

On récupère en premier temps la librairie de Vue.js en tapant l'url suivant



Cela nous donne la dernière version de vue.js ainsi que sa librairie

```
← → ↻  Sécurisé | https://unpkg.com/vue@2.5.17/dist/vue.js
```

```
/*!
 * Vue.js v2.5.17
 * (c) 2014-2018 Evan You
 * Released under the MIT License.
 */
(function (global, factory) {
  typeof exports === 'object' && typeof module !== 'undefined' ? module.exports = factory() :
  typeof define === 'function' && define.amd ? define(factory) :
  (global.Vue = factory());
})(this, (function () { 'use strict';

/* */

var emptyObject = Object.freeze({});

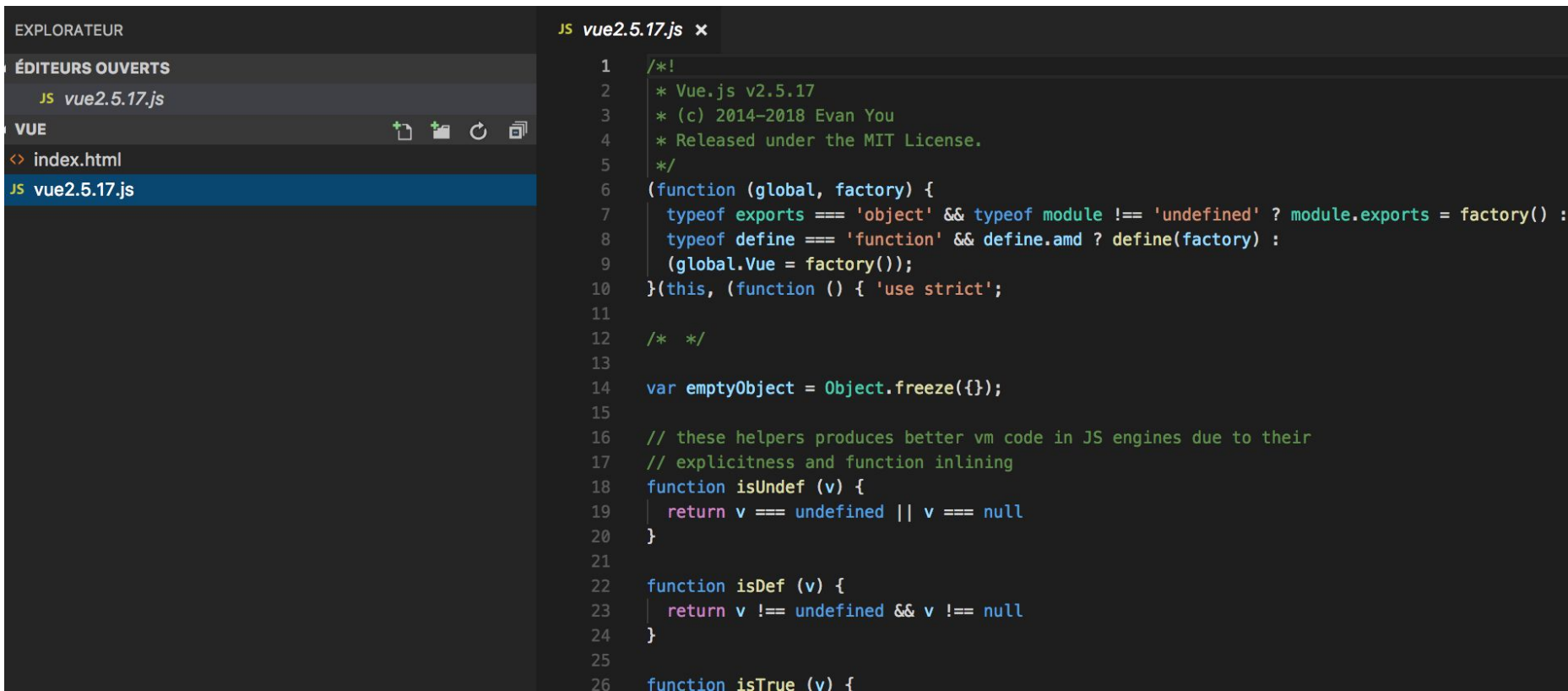
// these helpers produces better vm code in JS engines due to their
// explicitness and function inlining
function isUndef (v) {
  return v === undefined || v === null
}

function isDef (v) {
  return v !== undefined && v !== null
}

function isTrue (v) {
  return v === true
}

function isFalse (v) {
  return v === false
}
```

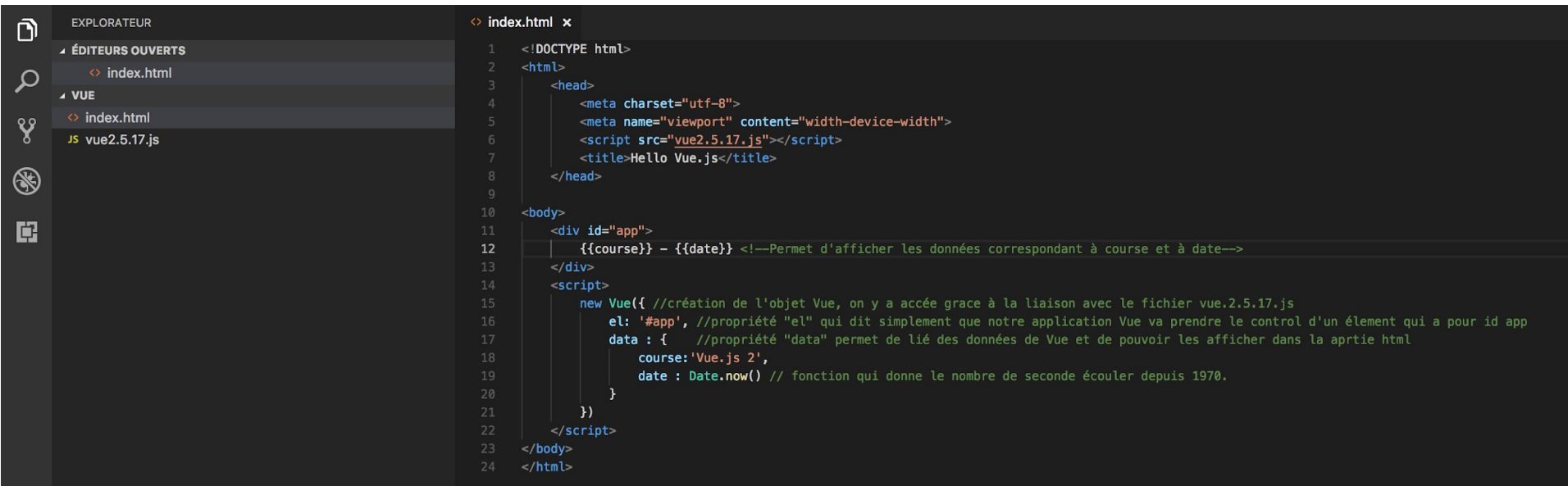
On copie la librairie et on la colle dans un nouveau fichier vue2.5.17.js



The image shows a code editor interface with a sidebar on the left and a main editor area on the right. The sidebar, titled 'EXPLORATEUR', contains a section 'ÉDITEURS OUVERTS' with two files: 'JS vue2.5.17.js' and 'VUE'. Below this, there are icons for file operations and a list of files including 'index.html' and 'vue2.5.17.js'. The main editor area shows the content of 'vue2.5.17.js', which is the Vue.js v2.5.17 library code. The code is written in JavaScript and includes a MIT license header, a factory function, and various helper functions like 'isUndef', 'isDef', and 'isTrue'.

```
1  /*!  
2  * Vue.js v2.5.17  
3  * (c) 2014-2018 Evan You  
4  * Released under the MIT License.  
5  */  
6  (function (global, factory) {  
7    typeof exports === 'object' && typeof module !== 'undefined' ? module.exports = factory() :  
8    typeof define === 'function' && define.amd ? define(factory) :  
9    (global.Vue = factory());  
10 })(this, (function () { 'use strict';  
11  
12  /* */  
13  
14  var emptyObject = Object.freeze({});  
15  
16  // these helpers produces better vm code in JS engines due to their  
17  // explicitness and function inlining  
18  function isUndef (v) {  
19    return v === undefined || v === null  
20  }  
21  
22  function isDef (v) {  
23    return v !== undefined && v !== null  
24  }  
25  
26  function isTrue (v) {
```

On créer un fichier index.html que l'on lie à notre fichier vue2.5.17.js



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width">
6     <script src="vue2.5.17.js"></script>
7     <title>Hello Vue.js</title>
8   </head>
9
10  <body>
11    <div id="app">
12      {{course}} - {{date}} <!--Permet d'afficher les données correspondant à course et à date-->
13    </div>
14    <script>
15      new Vue({ //création de l'objet Vue, on y a accée grace à la liaison avec le fichier vue.2.5.17.js
16        el: '#app', //propriété "el" qui dit simplement que notre application Vue va prendre le control d'un élément qui a pour id app
17        data : { //propriété "data" permet de lié des données de Vue et de pouvoir les afficher dans la aptie html
18          course: 'Vue.js 2',
19          date : Date.now() // fonction qui donne le nombre de seconde écoulé depuis 1970.
20        }
21      })
22    </script>
23  </body>
24 </html>
```

On instancie Vue().

On a accès à cet classe car notre fichier html est lié à notre fichier vue.js



Vue.js 2 - 1533720649749

Exécution du fichier index.html