

课程介绍

再次遇见 ECMAScript

目前我们掌握的是 ECMAScript 5 及之前的语法。

ES 6 已经是前端人员必备的知识技能。

系统化的学习 ECMAScript 是很有必要的。

内容概要

SUMMARY

- ECMAScript 与 JavaScript
- ECMAScript 的发展过程
- ECMAScript 2015 的新特性
- And more...

ECMAScript

JavaScript 的语言本身

通常将 ECMAScript 看作是 JavaScript 的标准规范。

实际上 JavaScript 是 ECMAScript 的扩展语言。

ECMAScript 只是提供了最基本的语法。

JavaScript 在语言基础上进行了扩展

JavaScript @ Web

ECMAScript

Web APIs

BOM

DOM

JavaScript @ Node.js

ECMAScript

Node APIs

fs

net

etc.

JavaScript 语言本身指的就是 ECMAScript

2015 年开始 ES 保持每年一个版本的迭代

名称	标准版本	发行时间
ECMAScript 2020 (ES2020)	11	2020年6月
ECMAScript 2019 (ES2019)	10	2019年6月
ECMAScript 2018 (ES2018)	9	2018年6月
ECMAScript 2017 (ES2017)	8	2017年6月
ECMAScript 2016 (ES2016)	7	2016年6月
ECMAScript 2015 (ES2015)	6	2015年6月
ECMAScript 5.1 (ES5.1)	5.1	2011年6月
ECMAScript 5 (ES5)	5	2009年12月
ECMAScript 4 (ES4)	4	被放弃
ECMAScript 3 (ES3)	3	1999年12月
ECMAScript 2 (ES2)	2	1998年6月
ECMAScript 1 (ES1)	1	1997年6月

ES2015 开始按照年份开始命名

很多人习惯将 ES2015 称之为 ES6

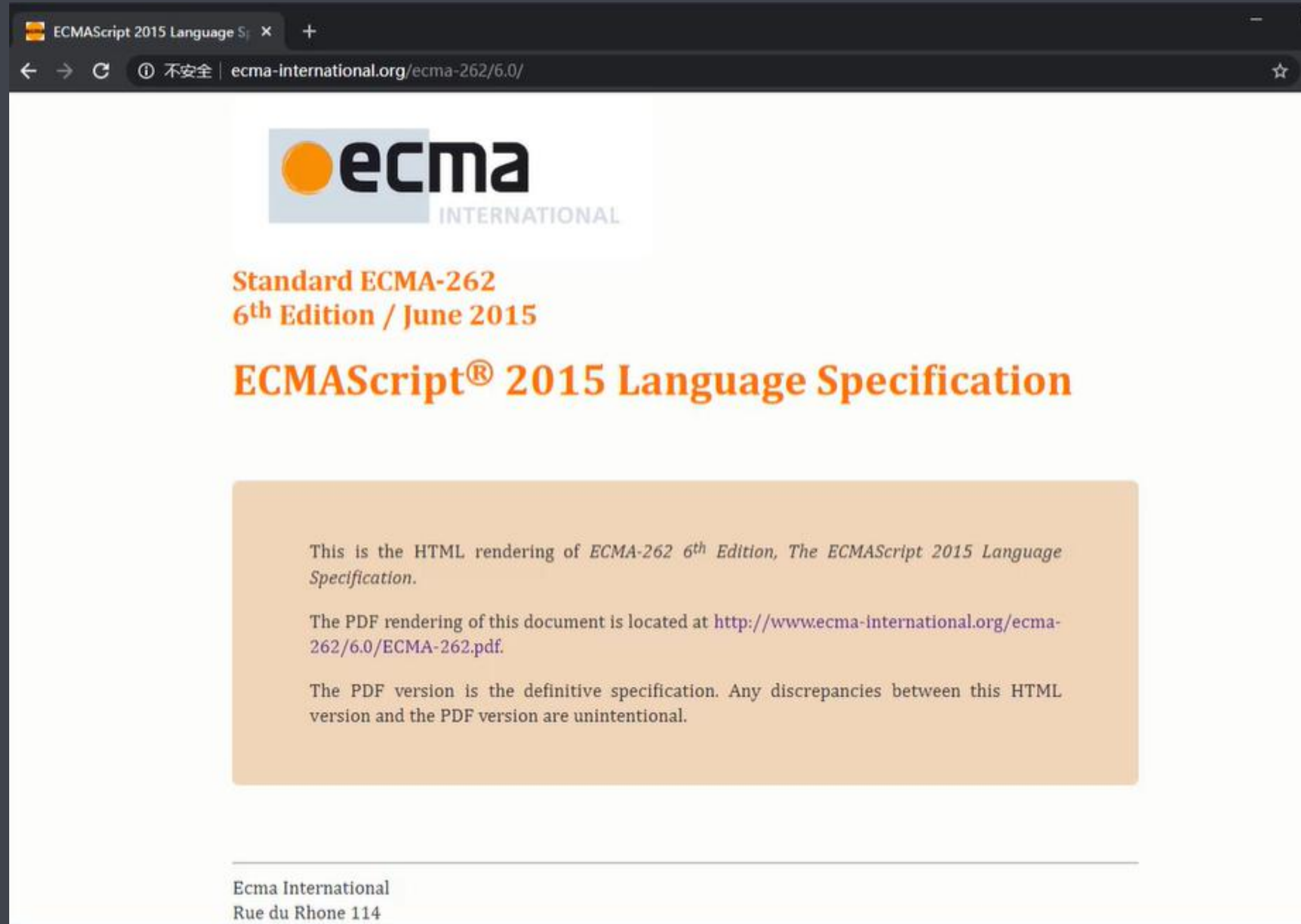
ECMAScript 2015

最新 ECMAScript 标准的代表版本

ES2015

用 ES6 来泛指所有的新标准

注意分辨用 ES6 是特指还是泛指



<http://www.ecma-international.org/ecma-262/6.0/>

重点了解在 ES5.1 基础之上的变化

- 解决原有语法上的一些问题或者缺陷
- 对原有语法进行增强
- 全新的对象、全新的方法、全新的功能
- 全新的数据类型和数据结构

准备工作

任何一个支持 ES2015 的环境都可以

我们在这里使用最新版本的 Chrome



Google Chrome



Google Chrome 已是最新版本
版本 87.0.4280.88 (正式版本) (32 位)

在 VScode 软件中使用 Chrome 调试

- Debugger for Chrome
- Live Server
- Browser Preview

let 与块级作用域

作用域 — 某个成员能够起作用的范围

在 ES2015 之前，ES 只有两种作用域

- 全局作用域
- 函数作用域
- 块级作用域

块，就是 {} 包裹起来的一个范围

```
if (true) {  
  console.log("hello")  
}
```

```
for (var i = 0 ; i < 10 ; i++) {  
  console.log(i);  
}
```

以前的块是没有作用域的

const

恒量 / 常量

特点：在 let 的基础上增加了一个【只读】效果

let + const

let + const + var

最佳实践：不用 `var`，主用 `const`，配合 `let`

数组的解构

Destructuring

对象的解构

Destructuring

模板字符串字面量

Template literals

模板字符串标签函数

Tagged templates

字符串的扩展方法

- `includes()`
- `startsWith()`
- `endsWith()`

参数默认值

Default parameters

剩余参数

Rest parameters

展开数组

Spread

箭头函数

Arrow functions

箭头函数与 this

对象字面量增强

Enhanced object literals

对象扩展方法

Object.assign

将多个源对象中的属性复制到一个目标对象中

对象扩展方法

Object.is

class 类

Classes

静态成员

static

静态方法 vs 实例方法

ES2015 中新增添加静态方法的关键词 `static`

类的继承

extends

Set 数据结构

Map 数据结构

Symbol

一种全新的原始数据类型

最主要的作用就是为对象添加独一无二的属性标识符

Symbol

补充

for...of 循环

for

for.....in

一些对象的遍历方法

这些遍历数据方式都有一定的局限性

引入了一种全新的遍历方式 for...of 循环

作为遍历所有数据结构的统一方式

ES2015 其他内容

- 可迭代接口
- 迭代器模式
- 生成器
- Proxy 代理对象
- Reflect 统一的对象操作 API
- Promise 异步解决方案
- ES Modules 语言层面的模块化标准

ECMAScript 2016

拉勾教育

— 互联网人实战大学 —



下载「拉勾教育App」
获取更多内容