

Vue.js

Sommaire :

- Qu'est-ce que Vue.js ?
- Commencer Vue.js.
- Rendu déclaratif.
- Condition et boucles.
- Gestion des entrées utilisateurs.
- Conclusion.

FUTUR PDF SUR

VUE.JS

Lien :

Vidéo tuto <https://youtu.be/g7YKecZhFRA>

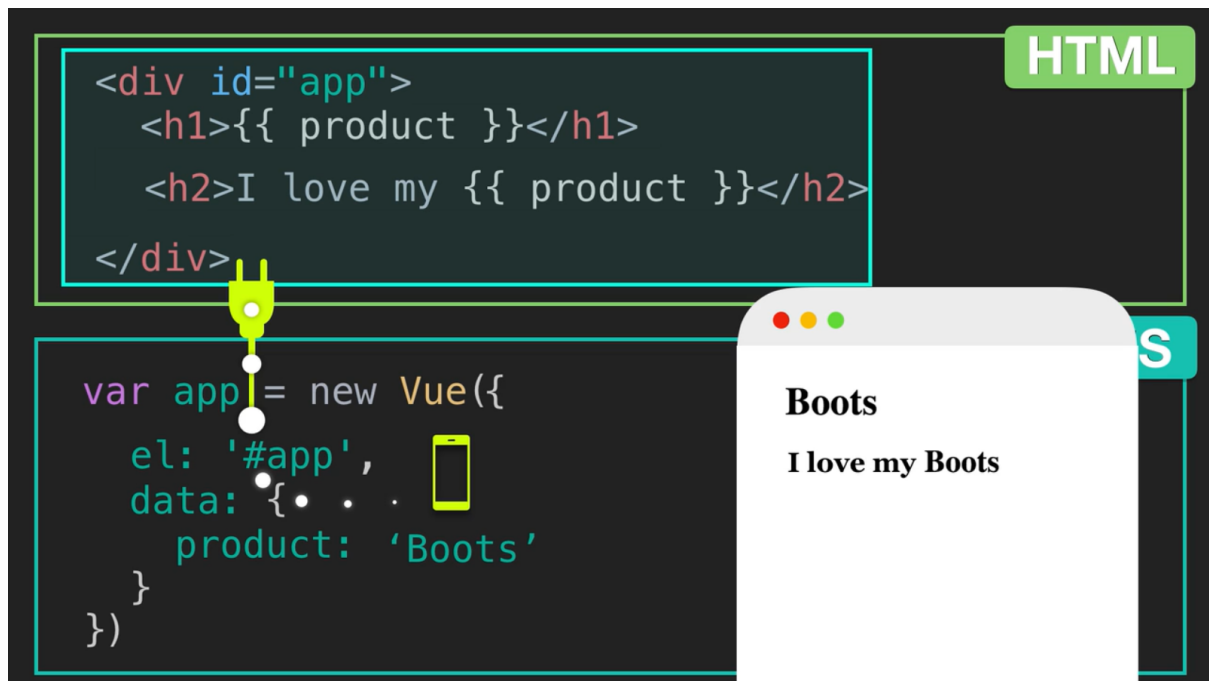
Guide du site officiel (tout y est) : <https://vuejs.org/v2/guide/>

Qu'est-ce que Vue.js ?

Vue est un framework progressif pour la construction d'interfaces utilisateur. Créé par Evan You et est maintenu par lui et le reste des membres actifs de l'équipe principale travaillant sur le projet et son écosystème.

La bibliothèque principale se concentre uniquement sur la couche d'affichage et est facile à récupérer et à intégrer à d'autres bibliothèques ou projets existants.

D'autre part, Vue est également capable d'alimenter des applications monopage sophistiquées lorsqu'il est utilisé en combinaison avec des outils modernes et des bibliothèques de support.



Commencer

Un moyen simple d'essayer Vue.js est de, comme tout autre langage de programmation, lui demander un “Hello World”. Tout d'abord, nous devons initialiser Vue.

```
<!-- development version, includes helpful console warnings -->
```

```
<script src="https://cdn.jsdelivr.net/npm/vue@2/dist/vue.js"></script>
```

OU :

```
<!-- production version, optimized for size and speed -->
```

```
<script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
```

Rendu déclaratif

Voici le programme pour renvoyer “Hello Vue!”

```
<div id="app"> |  
  {{ message }} | PARTIE HTML  
</div> |  
  
var app = new Vue({ |  
  el: '#app', |  
  data: { | PARTIE VUE  
    message: 'Hello Vue!' |  
  } |  
}) |
```

Hello Vue! |RENDU

Notez que nous n'avons plus à interagir directement avec le HTML. C'est désormais Vue qui gère les données de la page. Le HTML n'est plus que notre point d'entrée.

Conditions et boucles

On peut lier des données non seulement au texte et aux attributs, mais également à la structure. De plus, Vue fournit également un système de transition en temps réel des modifications.

Il existe d'autres directives, chacune avec sa propre fonctionnalité spéciale. Par exemple, la `v-for` directive peut être utilisée pour afficher une liste d'éléments en utilisant les données d'un tableau.

Gestion des entrées utilisateur

Pour permettre aux utilisateurs d'interagir avec votre application, nous pouvons invoquer des méthodes sur nos instances Vue avec

`v-on` :

```
<div id="app-5">
  <p>{{ message }}</p>
  <button v-on:click="reverseMessage">Reverse Message</button>
</div>
```

```
var app5 = new Vue({
  el: '#app-5',
  data: {
    message: 'Hello Vue.js!'
  },
  methods: {
    reverseMessage: function () {
      this.message = this.message.split('').reverse().join('')
    }
  }
})
```

```
}  
})
```

Bonjour Vue.js ! (Message inversé si on clique sur le bouton)

Conclusion

Vue.js est un Framework très intéressant, permettant d'interagir avec notre code JavaScript directement depuis une simple interface HTML. Il nécessite cela dit un apprentissage poussé du JavaScript, bien qu'il soit un peu plus simpliste que ce dernier. Lors de la réalisation d'un site, Vue.js sera un outil très utile et efficace.



Vue Mastery Socks

Cart(0)

Shipping: 0

In Stock

Details

- 80% cotton
- 20% polyester
- Gender-neutral

Colors:



Add to Cart