There are two phases:

1. Memory Allocation.
2. Execution.
3. Memory Allocation: Whenever we run any program in Browser it is already take Memory.

In this phase Hoisting occurs

|  |  |
| --- | --- |
| Graphical user interface, text, application  Description automatically generated | Graphical user interface, text  Description automatically generated with medium confidence |
|  | Chart  Description automatically generated with low confidence  CallStack contains Global Execution Content then sayHello Execution context |

Fig1(a) Debugger stop at line number 5Text

Description automatically generated

1. Fig2: (Break stop at line 8)

A picture containing table

Description automatically generated



1. Fig2: This anonymous is **Global Execution Context**. Since memory is allocated.
2. Graphical user interface, text

   Description automatically generated with medium confidence



1. Fig3: variable ‘a’ and sayHello take the memory in Global.

Hoisting case:

When let keyword is used, in console we will see an reference error because variable ‘a’ will be present under script object, which cannot be accessed without being initialised. It is called temporal Dead zone.

So does variable with let keyword hoist?

Ans: Yes, it hoists.

Text

Description automatically generatedFig: Outputs as reference error

