Chapter 101: Evaluating JavaScript

Parameter	Details
string	The JavaScript to be evaluated.

In JavaScript, the `eval` function evaluates a string as if it were JavaScript code. The return value is the result of the evaluated string, e.g. `eval('2 + 2')` returns 4.

`eval` is available in the global scope. The lexical scope of the evaluation is the local scope unless invoked indirectly (e.g. **var** `geval = eval; geval(s);`).

The use of `eval` is strongly discouraged. See the Remarks section for details.

Section 101.1: Evaluate a string of JavaScript statements

```javascript
var x = 5;
var str = "if (x == 5) {console.log('z is 42'); z = 42;} else z = 0; ";

console.log("z is ", eval(str));
```

The use of `eval` is strongly discouraged. See the Remarks section for details.

Section 101.2: Introduction

You can always run JavaScript from inside itself, although this is **strongly discouraged** due to the security vulnerabilities it presents (see Remarks for details).

To run JavaScript from inside JavaScript, simply use the below function:

```javascript
eval("var a = 'Hello, World!'");
```

Section 101.3: Evaluation and Math

You can set a variable to something with the `eval()` function by using something similar to the below code:

```javascript
var x = 10;
var y = 20;
var a = eval("x * y") + "<br>";
var b = eval("2 + 2") + "<br>";
var c = eval("x + 17") + "<br>";

var res = a + b + c;
```

The result, stored in the variable `res`, will be:

> 200
> 4
> 27

The use of `eval` is strongly discouraged. See the Remarks section for details.