1- CONVENTIONS As every language, JavaScript has many code style guides. Maybe the most used and recommended is the Google Code Style Guide for JavaScript, but we recommend you read

2- LINTING Nowadays the best tool for linting your JavaScript code is JSHint. We recommend that whenever possible you verify your code style and patterns with a Lint tool.

3- FUNCTIONS AS FIRST-CLASS OBJECTS Functions in JavaScript are first class objects. This means that JavaScript functions are just a special type of object that can do all the things that regular objects can do.

4- FUNCTION BINDING Function binding is most probably the least of your concerns when beginning with JavaScript, but when you realize that you need a solution to the problem of how to keep the context of this within another function, then you might realize that what you actually need is Function.prototype.bind(). COMMENT: Comment your code.

5- Include All Necessary Semicolons Most developers won't intentionally fail to put semicolons in the proper places. However, you need to be aware that the browser will usually put in semicolons where it thinks they are necessary. This can enable a bad habit that will come back to bite you down the road. In some instances, the compiler might assume that a semicolon is not needed, which will introduce tricky, hard-to-find bugs into your code. Avoid this by always adding the proper semicolons. A good tool to help you check your JavaScript for forgotten semicolons is JSLint.

6- Use Decimals Cautiously When is 0.1 + 0.2 not equal to 0.3? When you do the calculation in JavaScript. The actual value of 0.1 + 0.2 comes out to be something like 0.30000000000000004. The reason for this (nope, not a bug) is because JavaScript uses Binary Floating Point numbers. To get around this issue, you can multiply your numbers to remove the decimal portion. For instance, if you were to add up the cost of two items, you could multiply each price by 100 and then divide the sum by 100.

7- avoid global variables

8- beware of automatic type conversations

9- initialize variables

10- don’t use newobject()