1. Variable

Var name = ….;

Var type: string, number, boolean, null, undefined

Js automatively define and change the var type

Define:

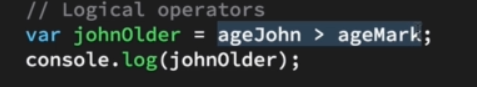
Var x,y,z;

X=…;

Y=…;

Z=…;

1. Console.log(x + y); type the x +y to console, x + y can be anything
2. Alert(x +y); type the x+y to pop-up window
3. Var lastName = prompt (‘what is his last name’);
4. Math operators: + - \* /
5. Logical operator: > <



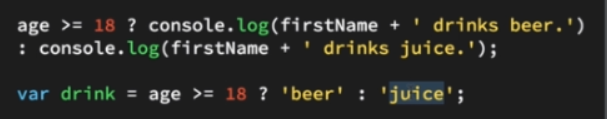
Result: false

1. Typeof operator:

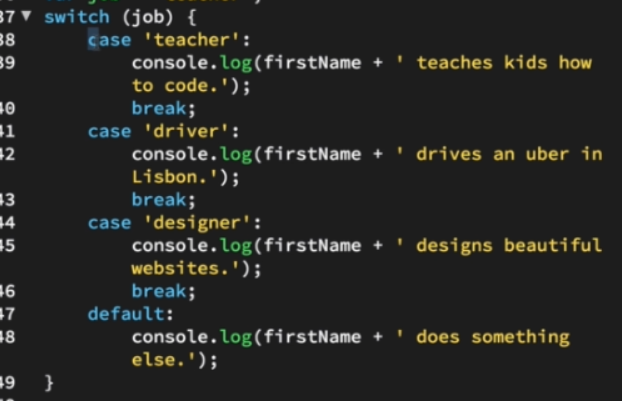


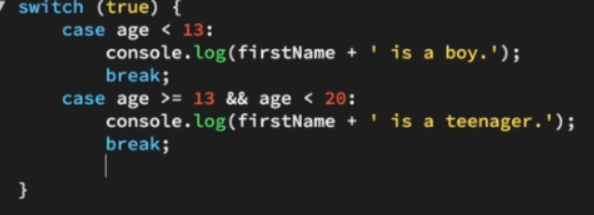
Resulat: boolean

1. If-else
2. Condition operator:

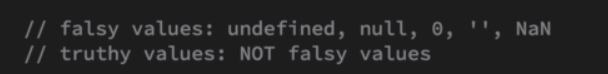


1. Switch

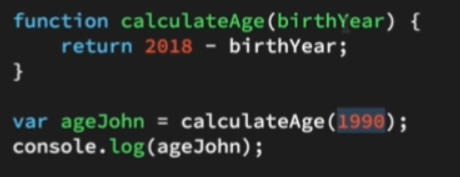




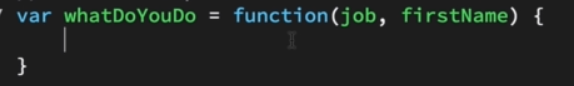
1. Value and true false



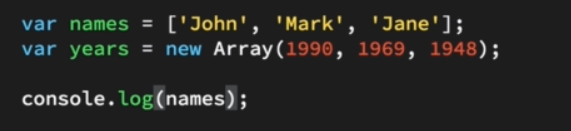
1. == and ===
2. Function

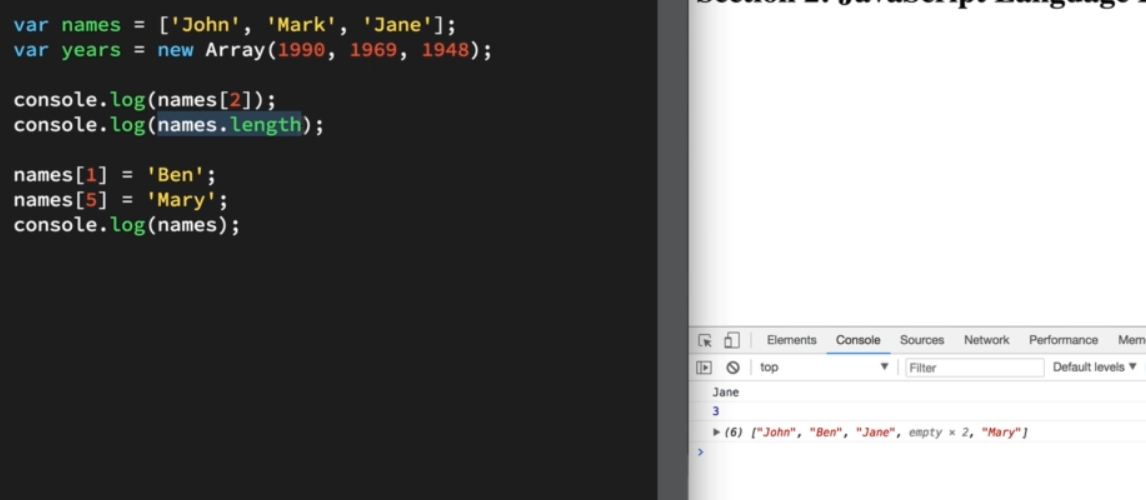
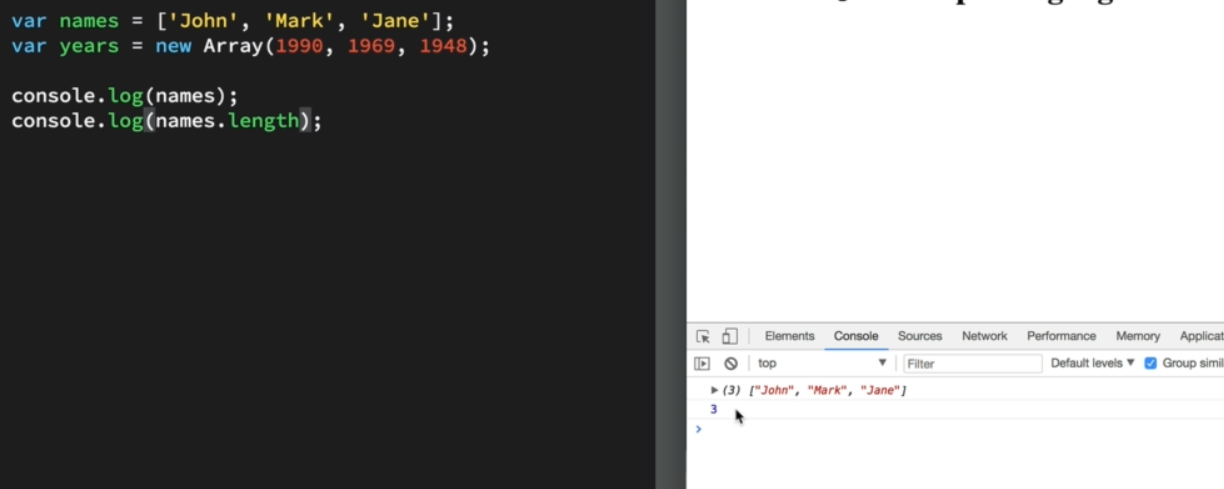


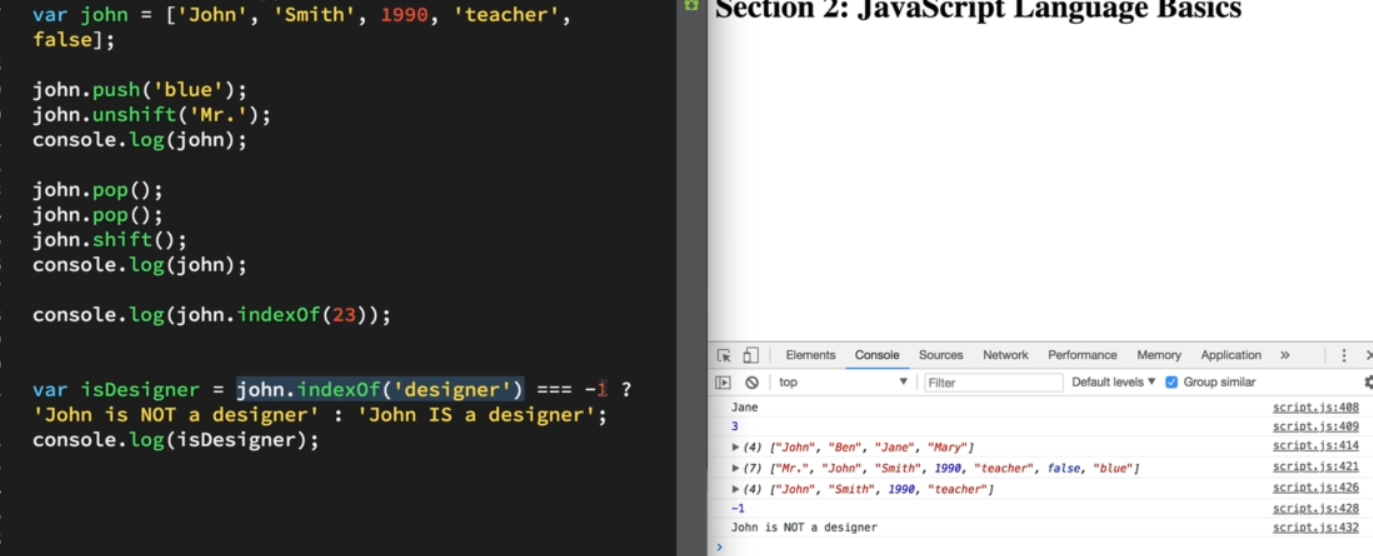
Chạm lệnh return -> thoát hàm



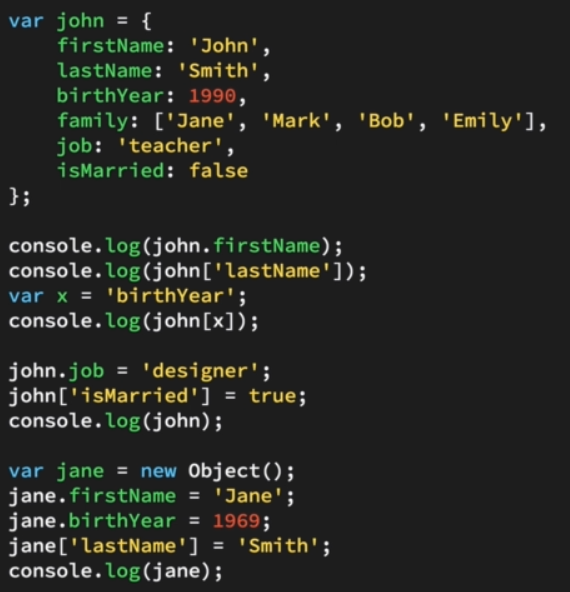
1. Array

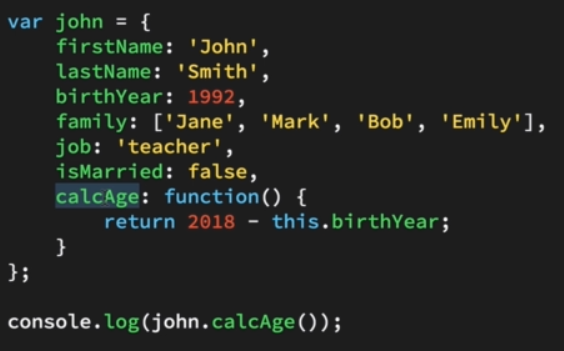






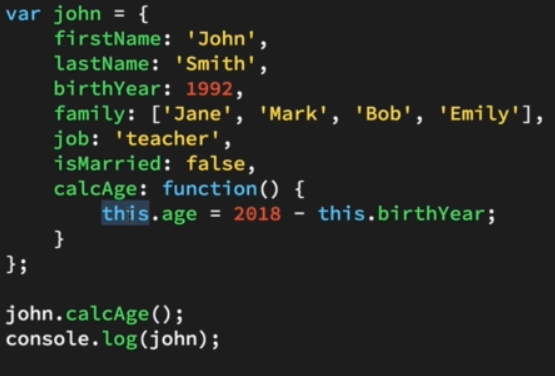
1. Object





Định nghĩa function trong object

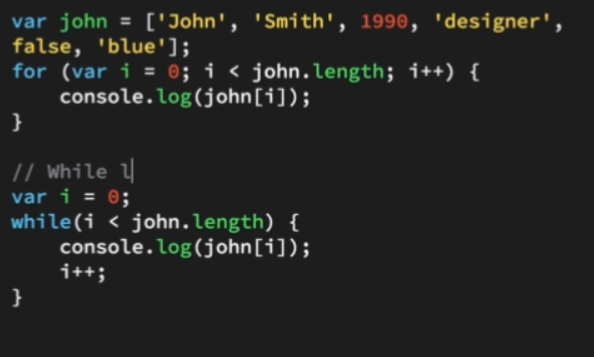
Sử dụng this



1. Loop

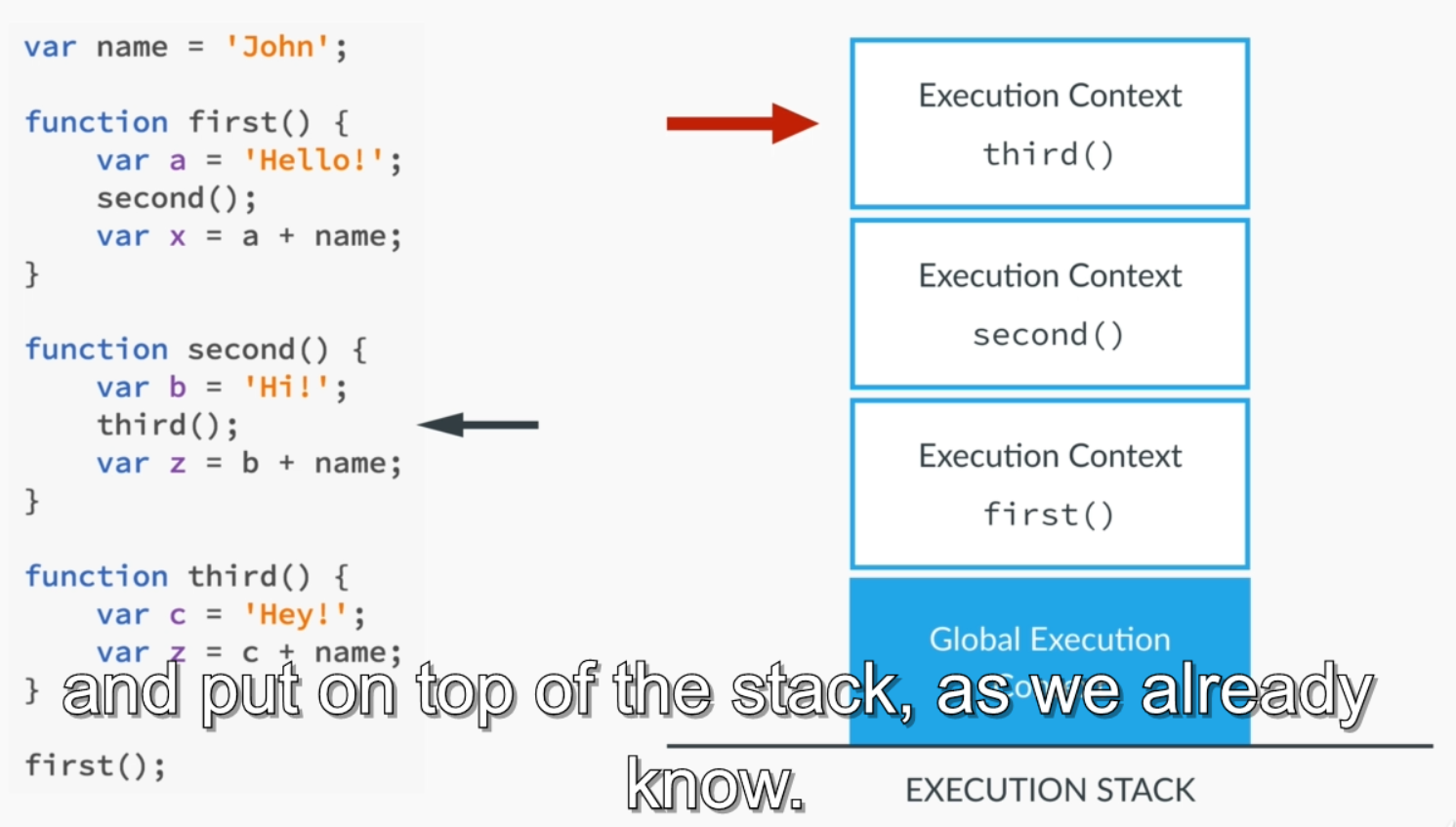
Continue and break

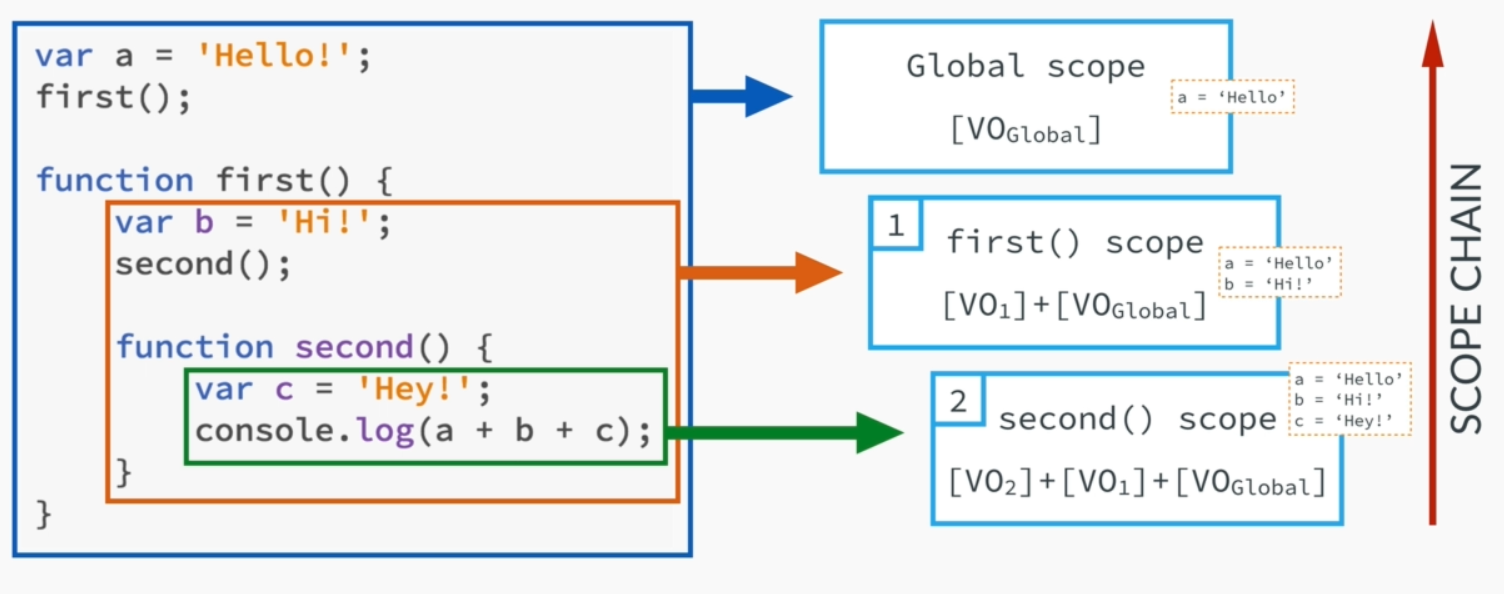
1. Funtion declation and function epression

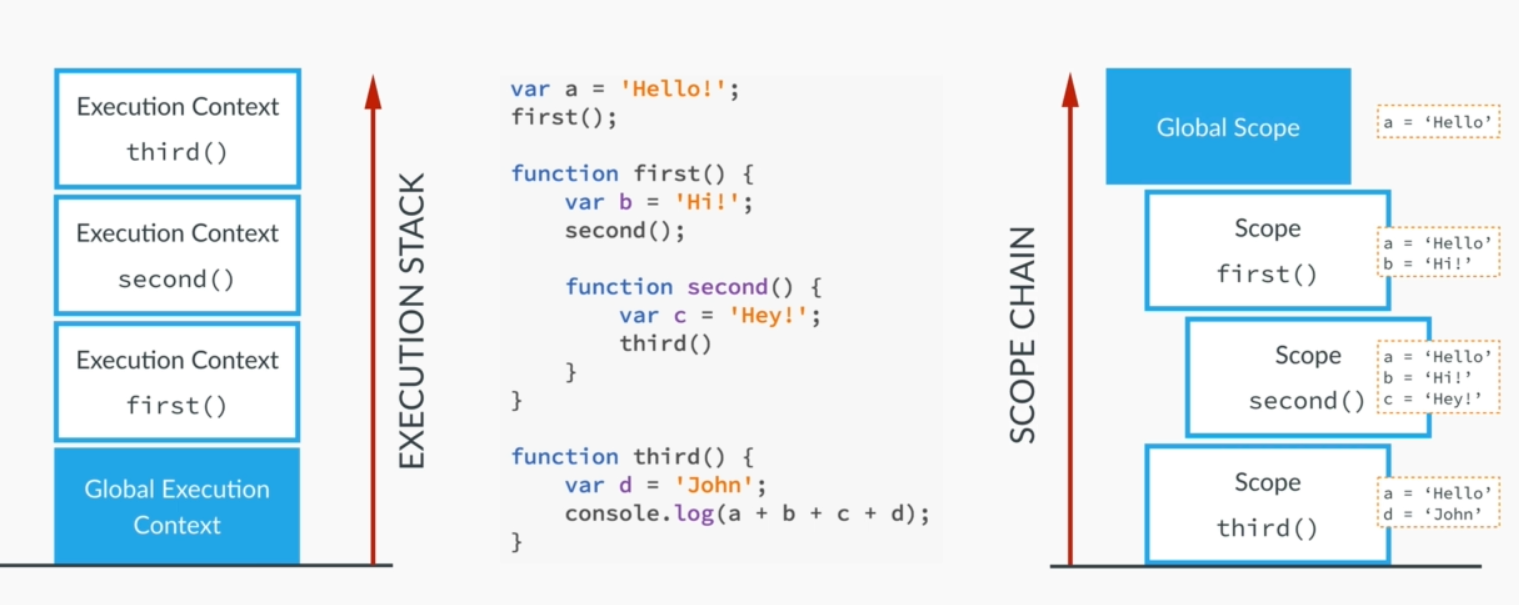


Với hàm có thể dùng trước rồi khai báo sau với decletion, nhưng với epression phải khai bái rồi mới dùng

1. Phạm vi biến







Hàm được khai báo( định nghĩa) trong hàm nào thì sẽ dùng được biến của hàm đó. Khai báo định nghĩa ngoài thì không dùng được

1. This

This trong main trỏ tới window

This trong hàm trỏ tới window

This trong method object trỏ thới object đó

This trong hàm trong method object trỏ tới window



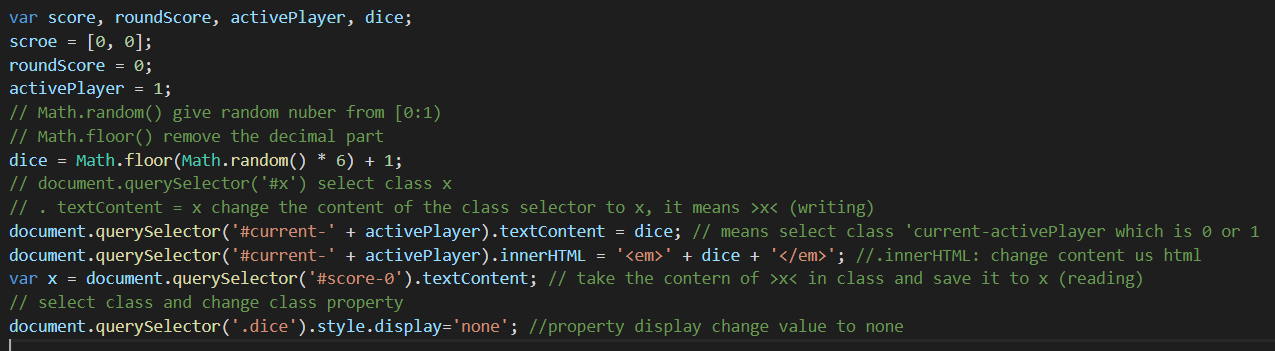
Borrow method

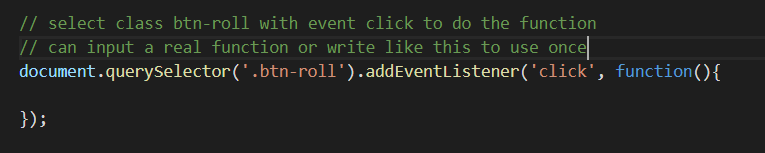
1. Math

Math.random() : random number form 0 to 1, [0,1)

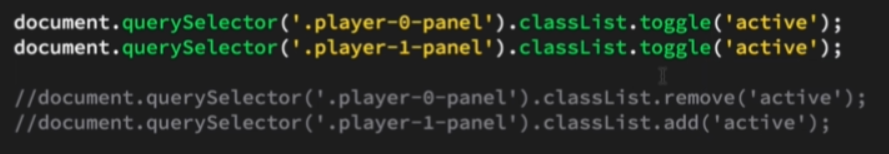
Math.floor(x) : remove the decima part

1. .DOM1





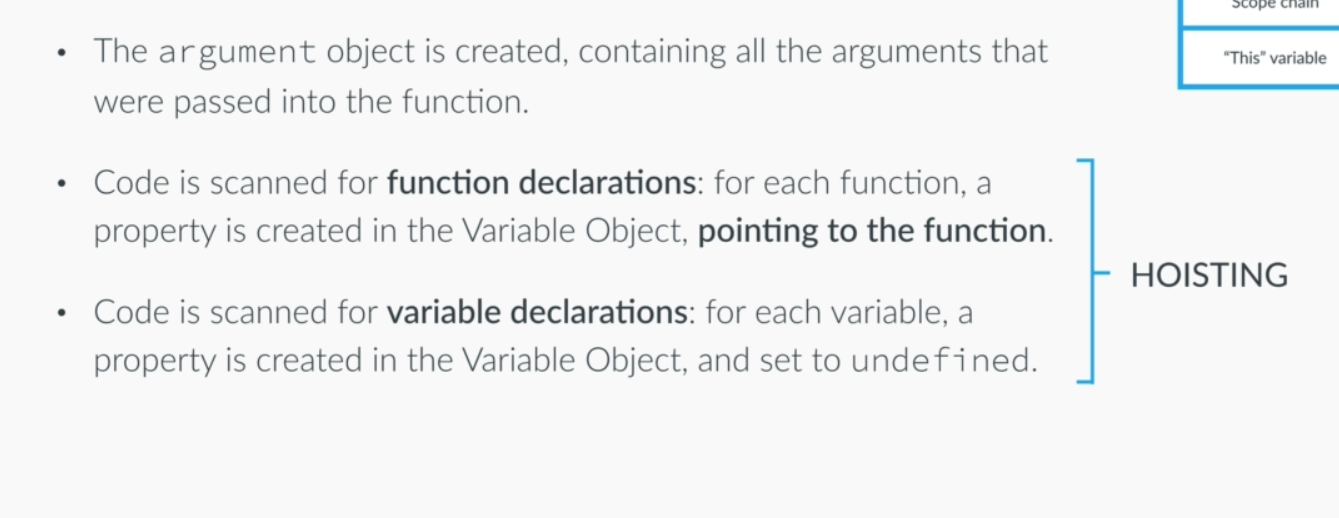






Take value from input element

// hoisting là các khai báo được kéo lên đầu chương trình và set underfine cho đến khi gặp câu lệnh gán giá trị thì nó được gán cho giá trị đó



Với hàm : hàm đó được định nghĩa như code ngay đầu chương trình

Với biến: hàm dó được định nghĩa và set về undefined cho đến khi gặp câu lệnh khai báo trong code hoặc gặp câu lệnh gán

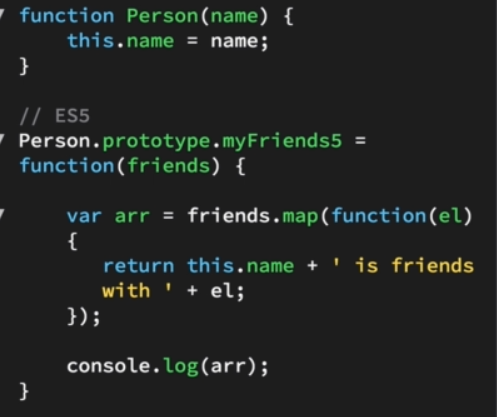
Funtion name() {} : function decletion: hoiting theo function

Var name = funtion(){} : funtion epression : hoisting theo biến

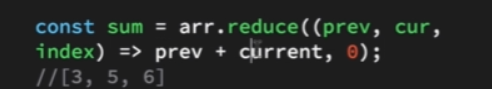


This trong object, trong method trỏ tới object

Tuy nhiên this trong hàm, có thể là hàm nằm trong method, hoặc hàm callback, hàm nằm trong object ( not method) trỏ tới window



Prototype kế thừa từ contructor của nó



Loop qua các phân từ mãng: arr.reduce((prev,cur,index)=>prev+curent,0)

Prev: giá trị của hàm ở previous ( chú ý là giá trị của hàm nghĩa là giá trị arr.reduce được return từ biểu thức)

Cur: phần từ mãng hiện tại

Index: index của phần tử mãng hiện tại

0: bắt đầu từ 0; ( nghĩa là khi bắt đầu thì prev=0)