@Louis_CAD

Writing your own: wrapping callbacks

```kotlin
override fun onResponse(call: Call<T>, response: Response<T>) {
```

```kotlin
suspend fun <T> Call<T>.await(): T = suspendCancellableCoroutine { c ->
```

```kotlin
override fun onFailure(call: Call<T>, t: Throwable) {
```

`cancel()` *// Cancels network operation* 👌

```
enqueue(object : Callback<T> {
```

```
c.invokeOnCancellation {
```

```
c.resumeWithException(t)
```

```kotlin
override fun onFailure(call: Call<T>, t: Throwable) {
```