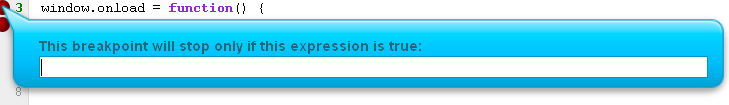
Debugging with Firebug

* Putting the breakpoint in highlighted number or simply we say that putting in the line which is a JavaScript statement.
* To activate debugging option like continue, step into reload the page again.
* Continue says go again till you hit another breakpoint.
* Step into step line by line through into the code.
* Step out means one level back up.
* Step over means execute this code or just do it and move on.
* Feature of JS



Common Errors in JS

* Issue 1: Syntax errors

function myFunction( {

console.log("You called myFunction);

}

window.onload = function() {

myFunction();

}

* Issue 2: calling a non-existent function

/\*

function myFunction() {

console.log("You called myFunction");

}

window.onload = function() {

myfunction();

}

\*/

* Issue 2b: typos very common with DOM methods

//var x = document.getElementByID("something");

* Issue 2c: using a non-existent object method

//var myArray = ["one","two","three"];

//console.log(myArray.revers());

* Issue 4: using document.getElementById before the element is part of the DOM.

// make sure the DOM is loaded - use window.load

//var myImage = document.getElementById("someImage");

* Issue 5 - assignment instead of equality

// This is NOT technically an error

/\*

var a = 10;

var b = 20;

if ( a = b) {

console.log("Something is wrong with the universe.");

} else {

console.log("This is what I expect!")

}

\*/

* Issue 6: missing parameters in function calls:

// This is NOT an error - it's just an unexpected result.

/\*

function calculateSum(a,b,c) {

return a + b + c;

}

var result = calculateSum(500,1000);

console.log(result);

\*/