

Что нового в JS и вебе



Мария Кондаурова

BIOCAD

BIOCAD

Biotechnology Company



telegram @Momomash

github @Momomash

twitter @mari_momos

О чем расскажу

- Автостопом по стандартам
- Пропозалы:
 - Что мы увидим в ECMAScript 2023
 - Что мы увидим еще, но позже
- Новые Web API



Ecma TC39

Ecma International, Technical Committee 39 - ECMAScript

4k followers The web <https://tc39.es> Verified

Overview Repositories 249 Projects Packages People 125

Pinned

[ecma262](#) Public

Status, process, and documents for ECMA-262

HTML 14k 1.3k

[ecma402](#) Public

Status, process, and documents for ECMA 402

HTML 470 124

[test262](#) Public

Official ECMAScript Conformance Test Suite

JavaScript 2k 457

[proposals](#) Public

Tracking ECMAScript Proposals

16.7k 723

[agendas](#) Public

TC39 meeting agendas

JavaScript 925 194

[notes](#) Public

TC39 meeting notes

JavaScript 476 100

Жизненный цикл пропозала

(почти как разработка игры)

Жизненный цикл пропозала

(почти как разработка игры)

0 - идея 💡

Есть проблема -> предложение решения

Жизненный цикл пропозала

(почти как разработка игры)

0 - идея 💡

Есть проблема -> предложение решения

1 - предложение

Концепт

Обоснование и описание

Жизненный цикл пропозала

(почти как разработка игры)

- | | |
|-----------------|--|
| 0 - идея 💡 | Есть проблема -> предложение решения |
| 1 - предложение | Концепт
Обоснование и описание |
| 2 - черновик | Прототип
Формальная реализация спецификации,
Первая имплементация (babel) |

Жизненный цикл пропозала

(почти как разработка игры)

- | | |
|-----------------|--|
| 0 - идея 💡 | Есть проблема -> предложение решения |
| 1 - предложение | Концепт
Обоснование и описание |
| 2 - черновик | Прототип
Формальная реализация спецификации,
Первая имплементация (babel) |
| 3- кандидат | Бета-тест
Отзывы от разработчиков и браузеров
Фикс багов |

Жизненный цикл пропозала

(почти как разработка игры)

- | | |
|-----------------|---|
| 0 - идея 💡 | Есть проблема -> предложение решения |
| 1 - предложение | Концепт
Обоснование и описание |
| 2 - черновик | Прототип
Формальная реализация спецификации,
Первая имплементация (babel) |
| 3- кандидат | Бета-тест
Отзызы от разработчиков и браузеров
Фикс багов |
| 4- финал | Предрелиз
Написаны юнит-тесты (test-262)
Практический опыт применения
-> Включение в ближайшую версию языка |

Web APIs



Web Incubator CG

attribution-reporting-api

Attribution Reporting API.

Incubating for 3 years and 11 months. Last updated March 2023 | ★ 276.

webpackage

Web packaging format.

Incubating for 6 years and 6 months. Last updated March 2023 | ★ 1174.

file-system-access

Expose the file system on the user's device, so Web apps can interoperate with the user's native applications.

Incubating for 7 years and 1 month. Last updated February 2023 | ★ 599.

pending-beacon

A better beaconing API.

Incubating for 1 year and 3 months. Last updated March 2023 | ★ 25.

sanitizer-api

No description available.

Incubating for 7 years and 3 months. Last updated February 2023 | ★ 197.

client-hints-infrastructure

Specification for the Client Hints infrastructure - privacy preserving proactive content negotiation.

Incubating for 4 years and 1 month. Last updated February 2023 | ★ 60.

turtledove

TURTLEDOVE.

Incubating for 3 years and 3 months. Last updated March 2023 | ★ 429.

nav-speculation

Proposal to enable privacy-enhanced preloading.

Incubating for 2 years and 8 months. Last updated March 2023 | ★ 114.

first-party-sets

No description available.

Incubating for 4 years and 8 months. Last updated March 2023 | ★ 165.

storage-buckets

API proposal for managing multiple storage buckets.

Incubating for 2 years and 9 months. Last updated March 2023 | ★ 22.

web-app-launch

Web App Launch Handler.

Incubating for 5 years and 7 months. Last updated March 2023 | ★ 57.

scroll-to-text-fragment

Proposal to allow specifying a text snippet in a URL fragment.

Incubating for 4 years and 3 months. Last updated February 2023 | ★ 538.

trust-token-api

Trust Token API.

Incubating for 3 years and 8 months. Last updated March 2023 | ★ 356.

webcomponents

Web Components specifications.

Incubating for 9 years and 5 months. Last updated March 2023 | ★ 4139.

view-transitions

No description available.

Incubating for 2 years and 5 months. Last updated March 2023 | ★ 705.

webmonetization

Proposed Web Monetization standard.

Incubating for 4 years and 7 months. Last updated March 2023 | ★ 349.

ua-client-hints

Wouldn't it be nice if `User-Agent` was a (set of) client hints?





Incubating for 4 years and 6 months. Last updated March 2023 | ★ 531.

shared-storage






Explainer for proposed web platform Shared Storage API.

Incubating for 1 year and 11 months. Last updated March 2023 | ★ 44.

B

- [Background Fetch API](#) 
- [Background Sync \(en-US\)](#) 
- [Background Tasks \(en-US\)](#)
- [Barcode Detection API \(en-US\)](#) 
- [Battery API \(en-US\)](#)
- [Beacon](#)
- [Bluetooth API \(en-US\)](#) 
- [Broadcast Channel API \(en-US\)](#)

C

- [CSS Counter Styles \(en-US\)](#)
- [CSS Custom Highlight API \(en-US\)](#) 
- [CSS Font Loading API \(en-US\)](#)
- [CSS Painting API \(en-US\)](#) 
- [CSS Properties and Values API \(en-US\)](#)
- [CSS Typed Object Model API \(en-US\)](#)
- [CSSOM](#)
- [Canvas API](#)
- [Channel Messaging API \(en-US\)](#)
- [Clipboard API](#)
- [Compression Streams API \(en-US\)](#)
- [Console API](#)
- [Contact Picker API \(en-US\)](#) 
- [Content Index API \(en-US\)](#) 
- [Cookie Store API \(en-US\)](#) 
- [Credential Management API](#)

D

- [DOM](#)
- [Device Orientation Events \(en-US\)](#)

Interop

Bocoup

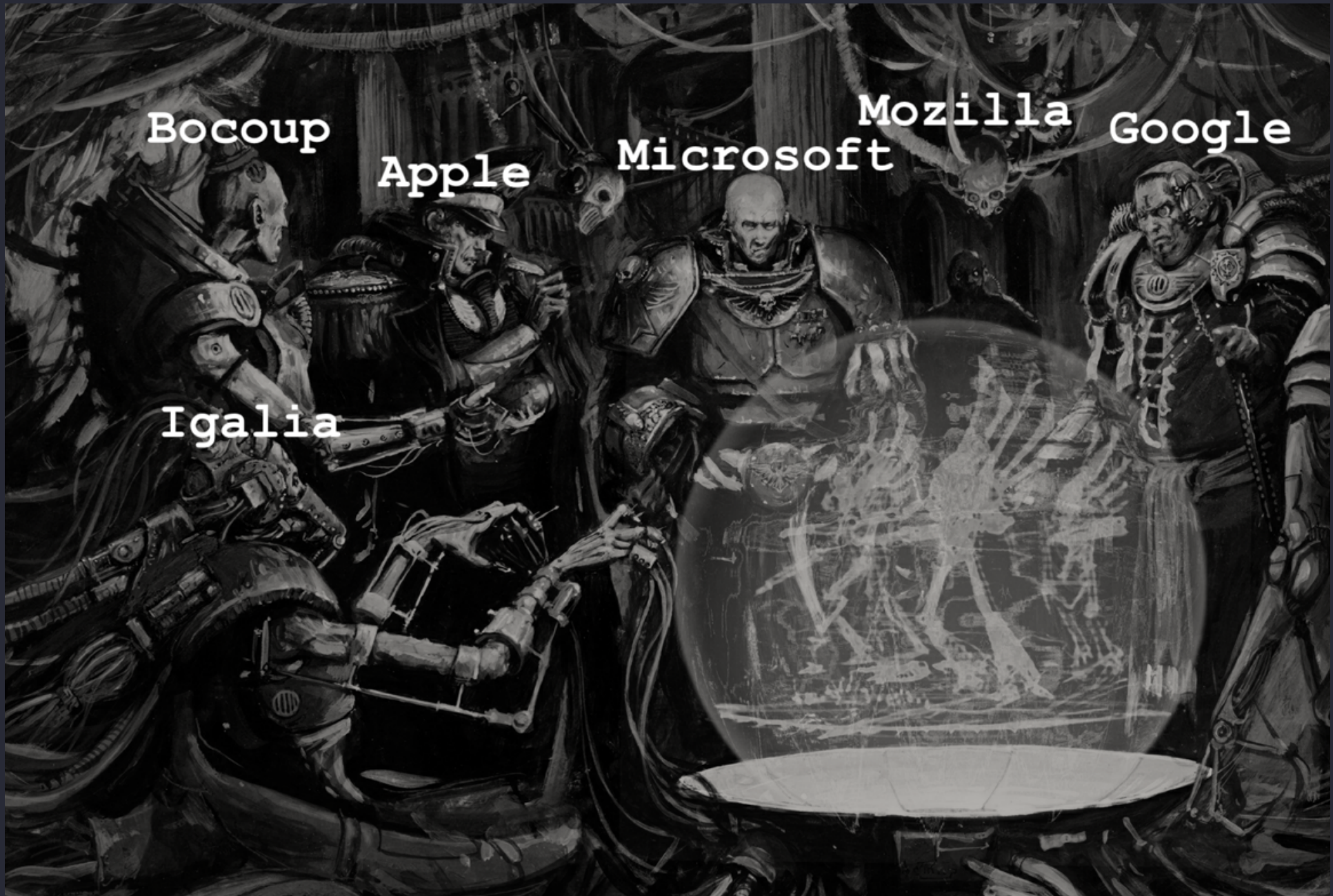
Apple

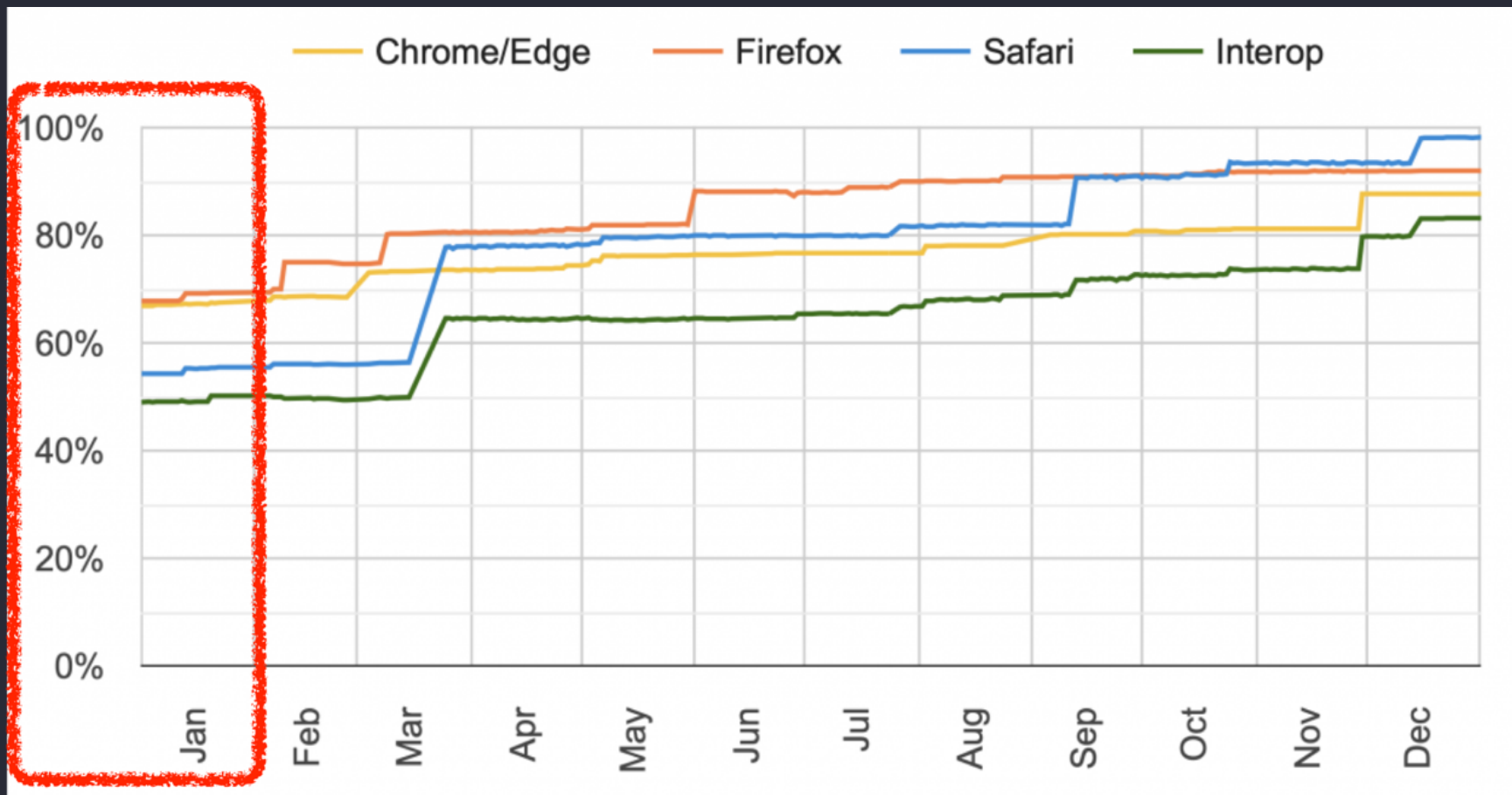
Microsoft

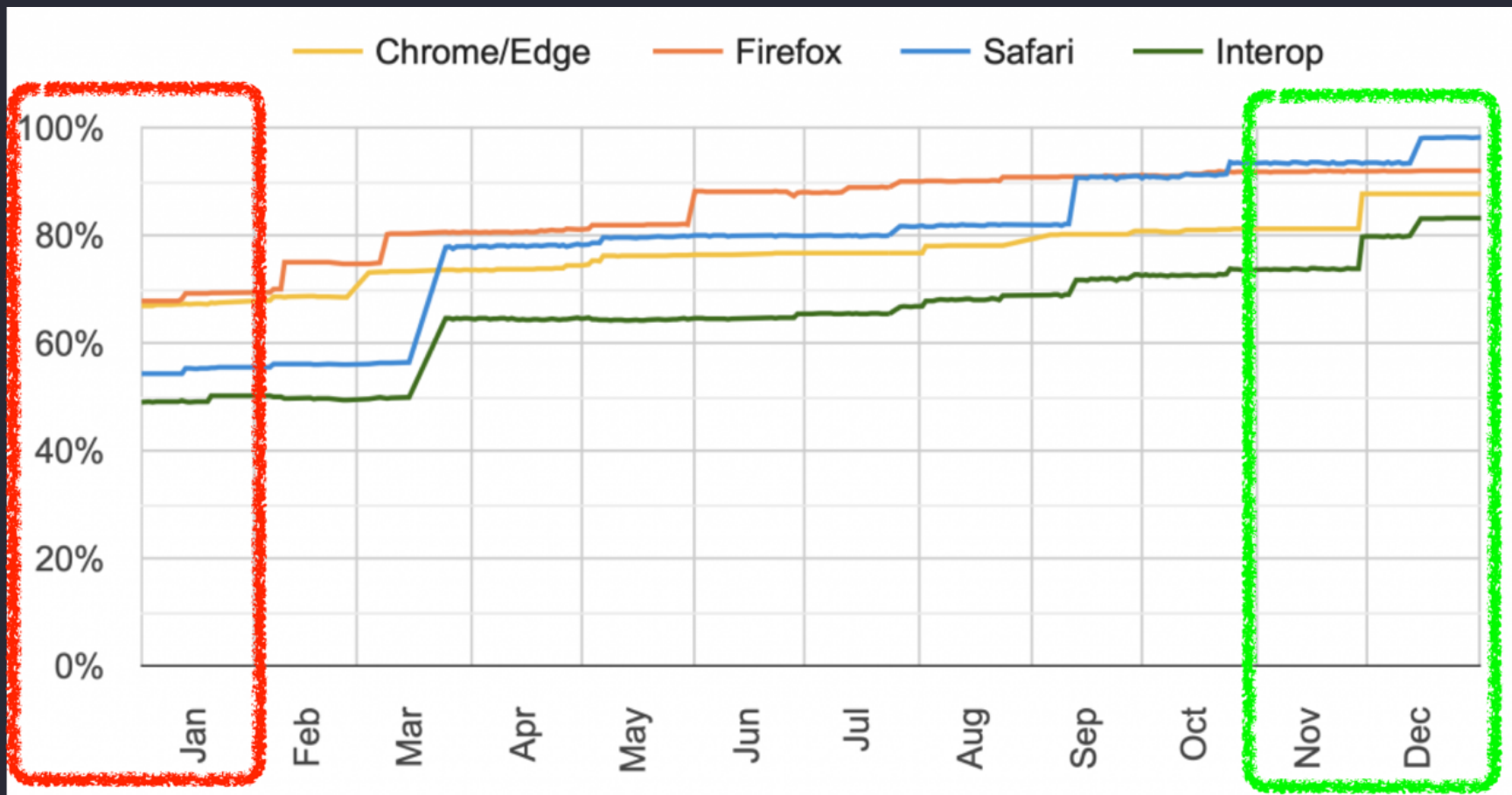
Mozilla

Google

Igalia







Interop 2023 Dashboard

STABLE

EXPERIMENTAL

59

INTEROP

30

INVESTIGATIONS

84



Chrome



Edge







Firefox

87



Safari

Active Focus Areas ^

				 INTEROP ^
Border Image	99.4%	92.0%	94.3%	87.9%
Color Spaces and Functions	99.7%	49.1%	99.5%	48.9%
Container Queries	97.8%	86.7%	84.7%	75.9%
Containment	97.4%	82.7%	73.7%	71.7%
CSS Math Functions	57.0%	52.9%	81.6%	51.9%
CSS Pseudo-classes	69.2%	42.0%	90.8%	8.6%
Custom Properties	88.5%	4.8%	93.7%	4.8%
Flexbox	99.3%	96.2%	93.5%	92.2%
Font Feature Detection and Palettes	96.2%	100%	93.5%	91.9%
Forms	85.9%	93.9%	86.1%	80.5%
Grid	99.3%	88.0%	90.6%	82.5%
:has()	92.1%	24.5%	93.9%	24.5%
Inert	100%	38.0%	89.4%	38.0%
Masking	68.2%	94.4%	86.1%	62.6%
Media Queries 4	99.0%	89.6%	99.9%	89.6%
Modules	99.3%	61.3%	86.6%	61.3%
Motion Path	55.7%	64.5%	91.5%	54.1%
Offscreen Canvas	98.0%	89.9%	95.9%	88.5%
Pointer and Mouse Events	63.3%	70.6%	46.6%	32.5%

Что мы увидим в ECMAScript 2023

proposal-symbols-as-weakmap-keys

	Map	WeakMap
Тип ключа	Any	Object
Препятствует сборке мусора?	true	false

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3 map.set(primarch, 'primarch');
4
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3
4 map.set(primarch, 'primarch');
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3 map.set(primarch, 'primarch');
4
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3
4 map.set(primarch, 'primarch');
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3 map.set(primarch, 'primarch');
4
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3
4 map.set(primarch, 'primarch');
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3 map.set(primarch, 'primarch');
4
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 let weakMap = new WeakMap();
3
4 weakMap.set(primarch, 'traitor');
5 primarch = null;
6
7 console.log(primarch) // {}
```

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3 map.set(primarch, 'primarch');
4
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 let weakMap = new WeakMap();
3
4 weakMap.set(primarch, 'traitor');
5 primarch = null;
6
7 console.log(primarch) // {}
```



```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3 map.set(primarch, 'primarch');
4
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 let weakMap = new WeakMap();
3
4 weakMap.set(primarch, 'traitor');
5 primarch = null;
6
7 console.log(primarch) // {}
```

```
1 let primarch = {name: 'Horus'};
2 const map = new Map();
3 map.set(primarch, 'primarch');
4
5 primarch = null;
6
7 console.log(map); // Map(1) {{name: 'Horus'}: 'primarch'}
```

```
1 let primarch = {name: 'Horus'};
2 let weakMap = new WeakMap();
3
4 weakMap.set(primarch, 'traitor');
5 primarch = null;
6
7 console.log(primarch) // {}
```

Map

WeakMap

Тип ключа

Any

Object, Symbol

Препятствует сборке
мусора?

true

false

proposal-array-find-from-last

```
let games = [  
  ...,  
  {year: 2015, author: 'CD Projekt RED', title: 'The Witcher 3  
  {year: 2020, author: 'CD Projekt RED', title: 'Cyberpunk 207  
  {year: 2022, author: 'From Software', title: 'Elden Ring'},  
  {year: 2023, author: 'Mundfish', title: 'Atomic Heart'},  
]
```

```
let games = [  
    ...,  
    {year: 2015, author: 'CD Projekt RED', title: 'The Witcher 3  
    {year: 2020, author: 'CD Projekt RED', title: 'Cyberpunk 207  
    {year: 2022, author: 'From Software', title: 'Elden Ring'},  
    {year: 2023, author: 'Mundfish', title: 'Atomic Heart'},  
]
```

```
let game, index;  
let i = games.length - 1;  
  
while (i > 0 && !game && !index) {  
    if (games[i].author === 'From Software'){  
        game = games[i]; index = i;  
    };  
    --i;  
};
```

```
let games = [
  ...,
  {year: 2015, author: 'CD Projekt RED', title: 'The Witcher 3'},
  {year: 2020, author: 'CD Projekt RED', title: 'Cyberpunk 2077'},
  {year: 2022, author: 'From Software', title: 'Elden Ring'},
  {year: 2023, author: 'Mundfish', title: 'Atomic Heart'},
]
```

```
let game, index;
let i = games.length - 1;

while (i > 0 && !game && !index) {
  if (games[i].author === 'From Software'){
    game = games[i]; index = i;
  };
  --i;
};
```

```
const game = games.findLast(
  ({author}) => author === 'author1'
);

const indexofGame = games.findLastIndex(
  ({author}) => author === 'author1'
);
```

proposal-change-array-by-copy

`.reverse()`

`.sort()`

`.splice()`

.reverse()

.sort()

.splice()

[...array].reverse()

[...array].sort()

[...array].splice()

.reverse()

.sort()

.splice()

[...array].reverse()

[...array].sort()

[...array].splice()

.toReversed()

.toSorted()

.toSpliced()

.with(index, value) -> Array

.with(index, value) -> Array

```
1 const array = [1,2,3];  
2  
3 array.with(1,10); // => [1,10,3]  
4  
5 console.log(array); // => [1,2,3]
```

.with(index, value) -> Array

```
1 const array = [1,2,3];  
2  
3 array.with(1,10); // => [1,10,3]  
4  
5 console.log(array); // => [1,2,3]
```

.with(index, value) -> Array

```
1 const array = [1,2,3];  
2  
3 array.with(1,10); // => [1,10,3]  
4  
5 console.log(array); // => [1,2,3]
```

Что мы увидим, но позже

Stage 3 - proposal-array-grouping

```
1 type Primarch = { name: string, isLoyal: string };
2
3 const primarchs: Primarch[] = [
4     { name: 'Horus', isLoyal: false },
5     { name: 'Leman Russ', isLoyal: true },
6     { name: 'Magnus the Red', isLoyal: true },
7     { name: 'Mortarion', isLoyal: false },
8     { name: 'Lion El Jonson', isLoyal: false },
9     { name: 'Conrad Curze', isLoyal: false },
10    { name: 'Corvus Corax', isLoyal: true },
11 ];
12
13 type GroupedPrimarchs = {
14     loyal: Primarch[], traitor: Primarch[]
15 };
```

```
1 type Primarch = { name: string, isLoyal: string };
2
3 const primarchs: Primarch[] = [
4     { name: 'Horus', isLoyal: false },
5     { name: 'Leman Russ', isLoyal: true },
6     { name: 'Magnus the Red', isLoyal: true },
7     { name: 'Mortarion', isLoyal: false },
8     { name: 'Lion El Jonson', isLoyal: false },
9     { name: 'Conrad Curze', isLoyal: false },
10    { name: 'Corvus Corax', isLoyal: true },
11 ];
12
13 type GroupedPrimarchs = {
14     loyal: Primarch[], traitor: Primarch[]
15 };
```

```
const groupedPrimarchs = primarchs.reduce(  
  (acc, primarch) => {  
    primarch.isLoyal ? acc.loyal = [...acc.loyal, primarch]  
      : acc.traitor = [...acc.traitor, primarch];  
    return acc;  
  },  
  {loyal: [], traitor: []}  
);
```

```
const groupedPrimarchs = primarchs.reduce(

  (acc, primarch) => {
    primarch.isLoyal ? acc.loyal = [...acc.loyal, primarch]
      : acc.traitor = [...acc.traitor, primarch];
    return acc;
  },
  {loyal: [], traitor: []}
);
```

```
const result = {loyal: [], traitor: []};

primarchs.forEach(() => {
  let target = i.isLoyal ? result.loyal : result.traitor;
  target.push(i);
})
```

НОВЫЕ МЕТОДЫ:

Array.**group()**

Array.**groupToMap()**

```
1 const groupedPrimarch = primarchs.group(
2   (item, array, index) =>
3     item.isLoyal ? 'loyal' : 'traitor';
4 )
5 // => { loyal: [ { name: 'Leman Russ', isLoyal: true },...],
6 //      traitor: [ { name: 'Horus', isLoyal: false },...]}
7 // }
```



```
1 const groupedPrimarch = primarchs.group(  
2   (item, array, index) =>  
3     item.isLoyal ? 'loyal' : 'traitor';  
4 )  
5 // => { loyal: [ { name: 'Leman Russ', isLoyal: true },...],  
6 //      traitor: [ { name: 'Horus', isLoyal: false },...]  
7 //      }
```

```
1 const loyal = {loyal: true};  
2 const traitor = {traitor: true};  
3  
4 const primarchsMap = primarchs.groupToMap(  
5   (item, array, index) => item.isLoyal ? loyal : traitor;  
6 ) // => Map  
7  
8 primarchsMap.get(loyal) // => [ { name: 'Leman Russ', isLoyal:  
9 primarchsMap.get(traitor) // => [ { name: 'Horus', isLoyal: fal
```

```
1 const groupedPrimarch = primarchs.group(
2   (item, array, index) =>
3     item.isLoyal ? 'loyal' : 'traitor';
4 )
5 // => { loyal: [ { name: 'Leman Russ', isLoyal: true },...],
6 //      traitor: [ { name: 'Horus', isLoyal: false },...]}
7 // }
```

```
1 const loyal = {loyal: true};
2 const traitor = {traitor: true};
3
4 const primarchsMap = primarchs.groupToMap(
5   (item, array, index) => item.isLoyal ? loyal : traitor;
6 ) // => Map
7
8 primarchsMap.get(loyal) // => [ { name: 'Leman Russ', isLoyal:
9 primarchsMap.get(traitor) // => [ { name: 'Horus', isLoyal: fal
```

```
1 const groupedPrimarch = primarchs.group(  
2   (item, array, index) =>  
3     item.isLoyal ? 'loyal' : 'traitor';  
4 )  
5 // => { loyal: [ { name: 'Leman Russ', isLoyal: true },...],  
6 //      traitor: [ { name: 'Horus', isLoyal: false },...]  
7 //      }
```

```
1 const loyal = {loyal: true};  
2 const traitor = {traitor: true};  
3  
4 const primarchsMap = primarchs.groupToMap(  
5   (item, array, index) => item.isLoyal ? loyal : traitor;  
6 ) // => Map  
7  
8 primarchsMap.get(loyal) // => [ { name: 'Leman Russ', isLoyal:  
9 primarchsMap.get(traitor) // => [ { name: 'Horus', isLoyal: fal
```

Stage 3 - proposal-json-modules

```
import config from './config.json';  
console.log(config.username);
```

```
import config from './config.json';  
console.log(config.username);
```

Failed to load module script: Expected a JavaScript module script but the server responded with a MIME type of "application/json". Strict MIME type checking is enforced for module scripts per HTML spec.


```
1 import config from './config.json' assert {type: 'json'};
```

```
1 const config = import('./config.json' assert {type: 'json'});  
2 config.then( data =>  
3   data.default.forEach(param => {  
4     console.log(param);  
5   })  
6 )
```

```
1 new Worker('app.wasm', {  
2   type: 'module',  
3   assert: {type: 'webassembly'}  
4 })
```


Stage 2 - proposal-record-tuple

8 ОСНОВНЫХ ТИПОВ В JavaScript:

Number

BigInt

String

Boolean

Null

Undefined

Symbol

Object

8 ОСНОВНЫХ ТИПОВ В JavaScript:

Number

BigInt

String

Boolean

Null

Undefined

Symbol

Object



8 ОСНОВНЫХ ТИПОВ В JavaScript:

Number

BigInt

String

Boolean

Null

Undefined

Symbol

Object

+

Record

Tuple


```
1 let alliances = #['Order', 'Chaos', 'Death', 'Destruction'];
2
3 let game = #{
4   name: 'Age of Sigmar',
5   tags: #['wargame', 'Games Workshop', 'Fantasy']
6 };
```

```
1 let alliances = #['Order', 'Chaos', 'Death', 'Destruction'];
2
3 let game = #{
4   name: 'Age of Sigmar',
5   tags: #['wargame', 'Games Workshop', 'Fantasy']
6 };
```



```
1 let alliances = #['Order', 'Chaos', 'Death', 'Destruction'];  
2  
3 let game = #{  
4   name: 'Age of Sigmar',  
5   tags: #['wargame', 'Games Workshop', 'Fantasy']  
6 };
```



```
1 const game = { name: 'Warhammer' };  
2  
3 console.log(game === { name: 'Warhammer' }); // false;
```

```
1 const game = { name: 'Warhammer' };  
2  
3 console.log(game === { name: 'Warhammer' }); // false;
```

```
1 const game = { name: 'Warhammer' };  
2  
3 console.log(game === { name: 'Warhammer' }); // false;
```

```
console.log(  
    JSON.stringify(game) === JSON.stringify({ name: 'Warhammer' }  
); // true;
```



```
1 const game = { name: 'Warhammer' };  
2  
3 console.log(game === { name: 'Warhammer' }); // false;
```

```
console.log(  
    JSON.stringify(game) === JSON.stringify({ name: 'Warhammer' }  
); // true;
```

```
const game1 = { name: 'Age of Sigmar', company: 'Games Workshop'  
const game2 = { company: 'Games Workshop', name: 'Age of Sigmar'  
  
console.log(JSON.stringify(game1) === JSON.stringify(game2));  
// false;
```

```
1 const game = { name: 'Warhammer' };  
2  
3 console.log(game === { name: 'Warhammer' }); // false;
```

```
console.log(  
    JSON.stringify(game) === JSON.stringify({ name: 'Warhammer' }  
); // true;
```

```
const game1 = { name: 'Age of Sigmar', company: 'Games Workshop'  
const game2 = { company: 'Games Workshop', name: 'Age of Sigmar'  
  
console.log(JSON.stringify(game1) === JSON.stringify(game2));  
// false;
```

```
1 const game1 = { name: 'Age of Sigmar', company: 'Games Worksho  
2 const game2 = { company: 'Games Workshop', name: 'Age of Sigma  
3  
4 // lodash  
5 .isEqual(game1, game2); // true
```

```
#{name: 'Sigmar'} === #{name: 'Sigmar'} // true  
#['Order', 'Chaos'] === #['Order', 'Chaos'] // true
```

Зачем?

IMMUTABLE



IMMUTABLE



**Когда работа с датами станет
удобнее?**

Moment.js

Date-fns

Day.js

Moment.js

Date-fns

Day.js

+

Temporal

```
const date = Temporal.Now.zonedDateTimeISO();

const { year, month, day, hour, minute, second, millisecond } = date;

console.log(
  year,      // 2023
  month,     // 4
  day,       // 19
  hour,      // 21
  minute,    // 48
  second,    // 50
  millisecond // 256
);
```

```
1 const { dayOfWeek, dayOfYear, daysInMonth, inLeapYear } = date;
2
3 console.log(
4     dayOfWeek,    // 3
5     dayOfYear,    // 109
6     daysInMonth,  // 30
7     inLeapYear    // false
8 );
```

```
1  const { dayOfWeek, dayOfYear, daysInMonth, inLeapYear } = date;
2
3  console.log(
4      dayOfWeek,    // 3
5      dayOfYear,    // 109
6      daysInMonth,  // 30
7      inLeapYear    // false
8  );
```

```
1 const { dayOfWeek, dayOfYear, daysInMonth, inLeapYear } = date;  
2  
3 console.log(  
4     dayOfWeek,    // 3  
5     dayOfYear,    // 109  
6     daysInMonth,  // 30  
7     inLeapYear    // false  
8 );
```

```
1 const { dayOfWeek, dayOfYear,daysInMonth,inLeapYear } = date;  
2  
3 console.log(  
4     dayOfWeek,    // 3  
5     dayOfYear,    // 109  
6     daysInMonth,  // 30  
7     inLeapYear    // false  
8 );
```

Сравнение

```
1 const now = Temporal.Now.plainDateISO();
2
3 const pastDate = Temporal.PlainDate.from('2021-03-12');
4 const futureDate = Temporal.PlainDate.from('2023-08-15');
5
6 Temporal.PlainDateTime.compare(now, pastDate) // 1
7 Temporal.PlainDateTime.compare(now, futureDate) // -1
8 Temporal.PlainDateTime.compare(now, now) // 0
```


Сравнение

```
1 const now = Temporal.Now.plainDateISO();  
2  
3 const pastDate = Temporal.PlainDate.from('2021-03-12');  
4 const futureDate = Temporal.PlainDate.from('2023-08-15');  
5  
6 Temporal.PlainDateTime.compare(now, pastDate)      // 1  
7 Temporal.PlainDateTime.compare(now, futureDate)    // -1  
8 Temporal.PlainDateTime.compare(now, now)           // 0
```

Добавление и вычитание + неизменяемость

```
1 const today = Temporal.PlainDate.from('2023-04-19'); // 2023-04-19
2
3 const tomorrow = today.add({days: 1}) // 2023-04-20
4 const monthAgo = today.subtract({months: 1}) // 2023-03-19
5 const yearAgo = today.subtract({years: 1}) // 2022-04-19
6
7 console.log(today.toJSON()) // 2023-04-19
```

Добавление и вычитание + неизменяемость

```
1 const today = Temporal.PlainDate.from('2023-04-19'); // 2023-04-19
2
3 const tomorrow = today.add({days: 1})                // 2023-04-20
4 const monthAgo = today.subtract({months: 1})          // 2023-03-19
5 const yearAgo = today.subtract({years: 1})            // 2022-04-19
6
7 console.log(today.toJSON())                            // 2023-04-19
```

Добавление и вычитание + неизменяемость

```
1 const today = Temporal.PlainDate.from('2023-04-19'); // 2023-04-19
2
3 const tomorrow = today.add({days: 1}) // 2023-04-20
4 const monthAgo = today.subtract({months: 1}) // 2023-03-19
5 const yearAgo = today.subtract({years: 1}) // 2022-04-19
6
7 console.log(today.toJSON()) // 2023-04-19
```

Добавление и вычитание + неизменяемость

```
1 const today = Temporal.PlainDate.from('2023-04-19'); // 2023-04-19
2
3 const tomorrow = today.add({days: 1}) // 2023-04-20
4 const monthAgo = today.subtract({months: 1}) // 2023-03-19
5 const yearAgo = today.subtract({years: 1}) // 2022-04-19
6
7 console.log(today.toJSON()) // 2023-04-19
```

Продолжительность и округление

```
1 const duration = Temporal.Duration.from({
2   minutes: 10,
3   seconds: 52
4 });
5
6 const durationInDays = duration.round({
7   smallestUnit: 'munites' // PT11M
8 })
```

Продолжительность и округление

```
1 const duration = Temporal.Duration.from({  
2   minutes: 10,  
3   seconds: 52  
4 });  
5  
6 const durationInDays = duration.round({  
7   smallestUnit: 'munites' // PT11M  
8 })
```

Преобразование в JSON и LocaleString

```
1 const today = Temporal.PlainDate.from('2023-04-19');  
2  
3 console.log(today.toLocaleString()); // 19.04.2023  
4 console.log(today.toJSON());        // 2023-04-19
```


Преобразование в JSON и LocaleString

```
1 const today = Temporal.PlainDate.from('2023-04-19');  
2  
3 console.log(today.toLocaleString()); // 19.04.2023  
4 console.log(today.toJSON());        // 2023-04-19
```

В чем подвох?



**Произвольное форматирование
пока не завезли**

Когда?

Когда?

Stage 3

Когда?

Stage 3

2 полифилла

Когда?

Stage 3

2 полифилла

"-Most likely it'll be a part of the next edition."

Ujjwal Sharma

Stage 2 - proposal-pipeline-operator

three(two(one(value))) **VS** value.one().two().three()

```
1 console.log(  
2   chalk.dim(  
3     `${Object.keys(envar)  
4       .map(envar) =>  
5         `${envar}=${envars[envar]}` )  
6       .join(' ' )  
7     }`,  
8     'node',  
9     args.join(' ' )));  
10  )  
11 )
```

```
1 console.log(  
2   chalk.dim(  
3     `${Object.keys(envvar)  
4       .map(envvar) =>  
5         `${envvar}=${envvars[envvar]}` )  
6       .join(' ')}  
7     }`,  
8     'node',  
9     args.join(' ')));  
10  )  
11 )
```

```
1 console.log(  
2   chalk.dim(  
3     `${Object.keys(envvar)  
4       .map(envvar) =>  
5         `${envvar}=${envvars[envvar]}` )  
6       .join(' ')}  
7     }`,  
8     'node',  
9     args.join(' ')));  
10  )  
11 )
```



```
1 console.log(  
2     chalk.dim(  
3         `${Object.keys(envvar)  
4             .map(envvar) =>  
5                 `${envvar}=${envvars[envvar]}`)  
6             .join(' ')}  
7         }`,  
8         'node',  
9         args.join(' ')));  
10 )  
11 )
```

```
1 console.log(  
2   chalk.dim(  
3     `${Object.keys(envar)  
4       .map(envar) =>  
5         `${envar}=${envars[envar]}` )  
6       .join(' ' )  
7     }`,  
8     'node',  
9     args.join(' ' )));  
10  )  
11 )
```

```
1 console.log(  
2   chalk.dim(  
3     `${Object.keys(envvar)  
4       .map(envvar) =>  
5         `${envvar}=${envvars[envvar]}` )  
6       .join(' ')}  
7     }`,  
8     'node',  
9     args.join(' ')));  
10  )  
11 )
```




```
1 console.log(  
2   chalk.dim(  
3     `${Object.keys(envvar)  
4       .map(envvar) =>  
5         `${envvar}=${envvars[envvar]}` )  
6       .join(' ' )  
7     }`,  
8     'node',  
9     args.join(' ' )));  
10  )  
11 )
```



```
1 Object.keys(envvars)  
2 .map(envvar) => `${envvar}=${envvars[envvar]}` )  
3 .join(' ' )  
4 |> `${ #}{%}`  
5 |> chalk.dim(%, 'node', args.join(' ' ))  
6 |> console.log(%)
```

```

1 console.log(
2   chalk.dim(
3     `${Object.keys(envvar)
4       .map(envvar) =>
5         `${envvar}=${envvars[envvar]}` )
6       .join(' ')}
7     }`,
8     'node',
9     args.join(' ')));
10  )
11 )

```



```

1 Object.keys(envvars)
2 .map(envvar) => `${envvar}=${envvars[envvar]}` )
3 .join(' ')
4 |> `${ #}%`
5 |> chalk.dim(%, 'node', args.join(' '))
6 |> console.log(%)

```

```

1 console.log(
2   chalk.dim(
3     `${Object.keys(envvar)
4       .map(envvar) =>
5         `${envvar}=${envvars[envvar]}` )
6       .join(' ')}
7     }`,
8     'node',
9     args.join(' ')));
10  )
11 )

```



```

1 Object.keys(envvars)
2 .map(envvar) => `${envvar}=${envvars[envvar]}` )
3 .join(' ')
4 |> `${ #}%`
5 |> chalk.dim(%, 'node', args.join(' '))
6 |> console.log(%)

```

```

1 console.log(
2   chalk.dim(
3     `${Object.keys(envvar)
4       .map(envvar) =>
5         `${envvar}=${envvars[envvar]}` )
6       .join(' ')}
7     }`,
8     'node',
9     args.join(' ')));
10  )
11 )

```



```

1 Object.keys(envvars)
2 .map(envvar) => `${envvar}=${envvars[envvar]}` )
3 .join(' ')
4 |> `${#}%`
5 |> chalk.dim(%, 'node', args.join(' '))
6 |> console.log(%)

```

```

1 console.log(
2   chalk.dim(
3     `${Object.keys(envvar)
4       .map(envvar) =>
5         `${envvar}=${envvars[envvar]}` )
6       .join(' ')}
7     }`,
8     'node',
9     args.join(' ')));
10  )
11 )

```



```

1 Object.keys(envvars)
2 .map(envvar) => `${envvar}=${envvars[envvar]}` )
3 .join(' ')
4 |> `${ #}%`
5 |> chalk.dim(%, 'node', args.join(' '))
6 |> console.log(%)

```

```

1 console.log(
2   chalk.dim(
3     `${Object.keys(envvar)
4       .map(envvar) =>
5         `${envvar}=${envvars[envvar]}` )
6       .join(' ')}
7     }`,
8     'node',
9     args.join(' ')));
10  )
11 )

```



```

1 Object.keys(envvars)
2 .map(envvar) => `${envvar}=${envvars[envvar]}` )
3 .join(' ')
4 |> `${#}%`
5 |> chalk.dim(%, 'node', args.join(' '))
6 |> console.log(%)

```

%

```
1 Object.keys(envvars)
2 .map(envar) => `${envar}=${envvars[envar]}`)
3 .join(' ')
4 |> `${#}%`
5 |> chalk.dim(%, 'node', args.join(' '))
6 |> console.log(%)
```

Stage 3 - proposal-decorators


```
1 function logDuration(fn) {  
2     return function decorator(...args) {  
3         let start = Date.now();  
4         let result = fn.apply(this, args);  
5         let duration = Date.now - start;  
6  
7         console.log(fn.name + '() duration: ' + duration);  
8         return result;  
9     };  
10 };
```

```
1 function logDuration(fn) {  
2     return function decorator(...args) {  
3         let start = Date.now();  
4         let result = fn.apply(this, args);  
5         let duration = Date.now - start;  
6  
7         console.log(fn.name + '() duration: ' + duration);  
8         return result;  
9     };  
10 };
```



4 итерации
7 лет разработки
Уже 2 legасу-имплементации



4 итерации

7 лет разработки

Уже 2 legасу-имплементации



Stage 3

Возможность через
прототип влиять на
объект декорирования

Проблемы
производительности

Возможность через
прототип влиять на
объект декорирования

Проблемы
производительности

Работают только с
объектом
декорирования, а не с
дескрипторами

Работают только с
классами и их
элементами

Новый объект
декорирования –
accessor

```
1  type Decorator = (  
2    value: Input,  
3    context: {  
4      kind: string;  
5      name: string | symbol;  
6      access: {  
7        get?(): unknown;  
8        set?(value: unknown): void;  
9      };  
10     private?: boolean;  
11     static?: boolean;  
12     addInitializer?(initializer: () => void): void;  
13   }  
14 ) => Output | void;
```



```
1  type Decorator = (  
2    value: Input,  
3    context: {  
4      kind: string;  
5      name: string | symbol;  
6      access: {  
7        get?(): unknown;  
8        set?(value: unknown): void;  
9      };  
10     private?: boolean;  
11     static?: boolean;  
12     addInitializer?(initializer: () => void): void;  
13   }  
14 ) => Output | void;
```

```
1  type Decorator = (  
2    value: Input,  
3    context: {  
4      kind: string;  
5      name: string | symbol;  
6      access: {  
7        get?(): unknown;  
8        set?(value: unknown): void;  
9      };  
10     private?: boolean;  
11     static?: boolean;  
12     addInitializer?(initializer: () => void): void;  
13   }  
14 ) => Output | void;
```

```
1  type Decorator = (  
2    value: Input,  
3    context: {  
4      kind: string;  
5      name: string | symbol;  
6      access: {  
7        get?(): unknown;  
8        set?(value: unknown): void;  
9      };  
10     private?: boolean;  
11     static?: boolean;  
12     addInitializer?(initializer: () => void): void;  
13   }  
14 ) => Output | void;
```



Демо

**Какие декораторы использовать
сейчас?**

BABEL

```
1 // .babelrc
2
3 "plugins": [
4   [
5     "@babel/plugin-proposal-decorators", {
6       "version": "2023-01"
7     }
8   ]
9 ]
```

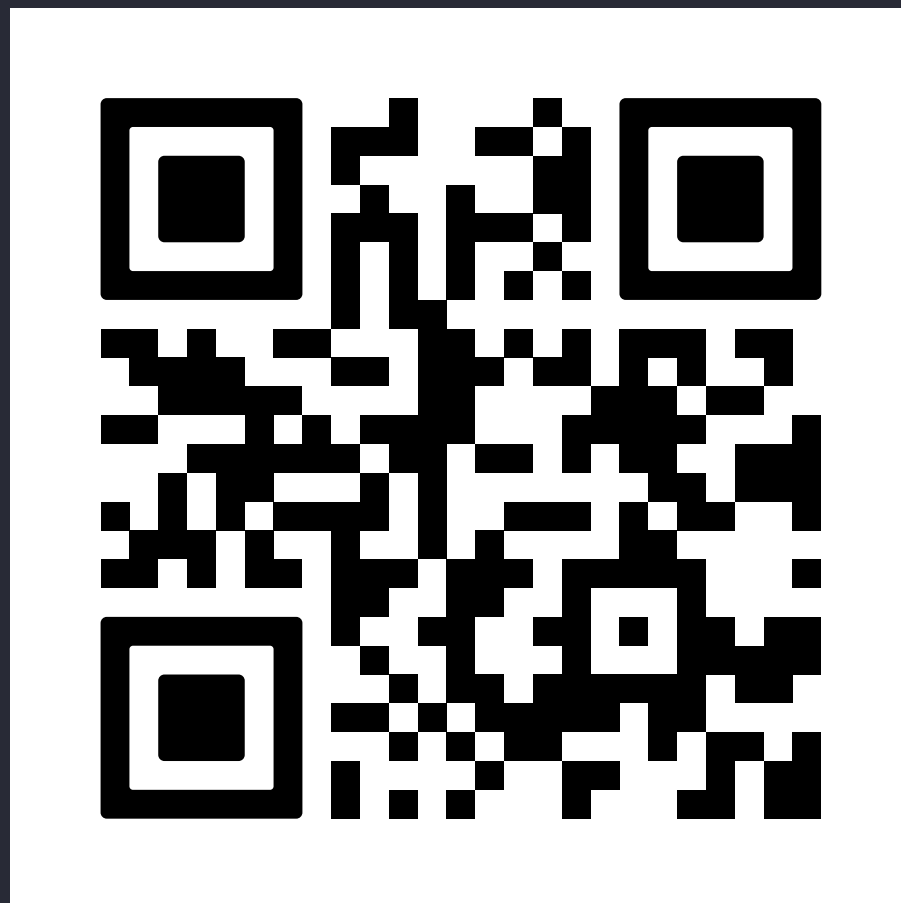
НОВЫЕ Web APIs

Дисклеймер

Следующие Web API поддерживаются еще не всеми браузерами 🧪.

Перед использованием убедитесь, что поддерживаются нужные вам платформы

Idle Detection API



Демо



94



94-95



No



80



No

Screen Wake Lock API

- Пользователь готовит по рецепту
- Использование навигатора
- Использование приложения с управлением голосом


```
1  if ("wakeLock" in navigator) {
2      let wakeLock = null;
3
4      try {
5          wakeLock = await navigator.wakeLock.request("screen");
6          console.log("Wake Lock is active!");
7      } catch (err) {
8          console.error(`${err.name}, ${err.message}`);
9      }
10
11 } else {
12     console.error("Wake lock is not supported by this browser.");
13 }
```



```
1 // отключить блокировку экрана
2
3 wakeLock.release().then(() => {
4     wakeLock = null;
5 });
```

```
1 // среагировать на пробуждение
2
3 wakeLock.addEventListener('release', () => {
4     console.log('Hello, user!');
5 });
```



84



94-84



No



70



16.4

HTML Sanitizer API





npm

🔍 sanitize html

182 packages found

Sort Packages

- ☐ Optimal
- ☒ Popularity
- ☐ Quality
- ☐ Maintenance

XSS

Sanitize untrusted HTML (to prevent XSS) with a configuration specified by a Whitelist

sanitization xss sanitize sanitisation input security escape encode filter va
whitelist

 leizongmin published 1.0.14 • 8 months ago

sanitize-html

Clean up user-submitted HTML, preserving allowlisted elements and allowlisted attributes on a per

html parser sanitizer sanitize

 boutell published 2.10.0 • 2 months ago

Element.setHTML

```
1 const initialString =  
2     "any <\script>alert('За Альянс!')<" + "/script> text";  
3  
4 const sanitizer = new Sanitizer();  
5  
6 const target = document.getElementById('target');  
7 target.setHTML(initialString, {sanitizer})  
8  
9 console.log(target.innerHTML); // any text
```

Element.setHTML

```
1 const initialString =  
2     "any <\script>alert('За Альянс!')<" + "/script> text";  
3  
4 const sanitizer = new Sanitizer();  
5  
6 const target = document.getElementById('target');  
7 target.setHTML(initialString, {sanitizer})  
8  
9 console.log(target.innerHTML); // any text
```

Element.setHTML

```
1 const initialString =  
2     "any <\script>alert('За Альянс!')<" + "/script> text";  
3  
4 const sanitizer = new Sanitizer();  
5  
6 const target = document.getElementById('target');  
7 target.setHTML(initialString, {sanitizer})  
8  
9 console.log(target.innerHTML); // any text
```


Element.setHTML

```
1 const initialString =  
2     "any <\script>alert('За Альянс!')<" + "/script> text";  
3  
4 const sanitizer = new Sanitizer();  
5  
6 const target = document.getElementById('target');  
7 target.setHTML(initialString, {sanitizer})  
8  
9 console.log(target.innerHTML); // any text
```

Element.sanitizeFor()

```
1  const initialString =  
2    "any <\script>alert('За Альянс!')<" + "/script> text";  
3  
4  const sanitizer = new Sanitizer();  
5  
6  const sanitizedString =  
7    sanitizer.sanitizeFor('div', initialString);  
8  
9  console.log(sanitizedString instanceof HTMLDivElement);  
10 // true  
11  
12 document.querySelector('div#target')  
13 .replaceChildren(sanitizedString)
```

Element.sanitizeFor()

```
1 const initialString =  
2   "any <\script>alert('За Альянс!')<" + "/script> text";  
3  
4 const sanitizer = new Sanitizer();  
5  
6 const sanitizedString =  
7   sanitizer.sanitizeFor('div', initialString);  
8  
9 console.log(sanitizedString instanceof HTMLDivElement);  
10 // true  
11  
12 document.querySelector('div#target')  
13 .replaceChildren(sanitizedString)
```

Element.sanitizeFor()

```
1  const initialString =  
2    "any <\script>alert('За Альянс!')<" + "/script> text";  
3  
4  const sanitizer = new Sanitizer();  
5  
6  const sanitizedString =  
7    sanitizer.sanitizeFor('div', initialString);  
8  
9  console.log(sanitizedString instanceof HTMLDivElement);  
10 // true  
11  
12 document.querySelector('div#target')  
13 .replaceChildren(sanitizedString)
```

Element.sanitizeForO

```
1  const initialString =  
2    "any <\script>alert('За Альянс!')<" + "/script> text";  
3  
4  const sanitizer = new Sanitizer();  
5  
6  const sanitizedString =  
7    sanitizer.sanitizeFor('div', initialString);  
8  
9  console.log(sanitizedString instanceof HTMLDivElement);  
10 // true  
11  
12 document.querySelector('div#target')  
13 .replaceChildren(sanitizedString)
```

Element.sanitize()

```
1  const sanitizer = new Sanitizer();  
2  
3  const iframe = document.getElementById('userFrame');  
4  const treeFromIframe = iframe.contentWindow.document;  
5  
6  const sanitizedIframeTree = sanitizer.sanitize(treeFromIframe);  
7  iframe.replaceChildren(sanitizedIframeTree);
```

Element.sanitize()

```
1  const sanitizer = new Sanitizer();
2
3  const iframe = document.getElementById('userFrame');
4  const treeFromIframe = iframe.contentWindow.document;
5
6  const sanitizedIframeTree = sanitizer.sanitize(treeFromIframe);
7  iframe.replaceChildren(sanitizedIframeTree);
```

Element.sanitize()

```
1  const sanitizer = new Sanitizer();  
2  
3  const iframe = document.getElementById('userFrame');  
4  const treeFromIframe = iframe.contentWindow.document;  
5  
6  const sanitizedIframeTree = sanitizer.sanitize(treeFromIframe);  
7  iframe.replaceChildren(sanitizedIframeTree);
```



```
dictionary SanitizerConfig {  
    sequence<DOMString> allowElements;  
    sequence<DOMString> blockElements;  
    sequence<DOMString> dropElements;  
    AttributeMatchList allowAttributes;  
    AttributeMatchList dropAttributes;  
    boolean allowCustomElements;  
    boolean allowUnknownMarkup;  
    boolean allowComments;  
};
```



Демо



105



105



83



79



No

CSS Custom Highlight API



Дока

/ Поиск



ESC

Фильтровать по:

HTML

CSS

JavaScript

Рецепты

Доступность

Веб-платформа

Введите текст в поле поиска, и появится список всего, что нашлось

3249 packages found

Sort Packages

- ☐ Optimal
- ☒ Popularity
- ☐ Quality
- ☐ Maintenance

highlight.js

Syntax highlighting with language autodetection.

highlight syntax



highlightjs_bot published 11.7.0 • 5 months ago

prismjs

Lightweight, robust, elegant syntax highlighting. A spin-off project from Dabblet.

prism highlight



rundevlopment published 1.29.0 • 8 months ago

ansicolors

Functions that surround a string with ansicolor codes so it prints in color.

ansi colors highlight string



thlorenz published 0.3.2 • 9 years ago

react-highlight

React component for syntax highlighting

react highlight.js syntax highlighting react-component

- Больше зависимостей
- Доп обертка в DOM

```
▼<a class="search-hit__link link"  
  href="/tools/react-and-alternatives/  
"> == $0  
  ::before  
  <mark class="search-hit__marked">  
    React</mark>  
  " и альтернативы "  
</a>
```




Демо



105



105



No



91



No

View Transitions API

Page 1 Page 2 Page 3 Page 4 Page 5 Page 6

REACT PAGE TRANSITIONS

A React component that makes it easy to use the page transitions from the [Codedrops Page Transitions Demo](#)

[View the usage instructions and source code on Github](#)

Preset:

fall

Enter Override:

None

Exit Override:

None



react transition

Search

698 packages found

Sort Packages

☐ Optimal

☐ Popularity

☐ Quality

☐ Maintenance

react-transition-group

A react component toolset for managing animations

react transition addons transition-group animation css transitions

 epsilon1lon published 4.4.5 • 9 months ago

react-simple-animate

React simple animate

react animation transition-animation animate-css animation-controller animation-sequence keyframes-animation



прт пакетов много не бывает

HTTP 203

All

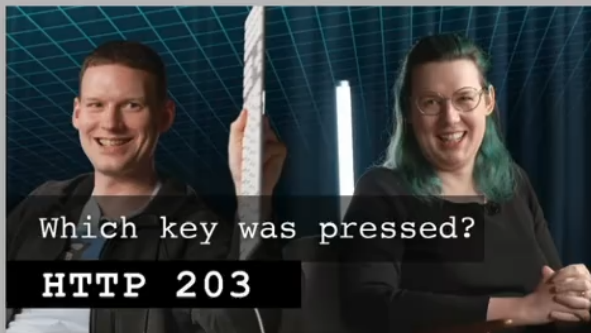
Ada

Surma

Paul



2022-05-03



2022-03-29



2022-03-15



2022-01-04



2021-12-27



2021-12-13

```
1  ::view-transition-old(root) {
2    animation: 90ms cubic-bezier(0.4, 0, 1, 1) both fade-out,
3             300ms cubic-bezier(0.4, 0, 0.2, 1) both slide-to-left;
4  }
5
6  ::view-transition-new(root) {
7    animation: 210ms cubic-bezier(0, 0, 0.2, 1) 90ms both fade-in
8             300ms cubic-bezier(0.4, 0, 0.2, 1) both slide-from-right;
9  }
```



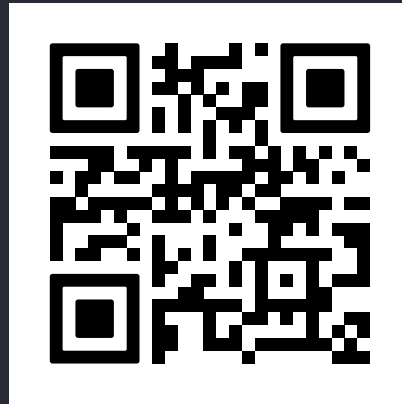
```
1  ::view-transition-old(root) {
2    animation: 90ms cubic-bezier(0.4, 0, 1, 1) both fade-out,
3      300ms cubic-bezier(0.4, 0, 0.2, 1) both slide-to-left;
4  }
5
6  ::view-transition-new(root) {
7    animation: 210ms cubic-bezier(0, 0, 0.2, 1) 90ms both fade-in
8      300ms cubic-bezier(0.4, 0, 0.2, 1) both slide-from-right;
9  }
```

```
<div
  style={{
    viewTransitionName: "box"
  }}
>
  <img
```

```
::view-transition-new(box) {
  animation-name: stretch;
  animation-duration: 0.5s;
  animation-timing-function: ease-out;
}
```



Без анимации



`startViewTransition()`



С анимацией

startViewTransition()

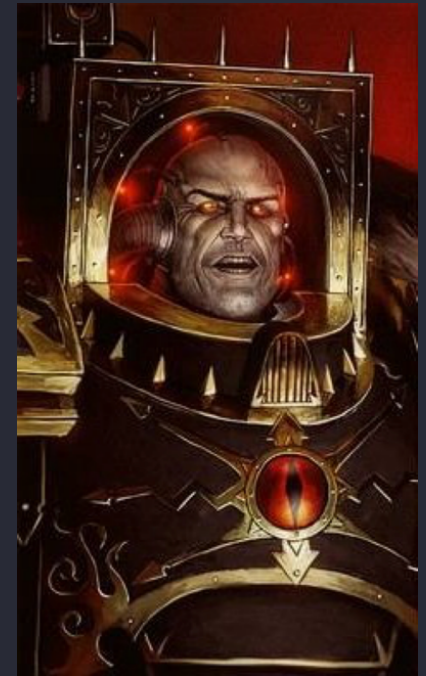




1. Вычисления...



1. Вычисления...
2. Построение
нового DOM





Opacity:1 ->
Opacity: 0

1. Вычисления...
2. Построение
нового DOM
3. Переход



Opacity:0 ->
Opacity: 1



Демо



111



111



No



97



No

Выводы...





BIOCAD

telegram @Momomash

github @Momomash

twitter @mari_momos

