

# 课程介绍

再次遇见 ECMAScript

# 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

# JavaScript @ Web

ECMAScript

## Web APIs

BOM

DOM

## JavaScript @ Node.js

ECMAScript

Node APIs

fs

net

etc.

名称	标准版本	发行时间
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

# ECMAScript 2015

- let 和 const
- Arrow functions
- Classes (类)
- Promises
- Generators
- Modules
- Template literals (模板字面量)
- Default parameters (默认参数)
- Enhanced object literals (对象字面量增强)
- Destructuring assignments (解构分配)
- Spread operator (展开操作符)
- for ... of 循环
- Map 和 Set
- Proxy



- let 和 const
- Arrow functions
- Classes (类)
- Promises
- Generators
- Modules
- Template literals (模板字面量)
- Default parameters (默认参数)
- Enhanced object literals (对象字面量增强)
- Destructuring assignments (解构分配)
- Spread operator (展开操作符)
- for ... of 循环
- Map 和 Set
- Proxy

- 对原有语法进行增强
- 解决原有语法上的一些问题或者缺陷
- 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

# let 与块级作用域

# const

恒量 / 常量

# 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

# 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

# 对象扩展方法

Object.is

# Proxy

代理对象

# Proxy

vs. Object.defineProperty()

handler 方法	触发方式
get	读取某个属性
set	写入某个属性
has	in 操作符
deleteProperty	delete 操作符
getProperty	Object.getPrototypeOf()
setProperty	Object.setPrototypeOf()
isExtensible	Object.isExtensible()
preventExtensions	Object.preventExtensions()
getOwnPropertyDescriptor	Object.getOwnPropertyDescriptor()
defineProperty	Object.defineProperty()
ownKeys	Object.keys() 、 Object.getOwnPropertyNames()、 Object.getOwnPropertySymbols()
apply	调用一个函数
construct	用 new 调用一个函数

# Reflect

统一的对象操作 API

handler 方法	默认调用
get	Reflect.get()
set	Reflect.set()
has	Reflect.has()
deleteProperty	Reflect.delete()
getProperty	Reflect.getPrototypeOf()
setProperty	Reflect.setPrototypeOf()
isExtensible	Reflect.isExtensible()
preventExtensions	Reflect.preventExtensions()
getOwnPropertyDescriptor	Reflect.getOwnPropertyDescriptor()
defineProperty	Reflect.defineProperty()
ownKeys	Reflect.ownKeys()
apply	Reflect.apply()
construct	Reflect.construct()

# Promise

一种更优的异步编程解决方案，在异步编程课程中专门介绍



# class 类

Classes

# 静态成员

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

# Symbol

补充

# for...of 循环



# 可迭代接口

Iterable

# 实现可迭代接口

Iterable

# 迭代器模式

Iterator

# 生成器

Generator

# 生成器应用

Generator

# ES Modules

语言层面的模块化标准

# ECMAScript 2016

# ECMAScript 2017



# ECMAScript 2018

# 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