

ES2016

Wat is ES2016?

ES is een afkorting voor ECMAScript.

ECMAScript is een specificatie voor een general-purpose scripting programmeertaal. De meest bekende implementatie van deze specificatie is JavaScript.

Om dit verder te verduidelijken, hieronder een termenlijst om vaak verwarde begrippen uit elkaar te houden.

Termenlijst

Ecma International

Een organisatie die standaarden creëert voor technologieën.

Wat wordt hier bedoeld met 'standaarden'? Als metafoor kun je hierbij denken aan toetsenborden.

Je hebt in je leven verschillende toetsenborden gebruikt. Waarschijnlijk waren deze allemaal volgens de QWERTY-**standaard** ingericht, maar niet elke toets zat altijd op precies dezelfde locatie. De toetsenborden waren gebaseerd op de QWERTY standaard, maar de implementatie per fabrikant is steeds net iets anders.

ECMAScript is een standaard zoals QWERTY dat is. In plaats van een toetsenbordindeling te definiëren, gaat het hier om de specificatie voor een scripting programmeertaal.

Een scripting taal

Een programmeertaal die specifiek is ontworpen om te draaien op een bestaande entiteit of systeem.

Het grootste verschil tussen een scripting taal en een 'niet-scripting' taal is dat een scripting taal geen expliciete compilatiestap vereist.

Bijvoorbeeld, voor een Java applicatie moet de code gecompileerd worden naar bytecode die op de JVM kan draaien. Voor JavaScript is deze stap niet nodig. De taal wordt *geïnterpreteerd* door een JavaScript engine, bijvoorbeeld in browsers.

ECMAScript

De specificatie gedefinieerd om een general purpose scripting taal te maken.

ECMAScript beschrijft de regels, details, en richtlijnen waaraan een scripting taal moet voldoen om ECMAScript-geschikt/compliant te zijn.

JavaScript

Een general purpose scripting taal die voldoet aan de ECMAScript specificatie.

JavaScript is *een van de* implementaties van de ECMAScript specificatie. JavaScript voldoet aan alle standaarden die in de ECMAScript specificatie zijn vastgelegd. Twee andere relatief bekende implementaties zijn JScript en ActionScript. Je zou zelf een ECMAScript-compliant scripting taal kunnen maken, door de ECMAScript specificatie door te lezen en je hieraan te houden.

Een JavaScript engine

Een programma of interpreter die JavaScript code begrijpt en uitvoert.

JavaScript engines worden vooral gebruikt in web browsers, zoals de V8 engine in Google Chrome, SpiderMonkey in Firefox, en Chakra in Edge. Verschillende engines voeren JavaScript op hun eigen manier uit, waardoor er verschillen (kunnen) zijn in (bijvoorbeeld) performance.

Omdat ECMAScript een specificatie is, zijn de nieuwste features zoals deze beschreven worden in de specificatie niet direct geïmplementeerd in browsers als Chrome of Edge. Dit kan een reden zijn dat je JavaScript programma wel werkt in de ene browser, maar niet in de andere.

Een JavaScript runtime

Een omgeving waarin JavaScript kan draaien en geïnterpreted wordt door een JavaScript engine.

Voor een server-side applicatie, in tegenstelling tot client-side (in de browser), kan JavaScript code uitgevoerd worden door de applicatie Nodejs. Nodejs gebruikt, net als de browser Chrome, de V8 JavaScript engine om JavaScript code uit te voeren.

ECMAScript 6 (ES6)

En dan zijn we nu terug aangekomen bij een soortgelijke vraag aan de initiële vraag: Wat is ES6?

ES6 is een synoniem voor ECMAScript 6.

In deze zesde editie van de ECMAScript specificatie zijn er wijzigingen doorgevoegd vanuit Ecma International. Nieuwere versies van ECMAScript noemen we vanaf ES6 naar hun jaartal: hier ECMAScript 2015. Dit omdat er jaarlijks nieuwe specificaties uit zouden komen.

ECMAScript 6 == ES6 == ES2015

ECMAScript 7 == ES7 == ES2016

De reden dat je ES7 nergens ziet, maar ES6 wel, is doordat vanaf dat punt de nieuwe naamgeving met het jaartal gebruikt wordt. ES2016 is dus een latere versie dan ES6/ECMAScript6/ES2015.

Babel

Een 'transpiler' die ES6 code kan vertalen naar ES5 code.

Omdat veel huidige implementaties van ECMAScript nog volgens de ES5 standaarden gemaakt zijn, was er vraag naar een transpiler die code van het nieuwe ES6 omzet naar ES5. Alle grote browsers ondersteunen de ES5 specificatie, dus hierdoor zou je met ES6 gewoon kunnen programmeren voor alle grote browsers.

'Transpilen' betekent niets meer dan het omzetten van code A naar code B. Het gebeurt op een hoger niveau dan 'compilen', omdat je source code omzet naar source code, in plaats van source code naar machine-readable code.