



Rounding numbers in Javascript

There are times when we need to round numbers to show in a web page. For example, we might want to round a number to two decimal places when showing currency values.

It is tempting just to truncate the values but that has its drawbacks. You may lose precision if you truncate the value without rounding it.

Enter `toFixed()`. This method takes a number and returns the rounded value to the number of places specified in the parameter.

The code below does the following:

1. set up a constant holding the number we want to round
2. use the `toFixed()` method to round the number to three decimal places
3. log the result to console
4. log the type of object that the result is

```
const num1 = 7.74325156;  
const result = num1.toFixed(3);  
console.log(result); // => 7.743  
console.log(typeof result); // => string
```

We get a surprise. Even though we used a number, running it through `toFixed()` returns a string.

To keep value as a number after running it through `toFixed()`, we use the `Number()` function.

The `Number` function is part of the [Number](#) primitive wrapper object used to represent and manipulate numbers. In this case we're converting the string `result2` to a number.

```
const num2 = 7.74325156;  
const result2 = Number(num2.toFixed(3));
```

```
console.log(result); // => 7.743  
console.log(typeof result2); // => number
```