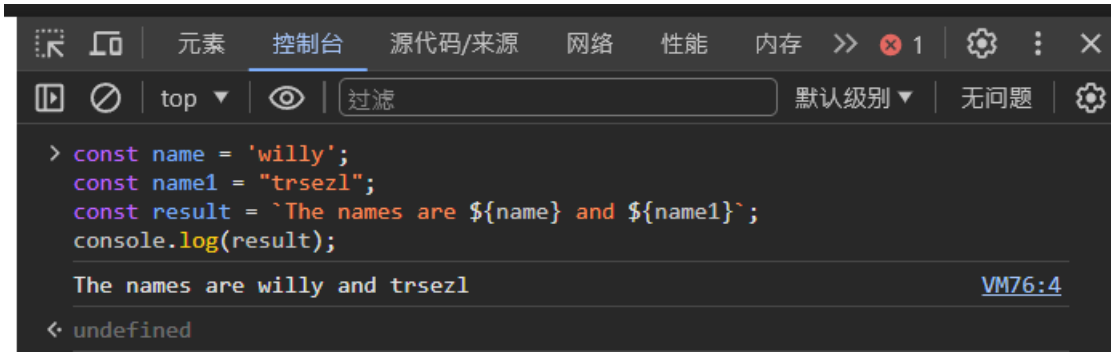
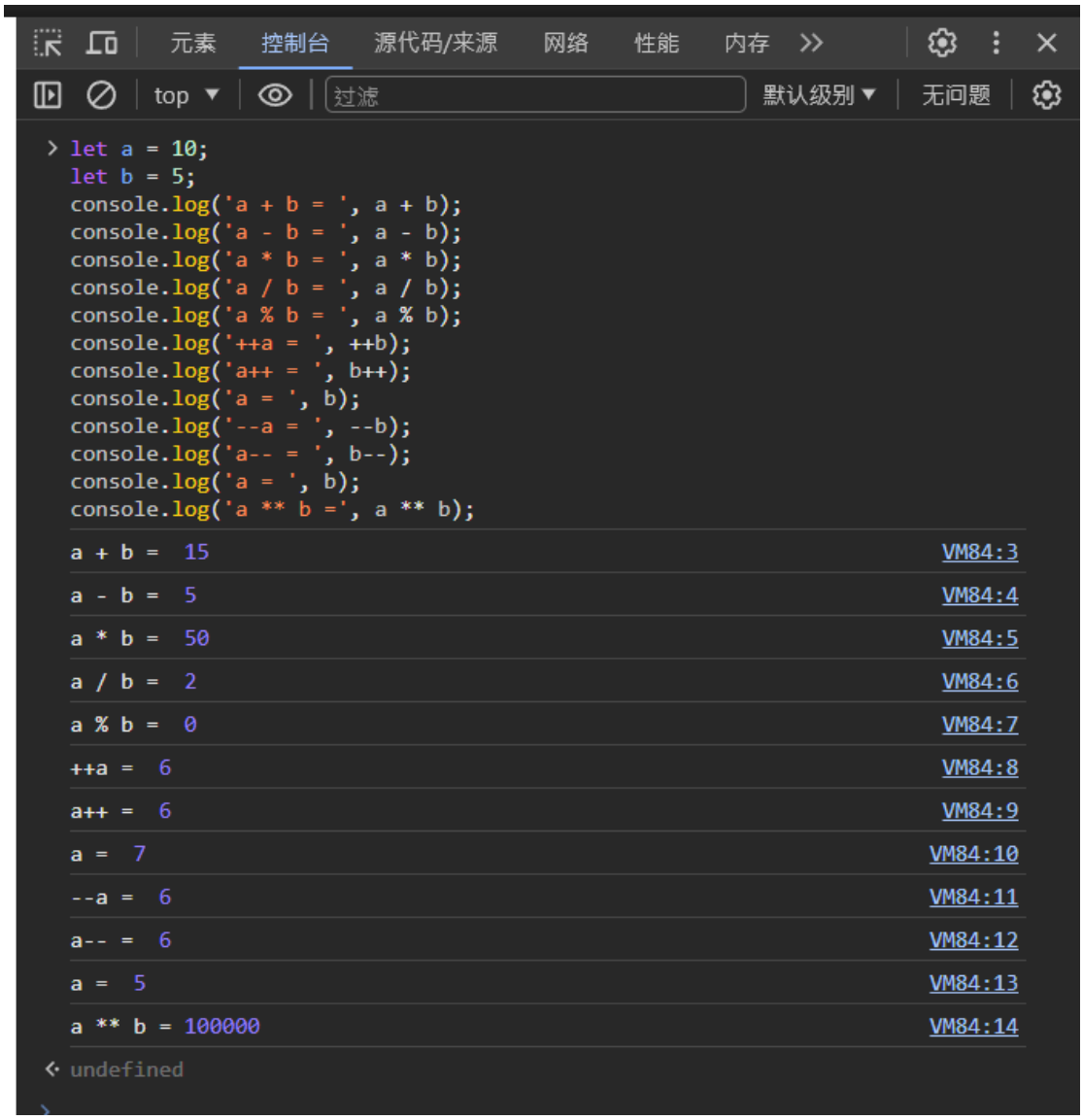
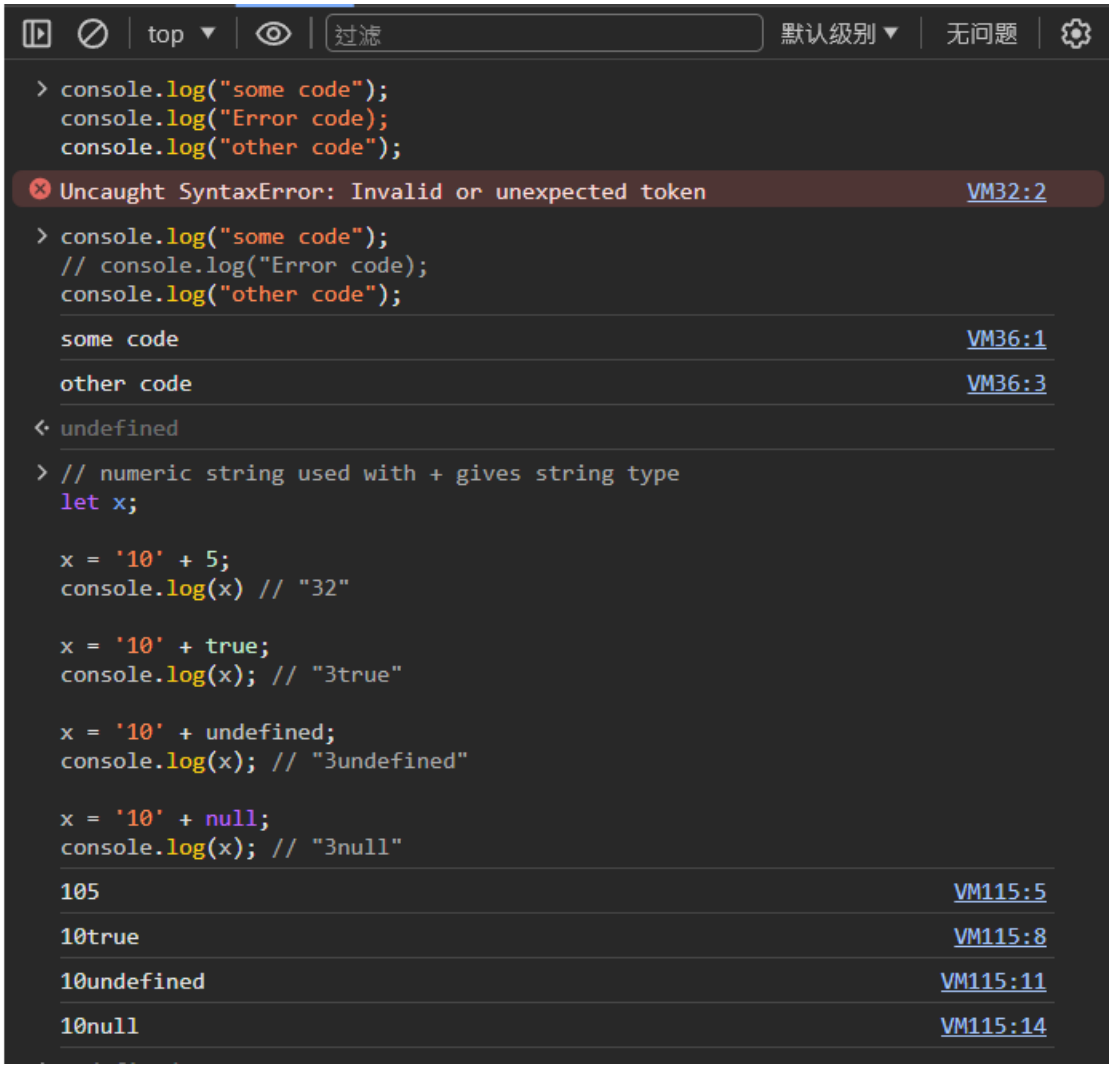
B11118024 黄靖哲




元素
控制台
源代码/来源
网络
性能
内存
top
过滤
默认级别
无问题
const x = 9999999999999999999n;
const y = x + 1n;
console.log(y);
const value2 = 900719925124740998n;
const result2 = value2 + 1n;
console.log(result2);
10000000000000000000n
VM72:3
900719925124740999n
VM72:6
undefined


元素
控制台
源代码/来源
网络
性能
内存
top
过滤
默认级别
无问题
let a = 10;
let b = 5;
console.log('a + b = ', a + b);
console.log('a - b = ', a - b);
console.log('a * b = ', a * b);
console.log('a / b = ', a / b);
console.log('a % b = ', a % b);
console.log('++a = ', ++b);
console.log('a++ = ', b++);
console.log('a = ', b);
console.log('--a = ', --b);
console.log('a-- = ', b--);
console.log('a = ', b);
console.log('a ** b =', a ** b);
a + b = 15
VM84:3
a - b = 5
VM84:4
a * b = 50
VM84:5
a / b = 2
VM84:6
a % b = 0
VM84:7
++a = 6
VM84:8
a++ = 6
VM84:9
a = 7
VM84:10
--a = 6
VM84:11
a-- = 6
VM84:12
a = 5
VM84:13
a ** b = 100000
VM84:14
undefined


top
过滤
默认级别
无问题
> console.log("some code");
console.log("Error code);
console.log("other code");
Uncaught SyntaxError: Invalid or unexpected token
VM32:2
> console.log("some code");
// console.log("Error code);
console.log("other code");
some code
VM36:1
other code
VM36:3
undefined
> // numeric string used with + gives string type
let x;
x = '10' + 5;
console.log(x) // "32"
x = '10' + true;
console.log(x); // "3true"
x = '10' + undefined;
console.log(x); // "3undefined"
x = '10' + null;
console.log(x); // "3null"
105
VM115:5
10true
VM115:8
10undefined
VM115:11
10null
VM115:14

```
> // numeric string used with - , / , * results number type

  let x;

  x = '10' - '5';
  console.log(x); // 2

  x = '10' - 5;
  console.log(x); // 2

  x = '10' * 5;
  console.log(x); // 8

  x = '10' / 5;
  console.log(x); // 2
```

```
5                                                    VM86:6
5                                                    VM86:9
50                                                   VM86:12
2                                                    VM86:15
< undefined
```

```
> // non-numeric string used with - , / , * results to NaN

  let x;

  x = 'hi' - 'ha';
  console.log(x); // NaN

  x = '10' - 'hi';
  console.log(x); // NaN
```

```
NaN                                                  VM133:6
NaN                                                  VM133:9
< undefined
>
```

```
> // if boolean is used, true is 1, false is 0

  let x;

  x = '10' - true;
  console.log(x); // 3

  x = 10 + true;
  console.log(x); // 5

  x = 10 + false;
  console.log(x); // 4
```

```
9                                                    VM82:6
11                                                   VM82:9
10                                                   VM82:12
< undefined
```

```
> // null is 0 when used with number
  let x;

  x = 10 + null;
  console.log(x);  // 4

  x = 10 - null;
  console.log(x);  // 4
```

```
10                                                   VM116:5
10                                                   VM116:8
```

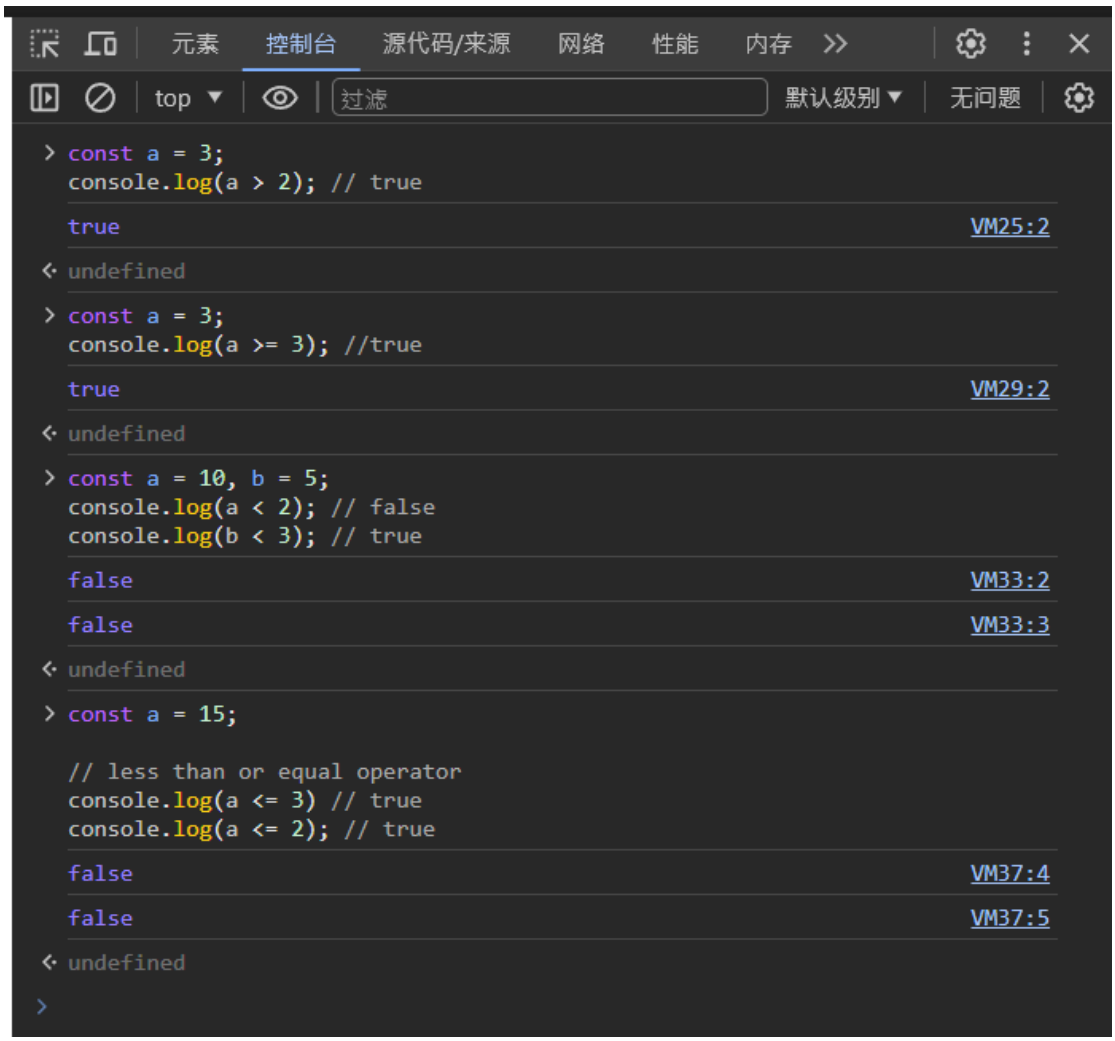
```
let x;

x = 10 + undefined;
console.log(x);  // NaN

result = 10 - undefined;
console.log(x);  // NaN

result = true + undefined;
console.log(x);  // NaN

result = null + undefined;
console.log(x);  // NaN
```

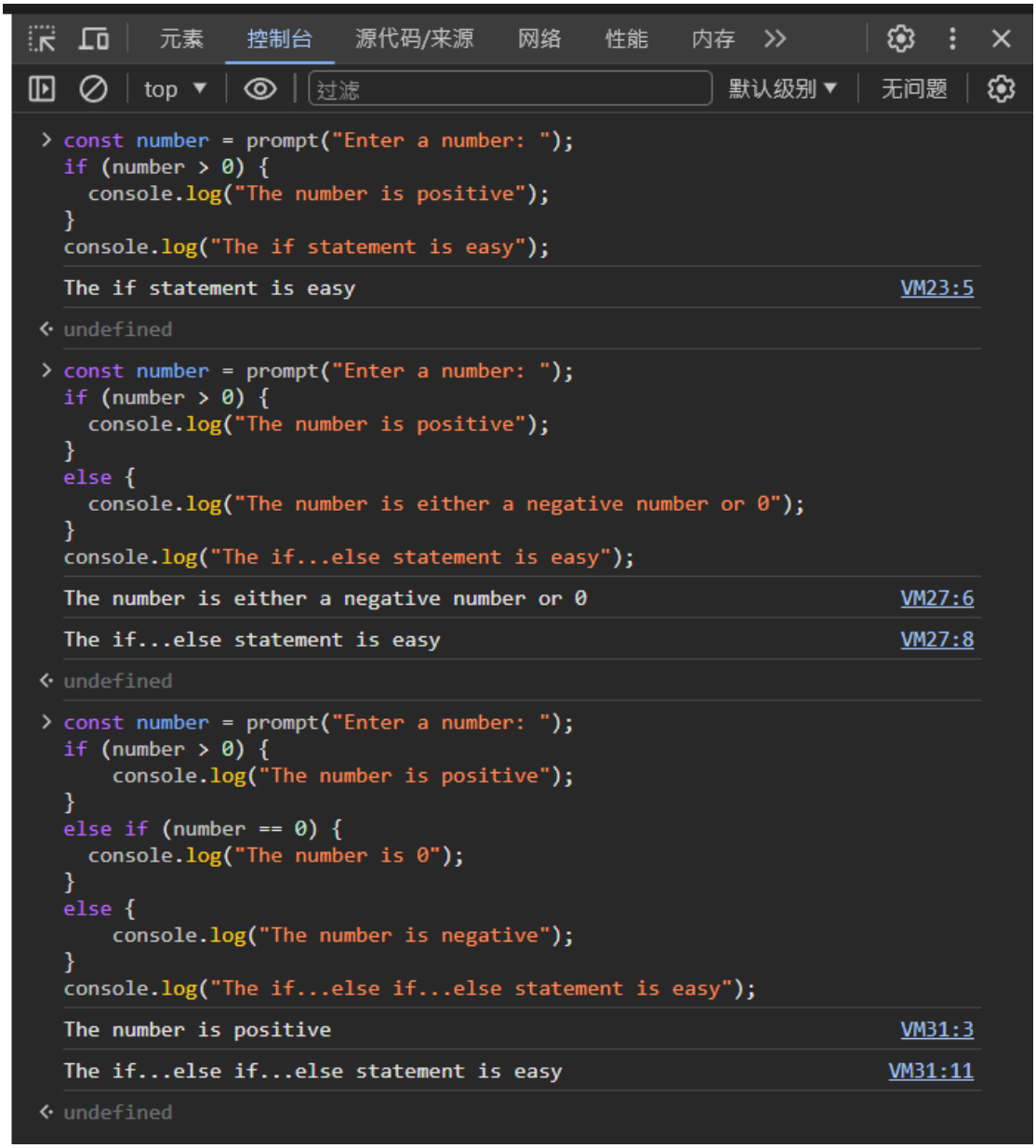
```
NaN                                                    VM183:6
NaN                                                    VM183:9
NaN                                                    VM183:12
NaN                                                    VM183:15
undefined
```

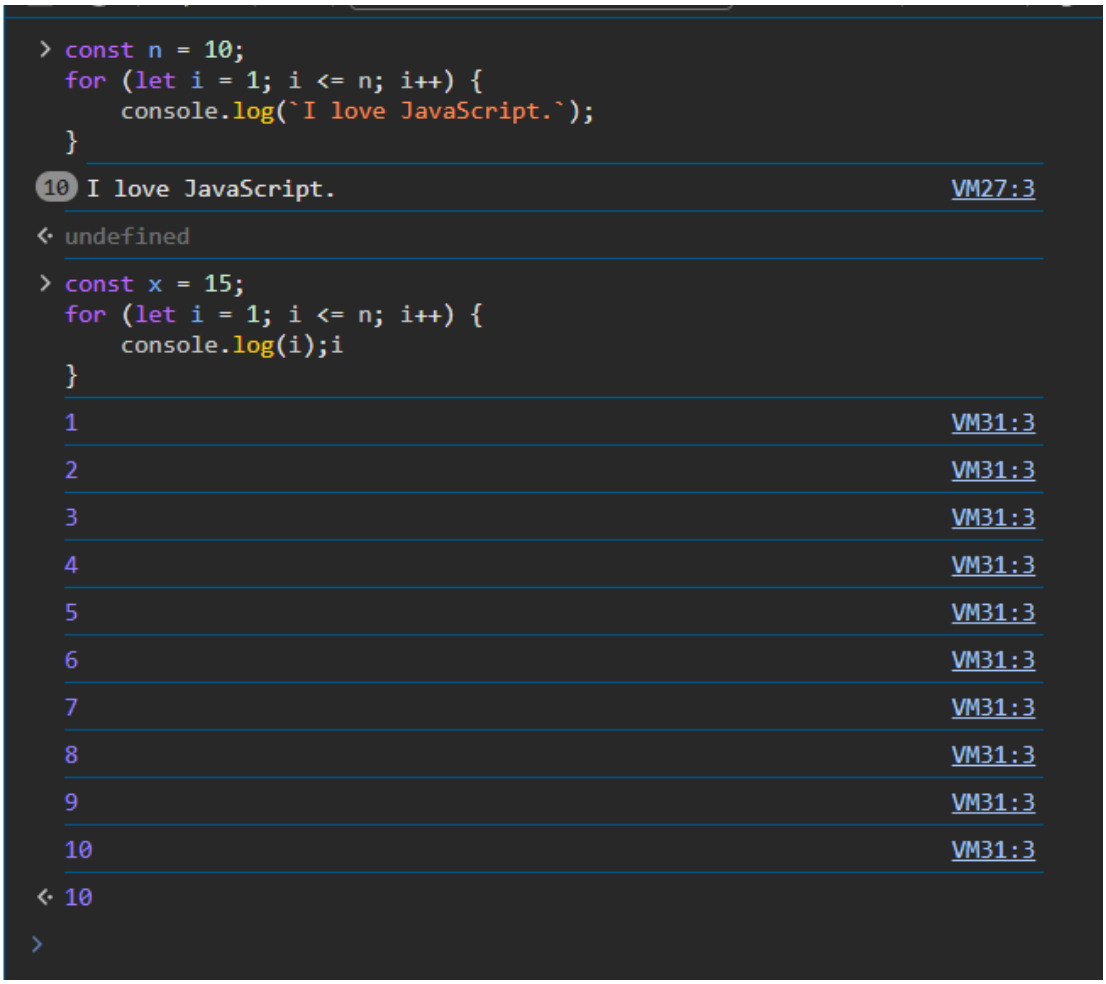
```
let z;

// string to number
z = Number('350');
console.log(z); // 324

z = Number('350e-1')
console.log(z); // 32.4

// boolean to number
z = Number(true);
console.log(z); // 1

z = Number(false);
console.log(z); // 0
```

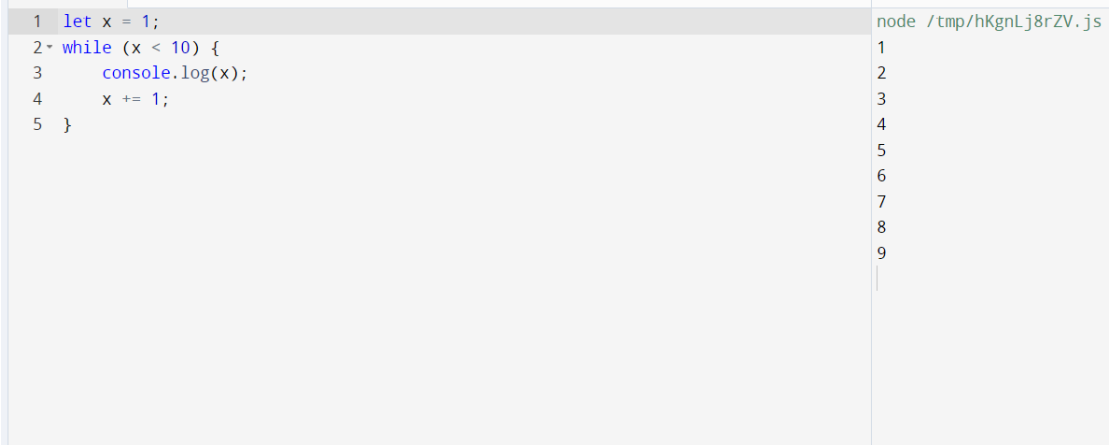
```
350                                                    VM238:5
35                                                     VM238:8
1                                                      VM238:12
0                                                      VM238:15
```


top
过滤
默认级别
无问题
> const a = 10, b = 5, c = 'hello';
console.log(a == 10);   // true
console.log(b == '5');   // true
console.log(c == 'Hello'); // false
true VM44:2
true VM44:3
false VM44:4
undefined
> const a = 5, b = 'hello';
console.log(a != 5); // true
console.log(b != 'Hello'); // true
false VM48:2
true VM48:3
undefined
> const a = 15;
console.log(a === 15); // true
console.log(a === '15'); // false
true VM52:2
false VM52:3
undefined
> const a = 11, b = 'hi';
console.log(a !== 11); // false
console.log(a !== '11'); // true
console.log(b !== 'Hi'); // true
false VM56:2
true VM56:3
true VM56:4
undefined
>


元素
控制台
源代码/来源
网络
性能
内存
top
过滤
默认级别
无问题
const a = 3;
console.log(a > 2); // true
true
VM25:2
undefined
const a = 3;
console.log(a >= 3); //true
true
VM29:2
undefined
const a = 10, b = 5;
console.log(a < 2); // false
console.log(b < 3); // true
false
VM33:2
false
VM33:3
undefined
const a = 15;
// less than or equal operator
console.log(a <= 3) // true
console.log(a <= 2); // true
false
VM37:4
false
VM37:5
undefined


元素
控制台
源代码/来源
网络
性能
内存
top
过滤
默认级别
无问题
> const number = prompt("Enter a number: ");
if (number > 0) {
console.log("The number is positive");
}
console.log("The if statement is easy");
The if statement is easy
VM23:5
undefined
> const number = prompt("Enter a number: ");
if (number > 0) {
console.log("The number is positive");
}
else {
console.log("The number is either a negative number or 0");
}
console.log("The if...else statement is easy");
The number is either a negative number or 0
VM27:6
The if...else statement is easy
VM27:8
undefined
> const number = prompt("Enter a number: ");
if (number > 0) {
console.log("The number is positive");
}
else if (number == 0) {
console.log("The number is 0");
}
else {
console.log("The number is negative");
}
console.log("The if...else if...else statement is easy");
The number is positive
VM31:3
The if...else if...else statement is easy
VM31:11
undefined


const n = 10;
for (let i = 1; i <= n; i++) {
console.log(`I love JavaScript.`);
}
10 I love JavaScript. VM27:3
undefined
const x = 15;
for (let i = 1; i <= n; i++) {
console.log(i);i
}
1 VM31:3
2 VM31:3
3 VM31:3
4 VM31:3
5 VM31:3
6 VM31:3
7 VM31:3
8 VM31:3
9 VM31:3
10 VM31:3
10


let x = 1;
while (x < 10) {
console.log(x);
x += 1;
}
node /tmp/hKgnLj8rZV.js
1
2
3
4
5
6
7
8
9

```
let num = 0, sum = 0;
while (num >= 0) {
    sum += num;
num = parseInt(prompt("Enter a number: "));
}
console.log(`The sum is ${sum}`);
```

```
node /tmp/mjYSGr5m5V.js
Enter a number: 20
Enter a number: 20
Enter a number:
```

```
let x = 10;
do {
    console.log(x);
    x--;
} while (x > 0);
```

```
node /tmp/pZ7ZP8SL9g.js
10
9
8
7
6
5
4
3
2
1
```

```
let sum = 0, num = 0;
do {
    sum += num;
    num = parseInt(prompt("Enter a number: "));
} while (num >= 0);
console.log(`The sum is ${sum}`);
```

```
node /tmp/jmaCOcBfPp.js
Enter a number: 10
Enter a number: 10
Enter a number: 10
Enter a number: -10
The sum is 30
```

```
function greet() {
    console.log("Hi");
}
greet();
```

```
node /tmp/rS0oPk
Hi
```

```
function greet(name) {
    console.log("Hello " + name + ":(");
}
let name = prompt("Enter a name: ");
greet(name);
```

```
node /tmp/75VUsbTf8j.js
Enter a name: willy
Hello willy:(
```

```
function add(x, y) {
    console.log(x + y);
}
add(10,100);
add(20,200);
```

```
node /tmp/4E6Tstc2Cy.js
110
220
```

```

function add(x, y) {
    return x + y;
}
let number1 = parseFloat(prompt("Enter first number: "));
let number2 = parseFloat(prompt("Enter second number: "));
let result = add(number1,number2);
console.log("The sum is " + result);
```

```
node /tmp/D6SxKgwMvM.js
Enter first number: 10
Enter second number: 20
The sum is 30
```