

* Shortest JS program

↳ The shortest JS program is the "Empty file".

↳ If the file is empty then also

— JS engine will still do its work.

* In case of chrome JS Engine is V8.

↳ What happens when we run an Empty file?

* A GEC is created & JS Engine also setup the memory space.

* JS Engine also creates something known as Window.

→ These functions & variables (which is inside window object) are created by JS Engine. & into the Global space.

It's like a big object with lot of functions & Variables

→ We can access all these variables and functions anywhere in the JS program.

→ Just like window JS Engine also creates 'this' keyword.

→ At the global level 'this' $\xrightarrow{\text{points to}}$ 'window' object.

Que:- What is window?

Ans:- Window is actually a global object which is created along with the GEC.

So, whenever any JS program runs a $\left. \begin{array}{l} \text{Global object} \\ \text{GEC} \end{array} \right\} \text{ is created.}$

& along with that execution context a 'this' variable is created.

* Global Object Created: —

→ The "Global Object" in case of browser is known as window.

→ Where ever JS is running there must be a JS Engine there.

→ All the JS Engine has a responsibility to create this Global Object.

→ In case of browsers it is known as "Window". In case of Node it is known as something else.

→ But always there is a Global Object created.

↳ An Even though our file is empty JS Engine will create this global object.

* At this Global level / Base level in the GEC.

this === window.

Note:- Whenever we create an execution context a "this" is created along with it, even for the functional context and GEC. { and at global level "this" is pointing to "Global Object" (i.e. window) }

Que:- What is Global Space?

→ Any code written inside JS which is not inside any function is global space.

e.g. `var a = 10;` → This is inside global space.
`function b() {`

`var x = 10;` → This x is not in global space
`}`

* Whenever we create any variable or function in global space, what happens?

→ These variable & functions will get attached to the "Global Object" (i.e. Window)

Example :-

```
var a = 10  
function b(){  
  var x = 5;  
}
```

console.log(x);

→ How to access the value of 'a'

```
console.log(a);  
console.log(window.a);  
console.log(window this.a);
```

These 3 are
Same.
Their o/p
will be: -
10

→ This will give ~~us~~ ^{us} an **Error** because
in global object (i.e. Windows) there is
not any variable 'x'. as we have not
declared 'x' inside global space.