

# index.html partie 1

```
<> index.html x # style.css
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     <link rel="stylesheet" href="./style.css">
8     <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css">
9     <title>Hello Vue.js</title>
10  </head>
11
12  <body>
13    <div id='app'>
14      <h2>{{title}}</h2>
15      <!--le formulaire a la responsabilité de gerer la soumission-->
16      <form v-on:submit.prevent='addTask()'> <!--IMPORTANT $event est un mots clé dans VUEJS-->
17        <input type='text' placeholder='tâche a faire' v-model='task'><br>
18        <button type="submit" >ajouter</button>
19      </form>
20      <ul>
21        <li
22          v-bind:class="{completed: task.done}"
23          v-on:click="task.done = !task.done"
24          v-for='task in tasks'><!-- v-for Nous permet de recrachter les task du tableau tasks-->
25          <span v-if="!task.isInEditMode"> <!-- v-if veut dire si l'objet n'est pas en editMode-->
26            <i class="fa fa-edit" v-on:click="switchMode(task)"> <!-- La fonction switchMode prend en argument la task courante-->
27            </i>{{task.description}}
28          </span>
29          <span v-else> <!-- v-else veut dire sinon-->
30            <i class="fa fa-thumbs-o-up"></i> <!-- v-model permet de faire du two-way binding du navigateur vers Vue -
31            <input type="text" v-model="task.description" v-on:keyup.enter="switchMode(task)"><!-- l'evenement keyup.enter est un helper de Vuejs -
32          </span>
33        </li>
34      </ul>
35    </div>
```

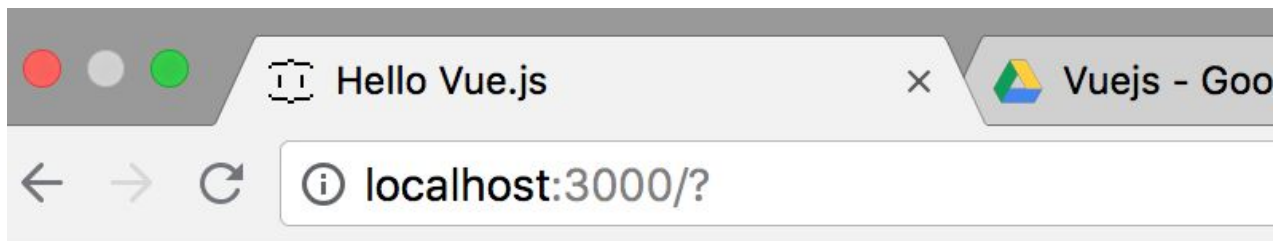
## index.html partie 1

```
<script>
  const vm = new Vue({ //création de l'objet Vue, on y a accée grace à la liaison avec le fichier vue.2.5.17.js
    el: '#app', //propriété "el" qui dit simplement que notre application Vue va prendre le control d'un élément qui a pour id app
    data : {    //propriété "data" permet de lié des données de Vue et de pouvoir les afficher dans la aptie html
      title: 'A faire',
      task: null,
      tasks: [] // création d'un tableau qui prend nos tâches
    },
    methods:{
      addTask :function(){ //fonction d'ajout de tâche
        this.tasks.push({
          id : Date.now(),
          description: this.task,
          done: false,
          isInEditMode : false
        });
        this.task = null;
      },
      switchMode :function(task){ // fonction permettant d'éditer les taches
        task.isInEditMode = !task.isInEditMode;
      }
    }
  })
</script>
</body>
</html>
```

<> index.html

# style.css x

```
1  li{
2      list-style-type: none; /* retire les points par défaut des listes*/
3      cursor: pointer; /* Transforme la souris en pointeur à l'approche de la liste */
4      margin-bottom: 10px;
5  }
6
7  .completed{
8      text-decoration: line-through; /*Permet de rayer la tâche*/
9      color: green;
10 }
```



# A faire

