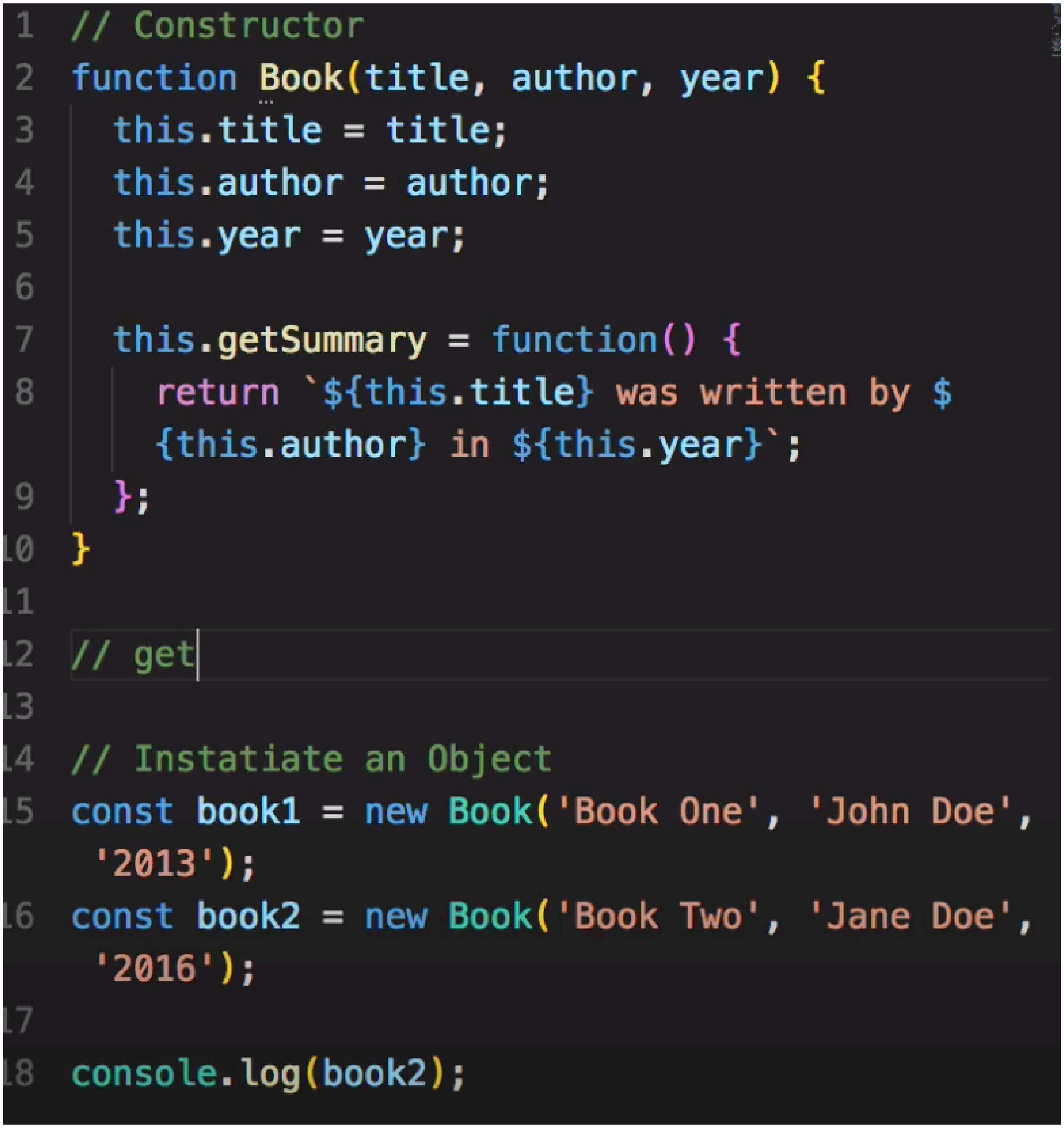
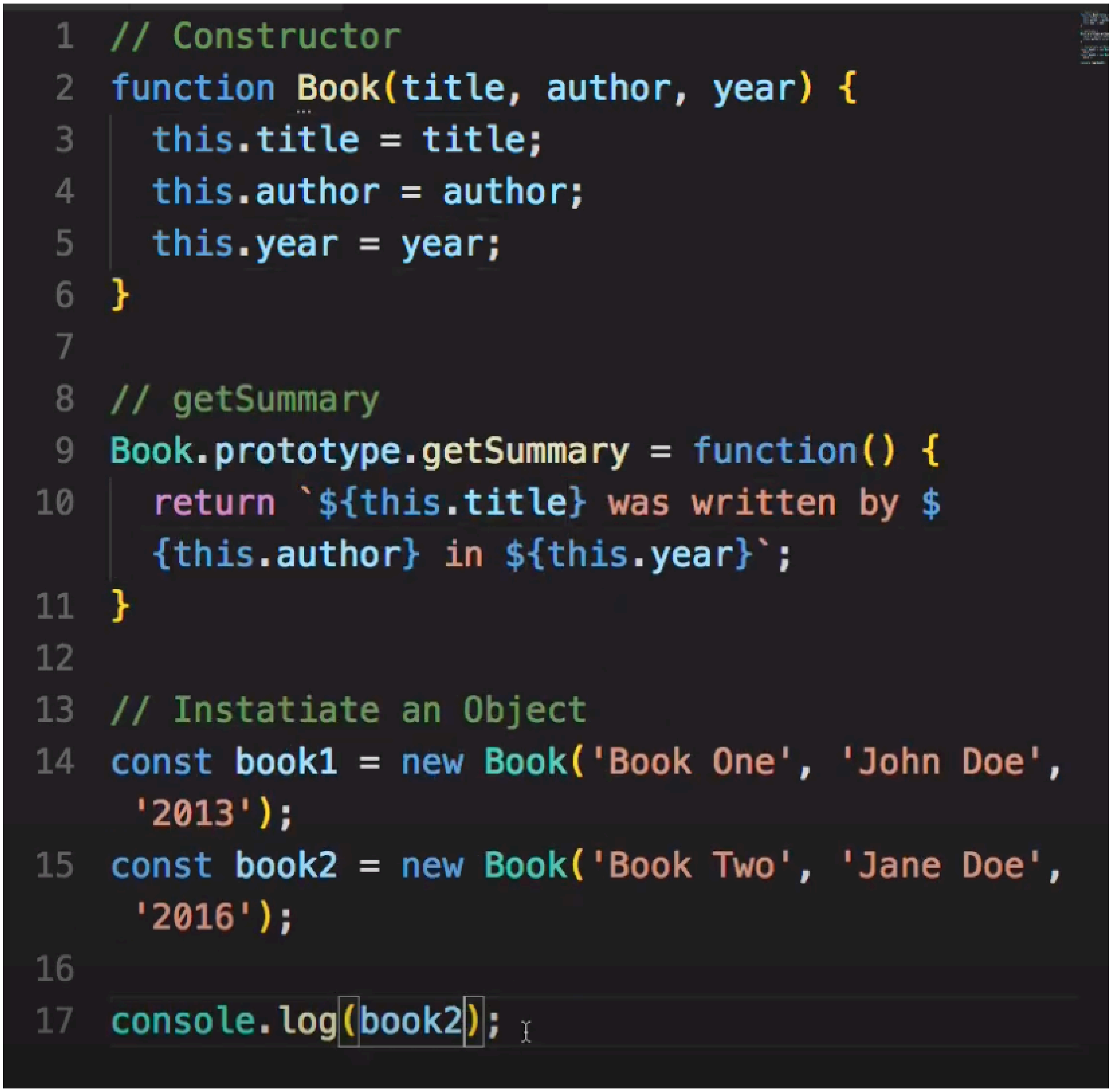
Object-oriented programming (OOP) is a computser programming model that organizes software design around data, or objects, rather than functions and logic.

constructor es5, is basically a function
When this keyword in trick mode, it doesn’t point to global object, undefined.

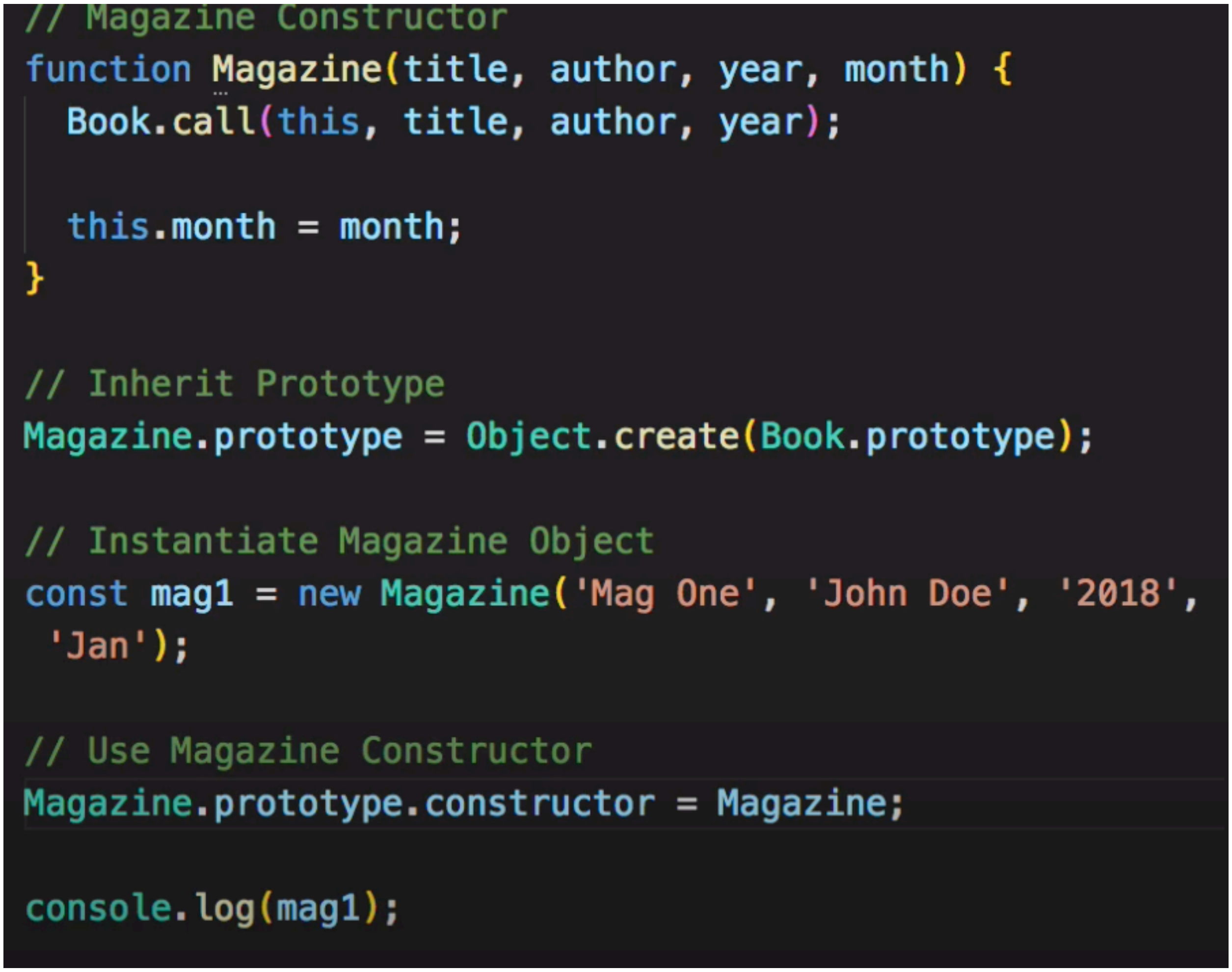
Store common functions inside prototype rather than the constructor object itselfs



```
// Constructor
function Book(title, author, year) {
  this.title = title;
  this.author = author;
  this.year = year;

  this.getSummary = function() {
    return `${this.title} was written by ${this.author} in ${this.year}`;
  };
}

// get

// Instatiate an Object
const book1 = new Book('Book One', 'John Doe', '2013');
const book2 = new Book('Book Two', 'Jane Doe', '2016');

console.log(book2);
```



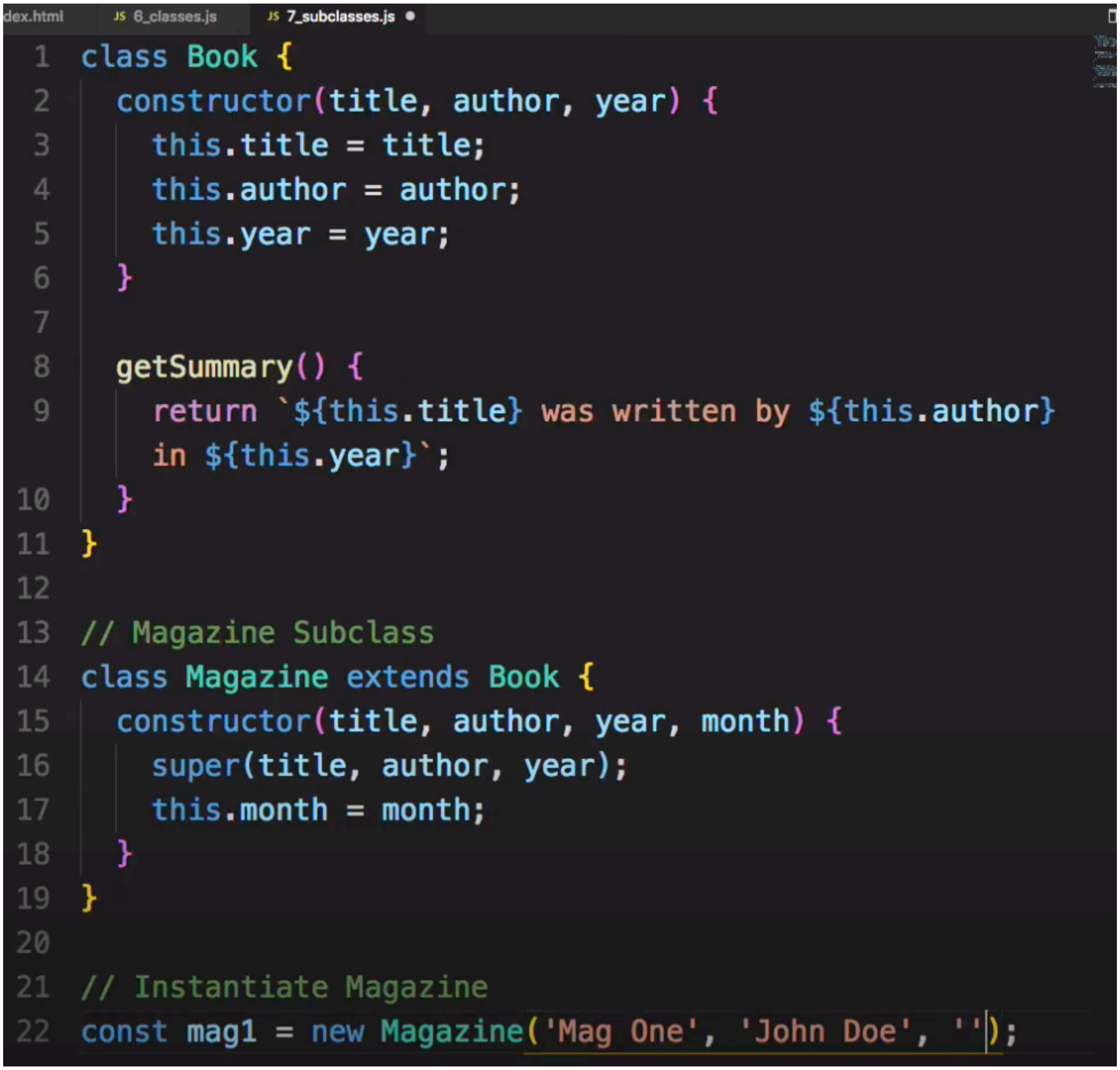
```javascript
// Constructor
function Book(title, author, year) {
  this.title = title;
  this.author = author;
  this.year = year;
}

// getSummary
Book.prototype.getSummary = function() {
  return `${this.title} was written by ${this.author} in ${this.year}`;
}

// Instatiate an Object
const book1 = new Book('Book One', 'John Doe', '2013');
const book2 = new Book('Book Two', 'Jane Doe', '2016');

console.log(book2);
```



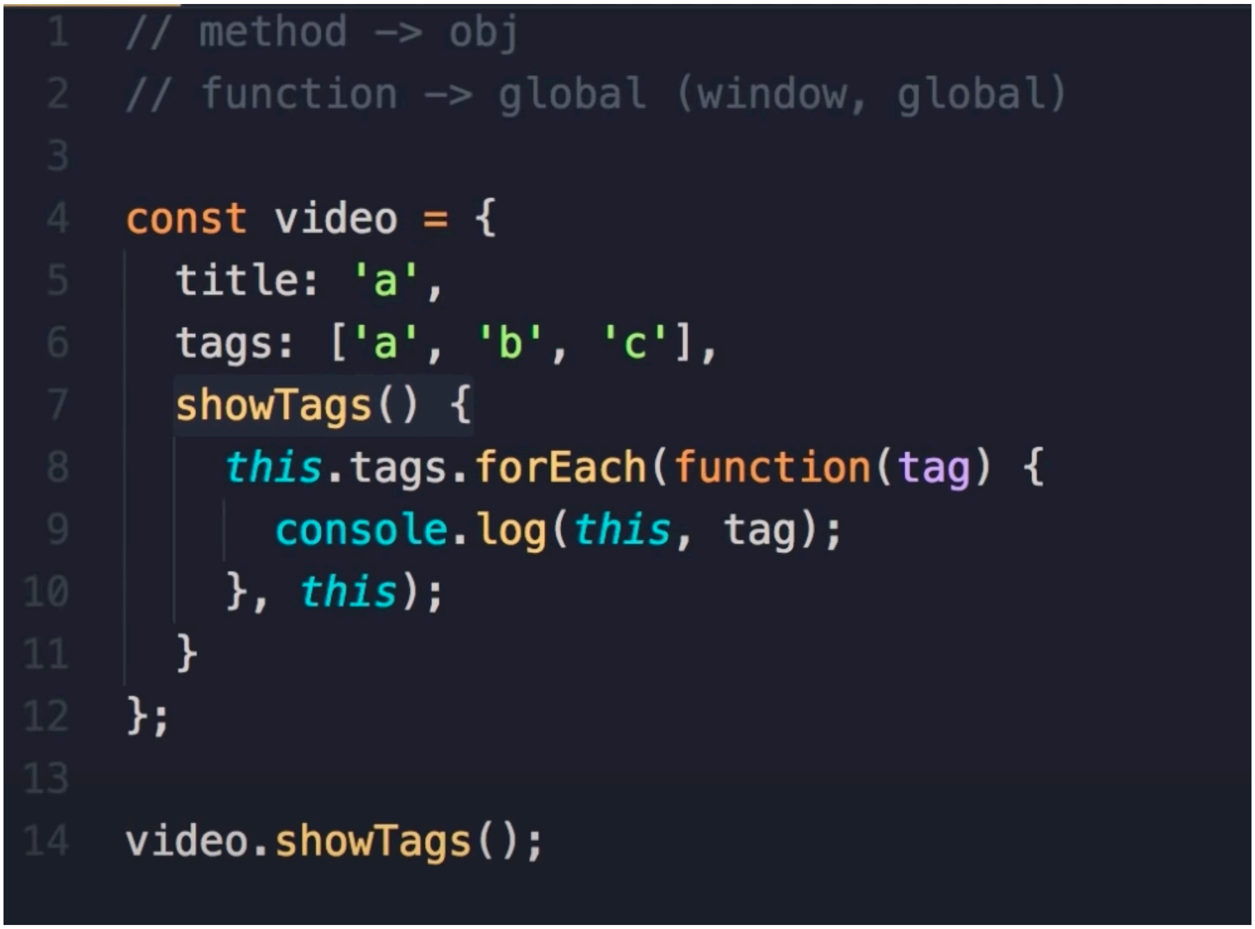
```
// Magazine Constructor
function Magazine(title, author, year, month) {
  Book.call(this, title, author, year);

  this.month = month;
}

// Inherit Prototype
Magazine.prototype = Object.create(Book.prototype);

// Instantiate Magazine Object
const mag1 = new Magazine('Mag One', 'John Doe', '2018',
 'Jan');

// Use Magazine Constructor
Magazine.prototype.constructor = Magazine;

console.log(mag1);
```

Class es6

Put prototype function inside class directly; static functions need no instantiate; static functions/properties belong to the class, not the objects



```javascript
class Book {
  constructor(title, author, year) {
    this.title = title;
    this.author = author;
    this.year = year;
  }

  getSummary() {
    return `${this.title} was written by ${this.author} in ${this.year}`;
  }
}

// Magazine Subclass
class Magazine extends Book {
  constructor(title, author, year, month) {
    super(title, author, year);
    this.month = month;
  }
}

// Instantiate Magazine
const mag1 = new Magazine('Mag One', 'John Doe', '');
```



```javascript
// method -> obj
// function -> global (window, global)

const video = {
  title: 'a',
  tags: ['a', 'b', 'c'],
  showTags() {
    this.tags.forEach(function(tag) {
      console.log(this, tag);
    }, this);
  }
};

video.showTags();
```

Call,apply, bind