Kotlin Basics

Kotlin Class Extensions

- Extends the functionality of an existing class

```
fun Int.addFive() : Int {
    return this + 5
}
```

- Does not actaully change the code of the class
- Provides a function that can be called on instances of the class

When are Class Extensions Useful in Ktor?

- To seperate business specific routes and logic from the rest of the routes

Coroutine Contexts

- DEFAULT: Number of threads = number of CPI cores - use this for calculations or if you are uncertain about which context to use
- IO: Number of threads = 64 or number of cores (whichever is larger) - use this for rest communication or storing data to a file or database
- MAIN: Number of threads = 1 - is mainly used in android apps to interact with user interface

Sample Code:

```
import kotlinx.coroutines.*
import kotlin.random.Random

fun main(args: Array<String>) = runBlocking {
    // 64 Threads in IO
    withContext(Dispatchers.IO) {
        repeat (100_000) { // 100_000 = 100,000
            launch {
                firstcoroutine(it) // 'it' will be the current iteration
            }
        }
        println("End of withContext")
    }
    println("End of main function")
}

suspend fun firstcoroutine(id: Int) {
    delay(Random.nextLong()%2000) // The delay is a random number less than 2 seconds
    println("first $id")
}
```

Running this code gives an output something like:

```
first 0
first 1
first 2
first 5
first 6
first 7
first 8
first 10
first 9
first 11
first 13
...
```

Notice how the sequence falls out of order? This is threading and Kotlin coroutines in action.

Revision #1
Created 17 April 2022 00:20:30 by Elkip
Updated 17 April 2022 01:02:09 by Elkip