

GetJson

Advanced Programming - Project 2024/2025

The goal is to develop a small framework in Kotlin addressing HTTP/GET endpoints that return JSON, similar to Spring Boot and the like. The framework will use the developed JSON library to automatically convert Kotlin values to JSON.

The framework should be launched by enumerating a list of REST controllers that define endpoints for HTTP/GET requests. Controllers define URL mappings through annotations, allowing programmers to map path segments and arguments into function arguments.

Attention: No external libraries/frameworks are allowed in the development, except for JUnit and a library for performing HTTP requests in the tests (for example, [OkHttp](#)).

The following code is a hypothetical example of the intended API. It can be developed in another style, as long as it allows similar flexibility.

```
fun main() {
    val app = GetJson(Controller::class, ...)
    app.start(8080)
}

@Mapping("api")
class Controller {
    @Mapping("ints")
    fun demo(): List<Int> = listOf(1, 2, 3)

    @Mapping("pair")
    fun obj(): Pair<String, String> = Pair("um", "dois")

    @Mapping("path/{pathvar}")
    fun path(
        @Path pathvar: String
    ): Pair<String, String> = pathvar + "!"

    @Mapping("args")
    fun args(
        @Param n: Int,
        @Param text: String
    ): Map<String, String> = mapOf(text to text.repeat(n))
}
```

In this example, the following endpoints would be available.

Path	Response
------	----------

/api/ints	[1, 2, 3]
/api/pair	{"first": "um", "second": "dois"}
/api/path/a	"a!"
/api/path/b	"b!"
/api/args?n=3&text=PA	{"PA": "PAPAPA"}