

Bài: Bài tập về kiểu dữ liệu Boolean trong JavaScript

Xem bài học trên website để ủng hộ Kteam: [Bài tập về kiểu dữ liệu Boolean 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 [Kiểu dữ liệu Boolean 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 kiểu dữ liệu Symbol 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 kiểu dữ liệu Boolean

Câu 1: Cú pháp khởi tạo boolean là gì ?

Câu 2: Chỉ ra các trường hợp mà khi dùng constructor Boolean trả về true, false.

Câu 3: Toán tử quan hệ là gì ? Các loại, công dụng của từng loại biểu thức quan hệ.

Đáp án bài kiểu dữ liệu chuỗi trong JavaScript

Bạn có thể xem chi tiết các câu hỏi ở [Bài tập về kiểu dữ liệu chuỗi trong JavaScript](#)

Đáp án câu 1:

A. Các dấu hiệu đặc trưng của chuỗi:

- Được đặt trong cặp dấu `"`, ``` hoặc `'''`.
- Dùng để biểu thị một hoặc một nhóm các kí tự.
- Các kí tự bên trong chuỗi không thể bị thay đổi bằng cách gán
- Có thể truy xuất các kí tự thông qua chỉ số của nó

B. Một vài cách khởi tạo chuỗi trong Js

- Dùng constructor String
- Dùng phương thức toString cho giá trị cần khởi tạo
- Đặt các giá trị trong cặp `"`, `'''` hoặc ```

Đáp án câu 2:

Đáp án và giải thích:

- a. Nối các chuỗi lại với nhau à `'1212'`
- b. Thực hiện phép toán cộng giữa số và chuỗi à tương tự phép cộng chuỗi à `'1212'`
- c. Number-String nhân với số à tương tự như nhân 2 số với nhau à 144
- d. Một chuỗi thông thường nhân với số à NaN
- e. Một chuỗi thông thường cộng với số à nối chuỗi à `'a13'`
- f. Thực hiện phép nhân trước, ta được 36. Sau đó, nối `'aa'` vào à `'aa36'`

Đáp án câu 3:

Dưới đây là chương trình tham khảo:

- **Câu a:**

Javascript:

```
console.log('.'.repeat(9))
console.log('.'.repeat(4) + '+' + '.'.repeat(4))
console.log('.'.repeat(3) + '+' + '.'.repeat(3) + '+' + '.'.repeat(3))
console.log('.'.repeat(2) + '+' + '.'.repeat(5) + '+' + '.'.repeat(2))
console.log('.' + '+' + '.'.repeat(7) + '+')
console.log('.'.repeat(9))
```

- **Câu b:**

Javascript:

```
console.log('.'.repeat(12))
console.log('.'.repeat(3) + '+' + '.'.repeat(2) + '+' + '.'.repeat(2) + '+' + '.'.repeat(3))
console.log('.'.repeat(5) + '-' + '.'.repeat(2) + '+' + '.'.repeat(5))
console.log('.'.repeat(4) + '_' + '.'.repeat(4) + '+' + '.'.repeat(4))
console.log('.'.repeat(12))
```

Đáp án câu 4:

Kết quả của các phương thức được thể hiện trong phần code bên dưới:

Javascript:

```
> a = 'howKteam and kter'
'howKteam and kter'
> a.split('t') // Tách các phần từ nằm giữa các kí tự 't'
[ 'howK', 'eam and k', 'er' ]
> a.replace('a', '__') // thay thế 'a' bằng '__'
'howKteam__nd kter'
> a.slice(-13, 8) // Lấy các kí tự từ -13 đến 8 (tức là từ 4 đến 7)
'team'
```

Đáp án câu 5:

Kết quả của các phép so sánh:

Javascript:

```
> 'cha' < 'con' // ở kí tự thứ 2, 'h' bé hơn 'o'
true
> 'toan' < 'TOAN' // 't' lớn hơn 'T', nên trả về false.
false
> 'ton' == 'tON' // ở kí tự thứ 2, 'o' > 'O'
false
> 'anh' > 'em' // 'a' < 'e'    'anh' > 'em' = false
false
> 'small' >= 'big' // 's' > 'b'    'small' > 'big' = true
True
```

Đáp án câu 6:

Một vài đặc điểm cơ bản:

- Được dùng trong cặp ``

- Cho phép khởi tạo chuỗi trên nhiều dòng
- Cho phép nhúng các biến, biểu thức, hoặc một function vào trong chuỗi

