Oc Vue.js

Part 1

1. Create your first vue app

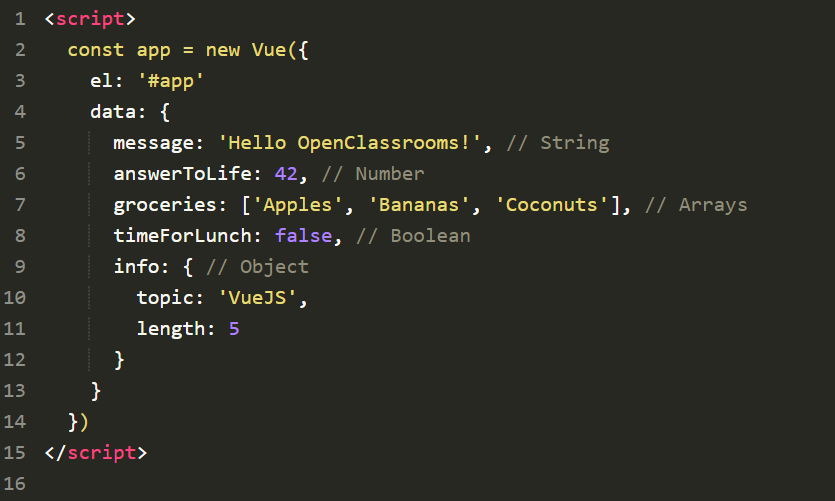
Ajouter Vue à ton projet peut se faire n’importe quand, il suffit d’une balise script dans ton html :



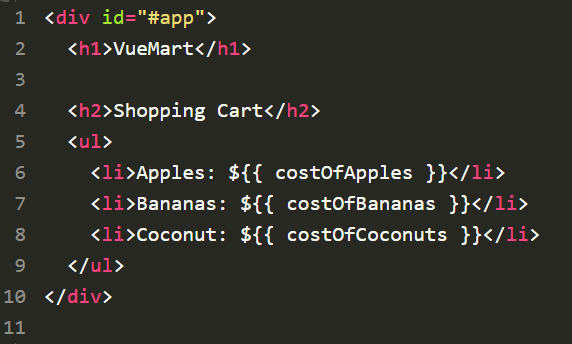
Vérifier que vue est bien là :



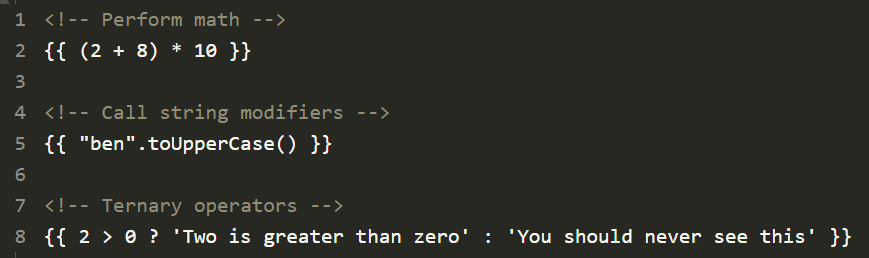
Store data - comme dans toute app, on a besoin d’une manière de store des data afin de les réutiliser. En vue, on ajoute une propriété data avec pour value un objet. (It can take any normal key-value pair that you would typically define in JavaScript.) :



Render data – handlebar expression to render data ( {{ miaou }} ) :

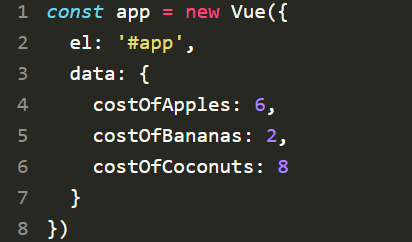


Il est possible de render des expressions js basique entre les handlebars :

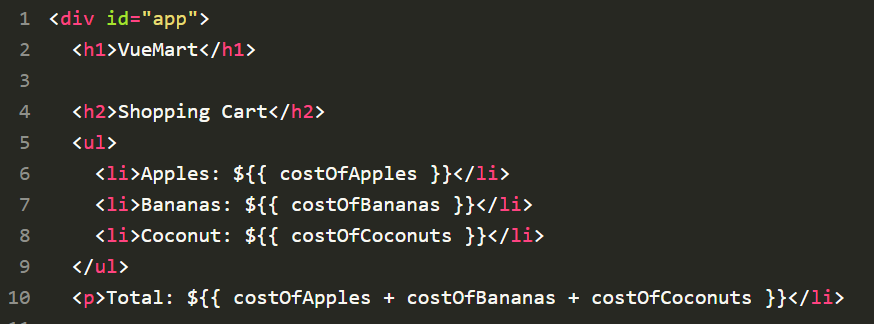


1. Store and manage data

Pour avoir des valeurs non statiques, on se sert de la handlebar écriture et boom c’est nice. L’exemple ici est un shop en ligne :



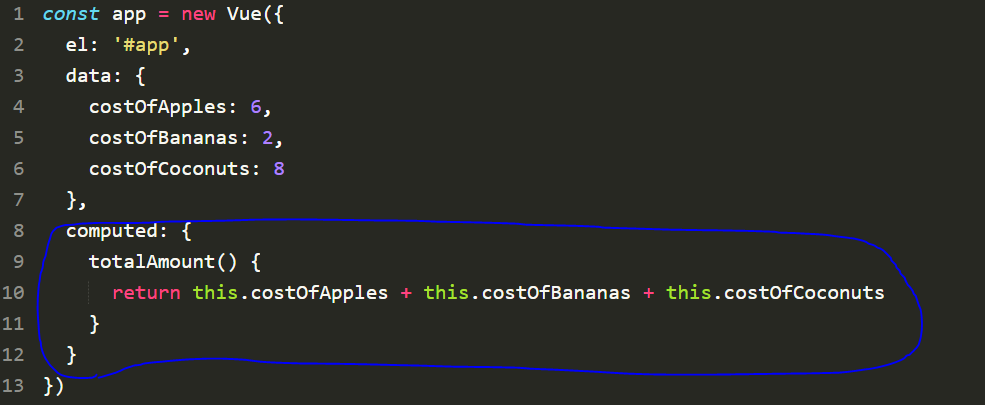
On fait des variables pour chaque fruit et ensuite on gère l’affichage/le rendu dans le html :



Ça marche mais ce n’est pas la solution la plus propre. Il serai plus propre de pouvoir écrire directement {{ totalAmount }}. Pcq t’imagine t’a plus de trois items dans ton panier ? et même la solution ci-dessus est vraiment à chier.

Computed properties – way to define a reusable value that is updated based on other data properties. Contrairement au data property, toutes les computed property doivent être des fonctions qui retournent une valeur. (it is performing calculations and returning the desired result).

Pour simplifier le total de tout à l’heure en {{ totalAmount }}, on peut faire ceci :



Do not use arrow functions, it will break the app.

1. Manage common scenarios with directives

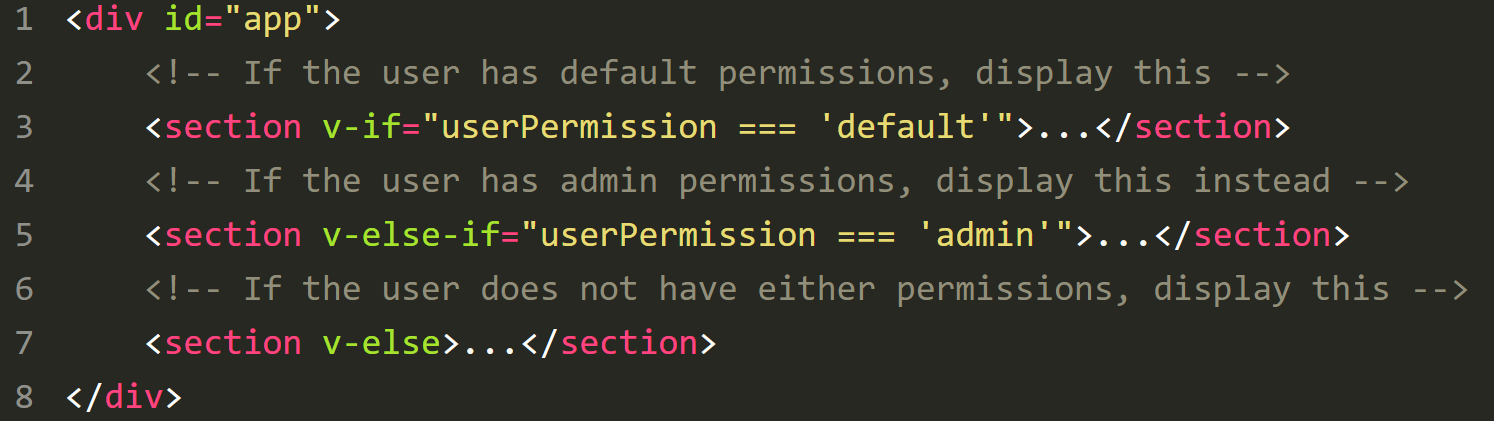
Ex de scenario : décider si on montre un truc ou pas, gérer une liste de données

Common directives : v-if, v-else-if, v-else, v-show, v-for, v-bind, v-on, v-model

Scenario 1 : toggle content visibility

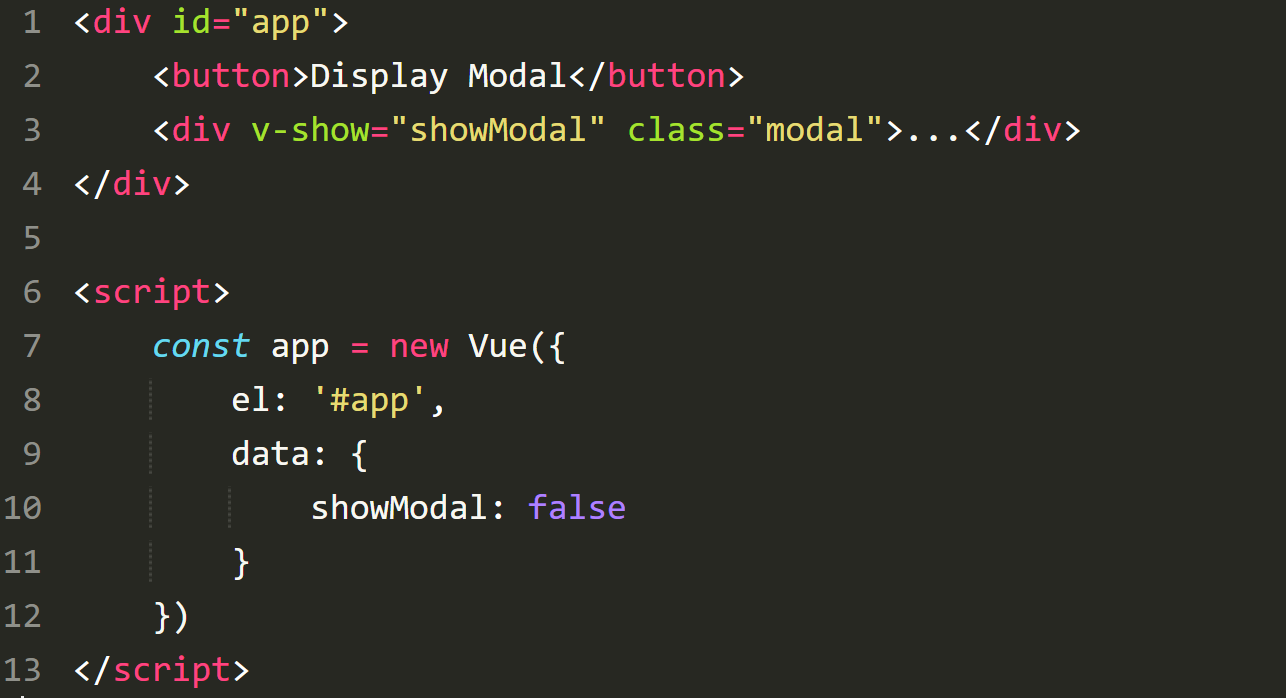
On utilisera two primary directives

-> v-if



-> et v-show

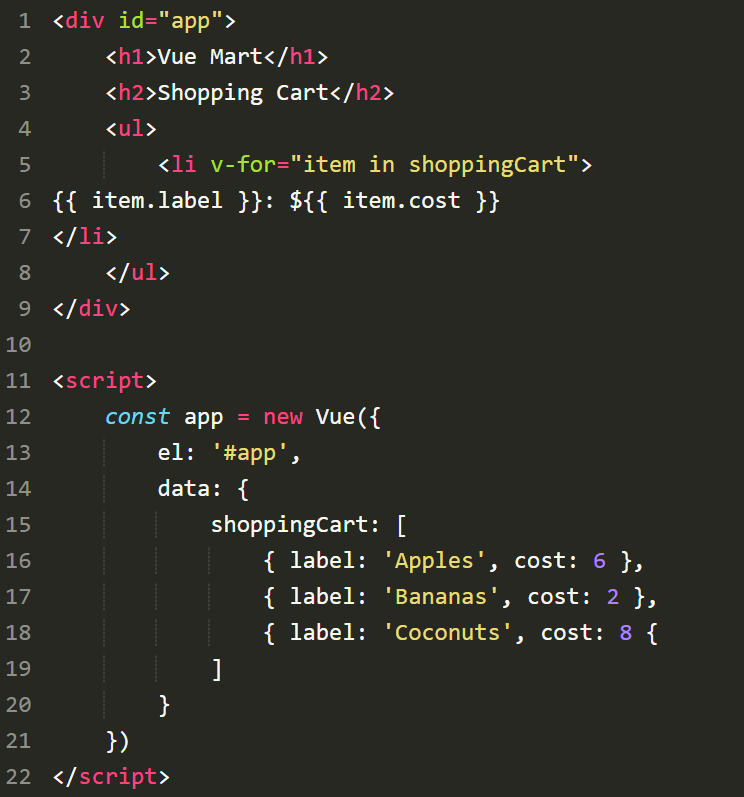
Utilisé pour contrôler la visibilité d’un élément.



La diff entre les deux est que v-show gère la visibilité de l’élément HTML en utilisant le css alors que v-if enlève entièrement l’élément du DOM.

Sceanario 2 : loop

Repeating an HTML element to render data. Ex render a list :



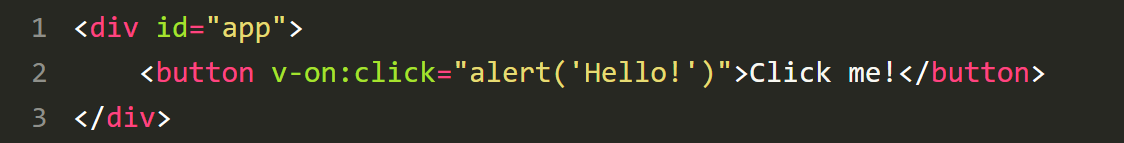
Scenario 3 : dynamically define HTML attributes



Utilisable sous la forme  :href= ‘’item.url’’ (pour le v-bind)

Scenario 4 : listening to events and responding to them

V-on directive



Tu peux remplacer v-on par @ \_ @click= ‘’alert(‘hello !’)’’

« However, you will often want to call functions much more complex than a single line of JavaScript. And that's where methods come into play. Methods allow you to define functions that your Vue app will have access to. It is defined just like the data property. »

Methods - Create JavaScript functions that Vue.js can use



Scenario 5 : Update data with form inputs

v-model (video) - Configure form elements to update your data store as the user interacts with it



Quiz de fin : qu4 – data is used for creating a reactive data store.

Notes du quiz : 8/9

Explanations :

A single-page application (SPA) is a web application or web site that dynamically generates the page content rather than loading new pages from the server.

The v-for directive allows you to render lists in the form of an array or object that uses similar syntax to the for ... in ... syntax for JavaScript.

While Vue.js has its own build tools for large scale applications, the simplest path forward is to use the CDN link for Vue, and you're ready to create a full application!

The data property on a Vue instance allows you to create a reactive data store that tracks and updates the use of each reactive data property.

The v-show directive toggles the visibility of an element with CSS. This is more performant than v-if, where the entire HTML element is destroyed and recreated with each toggle.

The v-bind directive allows you to manipulate HTML properties by telling Vue to interpret it as JavaScript rather than static string.

The v-model directive is the most common way to ensure that form elements are connected to the correct data store property. In this case, it should be <input type="text" id="first-name" v-model="firstName" /> .

When you want to refer to a property within the Vue instance, you need to provide context with this to ensure that it refers to the current Vue instance.

Computed properties are an excellent way to remove bloat from HTML since it allows you to dynamically assemble data in an easy to read way in your HTML.

Part 2

1. Create a new project with vue CLI