(/)

100%

JavaScript - Objects, Scopes and Closures

JavaScript

Amateur

By: Guillaume, CTO at Holberton School

Weight: 1

Your score will be updated once you launch the project review.

Resources

Read or watch:

- JavaScript object basics (/rltoken/GLuVTw6uPEtJoE48v6hh6Q)
- Object-oriented JavaScript (/rltoken/yJvx-KSQmjeLGL_zt2zSRg) (***read all examples!***)
- Class - ES6 (/rltoken/uQ0wKfDktBpsS52MD_V7Fg)
- super - ES6 (/rltoken/or2hUFeNS5YYJRbL_7a9Vw)
- extends - ES6 (/rltoken/DCDrR-o8mV3gaoaxQawYLg)
- Object prototypes (/rltoken/tv0EI26HJfIBpYITYpZswQ)
- Inheritance in JavaScript (/rltoken/yJvx-KSQmjeLGL_zt2zSRg)
- Closures (/rltoken/VUQSL4MeY58nNfFL6yPA_g)
- this/self (/rltoken/uduR9j8AJ4jUrKYiucKGIg)
- Modern JS (/rltoken/LdHi8ovDOIBTHdTBRhVCww)

Learning Objectives

At the end of this project, you are expected to be able to explain to anyone (/rltoken/3qhPKwdfmUr62ypWy-5hgg), **without the help of Google**:

General

- Why JavaScript programming is amazing
- How to create an object in JavaScript

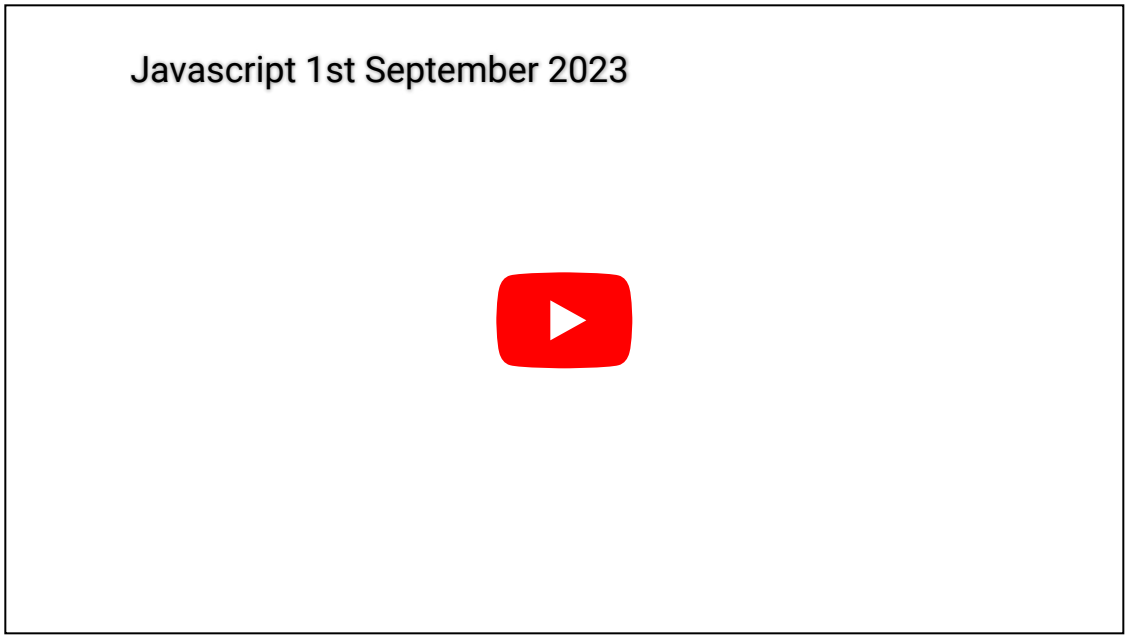
- What `this` means
- What `undefined` means
- Why the variable type and scope is important
- What is a closure
- What is a prototype
- How to inherit an object from another

Requirements

General

- Recommended editors: `Visual studio code`
- All your files will be interpreted on Ubuntu 20.04 LTS using `node` (version 14.x)
- All your files should end with a new line
- A `README.md` file, at the root of the folder of the project, is mandatory
- Your code should be `semistandard` compliant. Rules of Standard (/rltoken/GtopWkNbtPuUsJmuABueTw) + semicolons on top (/rltoken/bbBKzz198LWBw9vx7YFQqA). Also as reference: AirBnB style (/rltoken/1i7dhXHf_z8QlZFkY-y5PA)
- The length of your files will be tested using `wc`
- You are not allowed to use `var`

Intro session for this project



Quiz questions

Great! You've completed the quiz successfully! Keep going! (Show quiz)

Tasks

0. Rectangle #0

mandatory

Write an empty class `Rectangle` that defines a rectangle:

- You must use the `class` notation for defining your class

```
guillaume@ubuntu:~/$ cat 0-main.js
#!/usr/bin/node
const Rectangle = require('./0-rectangle');

const r1 = new Rectangle();
console.log(r1);
console.log(r1.constructor);

guillaume@ubuntu:~/$ ./0-main.js
Rectangle {}
[Class: Rectangle]
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx_javascript`
- Directory: `javascript_objects_scopes_closures`
- File: `0-rectangle.js`

Help | Check your code | >_ Get a sandbox

6/6 pts

1. Rectangle #1

mandatory

Write a class `Rectangle` that defines a rectangle:

- You must use the `class` notation for defining your class
- The constructor must take 2 arguments `w` and `h`
- Initialize the instance attribute `width` with the value of `w`
- Initialize the instance attribute `height` with the value of `h`

0)

```
guillaume@ubuntu:~/$ cat 1-main.js
#!/usr/bin/node
const Rectangle = require('./1-rectangle');

const r1 = new Rectangle(2, 3);
console.log(r1);
console.log(r1.width);
console.log(r1.height);

const r2 = new Rectangle(2, -3);
console.log(r2);
console.log(r2.width);
console.log(r2.height);

const r3 = new Rectangle(2);
console.log(r3);
console.log(r3.width);
console.log(r3.height);

guillaume@ubuntu:~/$ ./1-main.js
Rectangle { width: 2, height: 3 }
2
3
Rectangle { width: 2, height: -3 }
2
-3
Rectangle { width: 2, height: undefined }
2
undefined
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx_javascript`
- Directory: `javascript_objects_scopes_closures`
- File: `1-rectangle.js`

Help | Check your code | >_ Get a sandbox

10/10 pts

2. Rectangle #2

mandatory

Write a class `Rectangle` that defines a rectangle:

- You must use the `class` notation for defining your class
- The constructor must take 2 arguments `w` and `h`
- Initialize the instance attribute `width` with the value of `w`
- Initialize the instance attribute `height` with the value of `h`
- If `w` or `h` is equal to 0 or not a positive integer, create an empty object

0)

```
guillaume@ubuntu:~/$ cat 2-main.js
#!/usr/bin/node
const Rectangle = require('./2-rectangle');

const r1 = new Rectangle(2, 3);
console.log(r1);
console.log(r1.width);
console.log(r1.height);

const r2 = new Rectangle(2, -3);
console.log(r2);
console.log(r2.width);
console.log(r2.height);

const r3 = new Rectangle(2);
console.log(r3);
console.log(r3.width);
console.log(r3.height);

const r4 = new Rectangle(2, 0);
console.log(r4);
console.log(r4.width);
console.log(r4.height);

guillaume@ubuntu:~/$ ./2-main.js
Rectangle { width: 2, height: 3 }
2
3
Rectangle {}
undefined
undefined
Rectangle {}
undefined
undefined
Rectangle {}
undefined
undefined
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx_javascript`
- Directory: `javascript_objects_scopes_closures`
- File: `2-rectangle.js`

Help | Check your code | >_ Get a sandbox

10/10 pts

3. Rectangle #3

mandatory

Write a class `Rectangle` that defines a rectangle:
(/)

- You must use the `class` notation for defining your class
- The constructor must take 2 arguments: `w` and `h`
- Initialize the instance attribute `width` with the value of `w`
- Initialize the instance attribute `height` with the value of `h`
- If `w` or `h` is equal to 0 or not a positive integer, create an empty object
- Create an instance method called `print()` that prints the rectangle using the character `X`

```
guillaume@ubuntu:~/$ cat 3-main.js
#!/usr/bin/node
const Rectangle = require('./3-rectangle');

const r1 = new Rectangle(2, 3);
r1.print();

const r2 = new Rectangle(10, 5);
r2.print();

guillaume@ubuntu:~/$ ./3-main.js
XX
XX
XX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx_javascript`
- Directory: `javascript_objects_scopes_closures`
- File: `3-rectangle.js`

Help | Check your code | >_ Get a sandbox **10/10** pts

4. Rectangle #4

mandatory

Write a class `Rectangle` that defines a rectangle:

- You must use the `class` notation for defining your class
- The constructor must take 2 arguments: `w` and `h`
- Initialize the instance attribute `width` with the value of `w`
- Initialize the instance attribute `height` with the value of `h`
- If `w` or `h` is equal to 0 or not a positive integer, create an empty object
- Create an instance method called `print()` that prints the rectangle using the character `X`

(/)

- Create an instance method called `rotate()` that exchanges the `width` and the `height` of the rectangle
- Create an instance method called `double()` that multiples the `width` and the `height` of the rectangle by 2

```
guillaume@ubuntu:~/$ cat 4-main.js
#!/usr/bin/node
const Rectangle = require('./4-rectangle');

const r1 = new Rectangle(2, 3);
console.log('Normal:');
r1.print();

console.log('Double:');
r1.double();
r1.print();

console.log('Rotate:');
r1.rotate();
r1.print();

guillaume@ubuntu:~/$ ./4-main.js
Normal:
XX
XX
XX
Double:
XXXX
XXXX
XXXX
XXXX
XXXX
XXXX
Rotate:
XXXXXX
XXXXXX
XXXXXX
XXXXXX
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx_javascript`
- Directory: `javascript_objects_scopes_closures`
- File: `4-rectangle.js`

Help | Check your code | >_ Get a sandbox **10/10** pts

5. Square #0

mandatory

Write a class `Square` that defines a square and inherits from `Rectangle` of `4-rectangle.js`:

- You must use the `class` notation for defining your class and `extends`
- The constructor must take 1 argument: `size`
- The constructor of `Rectangle` must be called (by using `super()`)

```
guillaume@ubuntu:~/$ cat 5-main.js
#!/usr/bin/node
const Square = require('./5-square');

const s1 = new Square(4);
s1.print();
s1.double();
s1.print();

guillaume@ubuntu:~/$ ./5-main.js
XXXX
XXXX
XXXX
XXXX
XXXXXXXX
XXXXXXXX
XXXXXXXX
XXXXXXXX
XXXXXXXX
XXXXXXXX
XXXXXXXX
XXXXXXXX
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx_javascript`
- Directory: `javascript_objects_scopes_closures`
- File: `5-square.js`

Help | Check your code | >_ Get a sandbox

10/10 pts

6. Square #1

mandatory

Write a class `Square` that defines a square and inherits from `Square` of `5-square.js`:

- You must use the `class` notation for defining your class and `extends`
- Create an instance method called `charPrint(c)` that prints the rectangle using the character `c`
  - If `c` is `undefined`, use the character `X`

7)

```
guillaume@ubuntu:~/$ cat 6-main.js
#!/usr/bin/node
const Square = require('./6-square');

const s1 = new Square(4);
s1.charPrint();

s1.charPrint('C');

guillaume@ubuntu:~/$ ./6-main.js
XXXX
XXXX
XXXX
XXXX
CCCC
CCCC
CCCC
CCCC
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx_javascript`
- Directory: `javascript_objects_scopes_closures`
- File: `6-square.js`

Help | Check your code | >_ Get a sandbox

10/10 pts

Score

100%

Congratulations! You made it!

The next project will be available on Saturday, Sep 9th.

Go to home (/)

Previous project (/projects/2079)

(/)