ECMAScript 2017

Key	Value
Name	ECMAScript 2017
Short name	ES2017
Was finalized in	June 2017

Feature list

#	Feature	Desc
1	`MAJOR` Async functions	higher level abstraction over promises
2	`MAJOR` Shared memory and atomics	shared memory between multiple workers
3	String padding	add characters to reach a specific length. source
4	Object.values()	get an array of object values
5	Object.entries()	get an array of pair [key, value]
6	Object.getOwnPropertyDescriptors()	get object property descriptor
7	Trailing commas	in function parameter lists and calls

1. `MAJOR` Async functions

The async and await keywords enable asynchronous, promise-based behavior to be written in a cleaner style. source

```
function main() {
  getAllStudents()
    .then(response => {
      const { data, pagination } = response;
      console.log(data, pagination);
    })
    .catch(error => {
      console.log(error);
    })
}
```

```
async function main() {
  try {
    const response = await getAllStudents();
    console.log(data, pagination);
  } catch (error) {
    console.log(error);
  }
}
```

2. MAJOR Shared memory and atomics

Since ES2017, you can **create a shared memory array** between **web workers and their creator**, using a **SharedArrayBuffer**. source

> Đọc thêm topic này: https://github.com/tc39/ecmascript_sharedmem/blob/master/TUTORIAL.md

3. String padding

- pad = add character until it reach a specific length.
- default pad character is **space**

```
'1234'.padStart(8);          // '    1234'
'1234'.padStart(8, '*');     // '****1234'
'123456'.padStart(8, '*');   // '**123456'

'1234'.padEnd(8, '*');       // '1234****'
'12'.padEnd(8, '*');         // '12******'
```

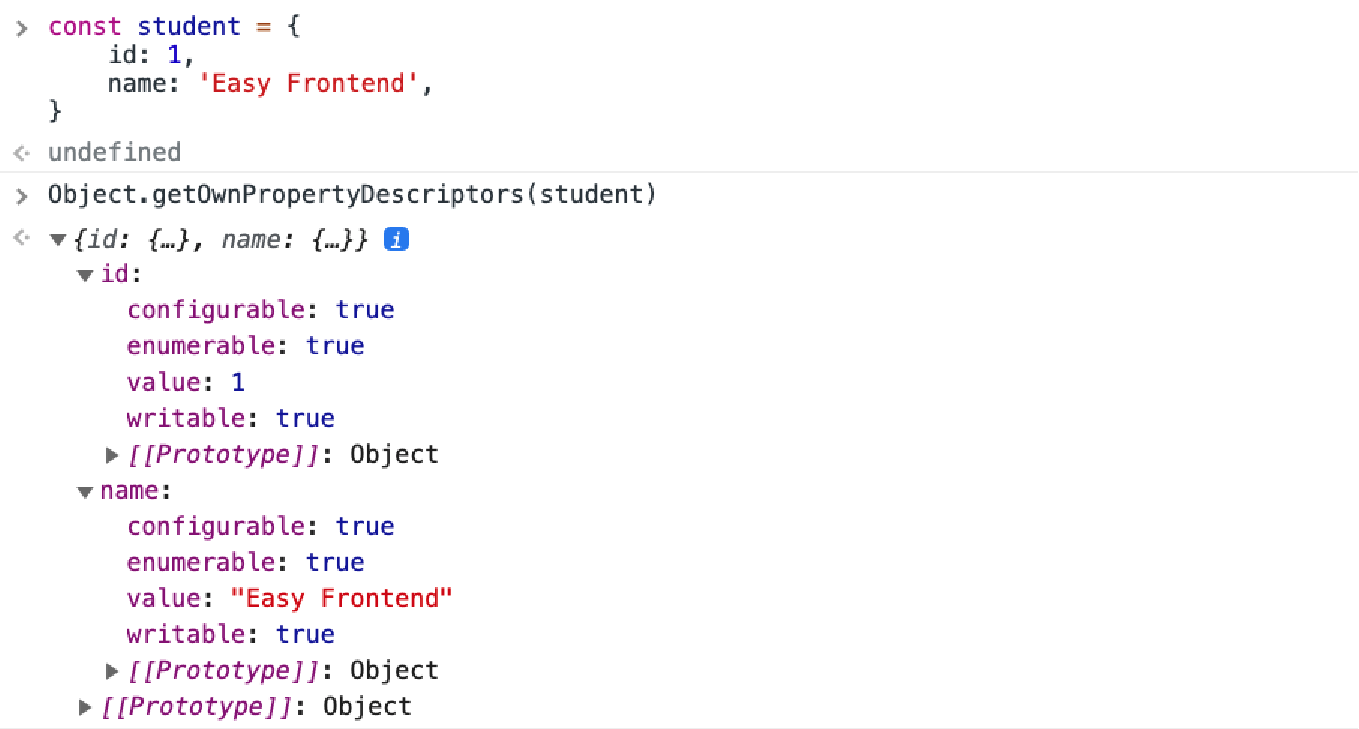
4. Object.values()

```
Object.values({
    id: 1,
    name: 'Easy Frontend',
    age: 18,
});

// [1, 'Easy Frontend', '18']
```

5. Object.entries()

```
Object.values({
    id: 1,
    name: 'Easy Frontend',
    age: 18,
});

// [
//   ['id', 1],
//   ['name', 'Easy Frontend'],
//   ['age', 18],
// ]
```

6. Object.getOwnPropertyDescriptors()



```
> const student = {
      id: 1,
      name: 'Easy Frontend',
  }
<· undefined
> Object.getOwnPropertyDescriptors(student)
<· ▼{id: {…}, name: {…}} i
    ▼id:
       configurable: true
       enumerable: true
       value: 1
       writable: true
      ▶[[Prototype]]: Object
    ▼name:
       configurable: true
       enumerable: true
       value: "Easy Frontend"
       writable: true
      ▶[[Prototype]]: Object
    ▶[[Prototype]]: Object
```

> Đọc thêm tại: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/getOwnPropertyDescriptors

7. Trailing commas in function parameter lists and calls

Trailing commas in Array

```
[1, 2, 3,].length;  // 3
[1, 2, 3,,].length; // 4

const numberList = [1, 2, 3,];
console.log(numberList); // [1, 2, 3]
```

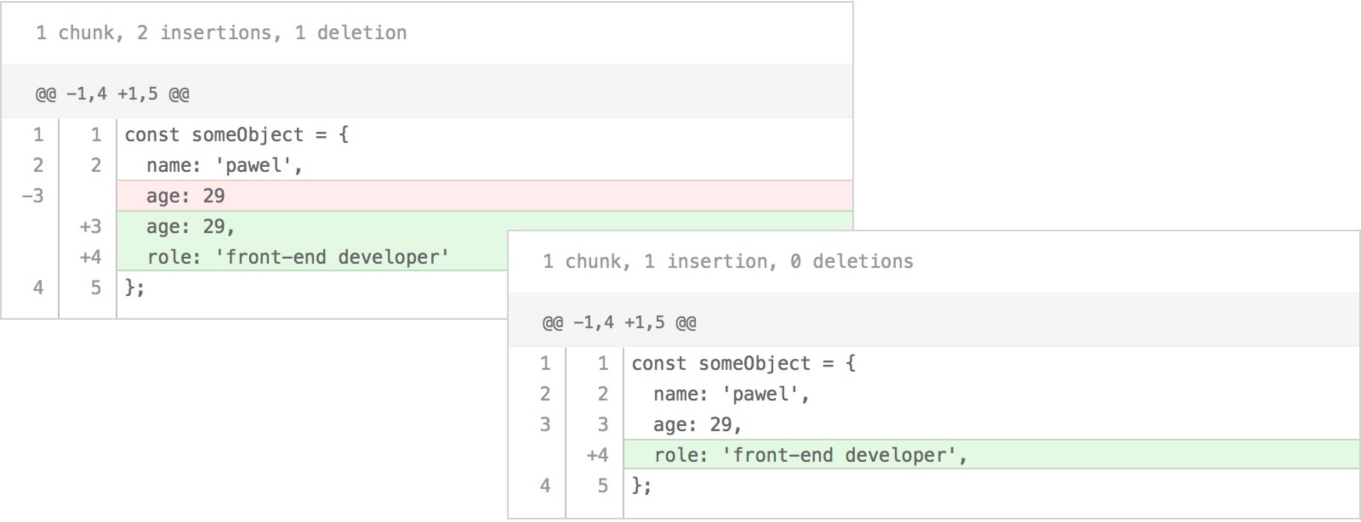
Trailing commas in object (from ECMAScript 5)

```
const student = {
  id: 1,
  name: 'Easy Frontend',
  age: 18, // <-- trailing comma
}
```

Trailing commas in functions (from ECMAScript 2017)

```
// function declaration
function createStudent({
  id,
  name,
  age,
  hobbies,
  isHero, // trailing comma
}) {
  console.log(id, name, age, hobbies, isHero);
}

// function call
createStudent({
  id: 1,
  name: 'Easy Frontend',
  age: 18,
  hobbies: ['music', 'coding'],
  isHero: false, // trailing comma
})
```

Trailing commas when using git



```
1 chunk, 2 insertions, 1 deletion

@@ -1,4 +1,5 @@
 1   1  const someObject = {
 2   2    name: 'pawel',
-3        age: 29
    +3    age: 29,
    +4    role: 'front-end developer'
 4   5  };
```

```
1 chunk, 1 insertion, 0 deletions

@@ -1,4 +1,5 @@
 1   1  const someObject = {
 2   2    name: 'pawel',
 3   3    age: 29,
    +4    role: 'front-end developer',
 4   5  };
```

> Source: https://pawelgrzybek.com/trailing-comma-in-ecmascript2017-function-parameter-list/

Tham khảo

- https://flaviocopes.com/es2017/
- https://exploringjs.com/es2016-es2017/ch_trailing-comma-parameters.html
- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Trailing_commas
- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/getOwnPropertyDescriptors
- https://github.com/tc39/ecmascript_sharedmem/blob/master/TUTORIAL.md

---

Khoá học Javascript cho người mới bắt đầu 2021 🎉

- Tác giả: **Hậu Nguyễn** - Founder Easy Frontend
- Khoá học chỉ được published trên Udemy, không thông qua trung gian.
- Khoá học không bán dạng videos upload trên Google Drive hay bất cứ hình thức nào tương tự.
- Khoá học có nhóm discord để hỗ trợ trong quá trình học tập.

☎️ Liên hệ tác giả để được hỗ trợ:

- ✅ Facebook: https://www.facebook.com/nvhauesmn/
- ✅ Fanpage: https://www.facebook.com/learn.easyfrontend
- ✅ Youtube Channel: https://www.youtube.com/easyfrontend