

VUE.JS

The Compassionate Framework

Ben Hong (@bencodezen)

 **Cream City Code 2019**

A little bit about me...





BEN HONG

Senior Frontend Engineer

Meltano in GitLab

[@bencodezen](#)

- Google Developer Expert
- Mozilla TechSpeaker
- Vue.js Community Partner
- Vue Mastery Instructor
- VuePress Core Team

Before we get started...

All resources will be available online

<https://www.twitter.com/bencodezen>

For the social media folks

[@bencodezen](https://twitter.com/bencodezen) - [#CreamCityCode](https://twitter.com/hashtag/CreamCityCode) - [#FallExperiment](https://twitter.com/hashtag/FallExperiment)

What is Vue.js?

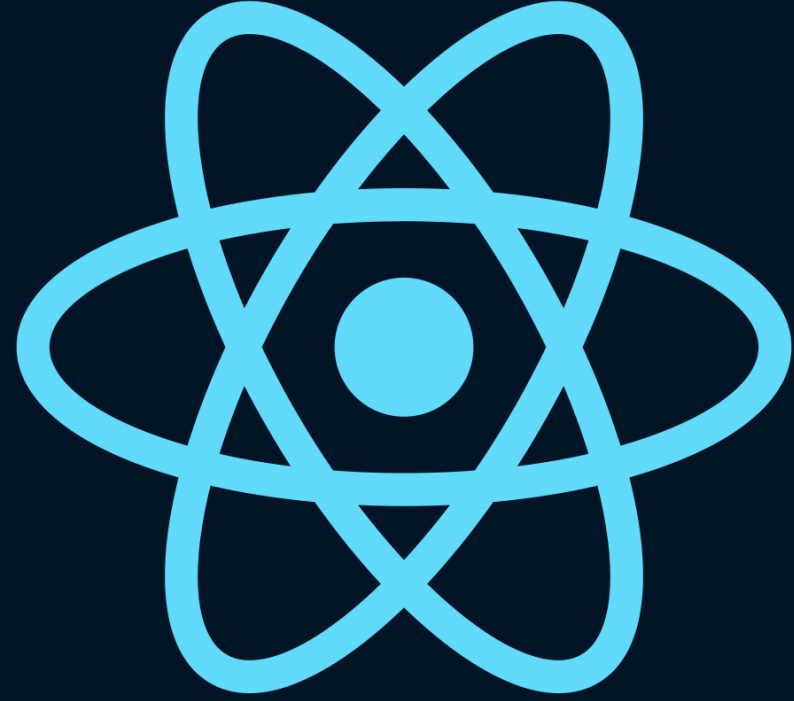
Vue.js is a progressive JavaScript framework

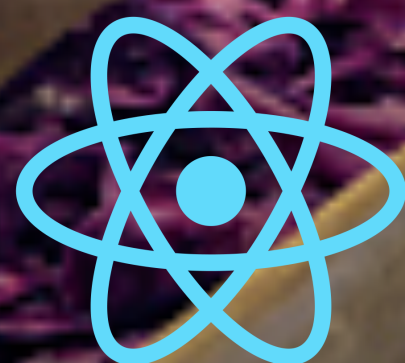
1. Approachable
2. Versatile
3. Performant

NUMBER OF DAYS SINCE



LAST NEW JS FRAMEWORK





A vibrant salad bar with various ingredients in glass bowls. Visible items include cherry tomatoes, shredded purple cabbage, shredded carrots, green leafy vegetables, and white rice. Each bowl has a wooden serving spoon. A small Santa Claus figurine is in the bottom left corner.

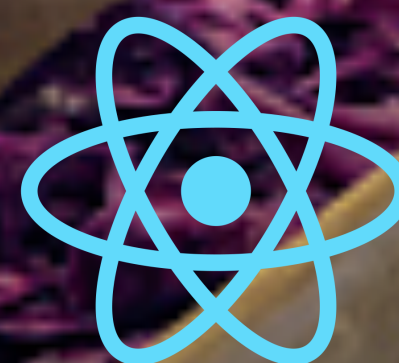
PROTEIN

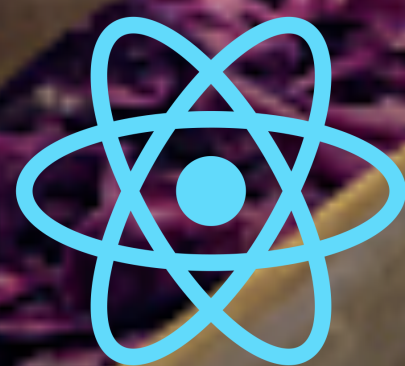
VEG

CARBS

SAUCE

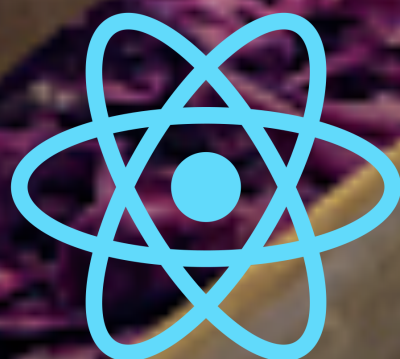
TOPPINGS







ES6







Banquet

Banquet

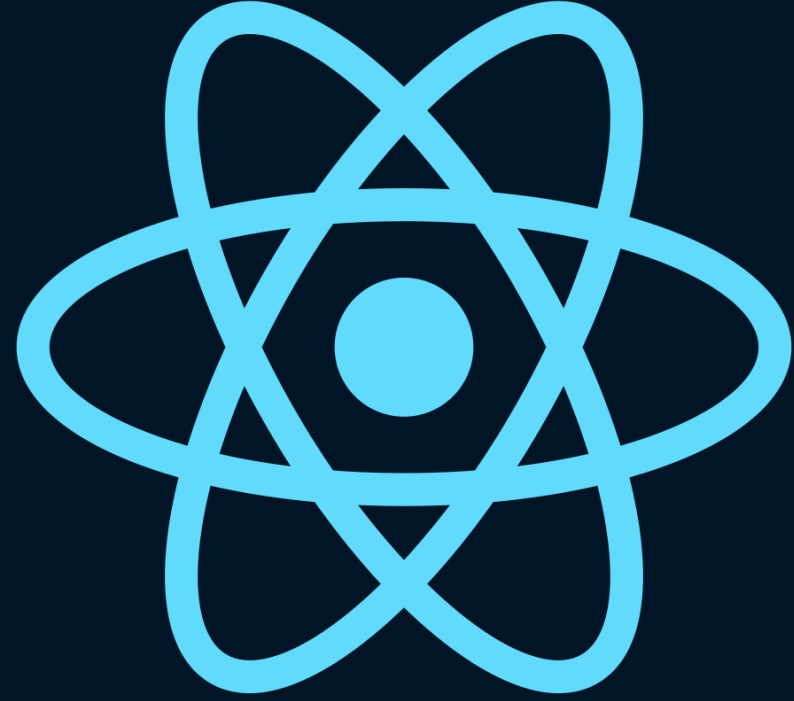
Meat Loaf Meal
Gravy with Meat Loaf,
Mashed Potatoes and Corn

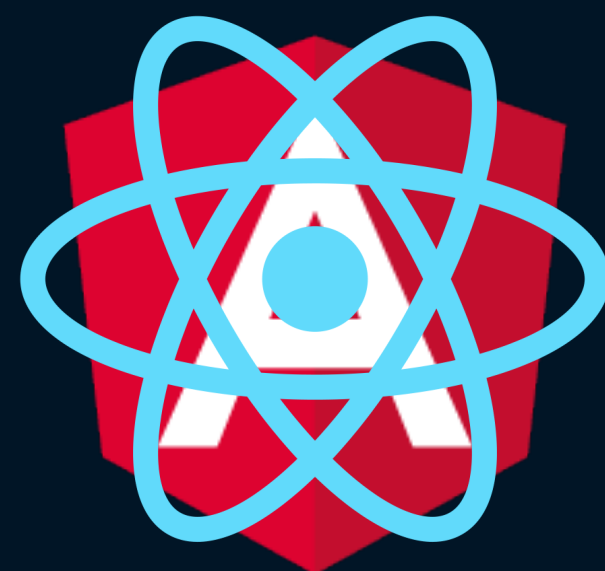
TS

Good
Source of
Protein

KEEP FROZEN
MUST BE COOKED THOROUGHLY

NET WT 9.5 OZ (269g)





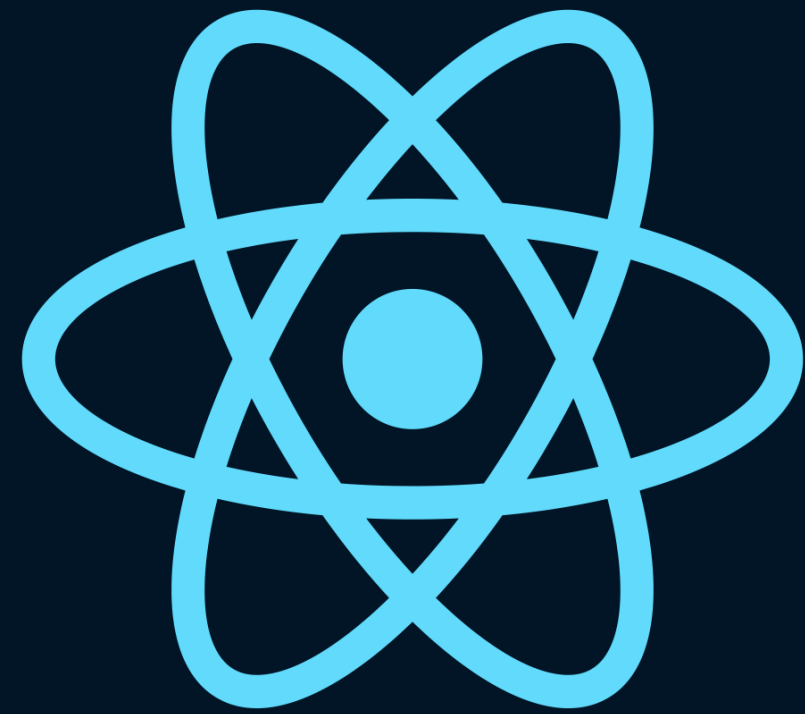












Facebook



Google



Community

Vue.js Fundamentals

An **instance** of Vue.js

```
const app = new Vue({  
  el: '#app',  
  data: {  
    firstName: 'Ben',  
    lastName: 'Hong',  
    event: 'Cream City Code'  
  }  
})
```

An instance of Vue.js

```
const app = new Vue({  
  el: '#app',  
  data: {  
    firstName: 'Ben',  
    lastName: 'Hong',  
    event: 'Cream City Code'  
  }  
})
```

Uses a CSS selector to determine where this Vue Instance will live.

An **instance** of Vue.js

```
const app = new Vue({  
  el: '#app',  
  data: {  
    firstName: 'Ben',  
    lastName: 'Hong',  
    event: 'Cream City Code'  
  }  
})
```

A data store for the Vue
to access in the app.

Expressions in Vue.js

```
const app = new Vue({  
  el: '#app',  
  data: {  
    firstName: 'Ben',  
    lastName: 'Hong',  
    event: 'Cream City Code'  
  }  
})
```

<!-- Using data in your template is easy! -->

```
<div id="app">  
  <p>{{ firstName }} {{ lastName }} is at {{ event }}!</p>  
</div>
```

Expressions in Vue.js

```
const app = new Vue({  
  el: '#app',  
  data: {  
    firstName: 'Ben',  
    lastName: 'Hong',  
    event: 'Cream City Code'  
  }  
})
```

<!-- This is how it renders! -->

```
<div id="app">  
  <p>Ben Hong is at Cream City Code!</p>  
</div>
```

Expressions in Vue.js

<!-- Using data in your template is easy! -->

<p>{{ firstName }} {{ lastName }} is at {{ conference }}!</p>

<!-- You can even do math! -->

<p>Let's perform calculations! {{ 12 * 12 / 4 }}</p>

<!-- Any simple expression just works -->

<p>{{ firstName.toUpperCase() }} IS EXCITED!</p>

<!-- Basic logic works too! -->

<p>{{ onSale ? price * 0.8 : price }}</p>

Computed properties

```
<p>{{ onSale ? price * 0.8 : price }}</p>
```

Computed properties

```
<script>
const app = new Vue({
  el: '#app',
  data: {
    price: 100,
    onSale: false
  },
  computed: {
    salePrice() {
      if (this.onSale) {
        return this.price * 0.8
      } else {
        return this.price
      }
    }
  }
})
</script>
```

Computed properties

```
const app = new Vue({
  el: '#app',
  data: {
    price: 100,
    onSale: false
  },
  computed: {
    salePrice() {
      if (this.onSale) {
        return this.price * 0.8
      } else {
        return this.price
      }
    }
  }
})
```

<p>{{ onSale ? price * 0.8 : price }}</p>

Computed properties

```
const app = new Vue({  
  el: '#app',  
  data: {  
    price: 100,  
    onSale: false  
  },  
  computed: {  
    salePrice() {  
      if (this.onSale) {  
        return this.price * 0.8  
      } else {  
        return this.price  
      }  
    }  
  }  
})  
  
<p>{{ salePrice }}</p>
```

Methods in Vue.js

```
const app = new Vue({
  el: '#app',
  data: {
    potion: 300,
    antidote: 100,
    revive: 1500,
    onSale: true
  },
  methods: {
    generatePrice(itemPrice) {
      if (this.onSale) {
        return itemPrice * 0.8
      } else {
        return itemPrice
      }
    }
  }
})
```

Let's talk about directives



Directives are the part of
Vue.js that are a bit **magical**...

WHAT ARE DIRECTIVES?

They are **Vue specific syntax** that allow you to accomplish common goals without worrying about how it is implemented

- ▶ `v-if`
- ▶ `v-else-if`
- ▶ `v-else`
- ▶ `v-bind`
- ▶ `v-for`
- ▶ `v-on`
- ▶ `v-model`

V-IF / V-ELSE-IF / V-ELSE

Allows you to determine whether an element on the page should exist

```
<article>
  <h1>Dashboard</h1>
  <section v-if="userPermission === 'admin'">
    ...
  </section>
  <section v-else-if="userPermission === 'user'">
    ...
  </section>
  <section v-else>
    <p>Please login.</p>
  </section>
</article>
```

V-BIND

Allows us to manipulate HTML attributes with dynamic data

<!-- The long form -->

```
<a v-bind:href="linkUrl">...</a>
```

<!-- The more common shorthand -->

```
<a :href="linkUrl">...</a>
```

<!-- It will render as... -->

```
<a href="www.vuejs.org">...</a>
```

V-ON

Allows us to attach methods to HTML elements when a certain event occurs

<!-- The long version -->

```
<button v-on:click="alert('Alohomora')">Unlock</button>
```

<!-- The more common shorthand -->

```
<button @click="alert('Alohomora')">Unlock</button>
```

<!-- Built in modifiers! -->

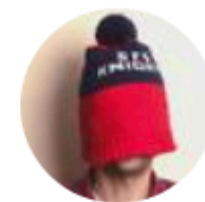
```
<input type="text" @keyup.enter="submit" />
```

V-MODEL

Allows us to use two-way binding

```
<div id="app">  
  <h1>Live Message Updates!</h1>  
  <p>{{ message }}</p>  
  <input type="text" v-model="message">  
</div>
```

```
<script>  
const app = new Vue({  
  el: '#app',  
  data: {  
    message: ''  
  }  
})  
</script>
```



Sam Saccone

@samcccone

Following



99.7% of software development in one requirement

A user should be able to view a list of items.

10:36 AM - 1 Dec 2017

610 Retweets 1,834 Likes



26



610



1.8K



V-FOR

Allows us to “render a list of items based on an array (or object).”

```
<script>
const app = new Vue({
  el: '#app',
  data: {
    Hogwarts: [
      'Ravenclaw',
      'Hufflepuff',
      'Slytherin',
      'Gryffindor'
    ]
  }
})
</script>
```

```
<!-- Given the data in the Vue instance -->
<div id="app">
  <ul>
    <li v-for="house in Hogwarts">
      {{ house }}
    </li>
  </ul>
</div>

<!-- Will be rendered as... -->
<ul>
  <li>Ravenclaw</li>
  <li>Hufflepuff</li>
  <li>Slytherin</li>
  <li>Gryffindor</li>
</ul>
</template>
```

Enough talking.
Let's see Vue.js in action!

What about
Vue.js 3.0?

What about

Vue.js 3.0?

DISCLAIMER



Evan You
@youyuxi



Just open sourced Vue 3 source code live during my [@vue_london](#) remote talk. Will be tweeting/blogging about some interesting stuff we've been doing for the Vue 3 compiler next week.



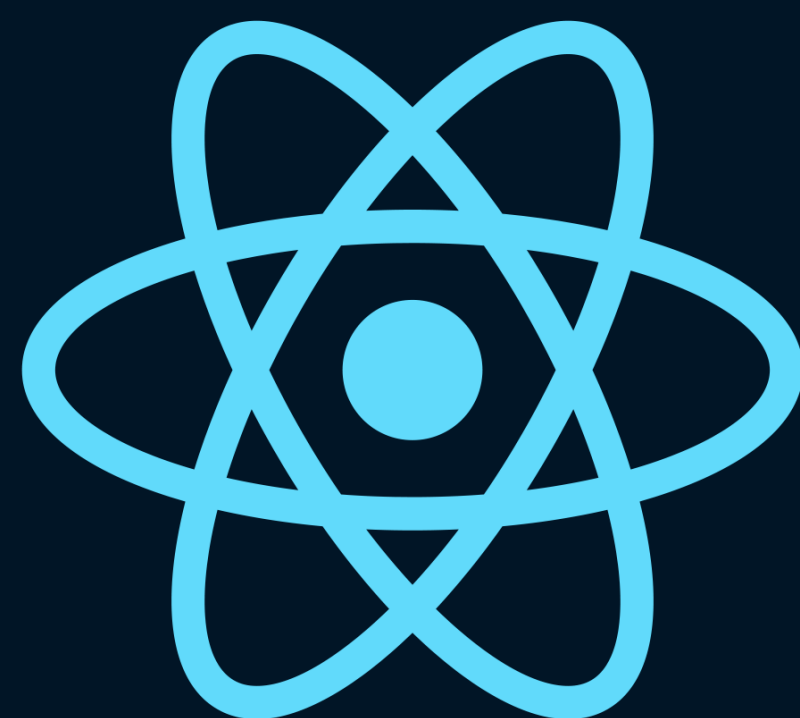
Vue.js @vuejs · 21h

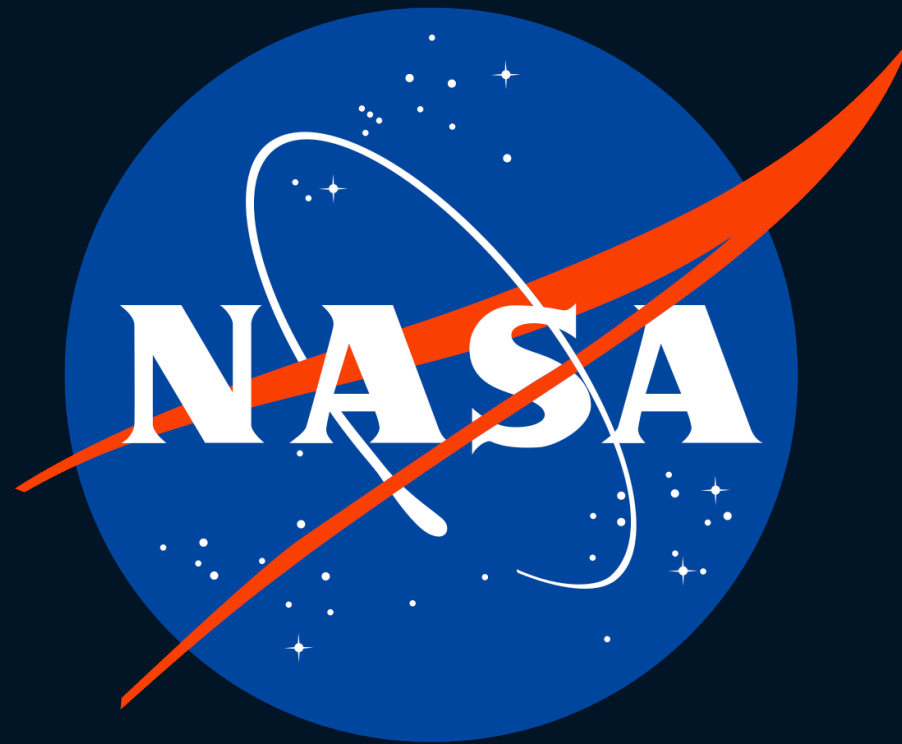
Vue 3 source code is now public! github.com/vuejs/vue-next

11:37 AM · Oct 4, 2019 · [Twitter Web App](#)

No need to wait.

Vue.js 3.0 is additive.





Creating and Combining Views

developer.apple.com/tutorials/swiftui/creating-and-combining-views

DeveloperDiscoverDesignDevelopDistributeSupportAccount

SwiftUITutorials

Creating and Combining Vie...

SwiftUI Essentials

Creating and Combining Views

This tutorial guides you through building *Landmarks* — an iOS app for discovering and sharing the places you love. You'll start by building the view that shows a landmark's details.

To lay out the views, Landmarks uses *stacks* to combine and layer the image and text view components. To add a map to the view, you'll include a standard MapKit component. As you refine the view's design, Xcode provides real-time feedback so you can see how those changes translate into code.

Download the project files to begin building this project, and follow the steps below.

40min

Estimated Time

Project files

Xcode 11 beta

Ready. Detected Vue 2.5.17.

Filter components

<Root>

<App>

<ProjectDetail> router-view: /tutorials/:id/*

<Project key='topic://com.apple.SwiftUI/creating-and-combining-views'>

<NavigationBar>

<ProjectSection key=0>

<Hero>

<ProjectSection key=1>

<TaskList>

<ProjectSection key=2>

<ProjectSection key=3>

Select a component instance to inspect.

ConsoleWhat's New

vue-devtools

Detected Vue v2.5.17

backend.js:1585



Photo by [Katie Moum](#) on [Unsplash](#)

DESIGNED TO BE
APPROACHABLE TO
DEVELOPERS OF
PRACTICALLY ALL
ABILITIES

JSX EXAMPLE

```
import React from 'react';
import { render } from 'react-dom';
import HelloWorld from './HelloWorld';

const styles = {
  fontFamily: 'Avenir, Helvetica, Arial, sans-serif',
  textAlign: 'center',
  color: '#2c3e50',
  marginTop: '60px'
};

const App = () => {
  <div style={styles} className="app">
    <h1>This is a React.js component file!</h1>
  </div>
};

export default App;
```

SFC EXAMPLE

```
<template>
  <div>
    <h1>This is a Vue Single File Component (SFC) file!</h1>
  </div>
</template>

<script>
export default {
  name: 'App'
}
</script>

<style>
  #app {
    font-family: 'Avenir, Helvetica, Arial, sans-serif';
    text-align: center;
    color: #2c3e50;
    margin-top: 60px;
  }
</style>
```

THEIR **DOCS** ARE
ONE OF THE BEST
OUT THERE





bit Share Code

Guide 2.x

Installation

What is Vue.js?

Getting Started

Declarative Rendering

Conditionals and Loops

Handling User Input

Composing with Components

Relation to Custom Elements

Ready for More?

The Vue Instance

Template Syntax

Computed Properties and Watchers

Class and Style Bindings

Conditional Rendering

List Rendering

Event Handling

What is Vue.js?

If you'd like to learn more about Vue before diving in, we **created a video** walking through the core principles and a sample project.

If you are an experienced frontend developer and want to know how Vue compares to other libraries/frameworks, check out the [Comparison with Other Frameworks](#).

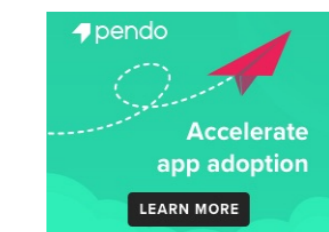
Getting Started



The official guide assumes intermediate level knowledge of HTML, CSS, and JavaScript. If you are totally new to frontend development, it might not be the best idea to jump right into a framework as your first step - grasp the basics then come back! Prior experience with other frameworks helps, but is not required.



[Become a Sponsor](#)



Accelerate App Adoption

ads via Carbon

Vue.js Docs

<https://vuejs.org/v2/guide/>

A close-up photograph of a dark asphalt surface. A large number '5' is drawn on the ground using white chalk. The chalk lines are slightly faded and uneven. Several dry, brown leaves are scattered around the drawing. The background is a solid dark blue color.

**GIVES YOU WHAT
YOU WANT **WHEN**
YOU NEED IT, THEN
IT GETS OUT OF
YOUR WAY.**

Paraphrased from Sarah Drasner

Order of words in component names

These rules help prevent errors, so learn and abide by them at all costs. Exceptions may exist, but should be very rare and only be made by those with expert knowledge of both JavaScript

ads via Carbon

<https://vuejs.org/v2/style-guide/>

SO WHAT ELSE IS THERE TO LEARN ABOUT?

- ▶ Props
- ▶ Slots
- ▶ Watch Property
- ▶ Custom Events
- ▶ Lifecycle Hooks
- ▶ Managing CSS in Vue.js
- ▶ State Management
- ▶ Animations
- ▶ Render Function
- ▶ Component Libraries
- ▶ Testing
- ▶ And much more!

NEXT STEPS...

RECOMMENDED RESOURCES

Free

- ▶ [Official Vue.js Guide](#)
- ▶ [VueMastery: Intro to Vue.js](#)
- ▶ [VueSchool Free Courses](#)

Paid

- ▶ [Vue Mastery](#)
- ▶ [Vue School](#)
- ▶ [Frontend Masters: Introduction to Vue.js](#)
- ▶ [Udemy: Vue.js 2 - The Complete Guide](#)

THANKS
EVERYONE!

@bencodezen



Questions?