

Bài: Bài tập về vòng lặp trong JavaScript

Xem bài học trên website để ủng hộ Kteam: [Bài tập về vòng lặp trong JavaScript](#)

Mọi vấn đề về lỗi website làm ảnh hưởng đến bạn hoặc thắc mắc, mong muốn khóa học mới, nhằm hỗ trợ cải thiện Website. Các bạn vui lòng phản hồi đến Fanpage [How Kteam](#) nhé!

Trong bài này, Kteam và bạn sẽ thực hiện một số bài tập củng cố kiến thức về [Vòng lặp trong JavaScript](#)

Để bài tập đạt hiệu quả tốt nhất bạn nên thực hiện theo các bước sau:

1. Đọc đề các câu hỏi và tự đưa ra đáp án/ lời giải của mình.
2. Tham khảo đáp án tại bài [Bài tập về Function trong JavaScript](#) và rà soát đáp án/ lời giải của bản thân
3. Xem video giải thích đáp án của bài này và note lại các ý chính để củng cố lại kiến thức.

Câu hỏi về vòng lặp trong JavaScript

Câu 1: Vòng lặp do-while và while khác nhau như thế nào ?

Câu 2: Viết một chương trình in ra các ước của n với n có sẵn.

Câu 3: Viết chương trình tính tổng các số chẵn, số lẻ từ 1 đến n (với n có sẵn).

Câu 4: Câu lệnh nào dưới đây tạo ra một vòng lặp vô tận (ta chưa xét trường hợp có break bên trong)?

- for(;;)
- while(false)
- for(let i = 1; i < 100; i++)

Câu 5: Viết một chương trình giúp thay đổi định dạng của 1 chuỗi theo quy tắc sau: **Kí tự đầu của từ được viết hoa, các kí tự còn lại viết thường.**
Đồng thời, sau câu phải đặt dấu chấm.

Ví dụ:

'we ARE kTeam' là 'We Are Kteam.'

'this is javascript' là 'This Is Javascript.'

Câu 6: Câu lệnh sau lặp bao nhiêu lần ?

Javascript:

```
for(let i = 0; i < 100; i++) {}
```

Đáp án Bài tập về câu lệnh điều kiện trong JavaScript

Bạn có thể xem chi tiết các câu hỏi ở [Bài tập về câu lệnh điều kiện trong JavaScript](#)

Đáp án Câu 1:

Về bản chất, ternary là một toán tử 3 ngôi và nó sẽ trả về một giá trị nào đó phụ thuộc vào điều kiện mà ta đưa cho nó. Nó có cú pháp như sau:

Javascript:

```
<condition> ? <value-1> : <value-2>
```

Nếu **<condition>** là **truthy**, thì lấy **<value-1>**, ngược lại lấy **<value-2>**

Ví dụ về ternary operator:

Javascript:

```
> a = 1
1
> a == 0 ? 'Kteam' : 'Kter'
'Kter'
> -a < 0 ? 'a is positive' : 'a is negative'
'a is positive'
```

Đáp án Câu 2:

Chương trình tham khảo:

Javascript:

```
// cho n là số bi mà Tun có
if(n >= 4) console.log('Cho anh hết !'); else {
  switch(n) {
    case 1: {
      console.log('Tun không cho');
      break;
    }
    case 2: {
      console.log('Mỗi người một nửa');
      break;
    }
    case 3: {
      console.log('Cho 1, giữ 2');
      break;
    }
  }
}
```

Đáp án Câu 3:

Chương trình sẽ so sánh 2 số a và b, sau đó chọn ra số lớn hơn:

Javascript:

```
if(a > b) console.log(a); else console.log(b);
```

Và ta có thể “tom gọn” đôi chút bằng cách sử dụng ternary operator:

Javascript:

```
a > b ? a : b;
```

Đáp án Câu 4:

Bài này thì gần tương tự như trên thôi, các bạn có thể tham khảo bài trên và comment bài làm của bạn vào phần Bình luận bên dưới nhé!

Đáp án Câu 5:

a. **Lệnh if-else:**

```
if (<condition>) <statement-1>; else <statement-2>
```

Trong đó:

- **<condition>**: thì vẫn giống như trên, chỉ khác là: nếu **<condition>** là **false** thì sẽ thực hiện **<statement-2>**
- **<statement-1>**: là lệnh (hoặc khối lệnh) sẽ được thực thi khi **<condition>** là **true**
- **<statement-2>**: là lệnh (khối lệnh) được thực hiện nếu **<condition>** là **false**.

b. **Lệnh switch-case:**

```
switch(<expression>) {
    case <value-1>:
        <statement-1>
        break;
    case <value-2>:
        <statement-2>
        break;
    case <value-3>:
        <statement-3>
        break;
    .....
    case <value-n>:
        <statement-n>
    default:
        <default-statement>
```

Trong đó:

- **<expression>**: là biểu thức, giá trị này sẽ được so sánh lần lượt với **<value>** của từng case, nếu như **<value>** mà bằng với **<expression>**, thì **<statement>** tương ứng được thực hiện
- **<value>**: là các giá trị được mang đi so sánh, lệnh (khối lệnh) bên trong nó sẽ được thực hiện nếu **<value>** bằng với **<expression>**.
- **<statement>**: là một lệnh hoặc một khối lệnh, sẽ được thực hiện nếu **<value>** tương ứng bằng với **<expression>**

- `<default-statement>`: là lệnh (khối lệnh) sẽ được thực hiện nếu `<expression>` khác với tất cả các `<value>`.

