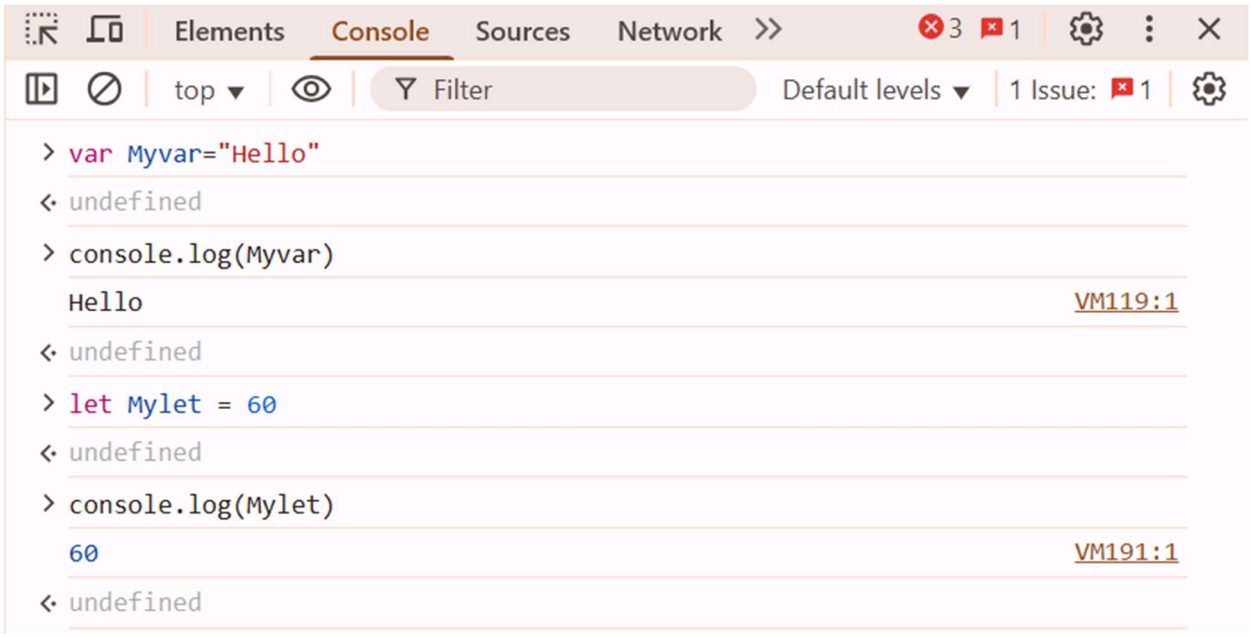
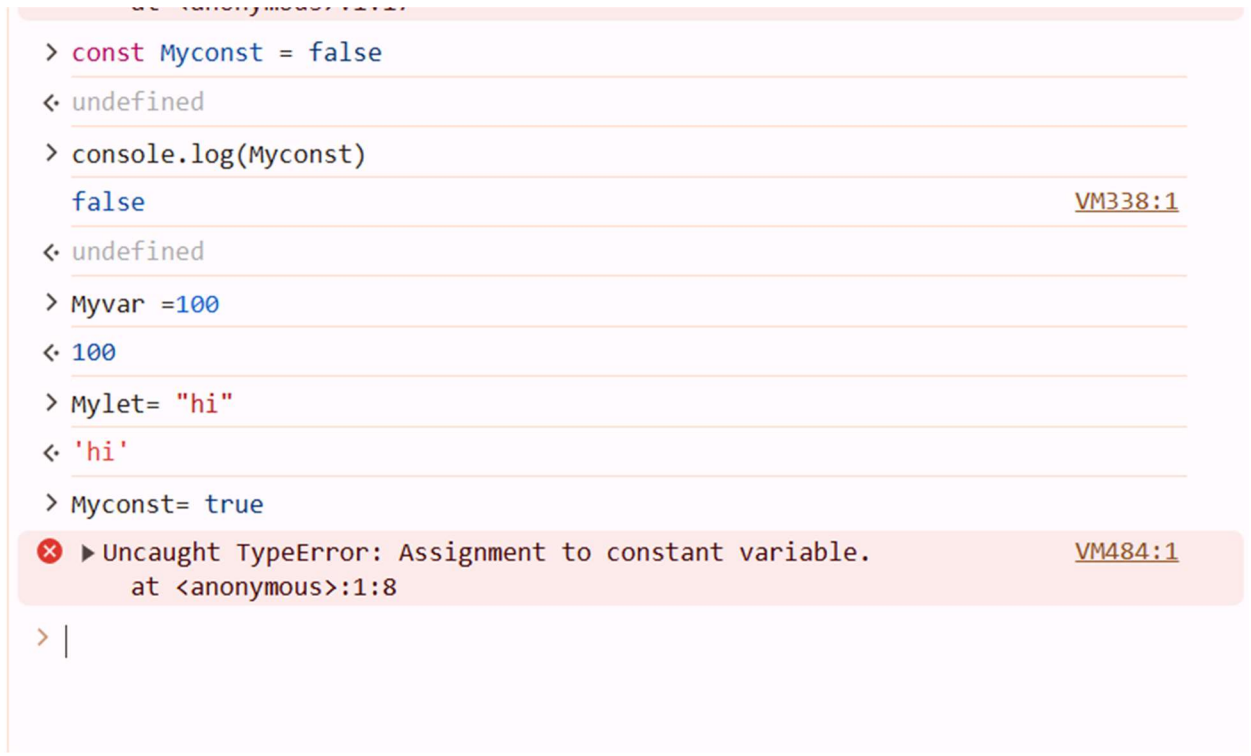


1)



It is possible to reassign and even alter the kinds of variables called var and let.

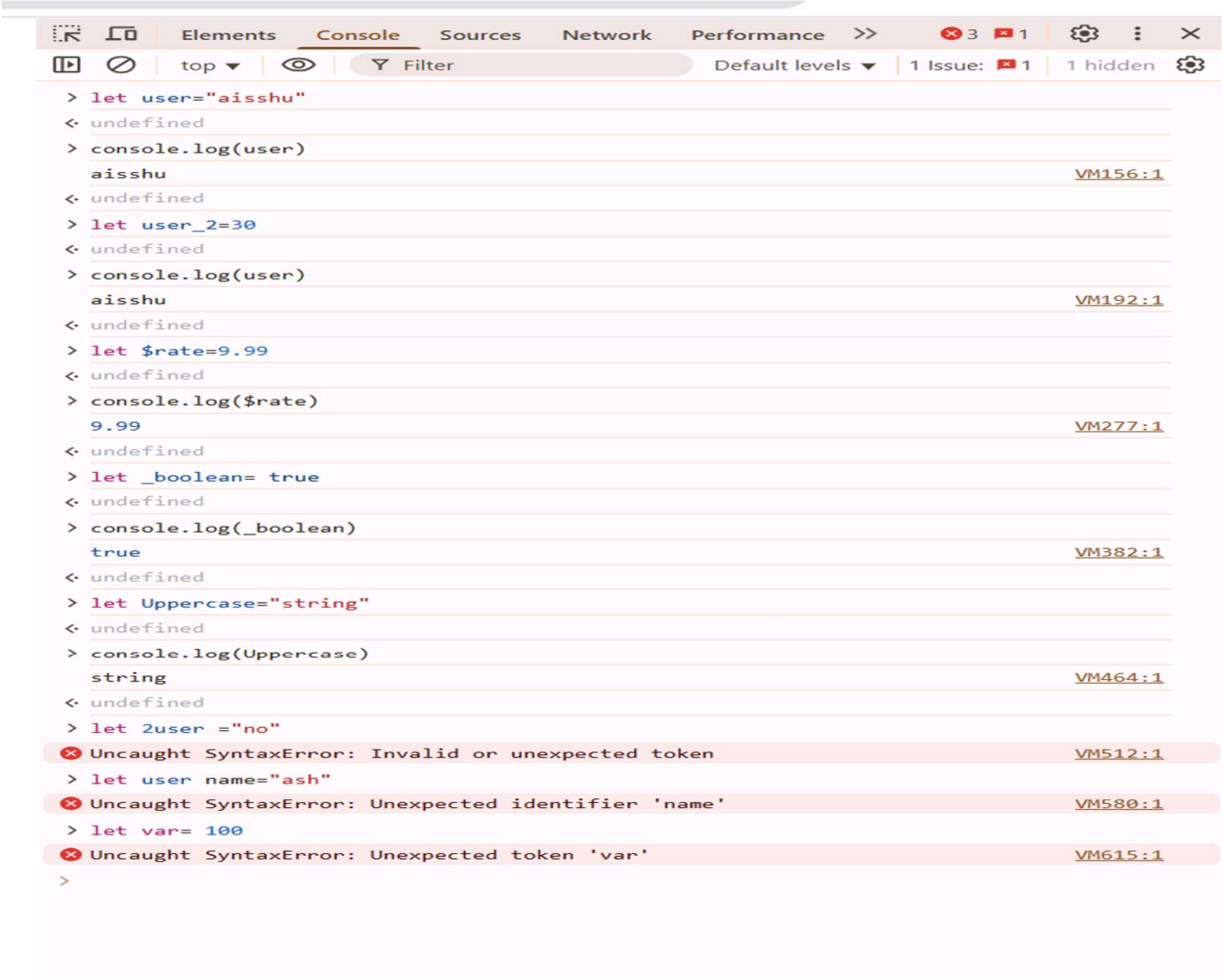
Once const is initialized, it cannot be reassigned. It must not change. Refer above screenshots

2) Let 2user = "no" Variable names cannot begin with a number.

let user name = "ash"→ Variable names cannot contain spaces.

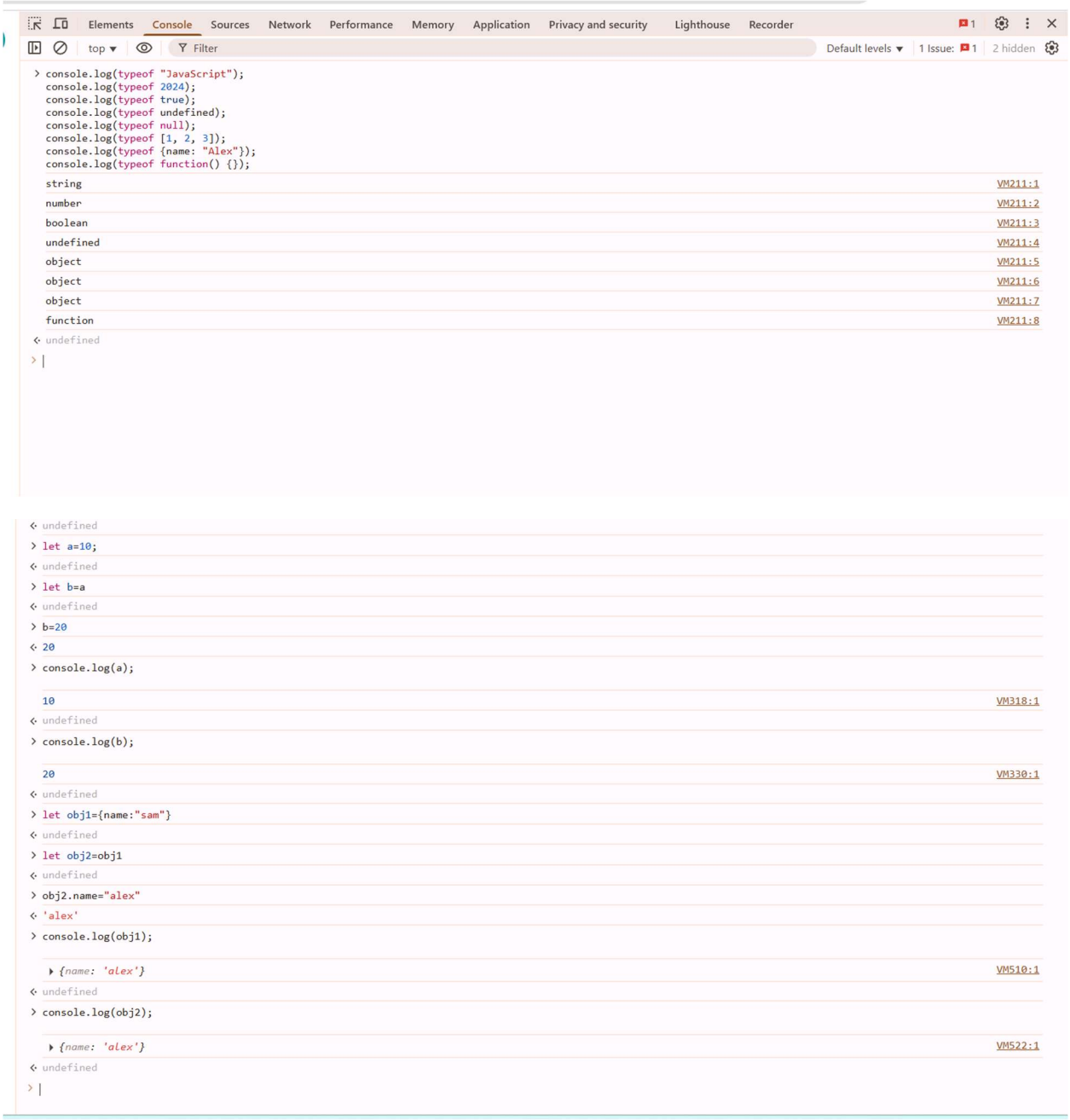
let var = 100 → var is not a valid variable name because it is a reserved keyword.

all others follow to correct JavaScript naming rules. Refer below screenshot



3) Typeof null results in "object".

Arrays and functions are technically objects, but typeof gives "function" for functions.
Below screenshot



4) Primitive types are copied by value, therefore changing b has no direct effect on a. Obj1 and Obj2 both refer to the same memory location (i.e..copied by reference). Refer above screenshot

5) JavaScript is dynamically typed — variable types can change during runtime. Below screenshot

5- II)above screenshots

a)Since 3 is forced into a string, "5" + "3" concatenates the strings. Thus, "53" type string(not addition) is the outcome.

b) It forces the string "5" to become a number. Numerical subtraction is done by JavaScript: 5 - 2 = 3.

c) The boolean true is coerced to 1, so the expression becomes 1 + 1, resulting in 2.

d) JavaScript's attempt to translate "hello" to a number is unsuccessful. The result of "hello" * 2 is NaN (Not a Number), yet its type is still number because "hello" is not a numeric.

e) null is coerced to 0 in numeric contexts. So, 0 + 5 results in 5. Number type