

Week 6 Homework

Kotlin Coroutines: Fundamentals (Beginning concurrency)

For this homework, you’ll be building up on the Week 5 assignment, to add threading & coroutines support to your database.

You have to remove the **allowMainThreadQueries()** call if you have it in your Room setup.You then have to use Kotlin Coroutines to wrap your database calls in Coroutines, to offload the main thread.

Using **launch()** and different **Dispatchers**, switch between the main and worker threads, and offload the main thread from the DB calls. Then post back on the main thread, to show the data.

# 

# Grading criteria

**Your app must have**:

* All the requirements from the previous [week’s assignment](https://docs.google.com/document/d/1CpuxckMjKrJdyOAJTczdIzxnMCkEHbsLSZ-89SU5OVQ/edit?pli=1).
* Removed **allowMainThreadQueries** call, to have Room work **only on background threads**.
* Kotlin Coroutines support (both core & Android library APIs)
* Implemented the [**LifecycleScope**](https://developer.android.com/topic/libraries/architecture/coroutines#viewmodelscope) from Android Jetpack Lifecycle components.
* Launch functions to start coroutines.
* Different dispatchers and switching from Main to BG or vice-versa.
* **IMPORTANT: Don’t use GlobalScope. Using GlobalScope will result in negative Homework grading! :]** (It’s a bad pattern, and should be avoided whenever/wherever possible)

**Extra credit ideas**:

* Custom CoroutineScope created for your calls. Check out the [documentation for examples of manual scopes](https://kotlin.github.io/kotlinx.coroutines/kotlinx-coroutines-core/kotlinx.coroutines/-coroutine-scope/).
* [Room with Kotlin Coroutines support](https://medium.com/androiddevelopers/room-coroutines-422b786dc4c5).
* Using the [Flow API](https://medium.com/androiddevelopers/room-flow-273acffe5b57) for reactive implementation of the Room Database.