById(id: Long) {

// TODO: do this in homework

}

}

Backend-разработчики наконец предоставили описание API для реализации:

1. Добавление лайка:

POST /api/posts/{id}/likes

2. Удаление лайка:

DELETE /api/posts/{id}/likes

Где {id} — это идентификатор поста.

В ответ на оба запроса сервер присылает JSON обновлённого поста, который можно использовать для отображения изменённого поста в ленте.

**Задача**

Реализуйте возможность ставить и снимать лайк в вашем проекте. Для этого используйте [код сервера с лекции](https://github.com/netology-code/andin-code/tree/master/02_threads/server).

Обратите внимание: после выполнения запроса список постов нужно обновить, чтобы отображалось актуальное количество лайков. Подумайте, как это правильно сделать.

**Задача №2. Swipe to Refresh\***

**Важно**: это необязательная задача. Она не влияет на получение зачёта.

**Легенда**

В списках в Android есть функция Swipe To Refresh:

1. Пользователь тянет сверху вниз список (или любое другое View).
2. Появляется иконка обновления.
3. Список обновляется.

Детали можно посмотреть на [странице документации](https://developer.android.com/training/swipe).

**Задача**

1. Добавьте [необходимую зависимость в build.gradle](https://developer.android.com/jetpack/androidx/releases/swiperefreshlayout).
2. Заверните свой RecyclerView в androidx.swiperefreshlayout.widget.SwipeRefreshLayout.
3. Реализуйте OnRefreshListener, который заново запрашивает все посты с сервера.