

0x00. ES6 Basics

[JavaScript Planning \(/planning/me\)](#)

Weight: 1

[Projects \(/projects/current\)](#)

Project over - took place from Jun 24, 2024 6:00 AM to Jun 25, 2024 6:00 AM



An auto-review will automatically make corrections (/to_review)

[Concepts Evaluation quizzes \(/dashboards/my_current_evaluation_quizzes\)](#)

For this project, we expect you to look at these concepts:



- JavaScript programming (/concepts/852)
- Curriculums (/dashboards/my_curriculums)
- Software Linter (/concepts/542)

[Concepts \(/concepts\)](#)[Conference rooms \(/dashboards/video_rooms\)](#)[Servers \(/servers\)](#)[Sandboxes \(/user_containers/current\)](#)[Tools \(/dashboards/my_tools\)](#)[Video on demand \(/dashboards/videos\)](#)

ES6 arrow functions

[Peers \(/users/peers\)](#)

Resources

[Read or Watch \(https://discord.com/app\)](https://discord.com/app)

- ECMAScript 6 - ECMAScript 2015 (/rltoken/HRvh-7X2k2JmPu2XMuvInQ)
- Statements and declarations (/rltoken/bu6OK8Wbzzxr04Si-qup-w)
- Arrow functions (/rltoken/kn70en_i7XsVI9PUhAK1fQ)
- Default parameters (/rltoken/e1-hBHivLFWOip87Lc4Jfw)
- Rest parameter (/rltoken/TB-tbhDM8tPkVIS4_Tw_rw)
- My profile (/users/my_profile)
- Demystifying ES6 Iterables & Iterators (/rltoken/MyAmrYqmTSHF66QmE5D8bA)



Learning Objectives

At the end of this project, you are expected to be able to explain to anyone (/rltoken/TyqtB2dhn7oqGwD-ZNvFwx), **without the help of Google**.



- What ES6 is
- New features introduced in ES6
- The difference between a constant and a variable



- Block Scoped Variables (/me)
- Arrow functions and function parameters default to them
- Rest and spread function parameters



- String Interpolation (/current)
- Object creation and their properties in ES6
- Iterators and for-of loops



QA Reviews I can make (/corrections/to_review)

Requirements



Evaluation quizzes (/dashboards/my_current_evaluation_quizzes)

General



- All your files will be executed on Ubuntu 18.04 LTS using NodeJS 12.11.x
- Allowed editors: vi, vim, emacs, Visual Studio Code
- 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 use the js extension
- Your code will be tested using the Jest Testing Framework (/rltoken/K0bBQgH1IKAABeUzTmzxAA)
- Your code will be analyzed using the linter ESLint (/rltoken/iKElBm_Xlj10nllm5OILKA) along with specific rules that we'll provide



Conference rooms (/dashboards/video_rooms)

- All of your functions must be exported



Servers (/servers)

Setup



Install NodeJS 12.11.x

(in your home directory):



Tools (/dashboards/my_tools)

```
curl -sL https://deb.nodesource.com/setup_12.x -o nodesource_setup.sh
sudo bash nodesource_setup.sh
```



```
sudo apt install nodejs
```

```
$ nodejs -v
v12.11.1
```



```
$ npm -v
6.11.3
```

Peers (/users/peers)



Discord (https://discord.com/app)

Install Jest, Babel, and ESLint



in your project directory, install Jest, Babel and ESLint by using the supplied package.json and run npm install.

My Profile (/users/my_profile)

Configuration files

Add the files below to your project directory

package.json

Home(/)

Click to show/hide file contents



babel.config.js

My Planning(/planning/me)



Click to show/hide file contents
Projects(/projects/current)

.eslintrc.js



QA Reviews I can make(/corrections/to_review)
Click to show/hide file contents

Finally, evaluation quizzes(/dashboards/my_current_evaluation_quizzes)

Don't forget to run `npm install` from the terminal of your project folder to install all necessary project dependencies.



Curriculumums(/dashboards/my_curriculumums)



Tasks

Concepts(/concepts)



Conferences

Conferences rooms(/dashboards/video_rooms)

mandatory



Modify
Servers(/servers)

- function `taskFirst` to instantiate variables using `const`
- function `taskNext` to instantiate variables using `let`



Sandboxes(/user_containers/current)



```
export function taskFirst() {  
  var task = 'I prefer const when I can.';  
  return task;  
}
```

Tools(/dashboards/my_tools)



```
export function getLast() {  
  return ' is okay';  
}
```

Video on demand(/dashboards/videos)



```
export function taskNext() {  
  var combination = 'But sometimes let';  
  combination += getLast();  
}
```

Peers(/users/peers)



```
Discord(https://discord.com/app)  
return combination;  
}
```

Discord(https://discord.com/app)



Execution example:

My Profile(/users/my_profile)

```
bob@dylan:~$ cat 0-main.js
import { taskFirst, taskNext } from './0-constants.js';

(
  console.log(`${taskFirst()} ${taskNext()}`);
)
```

Home(/)
bob@dylan:~\$
bob@dylan:~\$ npm run dev 0-main.js
I prefer const when I can. But sometimes let is okay
My Planing(/planning/me)

Repo: Projects(/projects/current)
✓ • GitHub repository: alx-frontend-javascript
• QA Reviews I can make(/corrections/to_review)
• Directory: 0x00-ES6_basic
• File: 0-constants.js
? Evaluation quizzes(/dashboards/my_current_evaluation_quizzes)
Check submission >_ Get a sandbox

1 Block Scope mandatory

Given what you've read about `var` and hoisting, modify the variables inside the function `taskBlock` so that the variables aren't overwritten inside the conditional block.

```
export default function taskBlock(trueOrFalse) {
  var task1 = true;
  var task2 = true;

  if (trueOrFalse){
    var task = true;
    var task2 = false;
  }

  return [task, task2];
}
```

Execution:
Video on demand(/dashboards/videos)
bob@dylan:~\$ cat 1-main.js
import taskBlock from './1-block-scoped.js';

```
console.log(taskBlock(true));
console.log(taskBlock(false));
bob@dylan:~$
bob@dylan:~$ npm run dev 1-main.js
[ false, true ]
[ false, true ]
bob@dylan:~$
```



Repo: My Profile(/users/my_profile)

- GitHub repository: alx-frontend-javascript
- Directory: 0x00-ES6_basic
- File: 1-block-scoped.js



Home(/)

Check submission

> Get a sandbox



Arrow functions

My Planning(/planning/me)

mandatory



Projects(/projects/current)

Rewrite the following standard function to use ES6's arrow syntax of the function `add` (it will be an anonymous function after)



QA Reviews(/can-make/corrections/to-review)

```
export default function getNeighborhoodsList() {
  this.sanFranciscoNeighborhoods = ['SOMA', 'Union Square'];

  const self = this;
  this.evaluateQuizzes(/dashboards/my_curriculum/quizzes)
    self.sanFranciscoNeighborhoods.push(newNeighborhood);
  return self.sanFranciscoNeighborhoods;
};
```



Curriculums(/dashboards/my_curriculums)

Execution:



Concepts(/concepts)

```
bob@dylan:~$ cat 2-main.js
import getNeighborhoodsList from './2-arrow.js';
```



Conference rooms(/dashboards/video_rooms)

```
const neighborhoodsList = new getNeighborhoodsList();
const res = neighborhoodsList.addNeighborhood('Noe Valley');
```



Servers(/servers)

```
console.log(res);
bob@dylan:~$
bob@dylan:~$ npm run dev 2-main.js
[ 'SOMA', 'Union Square', 'Noe Valley' ]
Sandboxes(/user_containers/current)
bob@dylan:~$
```



Tools(/dashboards/my_tools)

Repo:



Video on demand(/dashboards/videos)

- GitHub repository: alx-frontend-javascript
- Directory: 0x00-ES6_basic
- File: 2-arrow.js



Peers(/users/peers)

Check submission

> Get a sandbox



Discord(<https://discord.com/app>)

3. Parameter defaults

mandatory

Condense the internals of the following function to 1 line - without changing the name of each function.

My Profile(/users/my_profile)

Hint: The key here to define default parameter values for the function parameters.



```
export default function getSumOfHoods(initialNumber, expansion1989, expansion2019) {  
  if (expansion1989 === undefined) {  
    expansion1989 = 89;  
  }  
  
  if (expansion2019 === undefined) {  
    expansion2019 = 19;  
  }  
  
  return initialNumber + expansion1989 + expansion2019;  
}
```



QA Reviews I can make(/corrections/to_review)



```
bob@dylan:~$ cat 3-main.js  
import { getSumOfHoods } from './3-default-parameter.js';  
  
console.log(getSumOfHoods(34));  
console.log(getSumOfHoods(34, 3));  
console.log(getSumOfHoods(34, 3, 4));  
bob@dylan:~$  
bob@dylan:~$ npm run dev 3-main.js  
142  
56  
41  
bob@dylan:~$  
Conference rooms(/dashboards/video_rooms)
```



Servers(/servers)

- GitHub repository: alx-frontend-javascript
- Directory: 0x00-ES6_basic
- File: 3-default-parameter.js



Tools(/dashboards/my_tools)
Check submission Get a sandbox



Rest parameters syntax for functions

mandatory

Modify the following function to return the number of arguments passed to it using the rest parameter syntax



Peers(/users/peers)

```
export default function returnHowManyArguments() {  
  
}
```



Discord(<https://discord.com/app>)

Example:



```
> returnHowManyArguments("Hello", "Holberton", 2020);  
3  
> My Profile(/users/my_profile)
```

Execution:

```
bob@dylan:~$ cat 4-main.js
import returnHowManyArguments from './4-rest-parameter.js';

console.log(returnHowManyArguments("one"));
console.log(returnHowManyArguments("one", "two", 3, "4th"));
bob@dylan:~$
bob@dylan:~$ npm run dev 4-main.js
1
4
bob@dylan:~$
```

✓Repo: QA Reviews I can make(/corrections/to_review)

- GitHub repository: alx-frontend-javascript
- ? • Directory: 0x00-ES6_basic (/dashboards/my_current_evaluation_quizzes)
- File: 4-rest-parameter.js

Check submission Get a sandbox

5. The wonders of spread syntax

mandatory

Concepts(/concepts)

Using spread syntax, concatenate 2 arrays and each character of a string by modifying the function below. Your function body should be one line long.

```
export default function concatArrays(array1, array2, string) {
  // ...
}
```

Execution:

```
Sandboxes(/user_containers/current)
bob@dylan:~$ cat 5-main.js
import concatArrays from './5-spread-operator.js';

Tools(/dashboards/my_tools)
console.log(concatArrays(['a', 'b'], ['c', 'd'], 'Hello'));

bob@dylan:~$
Video on demand(/dashboards/videos)
bob@dylan:~$ npm run dev 5-main.js
[
  'a', 'b', 'c',
  'd', 'H', 'e',
  'l', 'l', 'o'
]
bob@dylan:~$

Discord(https://discord.com/app)
```

Repo:

- GitHub repository: alx-frontend-javascript
- Directory: 0x00-ES6_basic (/users/my_profile)

- File: 5-spread-operator.js

[Check submission](#)[Get a sandbox](#)[Home\(/\)](#)

6. Take advantage of template literals

mandatory

Rewrite the return statement to use a template literal so you can substitute the variables you've defined.



```
export default function getSanFranciscoDescription() {
  const year = 2017;
  const budget = {
    income: '$119,868',
    gdp: '$154.2 billion',
    capita: '$178,479',
  };
  return `As of ${year}, it was the seventh-highest income county in the United States, with a per capita personal income of ${budget.income}. As of 2015, San Francisco proper had a GDP of ${budget.gdp}, and a GDP per capita of ${budget.capita}.';
}
```

Execution:



Conference rooms(/dashboards/video_rooms)

```
bob@dylan:~$ cat 6-main.js
```

```
import getSanFranciscoDescription from './6-string-interpolation.js';
```



Servers(/servers)

```
console.log(getSanFranciscoDescription());
```



Sandbox(/user_containers/current)

```
bob@dylan:~$ npm run dev 6-main.js
```

```
As of 2017, it was the seventh-highest income county in the United States, with a per capita personal income of $119,868. As of 2015, San Francisco proper had a GDP of $154.2 billion, and a GDP per capita of $178,479.
```

```
bob@dylan:~$
```



Video on demand(/dashboards/videos)

Repo:



- GitHub repository: alx-frontend-javascript
- Directory: 0x00-ES6_basic
- File: 6-string-interpolation.js



Discord(<https://discord.com/app>)

[Check submission](#)[Get a sandbox](#)[My Profile\(/users/my_profile\)](#)

7. Object property value shorthand syntax

mandatory

Notice how the keys and the variable names are the same?

Modify the following function's `budget` object to simply use the keyname instead.



Home(/)



My Planning(/planning/me)
income: income,



gdp: gdp,
capita: capita,
Projects(/projects/current)



return budget;
QA Reviews I can make(/corrections/to_review)

Execution:



Evaluation quizzes(/dashboards/my_current_evaluation_quizzes)

bob@dylan:~\$ cat 7-main.js

```
import getBudgetObject from './7-getBudgetObject.js';
```



```
console.log(getBudgetObject(400, 700, 900));
```

bob@dylan:~\$



bob@dylan:~\$ npm run dev 7-main.js

```
{ income: 400, gdp: 700, capita: 900 }
```

bob@dylan:~\$



Conference rooms(/dashboards/video_rooms)



Repo:

Servers(/servers)

- GitHub repository: alx-frontend-javascript

- Directory: 0x00-ES6_basic



- Sandboxes(/user_containers/current)



File: 7-getBudgetObject.js



Tools(/dashboards/my_tools)

Check submission

Get a sandbox



No need to create empty objects before adding in properties

mandatory

Video on demand(/dashboards/videos)

Rewrite the `getBudgetForCurrentYear` function to use ES6 computed property names on the `budget` object




Peers(/users/peers)




Discord(<https://discord.com/app>)



My Profile(/users/my_profile)




```
function getCurrentYear() {
  const date = new Date();
  return date.getFullYear();
}
```



```
export default function getBudgetForCurrentYear(income, gdp, capita) {
  const budget = {};


  budget[`${income}-${getCurrentYear()}`] = income;
  budget[`${gdp}-${getCurrentYear()}`] = gdp;
  budget[`${capita}-${getCurrentYear()}`] = capita;

  return budget;
}
```




QA Reviews I can make(/corrections/to_review)

Execution:




```
bob@dylan:~$ cat 8-main.js
import getBudgetForCurrentYear from './8-getBudgetCurrentYear.js';


console.log(getBudgetForCurrentYear(2100, 5200, 1090));
```



```
bob@dylan:~$ npm run dev 8-main.js
```






```
{ 'income-2021': 2100, 'gdp-2021': 5200, 'capita-2021': 1090 }
```



Conference rooms(/dashboards/video_rooms)

Repo:


- 
- GitHub repository: alx-frontend-javascript
 - Directory: 0x00-ES6_basic
 - File: 8-getBudgetCurrentYear.js
- 
- Sandboxes(/user_containers/current)



Tools(/dashboards/my_tools)

9. ES6 method properties

mandatory



Video on demand(/dashboards/videos)

Rewrite `getFullBudgetObject` to use ES6 method properties in the `fullBudget` object



Peers(/users/peers)



Discord(<https://discord.com/app>)



My Profile(/users/my_profile)



```
import getBudgetObject from './7-getBudgetObject.js';

export default function getFullBudgetObject(income, gdp, capita) {
  const budget = getBudgetObject(income, gdp, capita);
  const fullBudget = {
    Home(/),
    getIncomeInDollars: function (income) {
      return `${income}`;
    }, My Planning(/planning/me)
    getIncomeInEuros: function (income) {
      return `${income} euros`;
    }, Projects(/projects/current)
  };

  return fullBudget;
}
```



Execution: Evaluation quizzes(/dashboards/my_current_evaluation_quizzes)



```
bob@dylan:~$ cat 9-main.js
import getFullBudgetObject from './9-getFullBudget.js';

const fullBudget = getFullBudgetObject(20, 50, 10);

console.log(fullBudget.getIncomeInDollars(fullBudget.income));
console.log(fullBudget.getIncomeInEuros(fullBudget.income));
```



```
bob@dylan:~$
bob@dylan:~$ npm run dev 9-main.js
$20
20 euros
bob@dylan:~$
```



Sandboxes(/user_containers/current)

Repo:



- GitHub repository: alx-frontend-javascript
- Directory: 0x00-ES6_basic
- File: 9-getFullBudget.js



Video on demand(/dashboards/videos)

Check submission

Get a sandbox



For of Loops

Peers(/users/peers)

mandatory



Rewrite the function `appendToEachArrayValue` to use ES6's `for...of` operator. And don't forget that `var` is not ES6-friendly.

Discord(<https://discord.com/app>)



My Profile(/users/my_profile)

```

export default function appendToArrayValue(array, appendString) {
  for (var idx in array) {
    (/) var value = array[idx];
    array[idx] = appendString + value;
  }
}

Home(/)
return array;
}

My Planning(/planning/me)

```

Execution:

```

Projects(/projects/current)
bob@dylan:~$ cat 10-main.js
import appendToArrayValue from './10-loops.js';

QA Reviews I can make(/corrections/to_review)
console.log(appendToArrayValue(['appended', 'fixed', 'displayed'], 'correctly
-'));

Evaluation quizzes(/dashboards/my_current_evaluation_quizzes)
bob@dylan:~$
bob@dylan:~$ npm run dev 10-main.js
[ 'correctly appended', 'correctly fixed', 'correctly displayed' ]
bob@dylan:~$

Curriculums(/dashboards/my_curriculums)

```

Repo:

- Concepts(/concepts)
 - GitHub repository: alx-frontend-javascript
 - Directory: 0x00-ES6_basic
- File: 10-loops.js

Servers(/servers) [Get a sandbox](#)

> 11. Iterator

mandatory

Sandboxes(/user_containers/current)

Write a function named `createEmployeesObject` that will receive two arguments:

- Tools(/dashboards/my_tools)
 - departmentName (String)
 - employees (Array of Strings)

```

Video on demand(/dashboards/videos)
export default function createEmployeesObject(departmentName, employees) {

}

```

Peers(/users/peers)
The function should return an object with the following format:

```

{
  Discord(https://discord.com/app)
  $departmentName: [
    $employees,
  ],
}

```




Execution: My Profile(/users/my_profile)



```
bob@dylan:~$ cat 11-main.js
import createEmployeesObject from './11-createEmployeesObject.js';


(
  console.log(createEmployeesObject("Software", [ "Bob", "Sylvie" ]));

```

```
bob@dylan:~$
bob@dylan:~$ npm run dev 11-main.js
{ Software: [ 'Bob', 'Sylvie' ] }
bob@dylan:~$
```

 **Repo:** Projects(/projects/current)


  QA Reviews I can make(/corrections/to_review)


 Evaluation quizzes(/dashboards/my_current_evaluation_quizzes)


12. Let's create a report object mandatory

Write a function named `createReportObject` whose parameter, `employeesList`, is the return value of the previous function `createEmployeesObject`.

```
export default function createReportObject(employeesList) {
  Conference rooms(/dashboards/video_rooms)
}
```

 `createReportObject` should return an object containing the key `allEmployees` and a method property called `getNumberOfDepartments`.

 `allEmployees` is a key that maps to an object containing the department name and a list of all the employees in that department. If you're having trouble, use the spread syntax.

 The method property receives `employeesList` and returns the number of departments. I would suggest thinking back to the ES6 method property syntax.

```
{
  Video on demand(/dashboards/videos)
  allEmployees: {
    engineering: [
      'John Doe',
      'Guillaume Salva',
    ],
    Peers(/users/peers)
  },
};
Discord(https://discord.com/app)
```

Execution:

My Profile(/users/my_profile)

```
bob@dylan:~$ cat 12-main.js
import createEmployeesObject from './11-createEmployeesObject.js';
import createReportObject from './12-createReportObject.js';

const employees = {
  // Home(/)
  ...createEmployeesObject('engineering', ['Bob', 'Jane']),
  ...createEmployeesObject('marketing', ['Sylvie'])
};

// My Planning(/planning/me)
const report = createReportObject(employees);
console.log(report.allEmployees);

// Projects(/projects/current)
console.log(report.projectsOfDepartments(report.allEmployees));

bob@dylan:~$
bob@dylan:~$ cat 13-review.js
import { createReportObject } from './12-createReportObject.js';
import { createEmployeesObject } from './11-createEmployeesObject.js';

const report = createReportObject(
  { engineering: [ 'Bob', 'Jane' ], marketing: [ 'Sylvie' ] }
);

// Evaluation quizzes(/dashboards/my_current_evaluation_quizzes)
```

Repo:

- Curriculum(/dashboards/my_curriculums)
- GitHub repository: alx-frontend-javascript
- Directory: 0x00-ES6_basic
- File: 12-projectsOfDepartments.js

Check some conference rooms (/dashboards/video_rooms)

Servers(/servers)

Sandboxes(/user_containers/current)

Done with the mandatory tasks? Unlock 2 advanced tasks now!

Tools(/dashboards/my_tools)

Video on demand(/dashboards/videos)

Copyright © 2024 ALX. All rights reserved.

Peers(/users/peers)

Discord(<https://discord.com/app>)



My Profile(/users/my_profile)