

A close-up photograph of several laboratory glass flasks filled with different colored liquids. The flask in the foreground is purple and has markings for 'APPROX. VOL' and numerical values 200, 300, 400, and 500. Behind it are flasks containing yellow, green, and red liquids. The background is blurred.

Array Mutator API

Os **mutator methods** quando invocados
modificam o array

- **push**: Adiciona um elemento no final
- **pop**: Remove um elemento do final
- **unshift**: Adiciona um elemento no início
- **shift**: Remove um elemento do início
- **splice**: Remove, substitui ou adiciona um ou mais elementos em uma determinada posição
- **sort**: Ordena os elementos de acordo com a função de ordenação
- **reverse**: Inverte a ordem dos elementos
- **fill**: Preenche os elementos de acordo com a posição de início e fim

The screenshot shows a macOS desktop environment with a terminal window open. The terminal window has a title bar "array_mutator_api_1.js — javascriptmasterclass". The main pane of the terminal displays the output of running the JavaScript file "array_mutator_api_1.js" using the command "node".

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_1.js
[ 'Python', 'C', 'Java' ]
4
5
[ 'Python', 'C', 'Java', 'Ruby', 'Go' ]
Go
Ruby
[ 'Python', 'C', 'Java' ]
rodrigobranas:javascriptmasterclass $
```

The screenshot shows a Mac OS X desktop environment with a terminal window open. The title bar of the terminal window reads "array_mutator_api_2.js — javascriptmasterclass". The main pane of the terminal displays the output of running the JavaScript file:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_2.js
[ 'Python', 'C', 'Java' ]
4
5
[ 'Go', 'Ruby', 'Python', 'C', 'Java' ]
Go
Ruby
[ 'Python', 'C', 'Java' ]
rodrigobranas:javascriptmasterclass $
```

The terminal window has a standard OS X interface with red, yellow, and green close buttons at the top left. It includes standard OS X window controls (minimize, maximize, close) at the top right. The status bar at the bottom right shows "2: bash". There are also standard OS X window controls (minimize, maximize, close) at the very top right of the window frame.

array_mutator_api_3.js — javascriptmasterclass

JS array_mutator_api_3.js ×

```
1 const languages = ["Python", "C", "Java"];
2 console.log(languages);
3 console.log(languages.splice(1, 1));
4 console.log(languages);
5 console.log(languages.splice(1, 0, "C++", "C#"));
6 console.log(languages);
7 console.log(languages.splice(1, 2, "C"));
8 console.log(languages);
```

TERMINAL ... 2: bash

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_3.js
[ 'Python', 'C', 'Java' ]
[ 'C' ]
[ 'Python', 'Java' ]
[]
[ 'Python', 'C++', 'C#', 'Java' ]
[ 'C++', 'C#' ]
[ 'Python', 'C', 'Java' ]
rodrigobranas:javascriptmasterclass $
```

A screenshot of a Mac OS X desktop environment. On the left is a code editor window titled "array_mutator_api_4.js — javascriptmasterclass". The code editor contains the following JavaScript code:

```
JS array_mutator_api_4.js x
array_mutator_api_4.js — javascriptmasterclass

1 const languages = [
2   {
3     name: "Python",
4     year: 1991
5   },
6   {
7     name: "C",
8     year: 1972
9   },
10  {
11    name: "Java",
12    year: 1995
13  }
14];
15languages.sort(function (a, b) {
16  return -1;
17});
18console.log(languages);
19
```

The code defines an array of objects representing programming languages, sorts it in ascending order by year, and logs the result to the console.

To the right of the code editor is a terminal window titled "TERMINAL" with tab "2: bash". The terminal shows the command being run and its output:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_4.js
[ { name: 'Python', year: 1991 },
  { name: 'C', year: 1972 },
  { name: 'Java', year: 1995 } ]
rodrigobranas:javascriptmasterclass $
```

A screenshot of a Mac OS X desktop environment. On the left is a code editor window titled "array_mutator_api_5.js — javascriptmasterclass". The code editor contains the following JavaScript code:

```
JS array_mutator_api_5.js x
array_mutator_api_5.js — javascriptmasterclass

1 const languages = [
2   {
3     name: "Python",
4     year: 1991
5   },
6   {
7     name: "C",
8     year: 1972
9   },
10  {
11    name: "Java",
12    year: 1995
13  }
14];
15languages.sort(function (a, b) {
16  return 1;
17});
18console.log(languages);
19
```

The code defines an array of objects representing programming languages, sorts it in ascending order by year, and logs the result to the console.

To the right of the code editor is a terminal window titled "TERMINAL" with the identifier "2: bash". The terminal shows the command being run and its output:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_5.js
[ { name: 'Java', year: 1995 },
  { name: 'C', year: 1972 },
  { name: 'Python', year: 1991 } ]
rodrigobranas:javascriptmasterclass $
```

The screenshot shows a Mac OS X desktop environment with a terminal window open. The terminal window has tabs labeled "array_mutator_api_6.js — javascriptmasterclass" and "2: bash". The code editor tab contains the following JavaScript code:

```
JS array_mutator_api_6.js ×
array_mutator_api_6.js — javascriptmasterclass
1 const languages = [
2   {
3     name: "Python",
4     year: 1991
5   },
6   {
7     name: "C",
8     year: 1972
9   },
10  {
11    name: "Java",
12    year: 1995
13  }
14];
15languages.sort(function (a, b) {
16  return a.year - b.year;
17});
18console.log(languages);
19
```

The terminal session shows the output of running the script:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_6.js
[ { name: 'C', year: 1972 },
  { name: 'Python', year: 1991 },
  { name: 'Java', year: 1995 } ]
rodrigobranas:javascriptmasterclass $
```

The screenshot shows a Mac OS X desktop environment with a terminal window open. The terminal window has tabs labeled "array_mutator_api_7.js — javascriptmasterclass" and "2: bash". The code editor tab contains the following JavaScript code:

```
JS array_mutator_api_7.js x
array_mutator_api_7.js — javascriptmasterclass
1 const languages = [
2   {
3     name: "Python",
4     year: 1991
5   },
6   {
7     name: "C",
8     year: 1972
9   },
10  {
11    name: "Java",
12    year: 1995
13  }
14];
15languages.sort(function (a, b) {
16  return b.year - a.year;
17});
18console.log(languages);
19
```

The terminal session shows the output of running the script:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_7.js
[ { name: 'Java', year: 1995 },
  { name: 'Python', year: 1991 },
  { name: 'C', year: 1972 } ]
rodrigobranas:javascriptmasterclass $
```

A screenshot of a Mac OS X desktop environment. On the left is a code editor window titled "array_mutator_api_8.js" with the file path "javascriptmasterclass". The code editor shows a script named "array_mutator_api_8.js" with the following content:

```
2  {
3      name: "Python",
4      year: 1991
5  },
6  {
7      name: "C",
8      year: 1972
9  },
10 {
11     name: "Java",
12     year: 1995
13 }
14 ];
15 languages.sort(function (a, b) {
16     return (a.name < b.name) ? -1 : 1;
17 });
18 console.log(languages);
19
```

The code uses ES6 syntax, including arrow functions and template literals. It defines an array of objects representing programming languages, sorts them by name, and then logs the sorted array to the console.

To the right of the code editor is a terminal window titled "TERMINAL" with the tab "2: bash". The terminal output is:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_8.js
[ { name: 'C', year: 1972 },
  { name: 'Java', year: 1995 },
  { name: 'Python', year: 1991 } ]
rodrigobranas:javascriptmasterclass $
```

A screenshot of a Mac OS X desktop environment. On the left is a code editor window titled "array_mutator_api_9.js — javascriptmasterclass". The code is written in JavaScript and defines an array of objects representing programming languages:

```
JS array_mutator_api_9.js ×
array_mutator_api_9.js — javascriptmasterclass
1 const languages = [
2   {
3     name: "Python",
4     year: 1991
5   },
6   {
7     name: "C",
8     year: 1972
9   },
10  {
11    name: "Java",
12    year: 1995
13  }
14];
15languages.sort(function (a, b) {
16  return (a.name > b.name) ? -1 : 1;
17});
18console.log(languages);
19
```

The code uses the `sort` method to sort the array by language name. On the right is a terminal window titled "TERMINAL" with tab "2: bash". The command `node array_mutator_api/array_mutator_api_9.js` is run, and the output shows the sorted array:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_9.js
[ { name: 'Python', year: 1991 },
  { name: 'Java', year: 1995 },
  { name: 'C', year: 1972 } ]
rodrigobranas:javascriptmasterclass $
```

The screenshot shows a Mac OS X desktop environment with a terminal window open. The terminal window has two tabs: 'array_mutator_api_10.js — javascriptmasterclass' and '2: bash'. The 'array_mutator_api_10.js' tab contains the following JavaScript code:

```
JS array_mutator_api_10.js x
1 const languages = [
2   {
3     name: "Python",
4     year: 1991
5   },
6   {
7     name: "C",
8     year: 1972
9   },
10  {
11    name: "Java",
12    year: 1995
13  }
14];
15 languages.sort(function (a, b) {
16   return a.name.localeCompare(b.name);
17 });
18 console.log(languages);
19
```

The 'bash' tab shows the output of running the script with Node.js:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_10.js
[ { name: 'C', year: 1972 },
  { name: 'Java', year: 1995 },
  { name: 'Python', year: 1991 } ]
rodrigobranas:javascriptmasterclass $
```

A screenshot of a macOS desktop environment showing a terminal window and a code editor.

The terminal window (top right) has the title "array_mutator_api_11.js — javascriptmasterclass". It contains the following text:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_11.js
[ 'Java', 'C', 'Python' ]
[ 'Python', 'C', 'Java' ]
rodrigobranas:javascriptmasterclass $
```

The code editor window (left side) has the title "JS array_mutator_api_11.js". It contains the following code:

```
1 const languages = ["Python", "C", "Java"];
2 languages.reverse();
3 console.log(languages);
4 languages.reverse();
5 console.log(languages);
6
```

A screenshot of a macOS desktop environment showing a terminal window and a code editor.

The terminal window is titled "array_mutator_api_12.js — javascriptmasterclass". It contains the following text:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_12.js
[ 'JavaScript', 'JavaScript', 'JavaScript' ]
rodrigobranas:javascriptmasterclass $
```

The code editor window has a tab titled "JS array_mutator_api_12.js". It contains the following JavaScript code:

```
1 const languages = ["Python", "C", "Java"];
2 languages.fill("JavaScript");
3 console.log(languages);
4
```

A screenshot of a Mac OS X desktop environment. On the left is a code editor window titled "array_mutator_api_13.js — javascriptmasterclass". The file contains the following JavaScript code:

```
JS array_mutator_api_13.js x
1 const languages = ["Python", "C", "Java"];
2 languages.fill("JavaScript", 1);
3 console.log(languages);
4
```

The code uses the `fill` method to replace the second element of the `languages` array with "JavaScript". The code editor has standard OS X window controls (red, yellow, green) and a toolbar with icons for search, refresh, and more.

To the right of the code editor is a terminal window titled "TERMINAL" with tab "2: bash". The terminal shows the command being run and its output:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_13.js
[ 'Python', 'JavaScript', 'JavaScript' ]
rodrigobranas:javascriptmasterclass $
```

A screenshot of a macOS desktop environment. On the left is a code editor window titled "array_mutator_api_14.js — javascriptmasterclass". The file contains the following JavaScript code:

```
JS array_mutator_api_14.js x
1 const languages = ["Python", "C", "Java"];
2 languages.fill("JavaScript", 0, 2);
3 console.log(languages);
4
```

The code uses the `fill` method to replace the first two elements of the `languages` array with "JavaScript". The code editor has standard OS X-style window controls (red, yellow, green buttons) at the top.

To the right of the code editor is a terminal window titled "TERMINAL" with a tab labeled "2: bash". The terminal output is:

```
rodrigobranas:javascriptmasterclass $ node array_mutator_api/array_mutator_api_14.js
[ 'JavaScript', 'JavaScript', 'Java' ]
rodrigobranas:javascriptmasterclass $
```