

JavaScript



CYBERSOFT
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

NHU CẦU SỬ DỤNG VÒNG LẶP

```
console.log("Xin chào, Cybersoft")
```

```
console.log("Xin chào, Cybersoft")
```

```
console.log("Xin chào, Cybersoft")
```

```
console.log("Xin chào, Cybersoft")
```

```
console.log("Xin chào, Cybersoft")
```

```
console.log("Xin chào, Cybersoft")
```

Xin chào, Cybersoft lần: 0
Xin chào, Cybersoft lần: 1
Xin chào, Cybersoft lần: 2
Xin chào, Cybersoft lần: 3
Xin chào, Cybersoft lần: 4
Xin chào, Cybersoft lần: 5

NHU CẦU SỬ DỤNG VÒNG LẶP



```
output

> Điểm lần này của bạn bao nhiêu ? 4
> Điểm lần này của bạn bao nhiêu ? 2
> Điểm lần này của bạn bao nhiêu ? 3
> Điểm lần này của bạn bao nhiêu ? 7

> CHÚC MỪNG BẠN ĐÃ ĐẬU
```

VÒNG LẶP

loop (điều kiện)

{

\\ các dòng lệnh xử lý

}



CÁC LOẠI VÒNG LẶP

Vòng lặp while

Vòng lặp do....while

Vòng lặp for

Vòng lặp while

Không biết chính xác số lần lặp

Cú pháp vòng lặp **while**

```
while (điều kiện)
```

```
{
```

```
// các dòng lệnh xử lý
```

```
}
```

Cách hoạt động của vòng lặp while

Bước lặp 1

1

khởi tạo giá trị cho biến bước nhảy

2 while (điều kiện) ✓

{

3 // các dòng lệnh xử lý

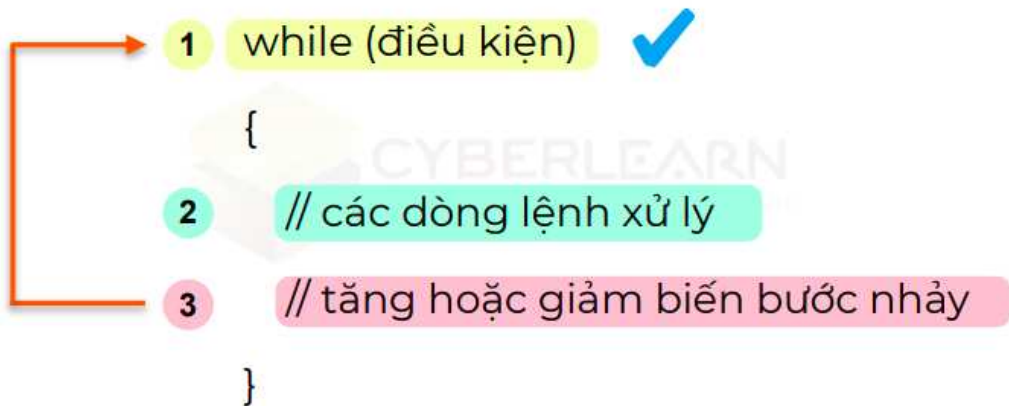
4 // tăng hoặc giảm biến bước nhảy

}

Cách hoạt động của vòng lặp while

Bước lặp 2

Giá trị khởi gán hoặc tạo biến bước nhảy



Cách hoạt động của vòng lặp while

Bước lặp 3

khởi tạo giá trị cho biến bước nhảy



1 while (điều kiện) ✗

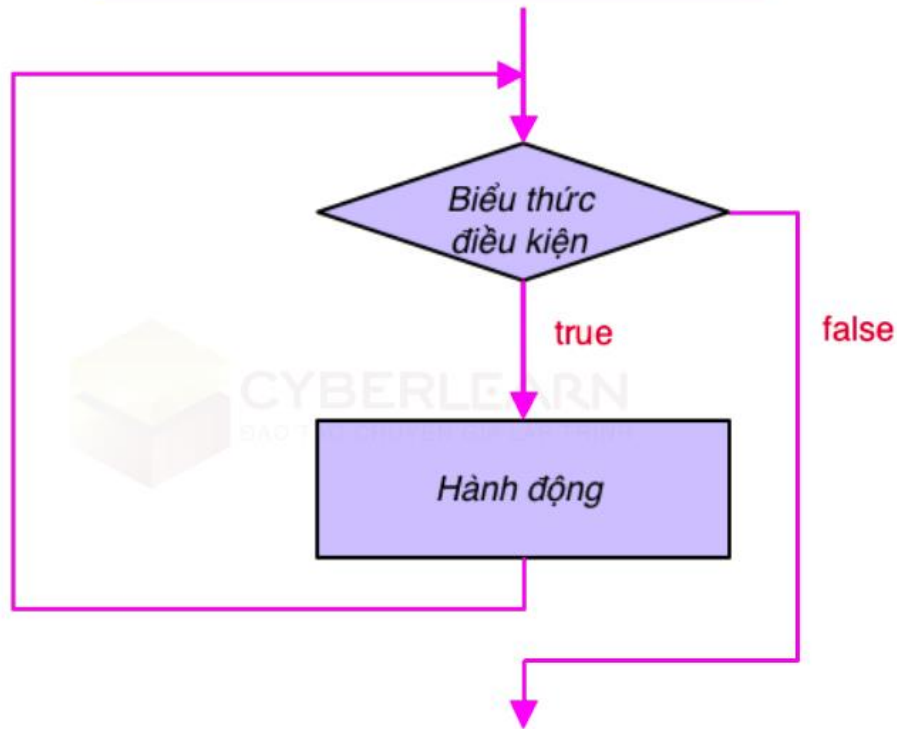
{

// các dòng lệnh xử lý

// tăng hoặc giảm biến bước nhảy

}

Lưu đồ hoạt động của vòng lặp while



DEMO VÒNG LẶP WHILE

- ❖ Nhập vào 1 số nguyên chẵn, chia dần số đó cho 2 và lặp lại việc đó đến khi nào giá trị nhỏ hơn hoặc bằng 1. In ra giá trị của mỗi lần lặp.

Nhập vào 1 số

Submit

count: 1 - num: 10

count: 2 - num: 5

count: 3 - num: 2

count: 4 - num: 1

```
<div class="container">
  <div class="row">
    <div class="col-sm-5">
      <div class="form-group">
        <label>Nhập vào 1 số</label>
        <input id="txtNum" type="text" class="form-control" />
        <button class="btn btn-success my-2" onclick="submit()">
          Submit
        </button>
        <div
          id="noti"
          class="alert alert-success"
          style="display: none"
        ></div>
      </div>
    </div>
  </div>
</div>
```

DEMO VÒNG LẶP WHILE

```
function submit() {  
    var num = parseInt(document.getElementById("txtNum").value);  
    var count = 0;  
    var contentHTML = "";  
  
    while (num > 1) {  
        count++;  
        num = parseInt(num / 2);  
        contentHTML += "<div>count: " + count + " - num: " + num + "</div>";  
        console.log("num: " + num, "-", "count: " + count);  
    }  
  
    document.getElementById("noti").style.display = "block";  
    document.getElementById("noti").innerHTML = contentHTML;  
}
```

num	num>1	count++	num = num/2
20	true	1	10
10	true	2	5
5	true	3	2
2	true	4	1
1	false	-	-

Cú pháp vòng lặp **do...while**

do

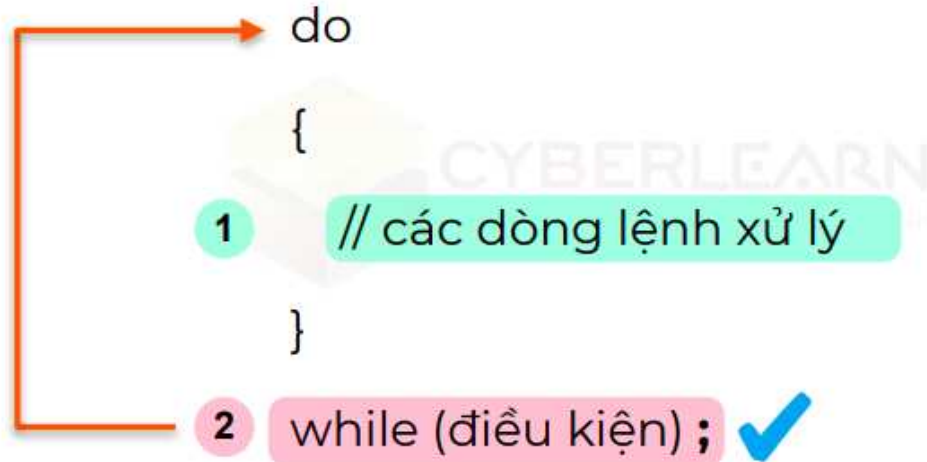
{

// các dòng lệnh xử lý

}

while (điều kiện) ;

Cách hoạt động của vòng lặp **do...while**



Cách hoạt động của vòng lặp do...while



```
do
```

```
{
```

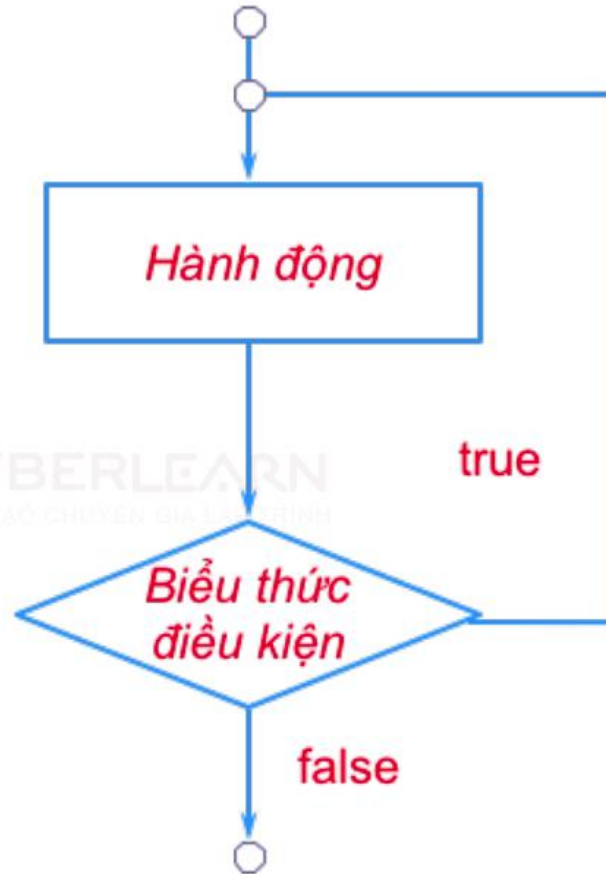
```
// các dòng lệnh xử lý
```

```
}
```

```
while (điều kiện);
```



Lưu đồ hoạt động của vòng lặp `do...while`



DEMO VÒNG LẶP DO WHILE

Tính tổng số nhập vào

Submit

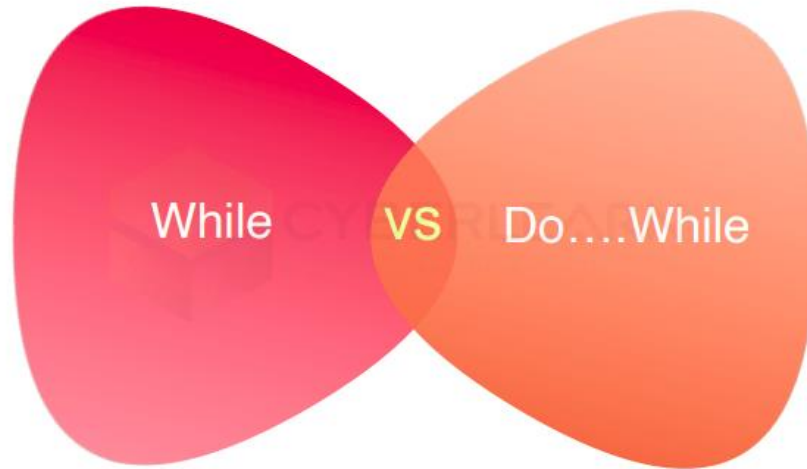
15

```
<div class="container">
  <div class="row">
    <div class="col-sm-5">
      <div class="form-group">
        <label>Tính tổng số nhập vào</label>
        <input id="txtNum" type="text" class="form-control" />
        <button class="btn btn-success my-2" onclick="submit()">
          Submit
        </button>
      </div>
      <div
        id="noti"
        class="alert alert-success"
        style="display: none"
      ></div>
    </div>
  </div>
</div>
```

```
function submit() {
  var num = parseInt(document.getElementById("txtNum").value);
  var i = 0;
  var total = 0;

  do {
    i++;
    total += i;
  } while (i < num);

  document.getElementById("noti").style.display = "block";
  document.getElementById("noti").innerHTML = total;
}
```



Vòng lặp While

```
while (điều kiện)
```

1

```
{
```

```
// các dòng lệnh xử lý
```

2

```
}
```

Vòng lặp Do...While

```
do
```

```
{
```

```
// các dòng lệnh xử lý
```

1

```
}
```

```
while (điều kiện);
```

2



Vòng lặp For

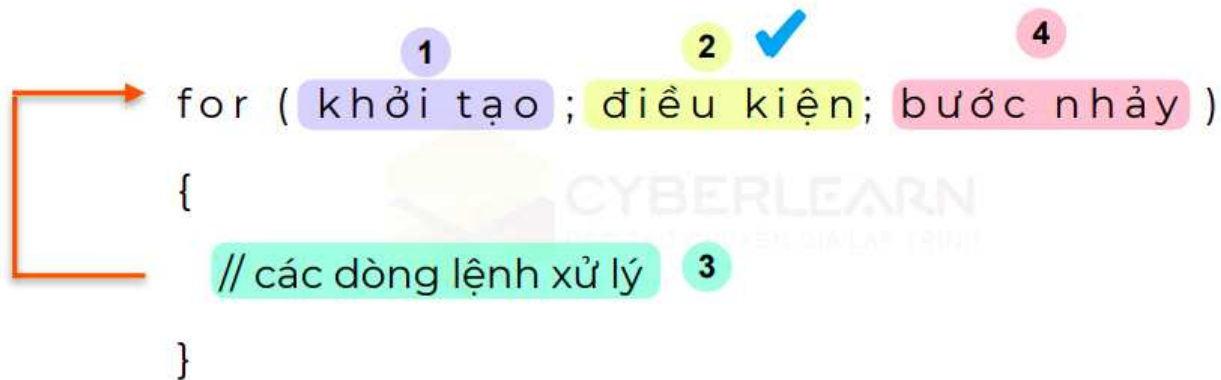
Cho biết trước số lần lặp lại

Cú pháp vòng lặp For

```
for ( khởi tạo ; điều kiện ; bước nhảy )  
{  
    // các dòng lệnh xử lý  
}
```

Cách hoạt động của vòng lặp For

Bước lặp 1

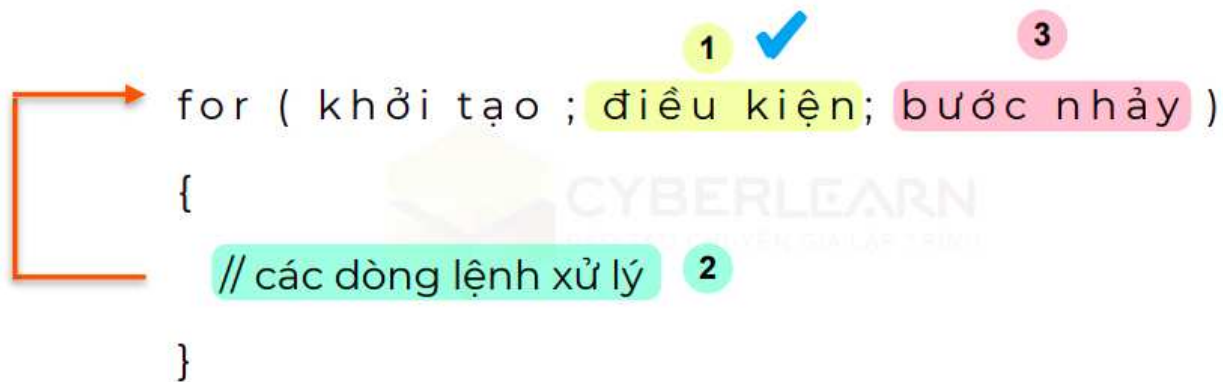


The diagram illustrates the components of a for loop. The code is: `for (khởi tạo ; điều kiện ; bước nhảy)`
`{`
`// các dòng lệnh xử lý`
`}`
An orange arrow on the left indicates the loop's flow. Numbered callouts identify: 1. 'khởi tạo' (initialization) in a purple box; 2. 'điều kiện' (condition) in a yellow box with a blue checkmark above it; 4. 'bước nhảy' (increment) in a pink box. A light green box containing 'CYBERLEARN' is in the background. A callout '3' points to the body of the loop, which contains the comment '// các dòng lệnh xử lý'.

```
for ( khởi tạo ; điều kiện ; bước nhảy )  
{  
    // các dòng lệnh xử lý  
}
```

Cách hoạt động của vòng lặp For

Bước lặp 2



The diagram illustrates the components of a for loop. The code is: `for (khởi tạo ; điều kiện; bước nhảy)`
`{`
`// các dòng lệnh xử lý`
`}`
Annotations:
1. A blue checkmark is placed above 'điều kiện' (condition).
2. A green circle with the number '2' is placed next to '// các dòng lệnh xử lý' (statements to be executed).
3. A pink circle with the number '3' is placed above 'bước nhảy' (step).
An orange bracket on the left side of the code block indicates the body of the loop.

```
for ( khởi tạo ; điều kiện; bước nhảy )  
{  
    // các dòng lệnh xử lý  
}
```


Cách hoạt động của vòng lặp For

Bước lặp 3

➔ ^{1 ✗} for (khởi tạo ; điều kiện; bước nhảy)
{
 // các dòng lệnh xử lý
}

MỘT SỐ GHI CHÚ

```
for ( khởi tạo ; điều kiện; bước nhảy )  
{  
    // các dòng lệnh xử lý  
}
```

```
for (var i = 0; i < 5; i++) {  
    console.log(i);  
}
```

Khởi tạo:

* Bắt đầu từ 0, 1, ...

Bước nhảy:

* ++, +=2, *=3, --, -=2

DEMO VÒNG LẶP FOR

```
for (var i = 0; i < 5; i++) {  
    console.log("*", i);  
}
```



CYBERSOFT
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

i = 0	i < 5	console.log("*")	i ++
0	true	console.log("*")	1
1	true	console.log("*")	2
2	true	console.log("*")	3
3	true	console.log("*")	4
4	true	console.log("*")	5
5	false	-	-

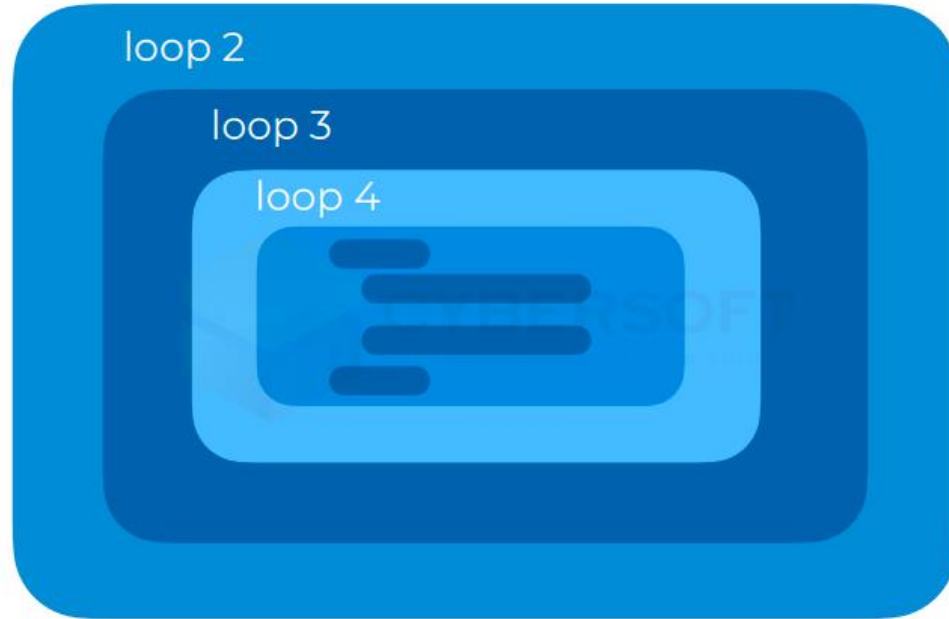
Luyện tập vòng lặp For

1. In tất cả số nguyên dương lẻ / chẵn nhỏ hơn 100. (Viết cả while và for) - Dùng bước nhảy và dùng % 2
2. Tính tổng các số chẵn từ 1 đến n. Với n nhập từ người dùng (While và for)
3. Có bao nhiêu số chia hết cho 3 từ 0-1000 (While và for)



Demo vòng lặp lồng

loop 1



Demo vòng lặp lồng

5 X 5

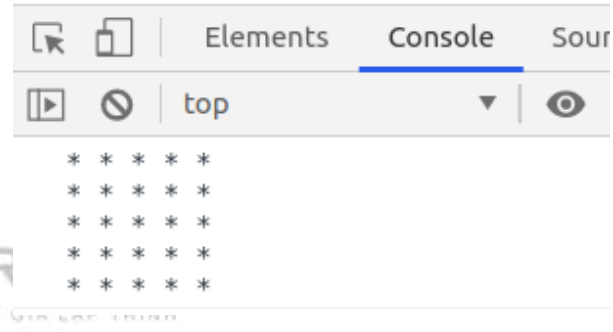
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*

DEMO VÒNG LẶP LỒNG

```
var content = "";

for (i = 0; i < 5; i++) {
  for (var j = 0; j < 5; j++) {
    content += "* ";
  }
  content += "\n";
}

console.log(content);
```



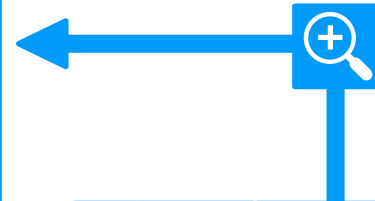
DEMO VÒNG LẶP LỒNG

j = 0	j < 5	console.log("*")	j ++
0	true	console.log("*")	1
1	true	console.log("*")	2
2	true	console.log("*")	3
3	true	console.log("*")	4
4	true	console.log("*")	5
5	false	-	-

```

* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```



i = 0	j < 5	Inner loop	console.log("\n")	j ++
0	true	✓	console.log("\n")	1
1	true	✓	console.log("\n")	2
2	true	✓	console.log("\n")	3
3	true	✓	console.log("\n")	4
4	true	✓	console.log("\n")	5
5	false	-	-	-

while (điều kiện)

{

if(<điều kiện> {

// các lệnh xử lý

break;



}

// các lệnh xử lý

}

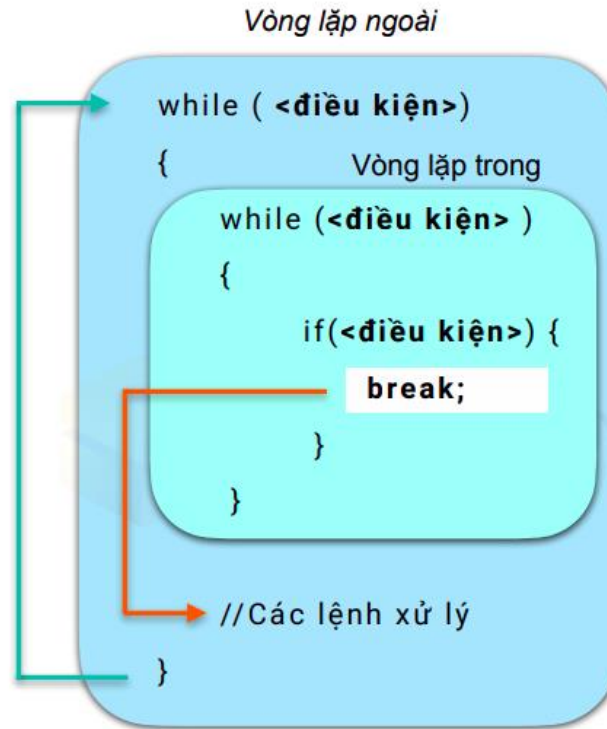
// các dòng lệnh xử lý tiếp theo

Lệnh Break

```
1  var i = 0;
2
3  while (true) {
4      console.log(i);
5      i++;
6      if (i > 5) {
7          break;
8      }
9  }
```



0
1
2
3
4
5



Lệnh Continue

index.html

JS main.js



main.js

```
for (i = 0; i < 10; i++) {  
  if (i === 3) {  
    continue;  
  }  
  console.log("The number is " + i);  
}
```

FT
RINH

The number is 0

The number is 1

The number is 2

The number is 4

The number is 5

The number is 6


The number is 7

The number is 8

The number is 9

Lệnh Continue

```
for(khởi tạo; điều kiện; bước nhảy){  
    if(điều kiện){  
        //Các lệnh xử lý  
        continue;  
    }  
    //Các lệnh xử lý  
}
```



LỆNH RETURN

JS main.js X <> index.html

JS main.js > ...

```
1  function demo() {  
2      //hàm demo dừng thực khi gặp lệnh return"  
3      console.log("code trước lệnh return sẽ được thực thi");  
4      if (1 === 1) return;  
5      console.log("Những dòng code bên dưới return sẽ không được thực thi");  
6  }  
7  
8  demo();|
```

Lệnh Return

```
function tenham(){  
    for(khởi tạo; điều kiện; bước nhảy){  
        if(điều kiện){  
            //Các lệnh xử lý  
            return;  
        }  
        //Các lệnh sau return  
    }  
}
```

→ //Code ngoài hàm

BÀI TẬP NỘI: LUYỆN VÒNG LẶP

1. Tìm số nguyên dương nhỏ nhất sao cho:
$$1 + 2 + \dots + n > 10000$$
2. Viết chương trình nhập vào 2 số x, n tính tổng: $S(n) = x + x^2 + x^3 + \dots + x^n$ (Sử dụng vòng lặp và hàm)
3. Nhập vào n . Tính giai thừa $1*2*\dots n$
4. Hãy viết chương trình khi click vào button sẽ in ra 10 thẻ div. Nếu div nào vị trí chẵn thì background màu **đỏ** và lẻ thì background màu **xanh**.
5. Viết chương trình có một ô input, một button. Khi click vào button thì in ra các số nguyên tố từ 1 tới giá trị của ô input