2023-10-20 - L05

Review:

* L01 Intro = Orga
* L02 Android Tools & Basic Architecture
* Views and navigaton
* Activity + (implicit/explicit) intents (how to pass data)
* ...versus services, broadcast receiver, service provider
* How code interacts with UI (XML), R, findViewById
* Lifecylce (onResume, onDestroy)
* L03 Kotlin
* Model data + service
* Coding (in scratch files)
* L04 Kotlin Advanced
* Concurrency, deferable functions (suspend), Coroutines (-Scopes), call async:
* suspend fun,    launch { ... },    join ... wait ...  waitAll
* Idiomatic Kotlin
* immutable data, Null Safety, properties, data classes, object, scope functions (apply..), extension functions, operator overloading
* Kotlin Week (Selected Feedback)

L05

* **Infos:**
* Android Jetpack:
* \* called "Architectural components"
* \* DataBinding, Lifecycles, LiveData, Navigation, Paging, Room, ViewModel, WorkManager,...
* More Details on the (Jetpack) libraries
* \* ViewModel
* \* LiveData
* \* DataBinding
* \* Repository
* \* persistence, local-db: Room
* \* web-service: Retrofit
* \* Workmanager
* \* Load in Background, Don't block UI
* \* Navigation (Flow)
* \* Declarative UI (code)
* \* Jetpack Compose
* \* Dependency Injektion
* \* Hilt
* \* Dagger
* Architecture
* <https://developer.android.com/topic/architecture/>
* Videos
* <https://www.youtube.com/c/AndroidDevelopers/videos> ... check out channels
* MAD: <https://developer.android.com/series/mad-skills>
* or Jetpack Compose: <https://www.youtube.com/playlist?list=PLWz5rJ2EKKc9Ty3Zl1hvMVUsXfkn93NRk>
* Demos
* from git Part4 Android UI
* <https://git-iit.fh-joanneum.at/omd/droid-devel/-/tree/master/Part-4-Android-APIs>
* ... old style (activities, intents)
* ... fragments
* ... compose
* structure:
* data
* service (network, persistence)
* ui
* **Coding**
* Compose - Tutorial <<https://developer.android.com/jetpack/compose/tutorial>>