Javascript Strings

Lets start with how to create strings and check their equality.

In software development, working with strings is a common problem. We often read, process, and write text files, perform logging on the activities of a system, or analyze user input.

Learning how to perform string operations is essential in any programming language.

Creating JavaScript strings

Strings contain a sequence of characters. You can either use single quotes (e.g. `'Javascript in Practice'`) or double quotes (e.g. `"Javascript in Practice"`) to create strings.

```
let firstString  = "Hugo's dog",
    secondString = 'Hugo\'s dog';
```

In the console, strings are displayed with double quotes.

As you can see in the previous example, it is possible to use a single quote inside a double quote, and it is also possible to use a double code inside a single quote. If you want to use a single quote inside a single quote, you have to escape it with a backslash.

The `\` tells the JavaScript interpreter that the next character should be treated as a literal character, not as a meta character. Therefore, `\'` does not signal the start or the end of a string.

Side note: if you want to place a backslash in a string, you have to escape it in the form of \ \.

Equality of strings

Two strings are equal whenever their values are equal:

```javascript
let firstString  = "Hugo's dog",
    secondString = 'Hugo\'s dog';
console.log(firstString == secondString);
//true
console.log(firstString === secondString)
//true
console.log(firstString === 'Hugo\'s dog');
//true
```