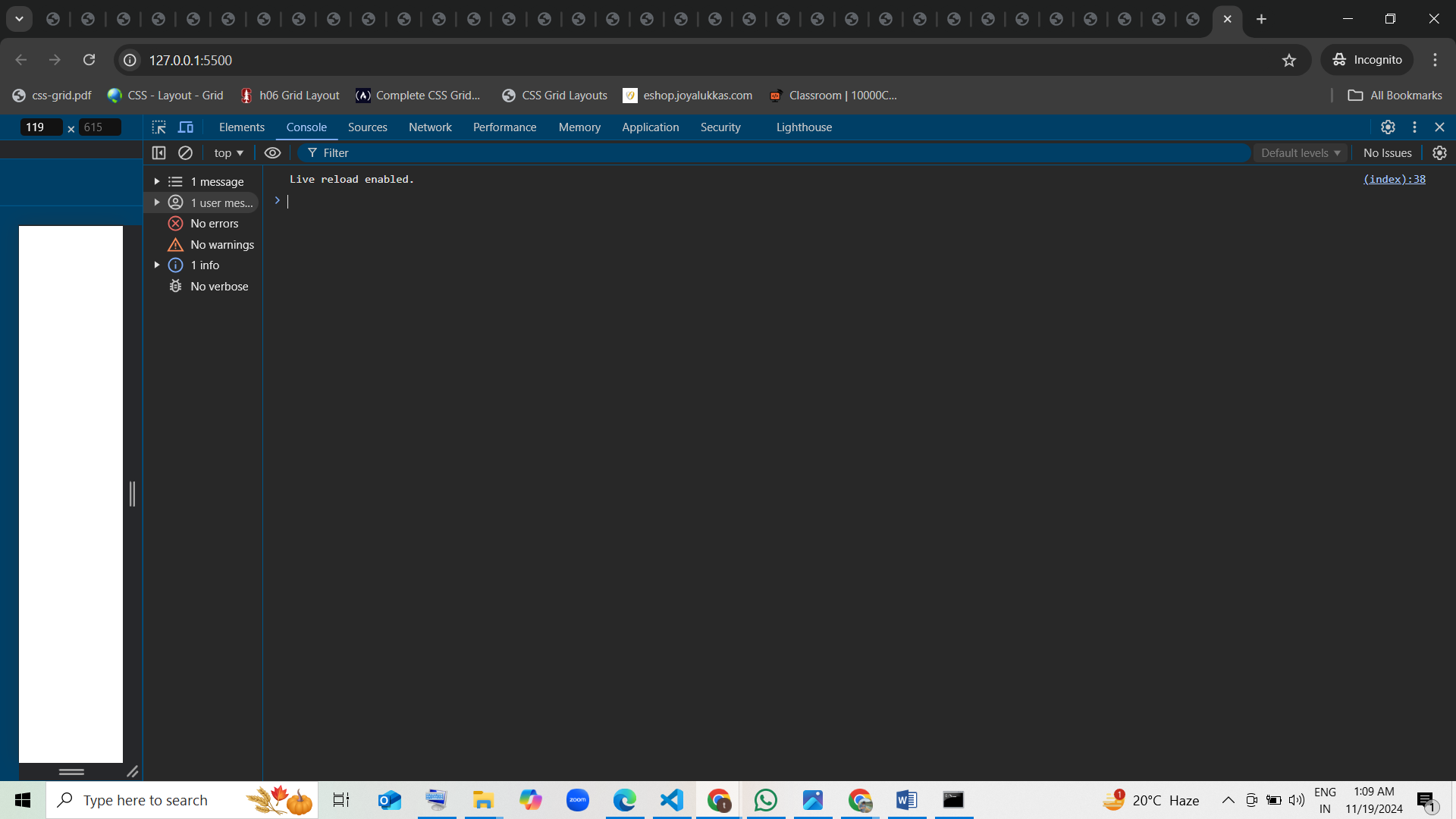
1. What is the value of x after the code executes?

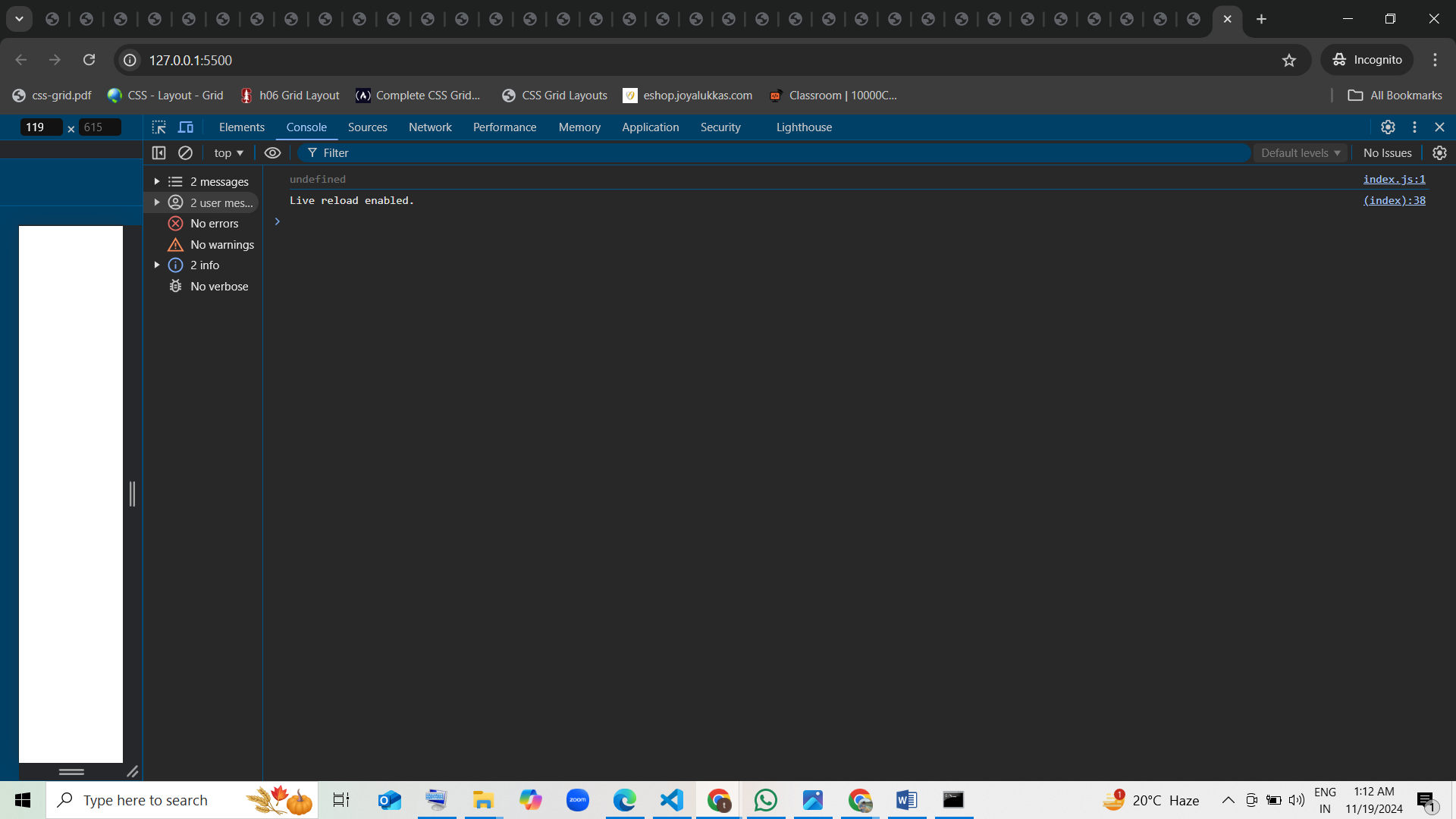
* var x;  
  x = 5;  
  var x = 10;
  + 1. 5
    2. 10 [guess]
    3. Undefined
    4. Error

A: empty



1. What is the output of the following code?

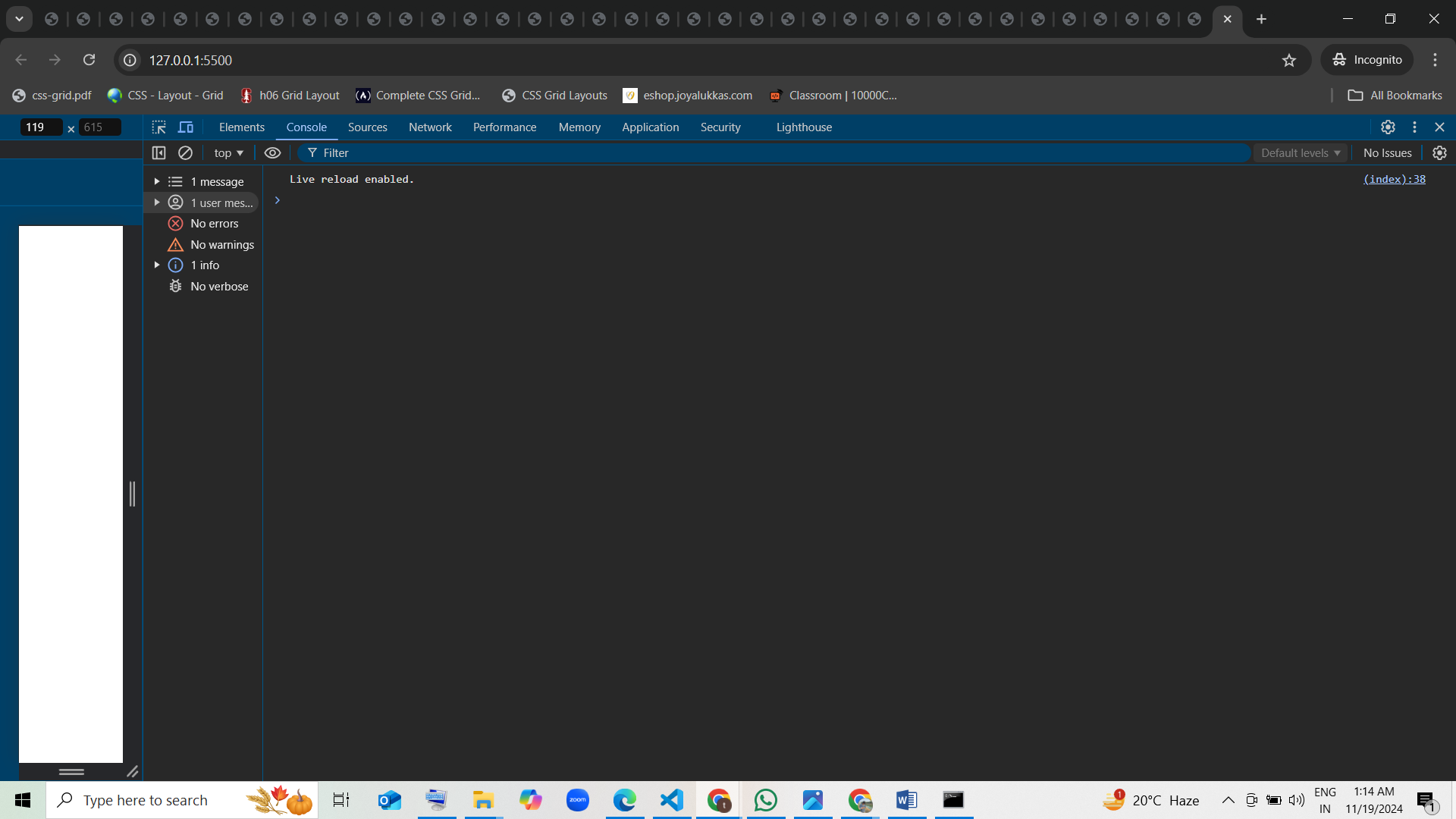
* console.log(x);  
  var x = 15;
  + 1. 15 [guess]
    2. Undefined [correct]
    3. ReferenceError
    4. Null



1. How many times is the variable y redeclared in this code?

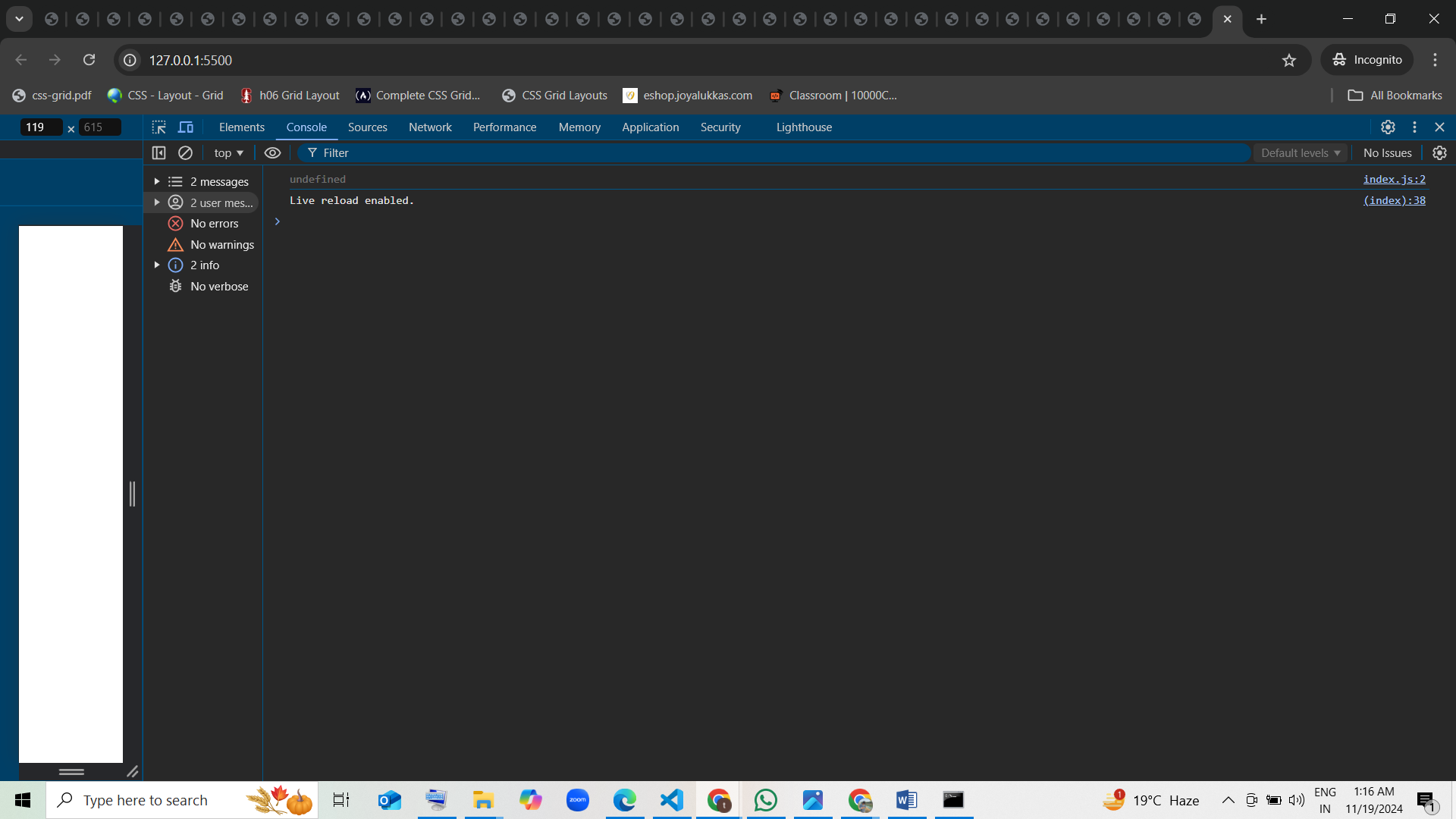
* var y = 1;   
  var y = 2;   
  y = 3;
  + 1. 1
    2. 2 [guess]
    3. 3
    4. 0

A: empty



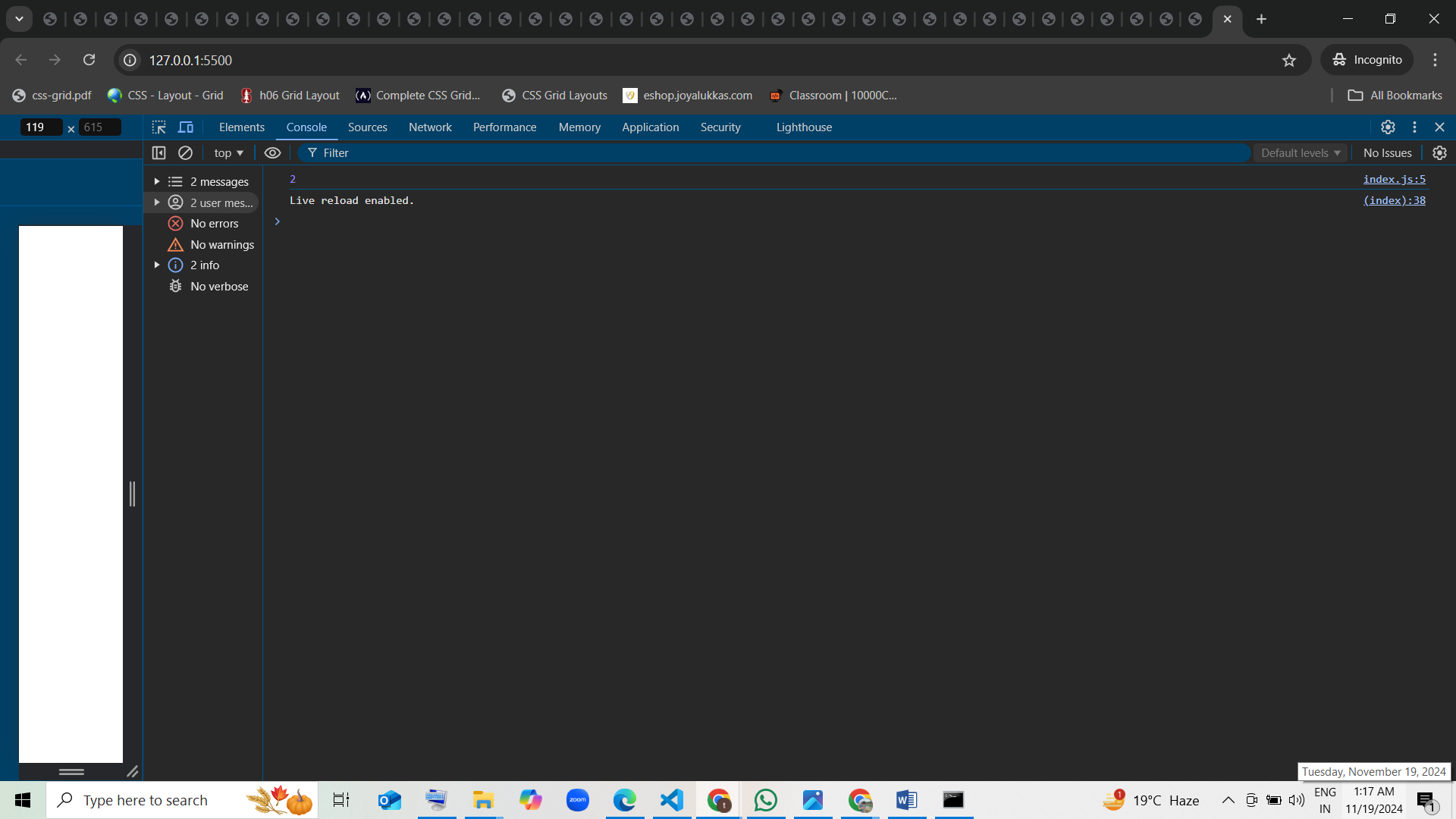
1. What happens when you declare a var variable without initializing it?

* var b;  
  console.log(b);
  + 1. Null
    2. Undefined [correct]
    3. Error [guess]
    4. 0



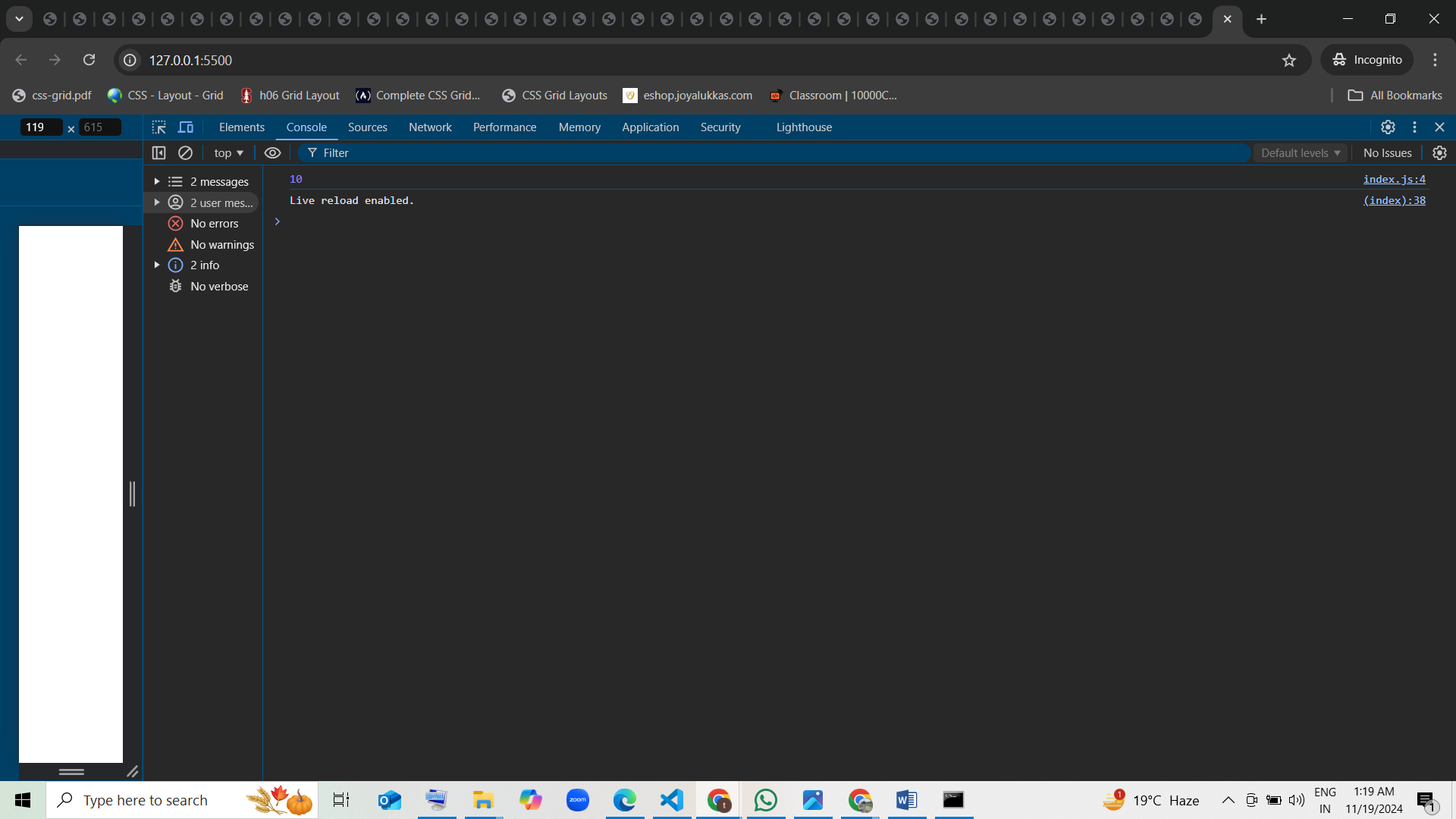
1. What will the following code print?

* var a = 1;  
  {  
   var a = 2;  
  }  
  console.log(a);
  + 1. 1
    2. 2 [guess] [correct]
    3. Undefined
    4. Error



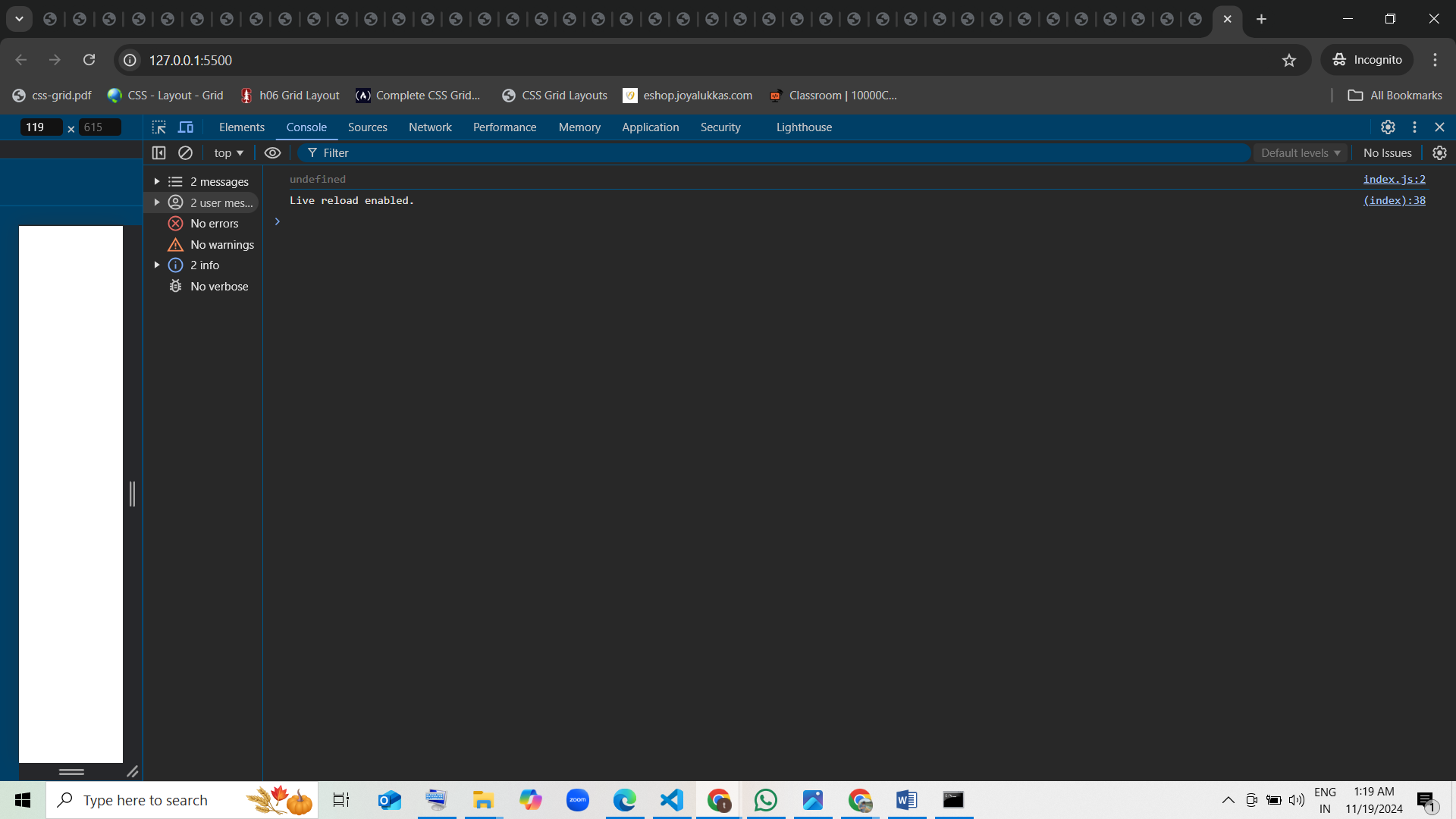
1. What happens when you declare a var variable inside a block and access it outside?

* {  
   var c = 10;  
  }  
  console.log(c);
  + 1. Error
    2. Undefined
    3. 10 [guess] [correct]
    4. Null



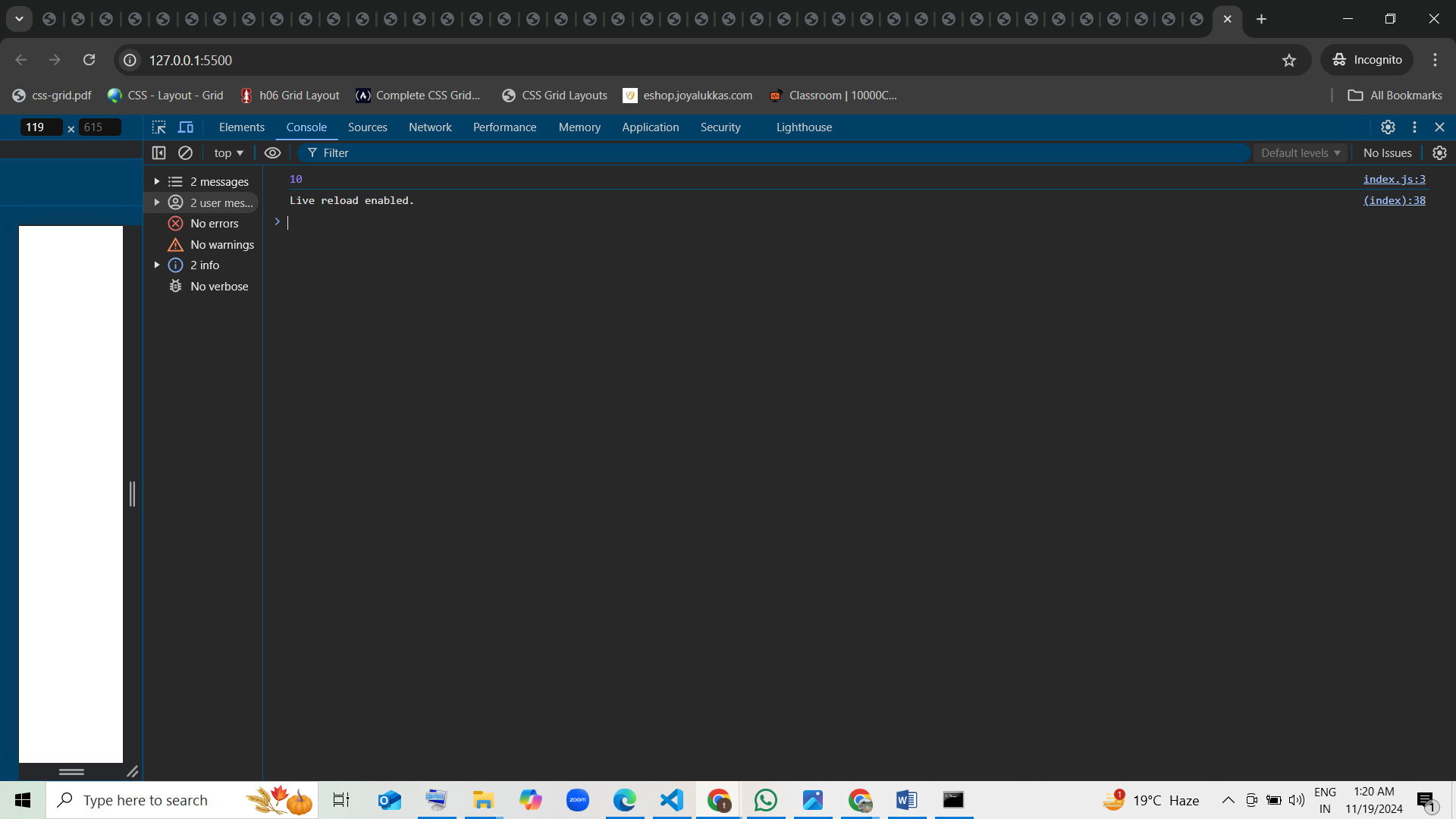
1. What is the output of the following code?

* var p;  
  console.log(p);  
  p = 20;
  + 1. Undefined [correct]
    2. 20 [guess]
    3. Null
    4. Error



1. What happens if you declare a var variable and later redeclare it in the same scope?

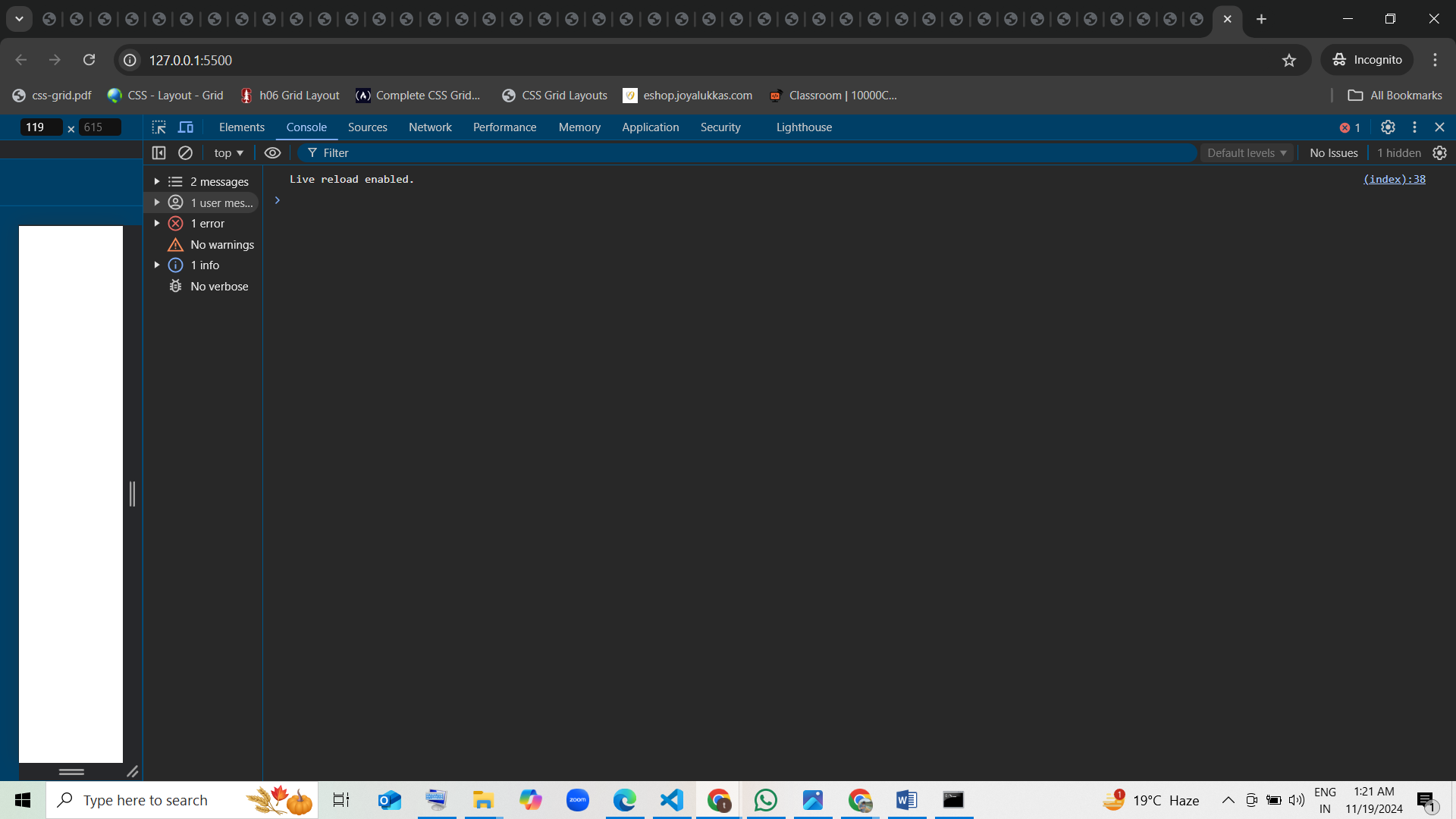
* var q = 5;   
  var q = 10;   
  console.log(q);
  + 1. 5
    2. 10 [guess] [correct]
    3. Error
    4. Undefined



1. What is the value of r after this code executes?

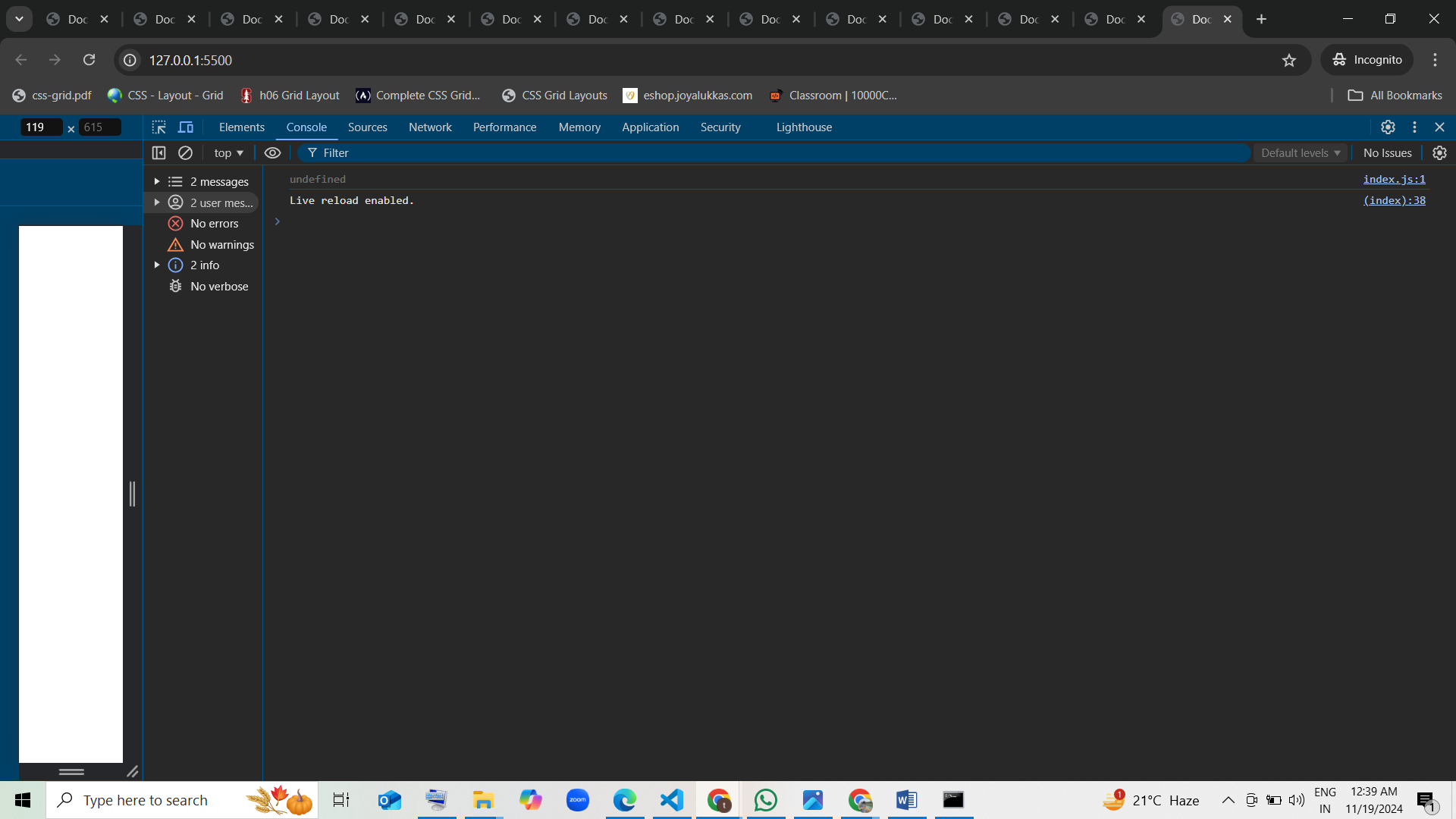
* var r;  
  r = 30;  
  var r = 50;
  + 1. 30 [guess]
    2. 50
    3. Undefined
    4. Error

A: empty



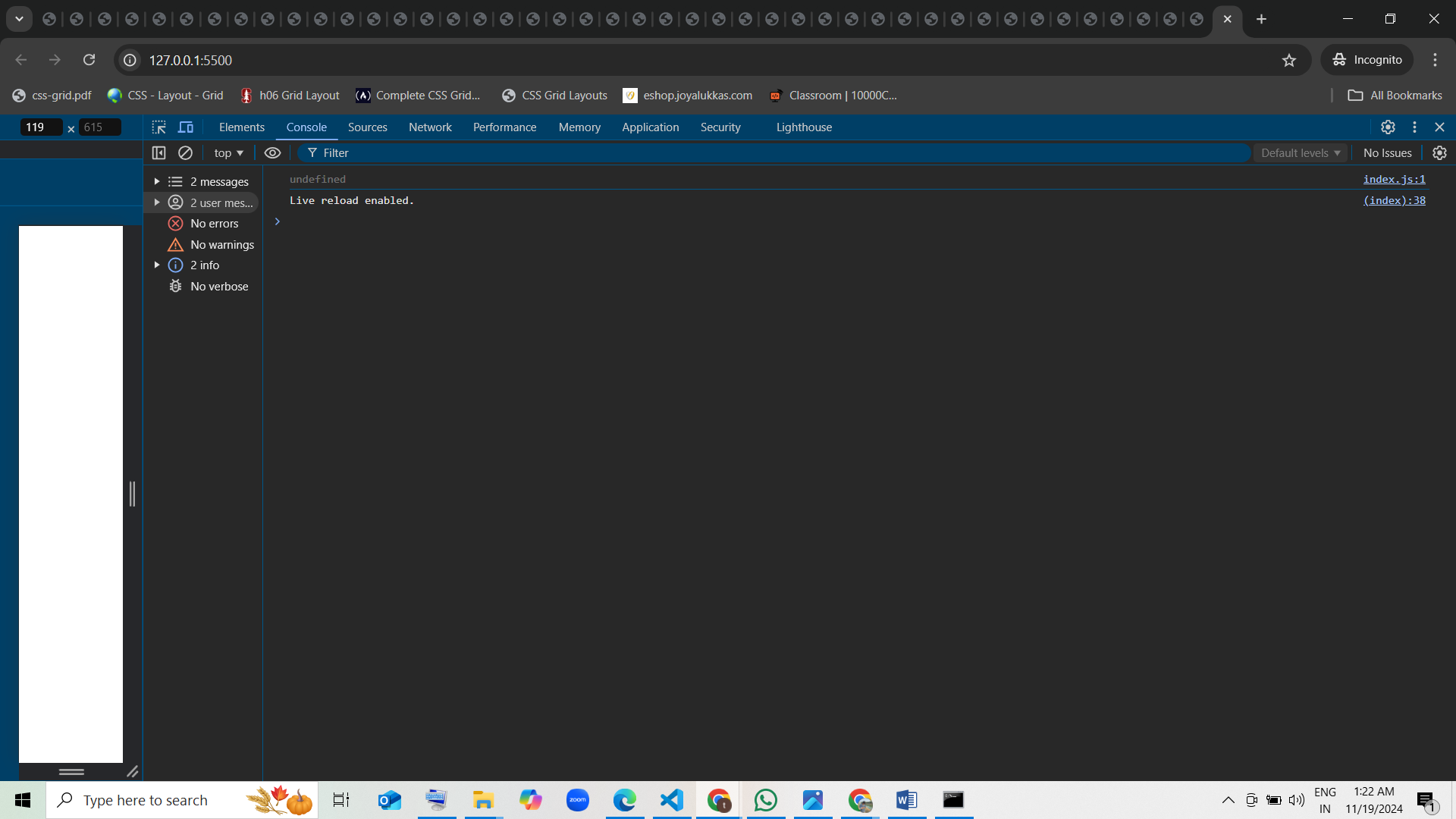
1. What will this code output?

* console.log(x);  
  var x = 25;
  + 1. Undefined [guess] [correct]
    2. 25
    3. Error
    4. Null



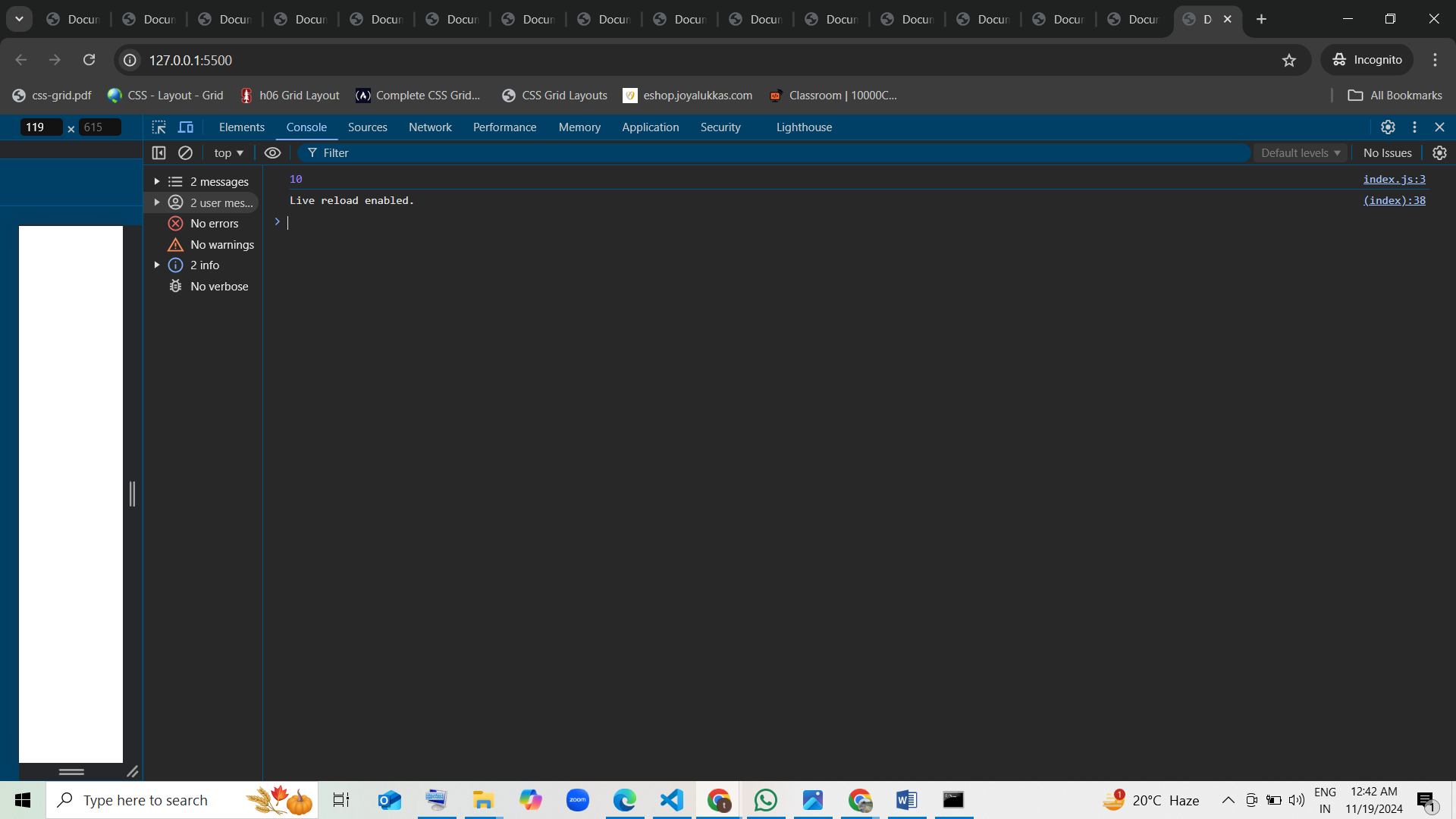
1. What happens if you try to access a var variable before declaring it?

* console.log(a);  
  var a = 40;
  + 1. Undefined [correct]
    2. ReferenceError [guess]
    3. 40
    4. Null



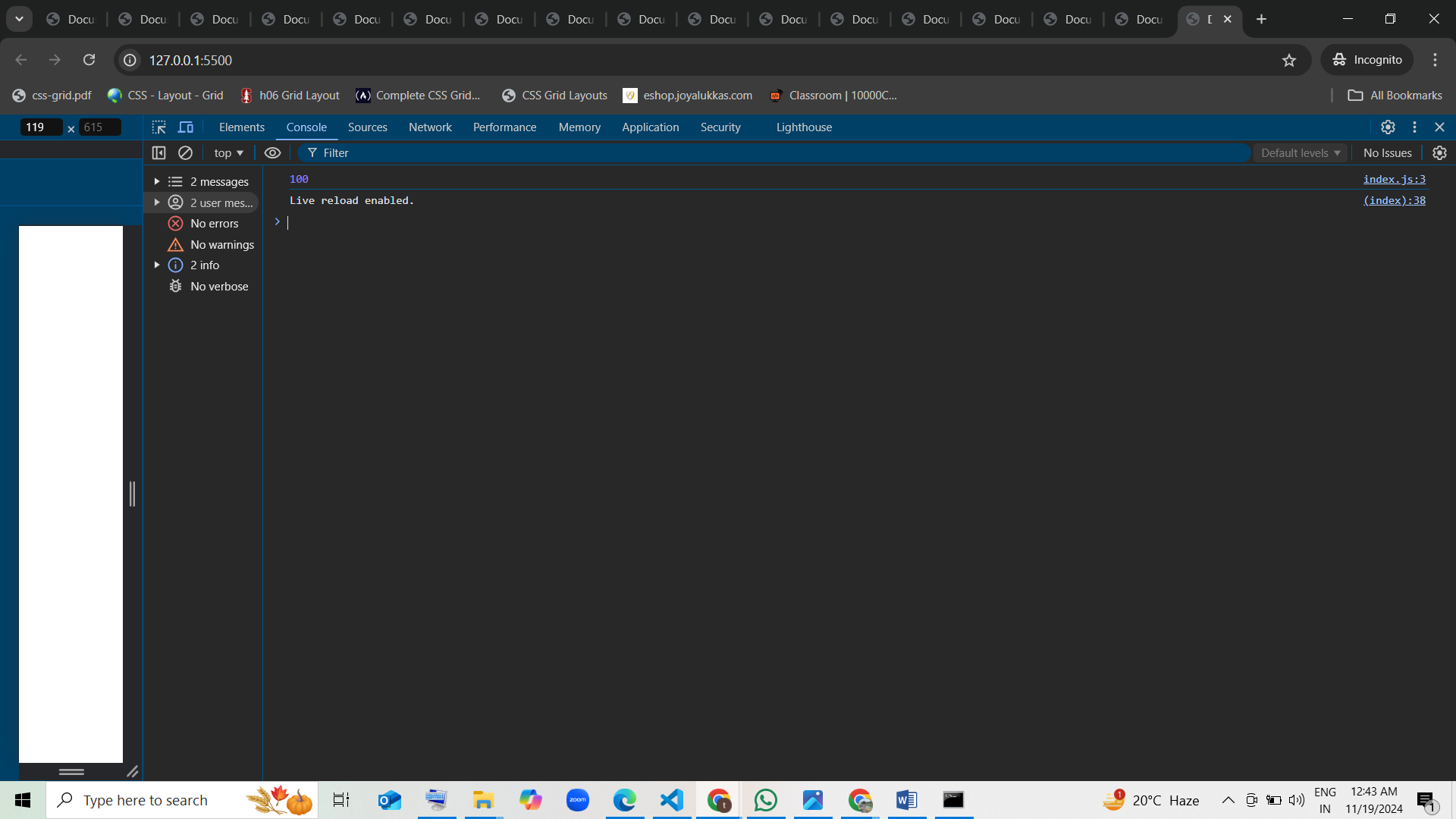
1. What happens in this code?

* var t = 10;   
  var t;   
  console.log(t);
  + 1. Undefined [guess]
    2. 10 [correct]
    3. Error
    4. Null



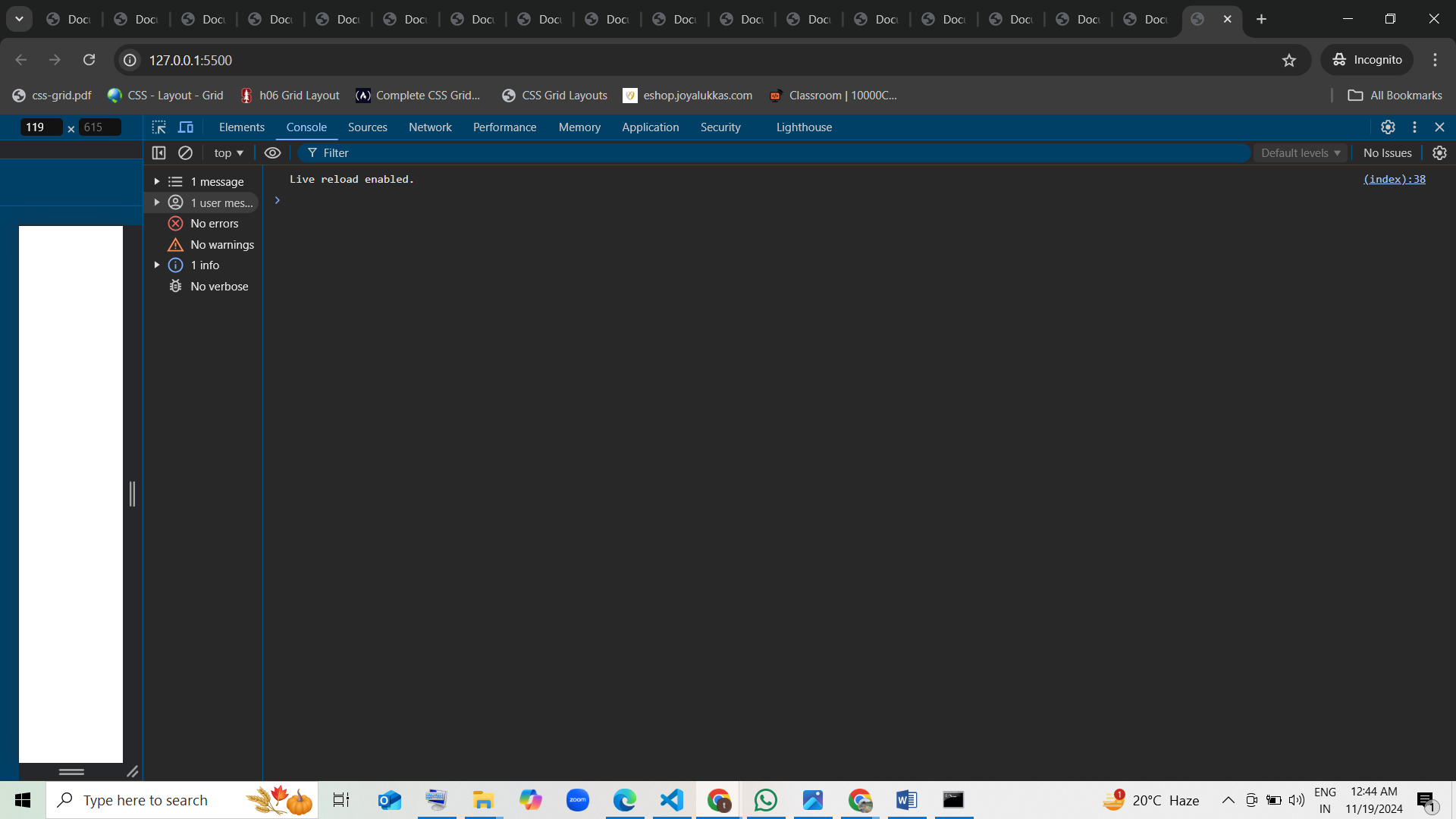
1. What happens if you assign a value to a var variable before declaring it?

* x = 100;   
  var x;   
  console.log(x);
  + 1. 100 [guess] [correct]
    2. Undefined
    3. Error
    4. Null



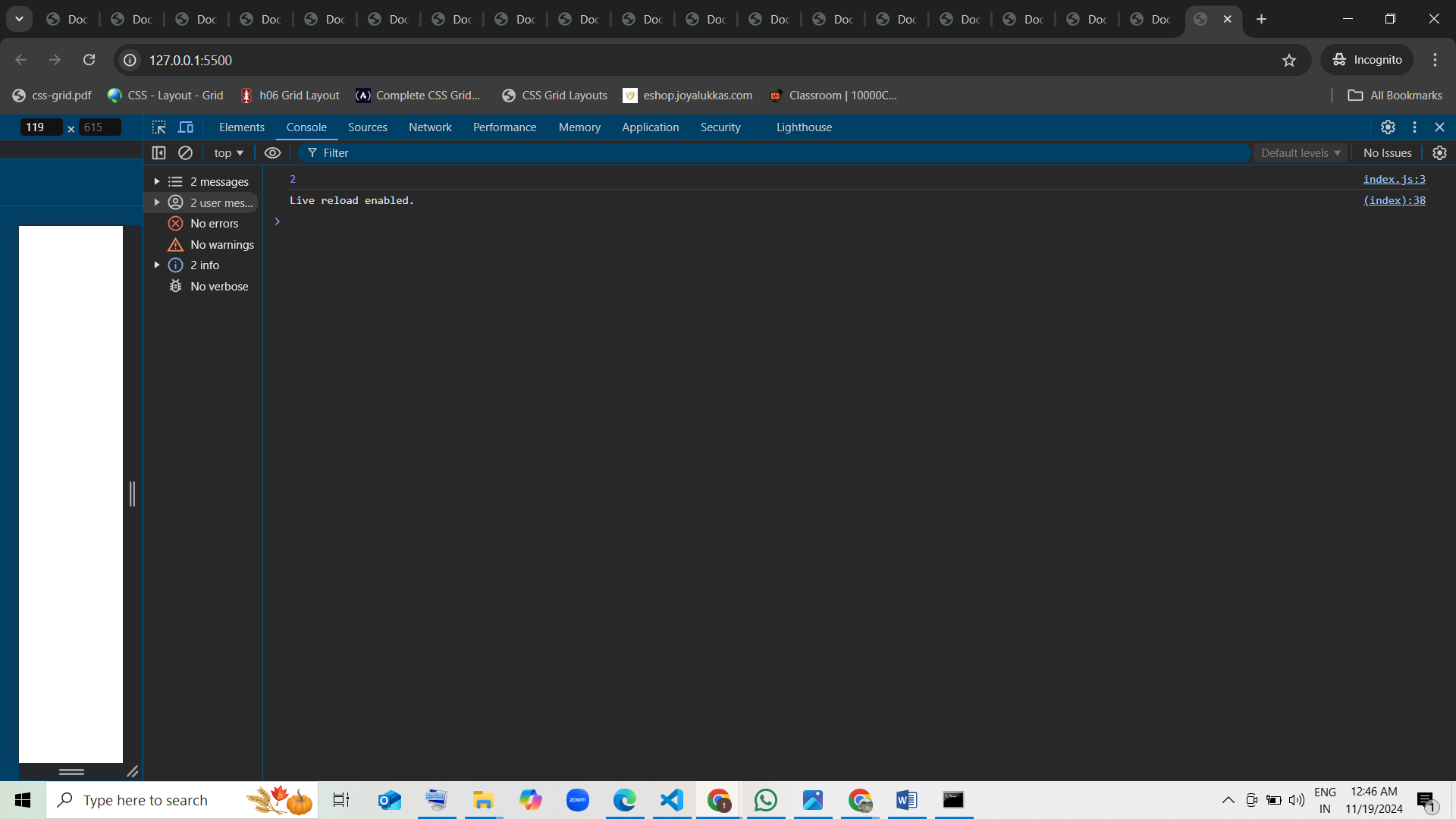
1. What happens when you declare a var variable twice in a global scope?

* var y = 5;   
  var y = 10;
  + 1. Error
    2. Redefines the value [guess]
    3. Ignored second declaration
    4. None of the above [correct]



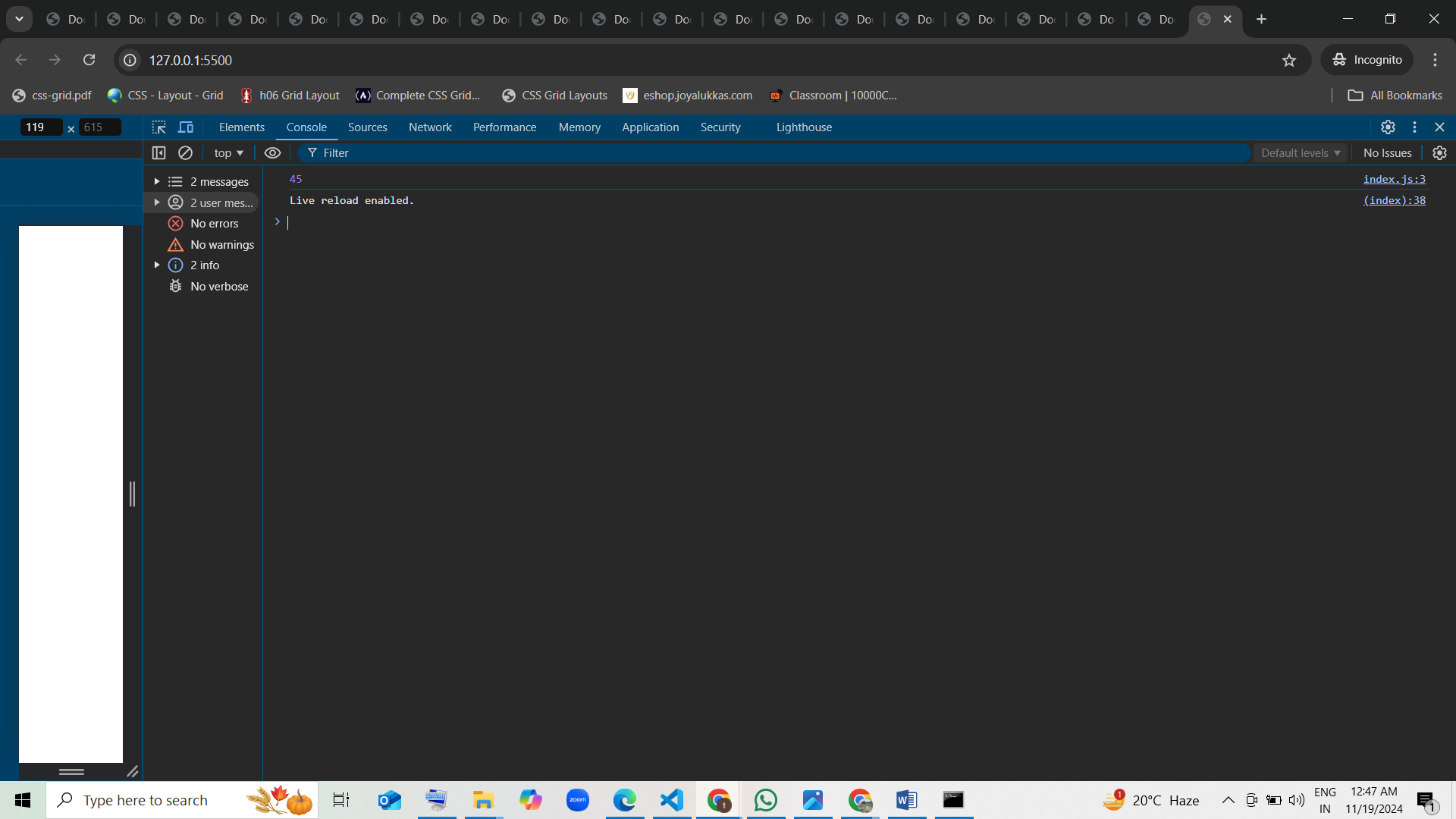
1. What is the output of this code?

* var z = 1;   
  z = 2;   
  console.log(z);
  + 1. 1
    2. 2 [guess] [correct]
    3. Undefined
    4. Error



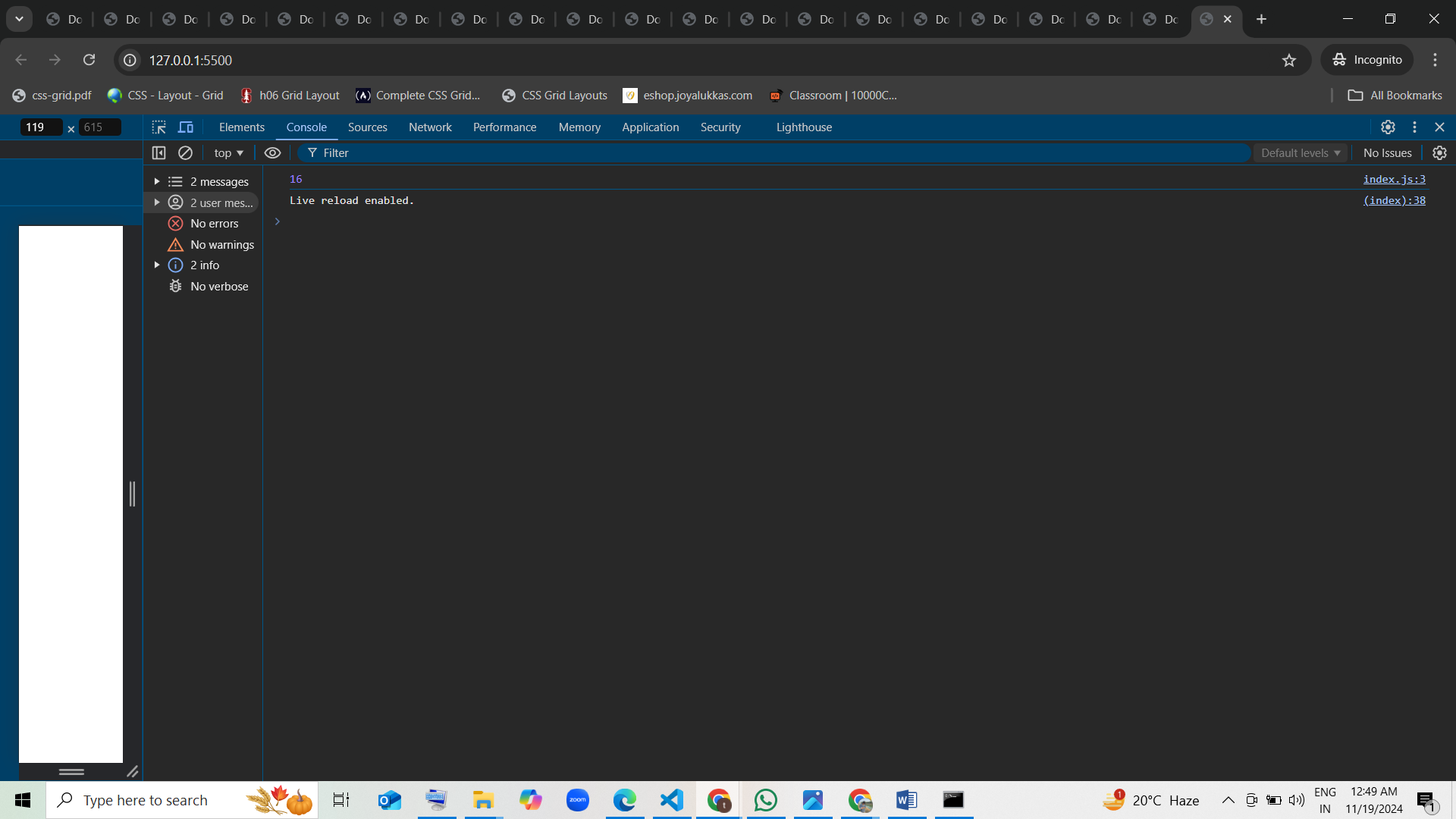
1. What is the value of \_Xyz12$ in this code?

* var \_Xyz12$;  
  \_Xyz12$ = 45;  
  console.log(\_Xyz12$);
  + 1. Undefined
    2. 45 [guess] [correct]
    3. Error
    4. Null



1. What will this code output?

* var $num\_Val = 8;   
  $num\_Val = 16;   
  console.log($num\_Val);
  + 1. 8
    2. 16 [guess] [correct]
    3. Undefined
    4. Error



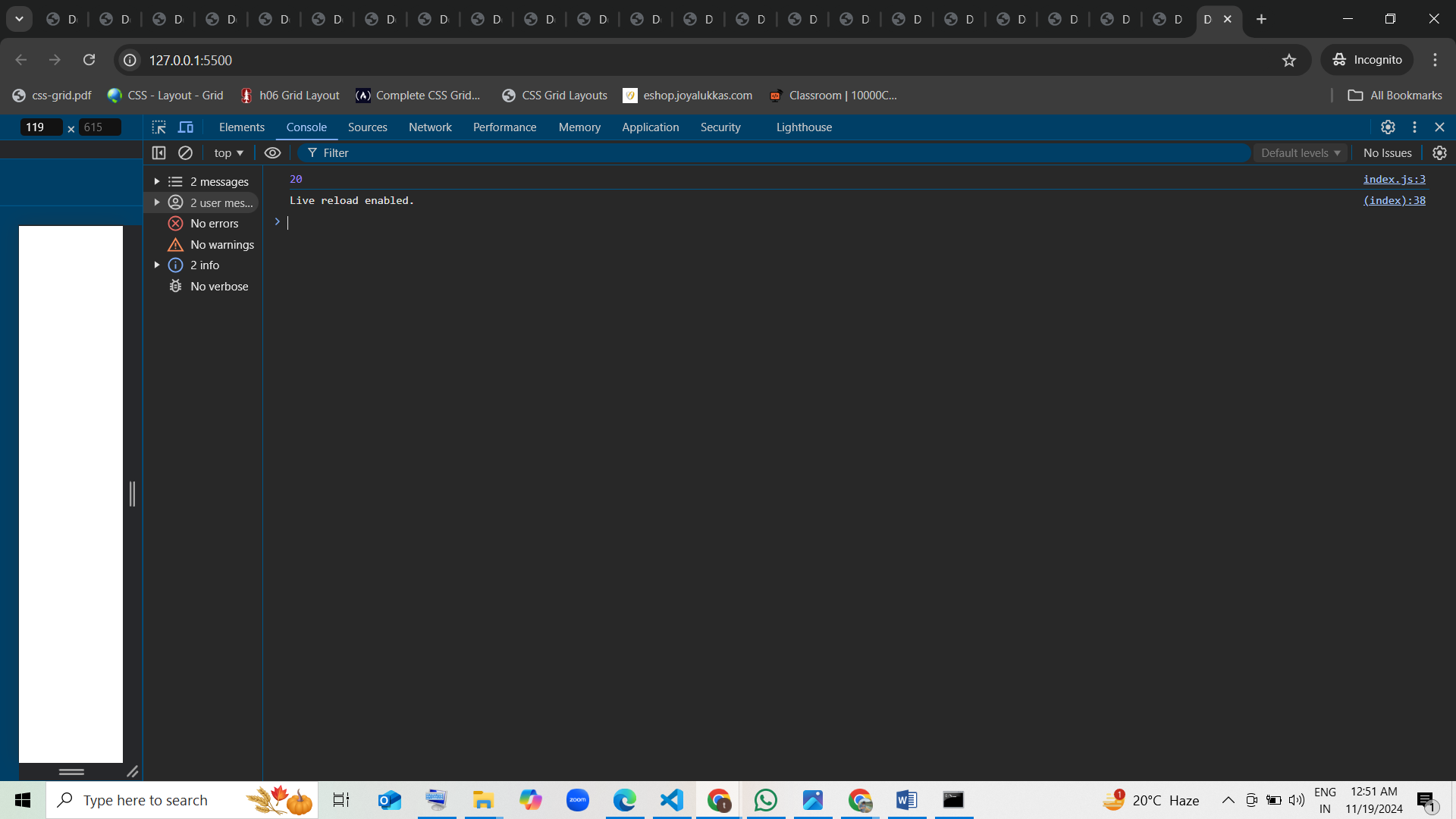
1. What happens in this code?

* console.log(\_x\_Val12);   
  var \_x\_Val12 = 27;
  + 1. Undefined [guess] [correct]
    2. 27
    3. Error
    4. Null



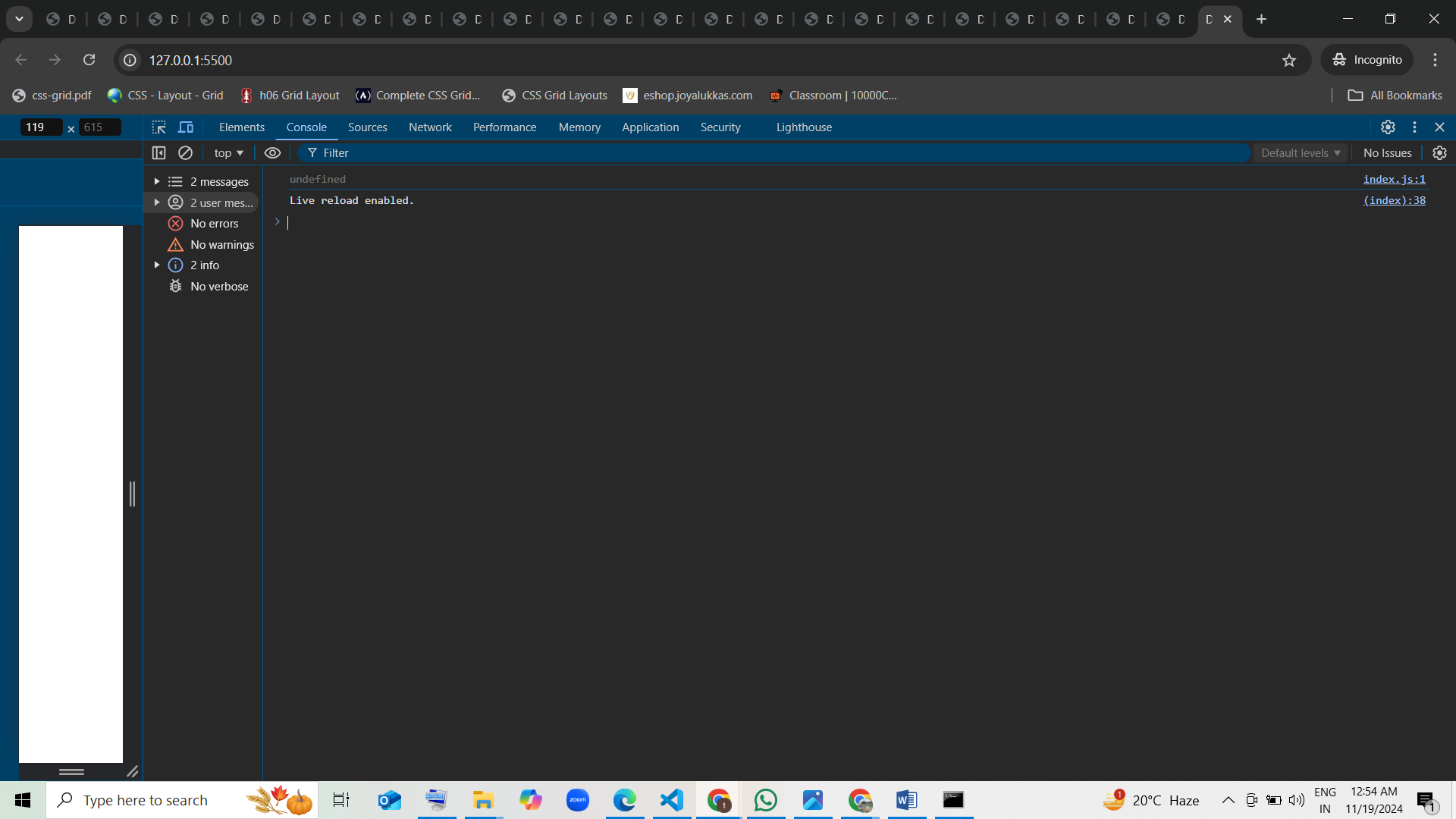
1. What is the final value of Val\_12X$?

* var Val\_12X$ = 10;   
  Val\_12X$ = 20;   
  console.log(Val\_12X$);
  + 1. 10
    2. 20 [guess] [correct]
    3. Error
    4. Null



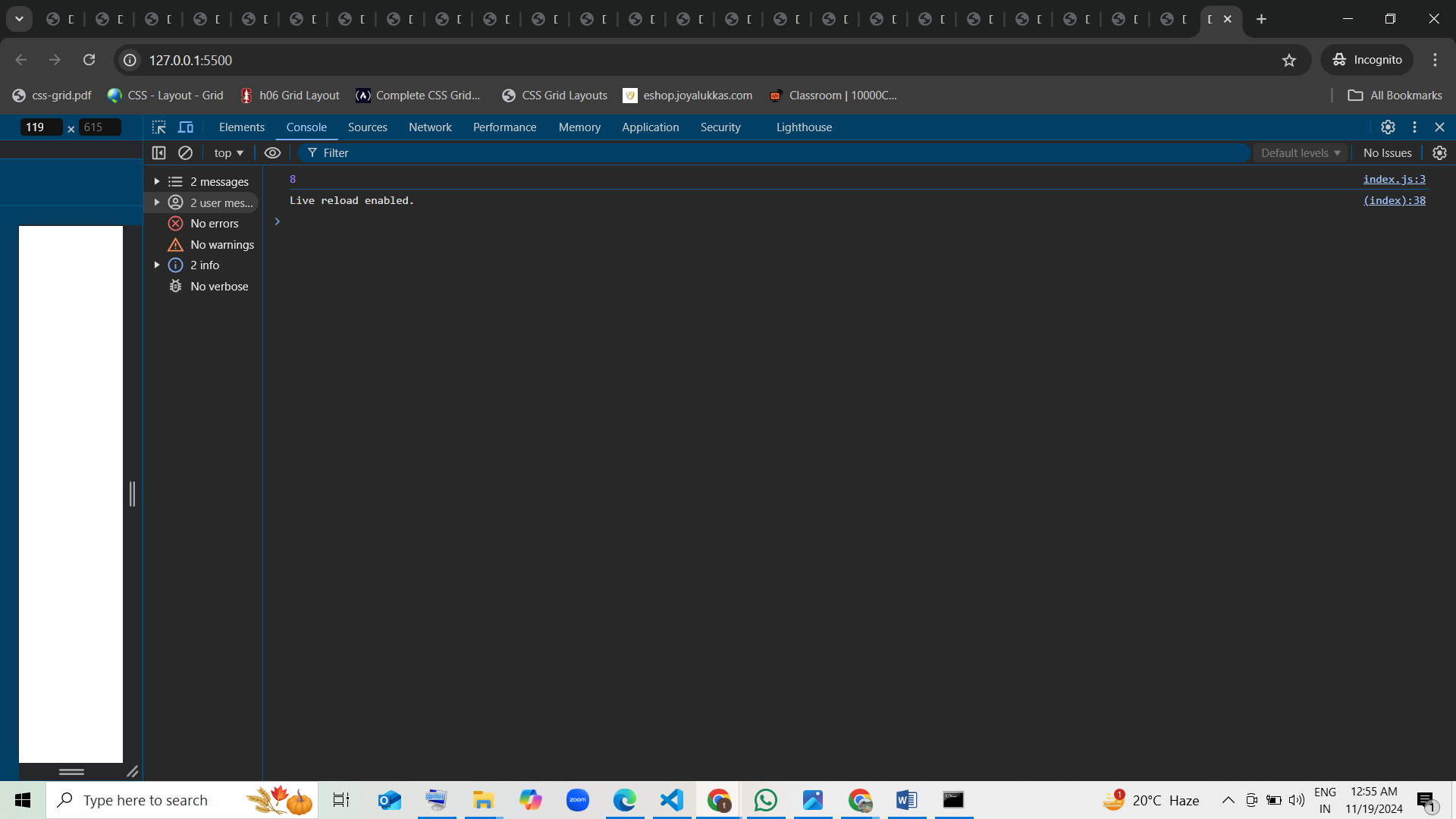
1. What happens here?

* console.log(VAL\_num);   
  var VAL\_num = "Test";
  + 1. Undefined [guess] [correct]
    2. Test
    3. Error
    4. Null



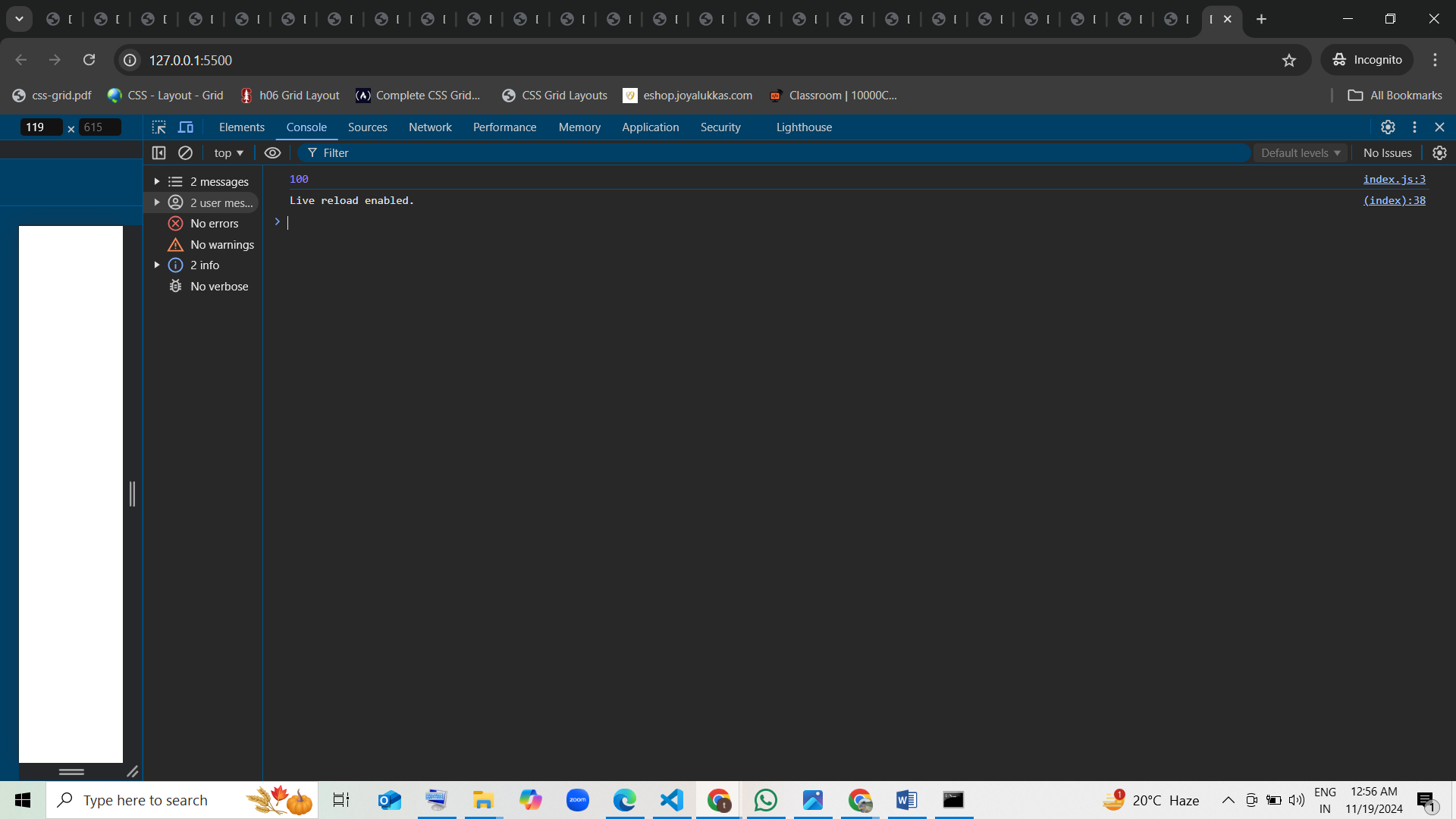
1. What does this code print?

* var $count = 5;   
  $count = $count + 3;   
  console.log($count);
  + 1. 5
    2. 8 [guess] [correct]
    3. Undefined
    4. Error



1. What is the result of this code?

* var \_alpha$ = 50;   
  var \_alpha$ = 100;   
  console.log(\_alpha$);
  + 1. 50
    2. 100 [guess] [correct]
    3. Error
    4. Undefined



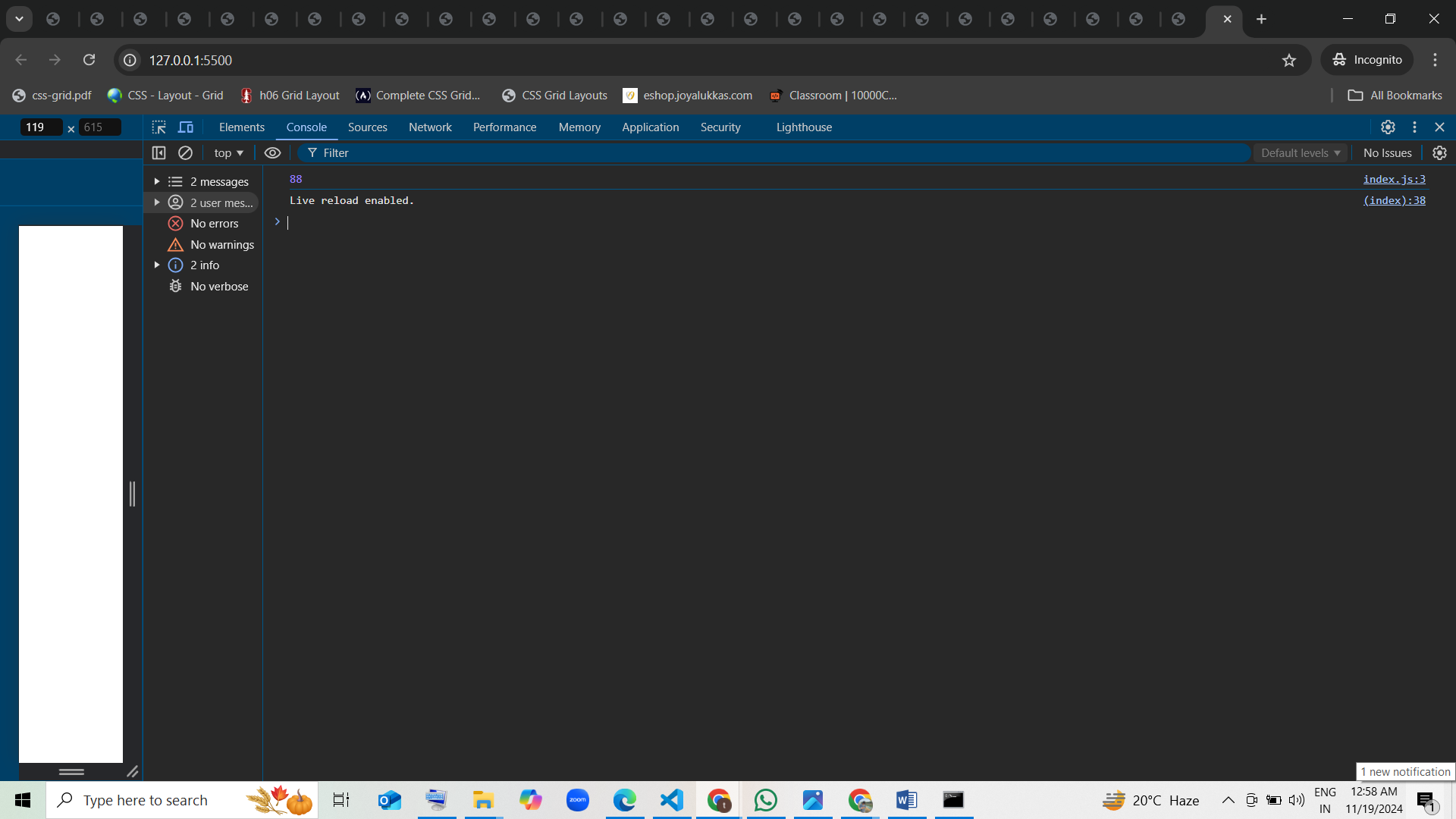
1. What happens in this code?

* var A1b\_2$ = 1;   
  console.log(A1b\_2$);
  + 1. 1 [guess] [correct]
    2. Undefined
    3. Error
    4. Null



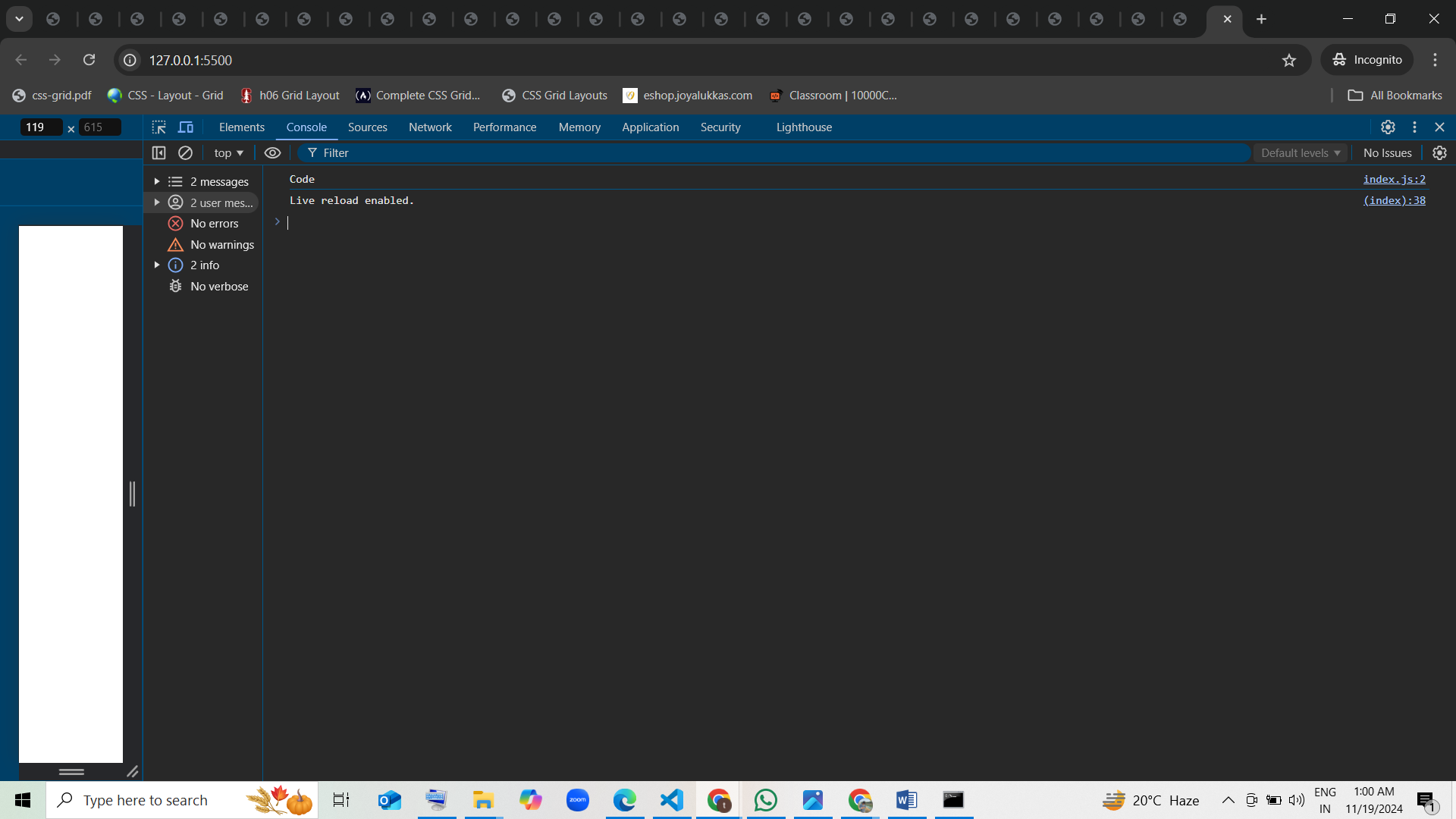
1. What does this print?

* var $$$ = 99;   
  $$$ = 88;   
  console.log($$$);
  + 1. 99 [guess]
    2. 88 [correct]
    3. Undefined
    4. Error



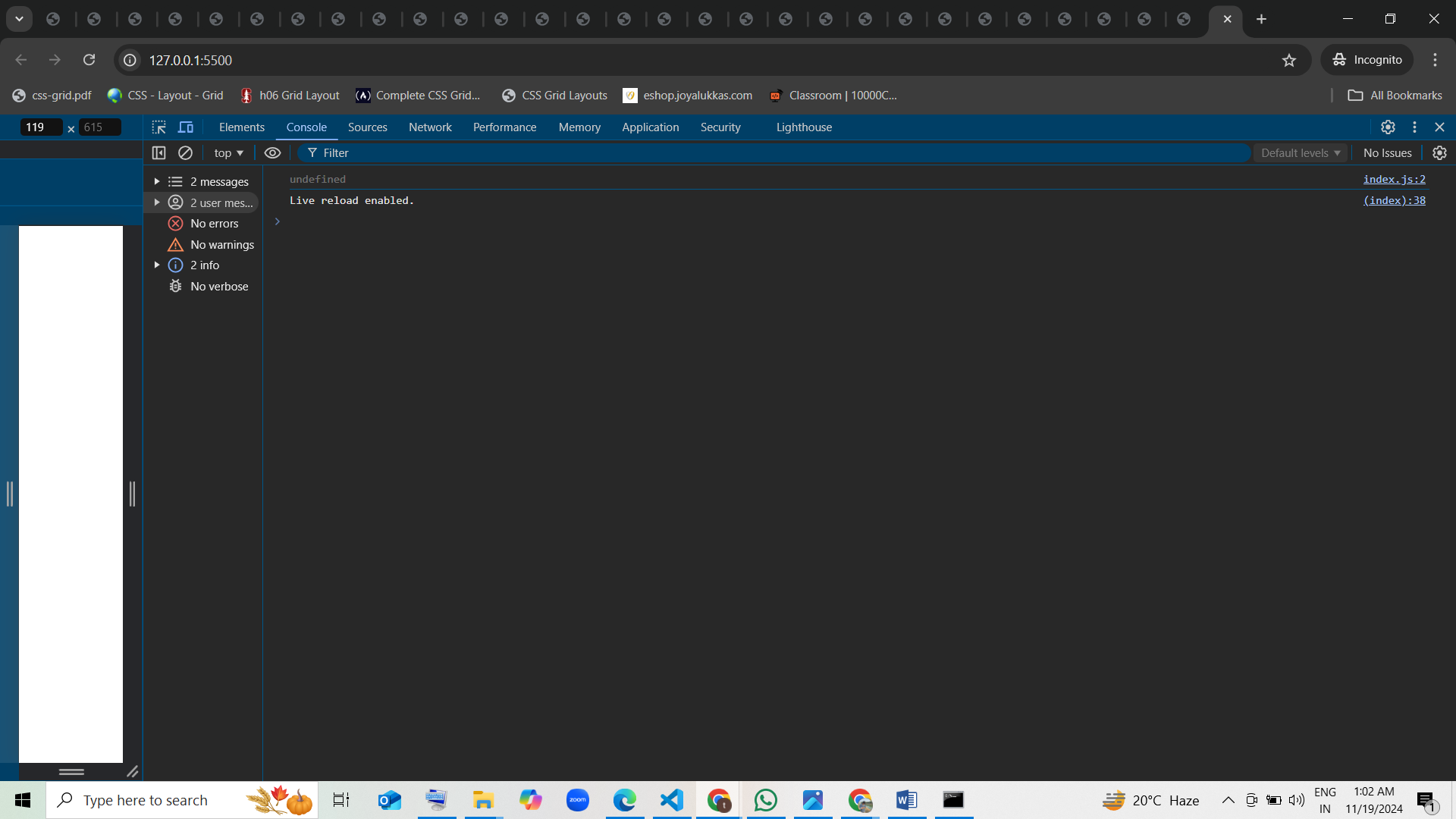
1. What is the output of this code?

* var \_vA\_rX = "Code";   
  console.log(\_vA\_rX);
  + 1. Undefined
    2. Code [guess] [correct]
    3. Error
    4. Null



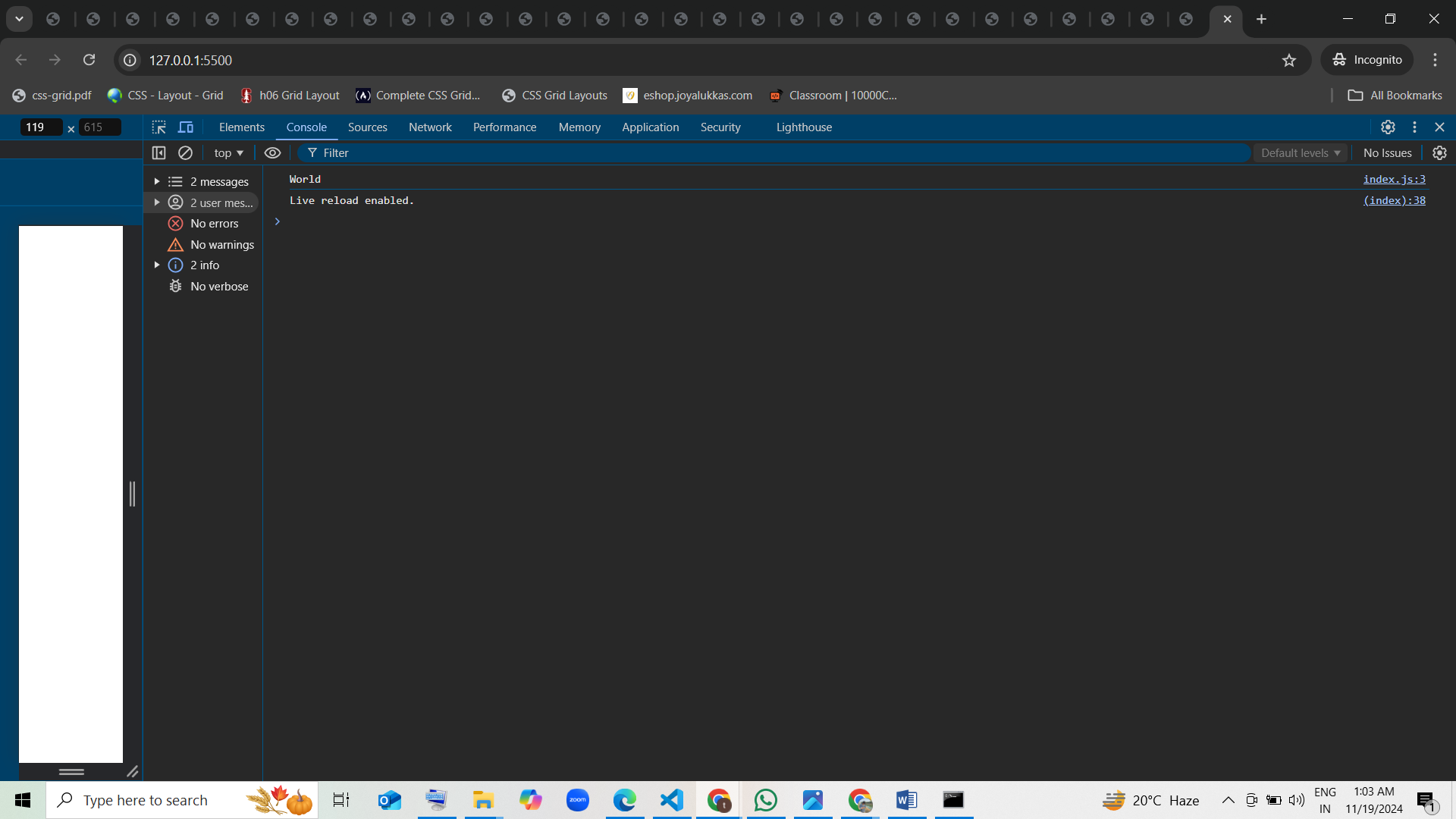
1. What happens when this executes?

* var X\_$12;   
  console.log(X\_$12);   
  X\_$12 = 123;
  + 1. Undefined [guess] [correct]
    2. 123
    3. Error
    4. Null



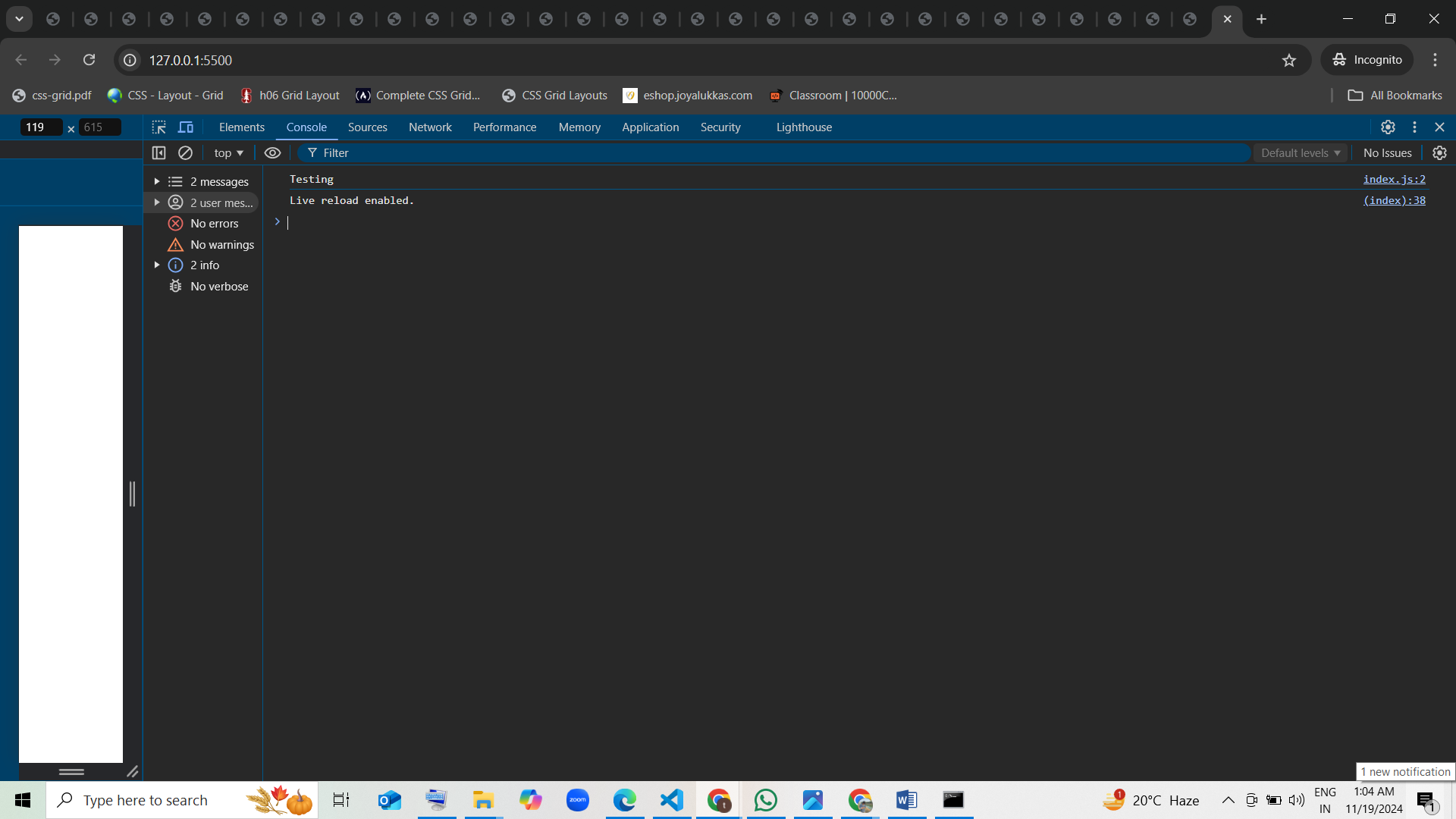
1. What is the behavior of \_\_X\_var$ in this scenario?

* var \_\_X\_var$ = "Hello";   
  var \_\_X\_var$ = "World";   
  console.log(\_\_X\_var$);
  + 1. Hello
    2. World [guess] [correct]
    3. Error
    4. Undefined



1. What will this output?

* var VaR\_$$ = "Testing";   
  console.log(VaR\_$$);
  + 1. Undefined
    2. Testing [guess] [correct]
    3. Error
    4. Null



1. What happens here?

* console.log($$$Alpha);   
  var $$$Alpha = 55;
  + 1. Undefined [guess] [correct]
    2. 55
    3. Error
    4. Null



1. What is the final value of 12x$var\_?

* var 12x$var\_ = 100;   
  console.log(12x$var\_); here var not started with **\_ or $**
  + 1. 100 [guess]
    2. Undefined
    3. Error
    4. Null [correct]

