**Sy̲n̲c̲h̲r̲o̲n̲o̲u̲s̲ request**, you make the request and stop executing your program until you get a response from the HTTP server (or an error if the server can't be reached, or a timeout if the sever is taking way, way too long to reply) The interpreter is *blocked* until the request is completed (until you got a definitive answer of what happened with the request: did it go well? was there an error? a timeout?... ).

**As̲y̲n̲c̲h̲r̲o̲n̲o̲u̲s̲ requests**, you *"launch"* the request, and you kind of *"forget about it"*, meaning: The interpreter continues executing the code after the request is made **without waiting** for the request to be completed.

This seems... rather pointless, right? You send the request *"to the void of space"*, and continue executing as usual? What happens when the server sends you its response? I made a request, and I wanna know what happened to it! Otherwise, I wouldn't have typed that in my code to begin with!!

Well, here's where the callback comes in. You launch the request *"to the void of space"* **BUT** you provide a callback function so when the HTTP server on the other end sends you its response, that function is run with said response as the first argument.