ما دیدم که چجوری میتونیم از کالکشن ها در کاتلین استفاده کنیم، حالا بیاین ببینیم چجوری میتونیم عمل های مختلف رو روشون انجام بدیم.

خب اینجا من یک سری کد از قبل درست کردم:

data class Album(val title: String, val year: Int, val chartUK: Int, val chartUS: Int, val tracks: List<Track> = *listOf*())  
  
data class Track(val title: String, val durationInSeconds: Int)  
  
val *albums* = *listOf*(  
 Album("The dark side of the moon", 1973,2,1,  
 *listOf*(Track("Speak To me",90),  
 Track("Breathe",163),  
 Track("On the run",216),  
 Track("Time",421),  
 Track("The great gig in the sky",276),  
 Track("Money",382),  
 Track("Us and them",462),  
 Track("Any color you like",205),  
 Track("Brain Damage",228),  
 Track("Eclipse",123)  
 )),  
 Album("The wall",1979,3,1),  
 Album("Wish you were here",1975,1,1),  
 Album("Animals",1977,2,3),  
 Album("The piper at the gates of dawn",1967,6,131),  
 Album("The final Cut",1983,1,6),  
 Album("The middle",1971,3,70),  
 Album("Atom heart mother",1970,1,55),  
 Album("Ummagumma",1969,5,74),  
 Album("More",1969,9,153))

همینطور که میبینین در انتها یک سری آلبوم داریم که به همراه تاریخ و رتبه چارت آمریکا و انگلیس اومده.

خب بیاید یکم با این کد بازی کنیم و در این بین هم من یک سری از توابعی که در دسترسه رو بهتون نشون میدم و البته که این ها همشون نیستن.

به عنوان مثال بیاین فرض کنیم که میخوایم آلبوم هایی که در انگلیس رتبه اول بودن رو پیدا کنیم.

من میدونم که یک سری آلبوم دارم و همینطور میدونم که هر آلبوم هم یک رتبه چارت انگلیس رو داخل خودش داره.

خب یک راه غیر Functional برای حل این مسئله اینه که یک حلقه درست کنیم و بین آرایه بگردیم:

fun main(args: Array<String>) {  
  
 for (album in *albums*)  
 if (album.chartUK == 1)  
 *println*("Found - ${album.title}")  
  
}

خب درواقع اینجا از یک for ساده استفاده کردیم. ولی بیاین یکم بهتر بنویسیم. خب ما این همه در مورد Functional Programming صحبت کردیم، بیایم یکم ازش استفاده کنیم.

*albums*.*forEach* **{** if (it.chartUK == 1) *println*("Found - ${it.title}") **}**

اولین چیزی که میتونیم ازش استفاده کنیم، همینطور که میبینین، بهره بردن از forEach هه ، درواقع اینجا همون عمل بالا رو داریم انجام میدیم و در انتها هم مثل هر تابع لاندای دیگه از it برای پاس کاری استفاده کردیم.

در واقع اگه کد forEach رو بررسی کنین متوجه میشین که تنها داره یک کالکشن رو دور میزنه و عمل رو انجام میده.

بیاین این کد دیگه رو نگا کنیم:

val albums1 = *albums*albums1.*filter* **{ it**.chartUK == 1 **}**albums1.*forEach* **{** *println*("Found - ${**it**.title}") **}**

در واقع اینجا اومدیم و از تابع filter استفاده کردیم و کاری که این تابع انجام میده اینه که یک آرایه رو دریافت میکنه و متناسب با عملی که دایم و روی اون آرایه ها انجام داده نتیجه رو برمیگردونه.

البته که من میتونم همین عبارت بالا رو زنجیرواری اینطوری بنویسم:

*albums*.*filter* **{** it.chartUK == 1 **}**.*forEach* **{** *println*("Found - ${**it**.title}") **}**

ادامه دارد...