For a array like:-

Var myList = [‘apples’,’óranges’,’bananas’];

myList.forEach(function(value,index){

//do something like

Alert(“my shopping list has ” + value + “in it”);

});

//this is absolutely valid but all the browsers don’t support it since forEach is a newer method

In JS you can access the html things due to the DOM model….and methods of the same…  
like getElementByID(‘p’);

//it returns an array of all the paragraphs …

getElementsByTagNames(‘p’);

for more on this:-

[https://www.youtube.com/watch?v=W3EK4MlZW4g HYPERLINK "https://www.youtube.com/watch?v=W3EK4MlZW4g&list=PLoYCgNOIyGACTDHuZtn0qoBdpzV9c327V&index=5"& HYPERLINK "https://www.youtube.com/watch?v=W3EK4MlZW4g&list=PLoYCgNOIyGACTDHuZtn0qoBdpzV9c327V&index=5"list=PLoYCgNOIyGACTDHuZtn0qoBdpzV9c327V HYPERLINK "https://www.youtube.com/watch?v=W3EK4MlZW4g&list=PLoYCgNOIyGACTDHuZtn0qoBdpzV9c327V&index=5"& HYPERLINK "https://www.youtube.com/watch?v=W3EK4MlZW4g&list=PLoYCgNOIyGACTDHuZtn0qoBdpzV9c327V&index=5"index=5](https://www.youtube.com/watch?v=W3EK4MlZW4g&list=PLoYCgNOIyGACTDHuZtn0qoBdpzV9c327V&index=5)

java script listeners:-

adding listeners to elements of html…:-

//first grabbing the element by id to a variable…Ex:-

Var firstTextBox = document.getById(‘first-t-box’);

firstTextBox.addEventListener(“click”,function() {

alert(“the element is clicked!!”);

});

//like this …whenever for the element that event is happened then the function beneath it is called

//list of some listeners:-

//all are in small

Click

Mouseenter

Mouseleave

Mousedown

Mouseup

Mousemove

Keyup

Keydown

Blur

Focus

//etc…

We can also put the function outside and put as a parameter, such as:-

Var firstTextBox = document.getById(‘first-t-box’);

firstTextBox.addEventListener(“click”,go);

Function go() {

alert(“the element is clicked!!”);

}

….

So that you could use the function go for multiple html elements

After getting any element by Id…we can use that variable’s one method called innerHtml = “something”

What is basically does is it puts that thing inside that pair of tags defined by our varbale.

---

no undeclared vars,

almost no == all ===,

.foreach preferred over traditional for loop,

no two variables sharing same declaration i.e. decalred on same line and therefore been provided different declarations on different lines.

strings should be in double quotes

last element in json should not have a seprating or trailing comma.

paranthesis of functions should not have a space between funtion keyword and paranthesis.

expects semicolon after end of every statement.

unused vars warn,

else on the line where if ends,

naming conventions:-

var names, property names, function names should be in camel case as expected.

constants are snakecased but with all alphas in caps.

filename used snake-case too.

convetions:-

prefer .bind(this) instead of declaring some variable as say self = this.

prefered and not compulsory since sometimes you need to context i.e. the original this at that scope and the outer this of the outer scope indeed, so at these type of situations we need to do something like var self= this or we might pass this in the paramter, as in we can technically do it too.

setTimeout does not say that that function would execute after that milli seconds, it says the minimum time in milli seconds after which it would be pushed ot event dispatcher, and after the clearance of call stack it would be pushed to call stack and executed. Generally the delay between the time specified and the actual time taken is very less , just few milli seconds so we hardly ever notice it. But it could be there for big projects