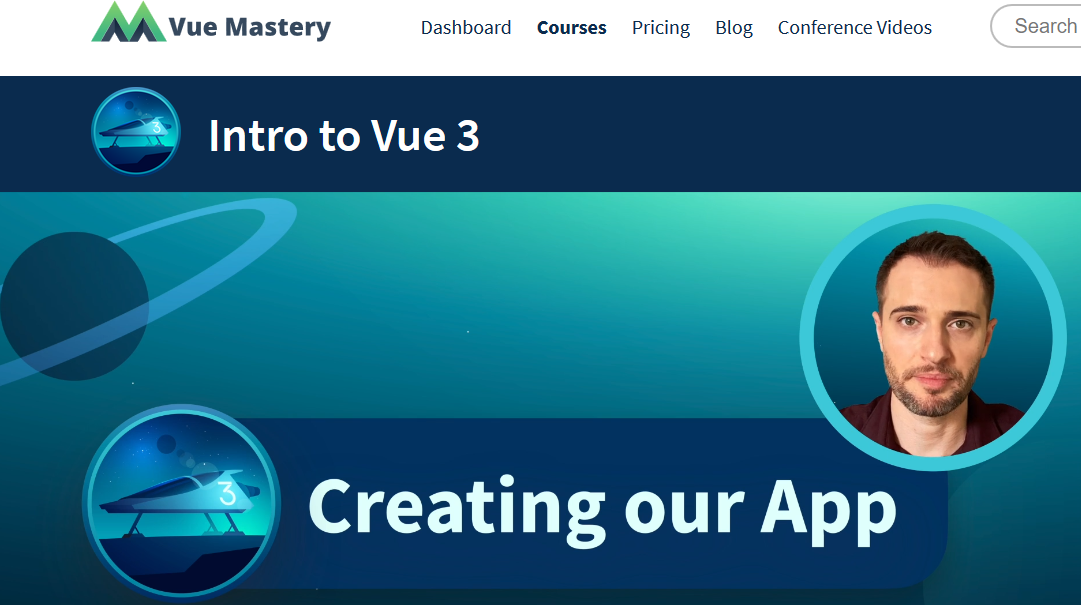
**VueJs 3**

**Introduction to Vue 3:**

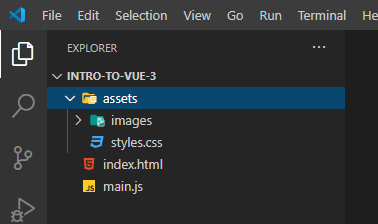
Download the github code:

Download editor VS Code and extension E6-string-html

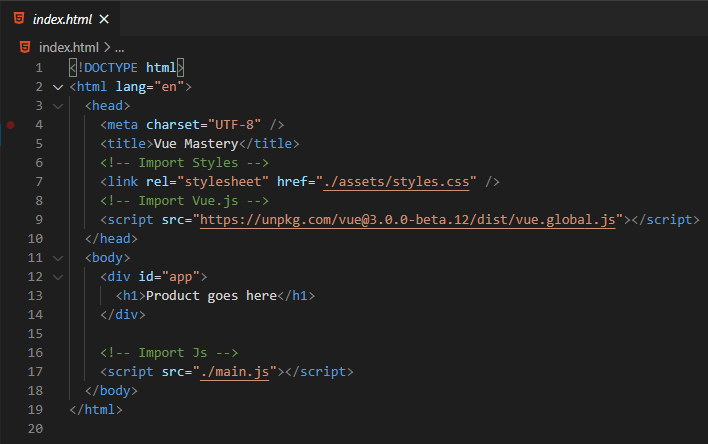


**In VS Code Open Folder:**

**Intro-to-vue-3**



Index.js page



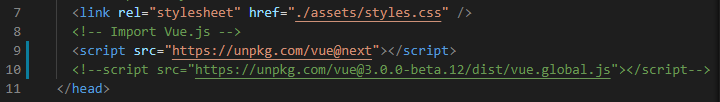
Get the non-Beta version of Vue3 from the Documentation:

**Copy the CDN link:**

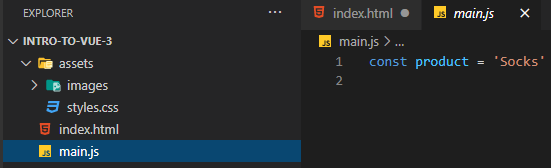
<script src="https://unpkg.com/vue@next"></script>

The css is imported – see Import Styles

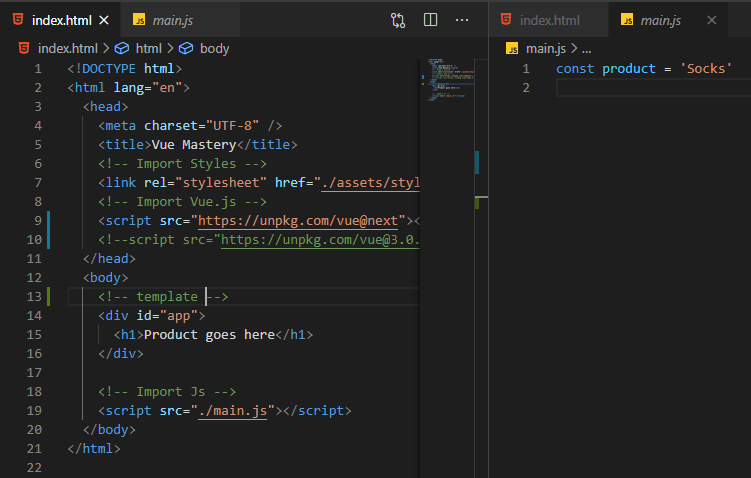
The vue is imported -- see Import Vue.js



**Notice the const type product is a the string ‘Socks’**



**The template is <h1>Product goes here</h1>**

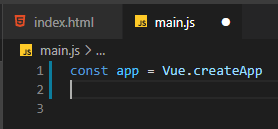


**How do we display the products using Vue.js?**

**1) Create a Vue app**

In main.js

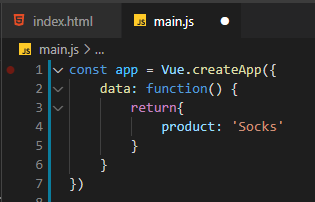
const app = Vue.createApp



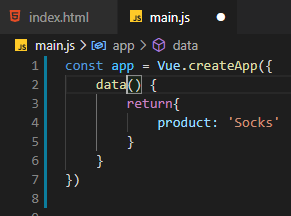
We need to create a data object {} which will be defined as a function

which will return another object {}, where we store the data items.

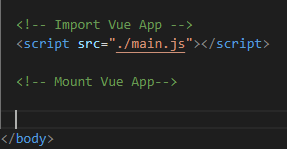
e.g. product: ‘Socks’



ESS shorthand allows the removal of the type of object (: function) => ()

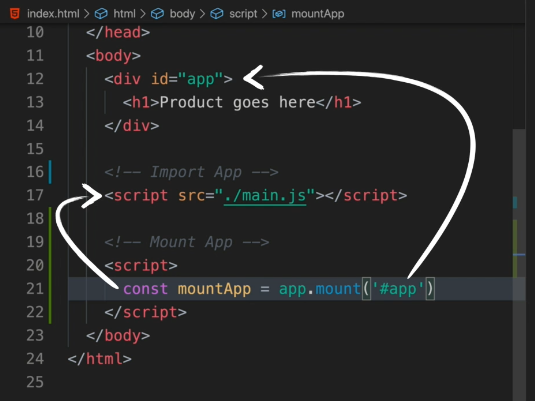


**2) With the Vue App created we can now import and mount the App:**

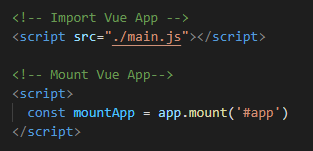


// Import app

// mount app



Resulting Import and Mount

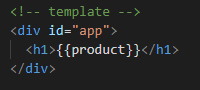


Why the # in front of app as a string?

# indicates a DOM selector for insertion into the div that contains the id=“app”

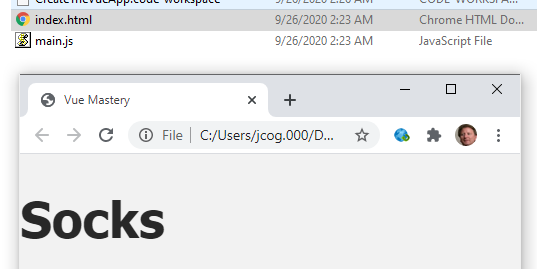
**3) We will use the Curly Brace Syntax** {{}} as the placeholder for the object product

{{}} means expose or reveal the value for product

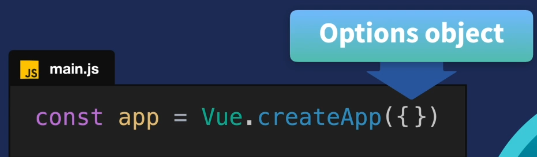


Now, let us see what the product object reveals in the Browser:

Right-click on open the index.html page



How did this happen?

We created an Options Object  


In go the properties

You always have to pass at least and empty {}

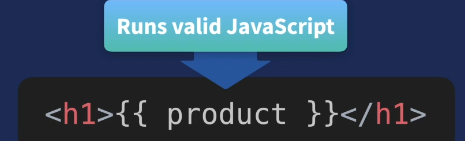
The data ‘Socks’ gets into the App

We then plug the app into the div with id=app

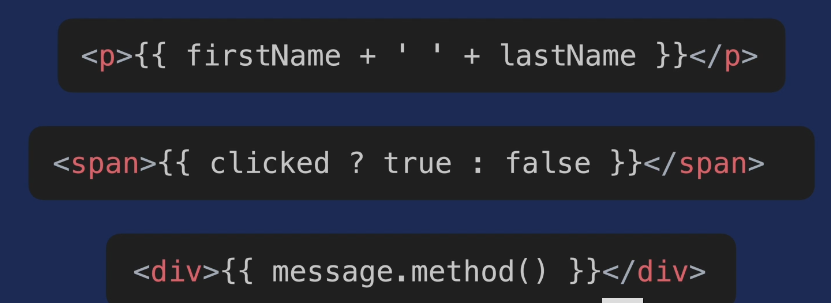
{{}} what is the value of product (this is also known as mustache notation)

App responds with ‘Socks’

Running valid javascript in HTML

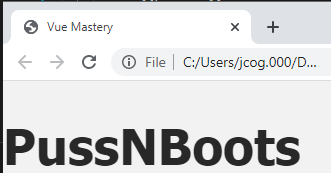


Other Expressions that can be run on data!!

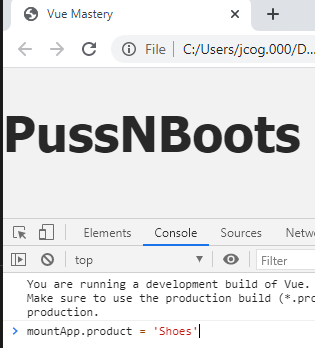


**View is referred to as Reactive:**

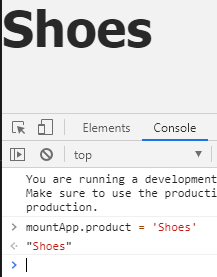
Change Socks to PussNBoots and change is immediate

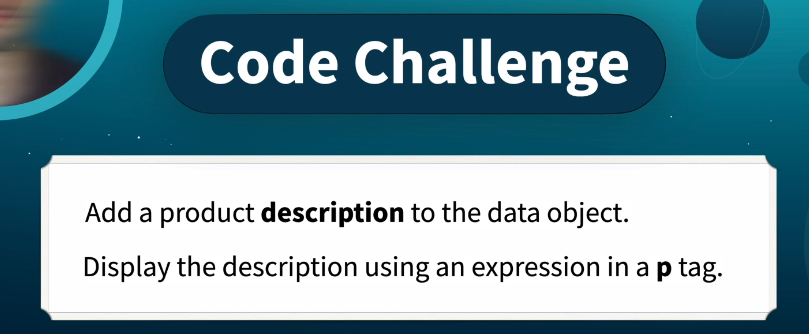


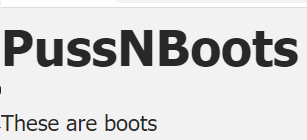
Using the Console in Chrome you can change the Data:

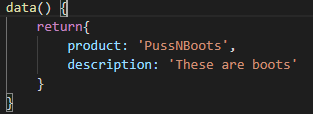


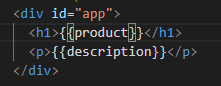
Press Enter:











Correct!!