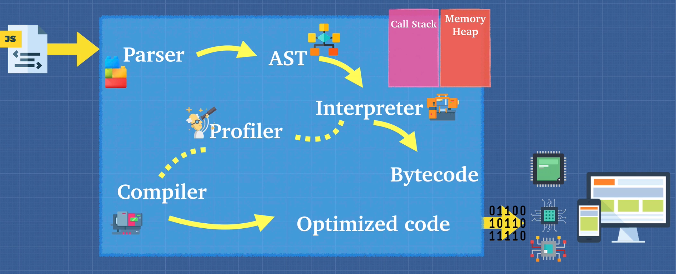
JAVASCRIPT ENGINE :

Js File - > Javascript Engine -> convert into machine code

//list of engines

<https://en.wikipedia.org/wiki/List_of_ECMAScript_engines>

Inside JS Engine :

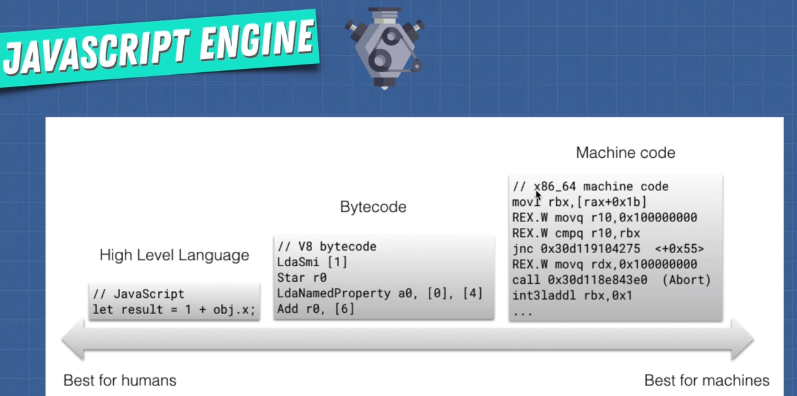


<https://astexplorer.net/>

2 ways -> intreperter & compiler

Interpretr -> translate & read files line by line

Compiler ->one pass through the code & understand what it does, writes a new program ( low level machine code )



Have you heard of Babel or TypeScript? They are heavily used in the Javascript ecosystem and you should now have a good idea of what they are:  
  
[Babel](https://babeljs.io/) is a Javascript compiler that takes your modern JS code and returns  browser compatible JS (older JS code).  
[Typescript](https://www.typescriptlang.org/) is a superset of Javascript that compiles down to Javascript.  
  
Both of these do exactly what compilers do: Take one language and convert into a different one!

Interpreter -> very fast

Cons : running same code again becomes slower

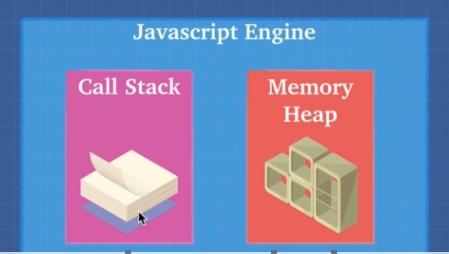
Compiler -> will helps here

* It will optimes the code

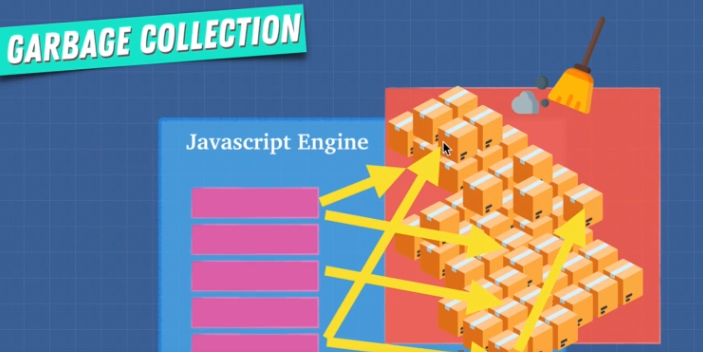
Combine Compiler + interpreter ( JIT Compiler )

Just In time compiler

Profiler : watches the code & thinks how can we optimise the code







JS is single Threaded -> only 1 callstack , never running functions in parallel (Synchronous )

Ex : alert functions

JS Runtime ( makes it asynchronous )

