

4.1/ Vòng lặp trong Javascript

4.1.2) Cách sử dụng vòng lặp for trong JavaScript

- Vòng lặp for dùng để lặp lại việc thực thi một đoạn mã nào đó một số lần. **Cú pháp:**

```
for(biểu thức 1; biểu thức 2; biểu thức 3){  
    //Đoạn mã mà bạn muốn được thực thi nhiều lần  
}
```

Trong đó:

- **Biểu thức 1** thường là một câu lệnh khai báo biến (biến này được dùng để tham gia vào biểu thức 2)
- **Biểu thức 2** là một biểu thức điều kiện (nếu điều kiện đúng thì đoạn mã sẽ được thực thi, nếu điều kiện sai thì vòng lặp kết thúc)
- **Biểu thức 3** thường là một biểu thức làm thay đổi giá trị của biến được khai báo trong biểu thức 1 (mục đích là để cho biểu thức điều kiện dần trở nên bị SAI, giúp vòng lặp được kết thúc)

4.1/ Vòng lặp trong Javascript

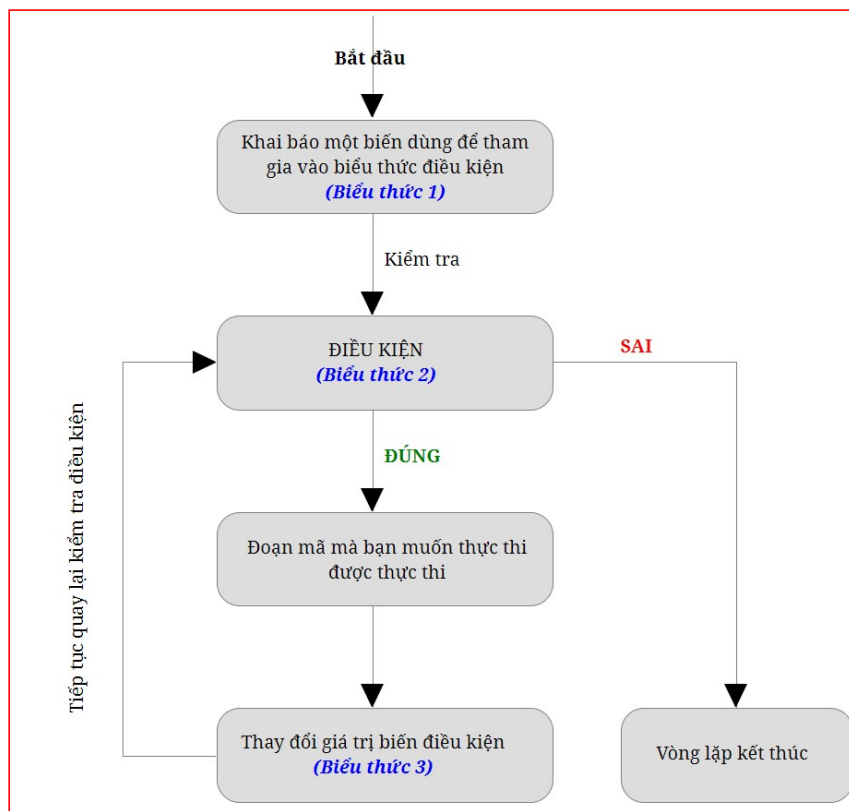
4.1.1) Vòng lặp là gì?

- Trong JavaScript, vòng lặp là một loại cú pháp giúp *ta lặp lại việc thực thi một đoạn mã nhiều lần*.
- Ví dụ, nếu ta muốn hiển thị lên màn hình **1000** dòng chữ "**Lập Trình Web**" thì đáng ra phải gõ **1000** câu lệnh `document.write("<p>Lập Trình Web</p>")`. Tuy nhiên, với việc sử dụng vòng lặp thì chỉ cần đến ba câu lệnh như bên dưới là đủ.

```
<script>
  for(var i=1; i<=1000; i++){
    document.write("<p>Lập Trình Web</p>");
  }
</script>
```

4.1/ Vòng lặp trong Javascript

4.1.1) Vòng lặp là gì?



```
<!DOCTYPE html>
<html>
<body>
  <script>
    for(var i=1; i<=3; i++){
      document.write("<p>Lập Trình Web</p>");
    }
  </script>
</body>
</html>
```

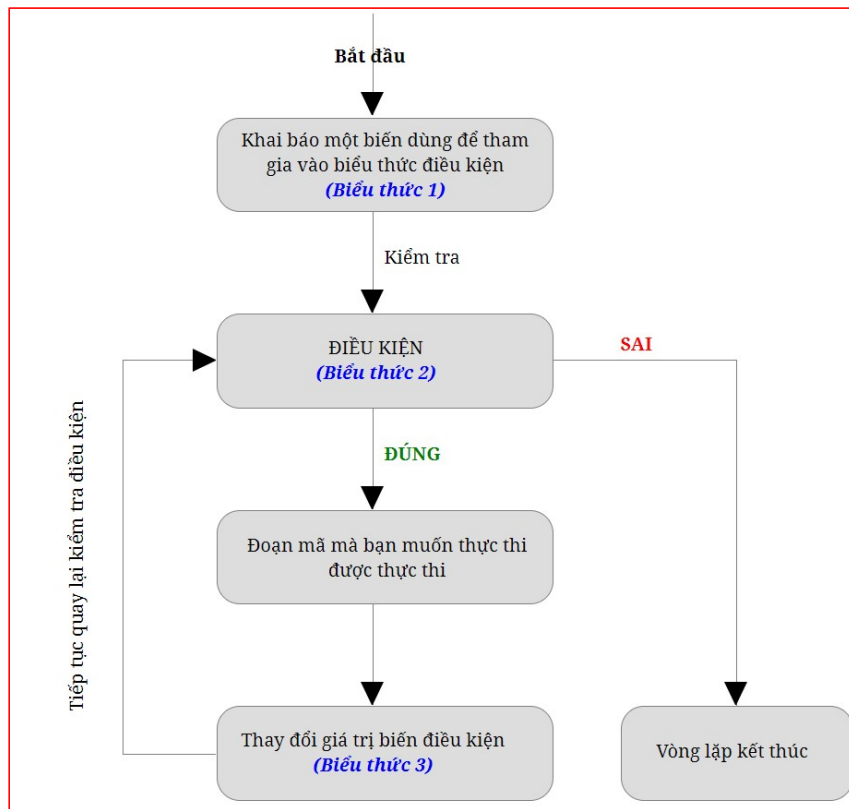
Lập Trình Web

Lập Trình Web

Lập Trình Web

4.1/ Vòng lặp trong Javascript

4.1.1) Vòng lặp là gì?



```
<!DOCTYPE html>
<html>
<body>
  <script>
    for(var i=9; i>1; i--){
      document.write("Số: " + i + "<br>");
    }
  </script>
</body>
</html>
```

Số: 9
Số: 8
Số: 7
Số: 6
Số: 5
Số: 4
Số: 3
Số: 2

4.1/ Vòng lặp trong Javascript

4.1.1) Vòng lặp for lồng nhau

- Thực chất vòng lặp for lồng nhau chỉ là cách sử dụng nâng cao của vòng lặp for thông thường để giúp cho số lần lặp được tăng theo cấp số nhân.
- Vòng lặp con được đặt vào bên trong vòng lặp cha. Khi điều kiện của vòng lặp cha đúng thì vòng lặp con sẽ được thực thi.

```
for(biểu thức 1; biểu thức 2; biểu thức 3){  
    for(biểu thức 1; biểu thức 2; biểu thức 3){  
        //Đoạn mã mà bạn muốn được thực thi nhiều lần  
    }  
}
```

4.1/ Vòng lặp trong Javascript


4.1.1) Vòng lặp for lồng nhau

- Ví dụ : Hiển thị ba dòng dãy số 1 2 3 4 5 6 7 8 9.

```
<!DOCTYPE html>
<html>
<body>

  <script>
    for(i = 1; i < 4; i++){
      var text = "";
      for(j = 1; j < 10; j++){
        text = text + j + " ";
      }
      document.write(text + "<br>");
    }
  </script>

</body>
</html>
```

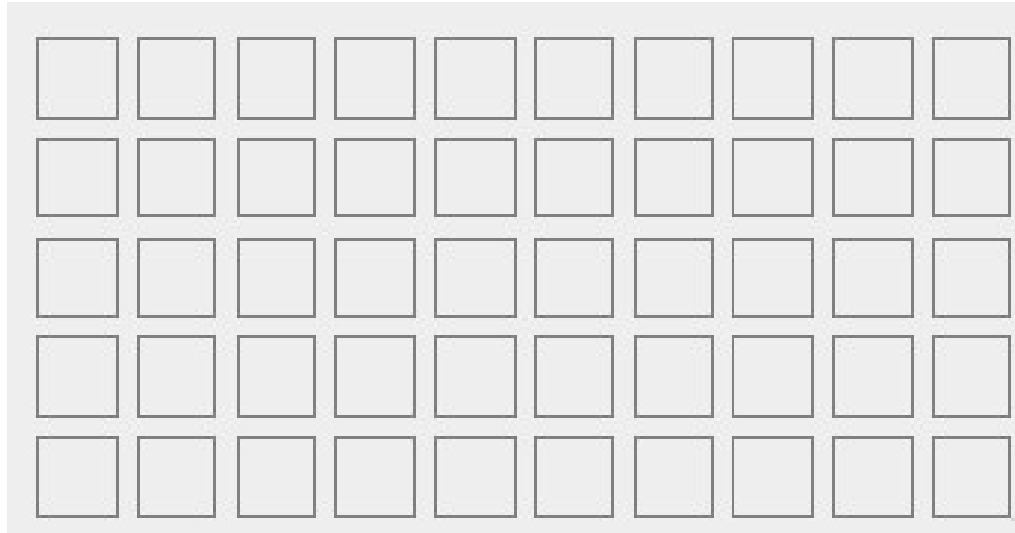


1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9
1 2 3 4 5 6 7 8 9

4.1/ Vòng lặp trong Javascript

4.1.1) Vòng lặp for lồng nhau

- Ví dụ : - Hiển thị một ma trận ô vuông giống hình bên dưới:



4.1/ Vòng lặp trong Javascript

4.1.1) Vòng lặp for lồng nhau

- Ví dụ : - Hiển thị một ma trận ô vuông giống hình bên dưới:

```
<!DOCTYPE html>
<html>
<body>

  <style>
    .square
    {
      height:20px; width:20px; float:left; border:1px solid gray;
      margin-left:5px; margin-bottom:5px;
    }
  </style>

  <script>
    for(var i = 0; i < 5; i++){
      for(var j = 0; j < 10; j++){
        document.write("<div class='square'></div>");
      }
      document.write("<div style='clear:both'></div>");
    }
  </script>

</body>
</html>
```


4.1/ Vòng lặp trong Javascript

4.1.1) Lệnh break

Lệnh **break** thường được đặt vào bên trong vòng lặp **for**. Khi lệnh **break** được thực thi, vòng lặp sẽ kết thúc mặc cho điều kiện của vòng lặp có còn đúng hay không:

```
<!DOCTYPE html>
<html>
<body>

  <script>
    var number = 1;
    for(var i = 1; i <= 10;i++){
      document.write(number + "<hr>");
      number++;
      if(number==5){
        break;
      }
    }
  </script>

</body>
</html>
```



1

2

3

4

4.1/ Vòng lặp trong Javascript

4.1.1) Lệnh continue

Lệnh **continue** thường được đặt vào bên trong vòng lặp **for**. Khi lệnh **continue** được thực thi, những câu lệnh còn lại của lần lặp hiện tại sẽ bị bỏ qua:

```
<!DOCTYPE html>
<html>
<body>

  <script>
    for(var i = 1; i <= 10; i++){
      if(i==2 || i==5 || i==9){
        continue;
      }
      document.write(i + "<br>");
    }
  </script>
</body>
</html>
```

1
3
4
6
7
8
10

4.1/ Vòng lặp trong Javascript

4.1.1) Lặp qua một lần các phần tử của mảng

```
<!DOCTYPE html>
<html>
<body>
  <script>

    var mobile = ["HTC", "Nokia", "SamSung", "LG", "Apple", "Lenovo"];

    for(var i = 0; i < mobile.length; i++){
      document.write(mobile[i] + "<br>");
    }

  </script>
</body>
</html>
```

4.1/ Vòng lặp trong Javascript

4.1.1) Cách sử dụng vòng lặp for in trong JavaScript

```
<!DOCTYPE html>
<html>
<body>
  <script>

    var SinhVien = {
      name:"Giang Hào Côn",
      year:1971,
      gender:"Nam",
      hometown:"Sài Gòn"
    }

    for(x in SinhVien){
      document.write(SinhVien[x]);
      document.write("<hr>");
    }

  </script>
</body>
</html>
```

Giang Hào Côn

1971

Nam

Sài Gòn

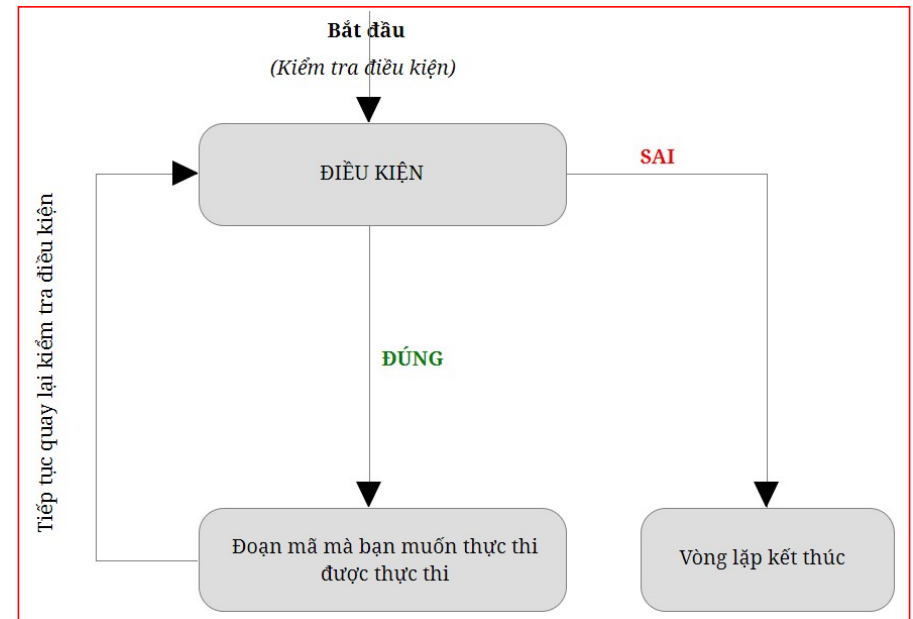
4.1/ Vòng lặp trong Javascript

4.1.2) Vòng lặp while trong JavaScript

Vòng lặp **while** dùng để lặp lại việc thực thi một đoạn mã nếu điều kiện mà ta đưa ra vẫn còn đúng.

Cú pháp:

```
while(điều kiện){  
    //Đoạn mã mà bạn muốn thực thi  
}
```



4.1/ Vòng lặp trong Javascript

4.1.2) Vòng lặp while trong JavaScript

Ví dụ 1: Dùng vòng lặp while để hiển thị dãy số tăng dần từ một đến chín

```
<!DOCTYPE html>
<html>
<body>

  <script>

    var i = 1;

    while(i < 10){
      document.write(i + "<br>");
      i++;
    }

  </script>

</body>
</html>
```

1
2
3
4
5
6
7
8
9

4.1/ Vòng lặp trong Javascript

4.1.2) Vòng lặp while trong JavaScript

Ví dụ 2: Dùng vòng lặp while để hiển thị dãy số giảm dần từ tám xuống hai

```
<!DOCTYPE html>
<html>
<body>

  <script>

    var i = 8;

    while(i > 1){
      document.write(i + "<br>");
      i--;
    }

  </script>

</body>
</html>
```

8
7
6
5
4
3
2

4.1/ Vòng lặp trong Javascript

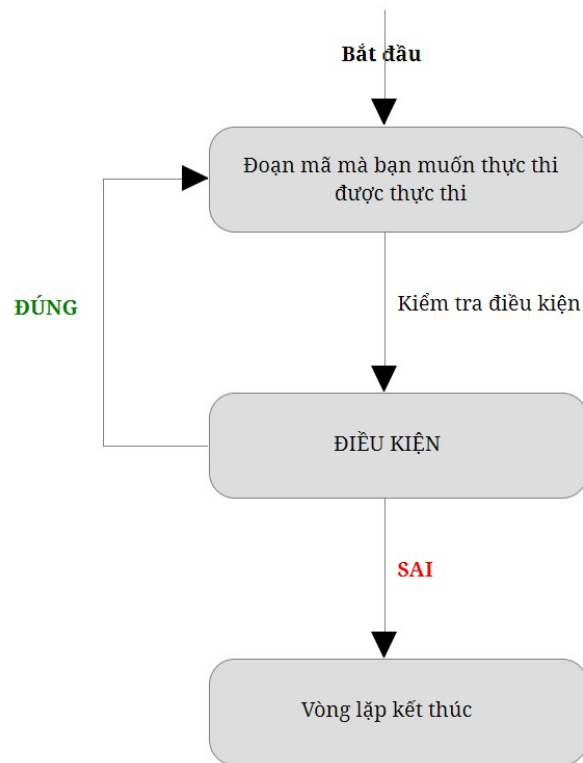
4.1.3) Vòng lặp do while trong JavaScript

- Tương tự như vòng lặp while, vòng lặp **do while** dùng để lặp lại việc thực thi một đoạn mã nếu điều kiện mà ta đưa ra vẫn còn đúng.
- Tuy nhiên, đối với vòng lặp do while thì ở lần đầu tiên đoạn mã sẽ được thực thi luôn mà không cần phải kiểm tra điều kiện (đó chính là điểm khác nhau giữa vòng lặp while và do while)
- **Cú pháp:**

```
do{  
    //Đoạn mã mà bạn muốn thực thi  
}while(điều kiện);
```


4.1/ Vòng lặp trong Javascript

4.1.3) Vòng lặp do while trong JavaScript



```
<!DOCTYPE html>
<html>
<body>

  <script>

    var i = 1;

    do{
      document.write(i + "<br>");
      i++;
    }while(i < 10);

  </script>

</body>
</html>
```

1
2
3
4
5
6
7
8
9

4.1/ Vòng lặp trong Javascript

4.1.4) Điểm khác nhau giữa vòng lặp while và do while

```
<script>
  var i = 11;
  while(i < 10){
    document.write(i);
    i++;
  }
</script>
```

```
<script>
  var i = 11;
  do{
    document.write(i);
    i++;
  }while(i < 10);
</script>
```

Bài 01

In bảng cửu chương

Nhập cửu chương	<input type="text" value="k"/>	<input type="button" value="In"/>
Số nhập không hợp lệ!		

Khi người dùng nhập 1 số và bấm nút “In” thì hiện ra bảng cửu chương tương ứng

In bảng cửu chương

Nhập cửu chương	<input type="text" value="9"/>	<input type="button" value="In"/>
$9 * 1 = 9$ $9 * 2 = 18$ $9 * 3 = 27$ $9 * 4 = 36$ $9 * 5 = 45$ $9 * 6 = 54$ $9 * 7 = 63$ $9 * 8 = 72$ $9 * 9 = 81$ $9 * 10 = 90$		

Bài 02

Mở rộng Bài 01, cho nhập số thứ X và Số thứ Y và bấm nút “In” thì sẽ in ra màn hình bảng cửu chương từ X đến Y.

Bài 03

Tìm các số chia hết cho 6 và 9

Nhập một số	<input type="text" value="90"/>	<input type="button" value="In"/>
Các số từ 1 đến 90 chia hết cho 6 và 9 là: 18, 36, 54, 72, 90.		

Bài 04

Tìm số đầu tiên chia hết cho 5 và 7

Nhập từ số	<input type="text" value="50"/>	đến số	<input type="text" value="105"/>	<input type="button" value="In"/>
Số đầu tiên nằm trong khoảng từ 50 đến 105 chia hết cho 5 và 7 là: 70				