Day-4

How javascript code executes in browser/node.js environment ?

javascript language :

- by default, all IO operations (read /write) are non-blocking/asynchronous
- by default, single-threaded process / light-weight process,
  but can execute concurrent actions due to non-blocking IO

Concurrency model and the event loop

JavaScript has a concurrency model based on an event loop, which is responsible for executing the code, collecting and processing events, and executing queued sub-tasks. This

https://developer.mozilla.org/en-US/docs/Web/JavaScript/EventLoop

http://latentflip.com/loupe/

Browser internal arch :

e.g chrome browser

1. javascript-engine/runtime (v8) ⟹ ECMAscript spec
    - single-call-stack (fixed size) ⇒ function execution
    - heap / free-memory (extendible memory) ⇒ to hold objects
2. browser apis (e.g DOM,Timer,XHR,...) ⟹ w3c recommendations
3. event/message/callback queue
4. event-loop-thread
