JavaScript, HTML and CSS Coding Conventions

Samengevat:

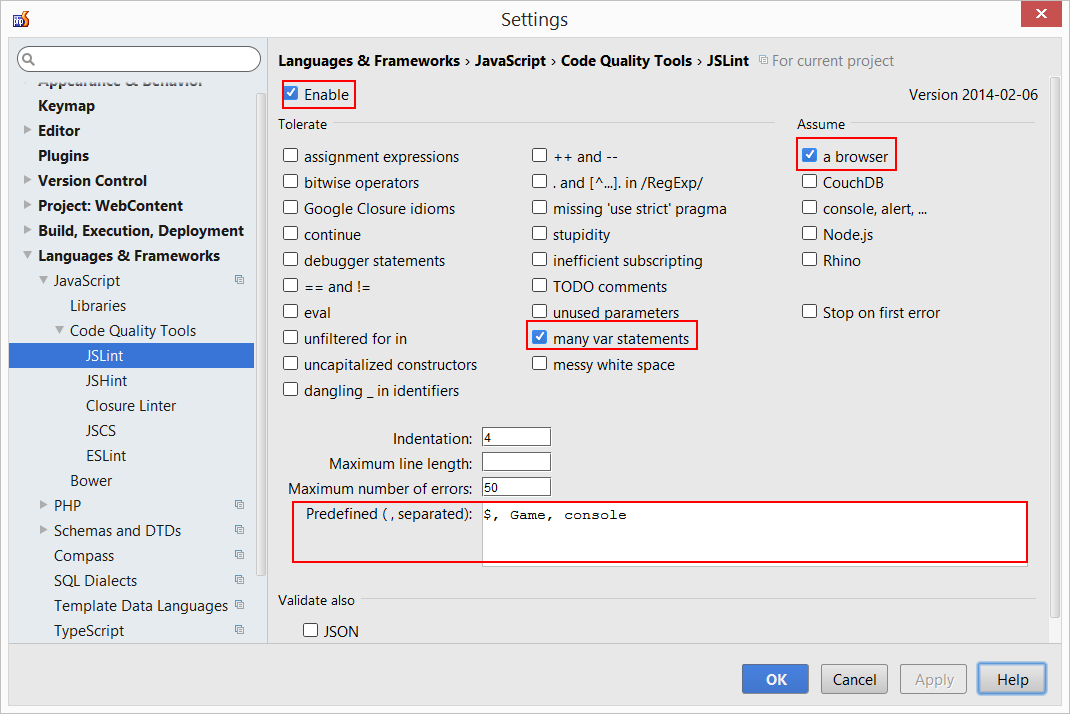
In Java, JavaScript, CSS en HTML schrijven we functie namen en variabelen in de **camelCase** notatie.

* var view = $(“#divHome”); /\* JavaScript \*/
* Game.addPlayer = function () {} /\* JavaScript \*/
* div.view = { background-color: black; } /\* CSS \*/
* .coolStyle = { text-decoration: line-through; } /\* CSS \*/
* private String name; /\* Java \*/
* public String getName() { return name; } /\* Java \*/
* …

**Uitzondering:** In Java en JavaScript schrijven we namespaces, classes en objecten waarvan er maar één instantie bestaat met **HoofdLetters**.

* Game = Game || {}; /\* JavaScript \*/
* Game.Views.HomeView.show(); /\* JavaScript \*/
* public class FestivalCard { } /\* Java \*/
* private Interface ICard; /\* Java \*/
* …

Voor JavaScript, gebruik JSLint.



Dit kan je configureren in phpstorm

Je schrijft je code in elk Javascript bestand dan in een closure. Dat beschermt je tegen ongewenst overschijven van variabelen en zorgt dat de globale namespace niet vervuilt raakt.

Voorbeeld van een minimum JavaScript code file is:

(function () { // Begin van bestand

“use strict”;

// CODE HIER

}()); // Einde bestand

Meer:

* <http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>
* <https://en.wikipedia.org/wiki/Coding_conventions>
* GOOGLE!