



Module 3

VUE EVENT HANDLING

JAVA – MODULE 3, DAY 12

ELEVATE  YOURSELF





START STRONG,
STAY STRONG, AND
FINISH STRONG BY
ALWAYS REMEMBERING
WHY YOU'RE DOING IT IN
THE FIRST PLACE.

{Ralph Marston}



RECAP

Review

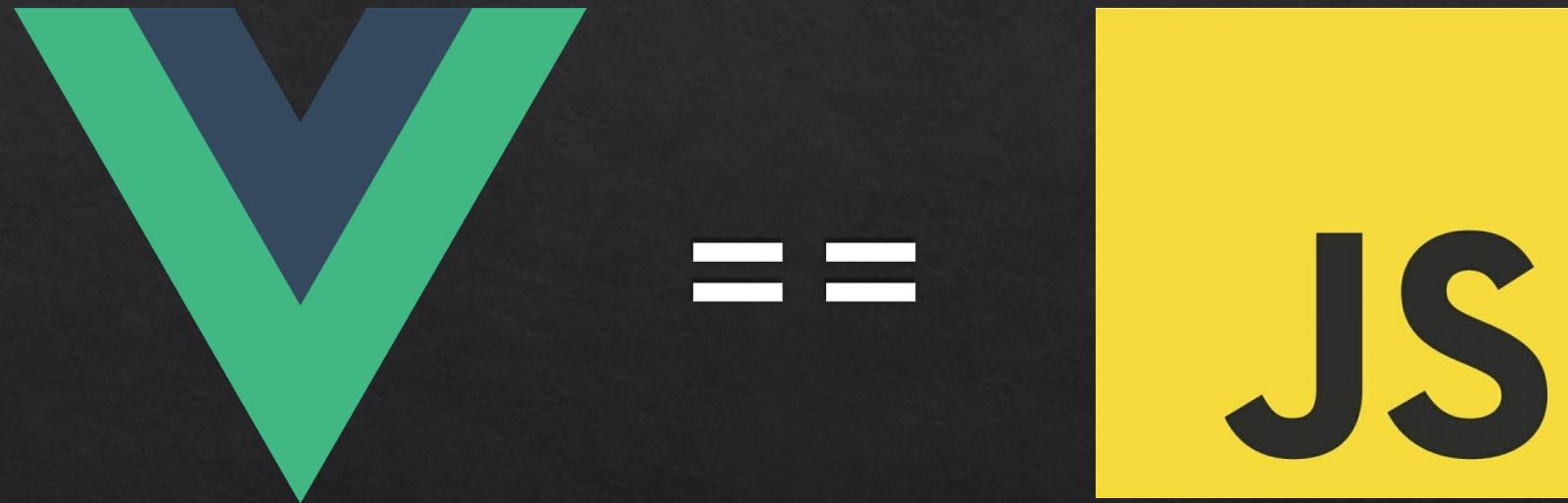




Review



VUE IS JAVASCRIPT



And More





VUE IS A JS FRAMEWORK

Review



Event Management

Easy DOM Manipulation

Established Patterns

Integration Points

Reusable Components

Data Binding

URL Management





New Vue Project

Review



```
> vue create project-name
```





Vue Projects

Review



> **npm install**

> **npm run serve**

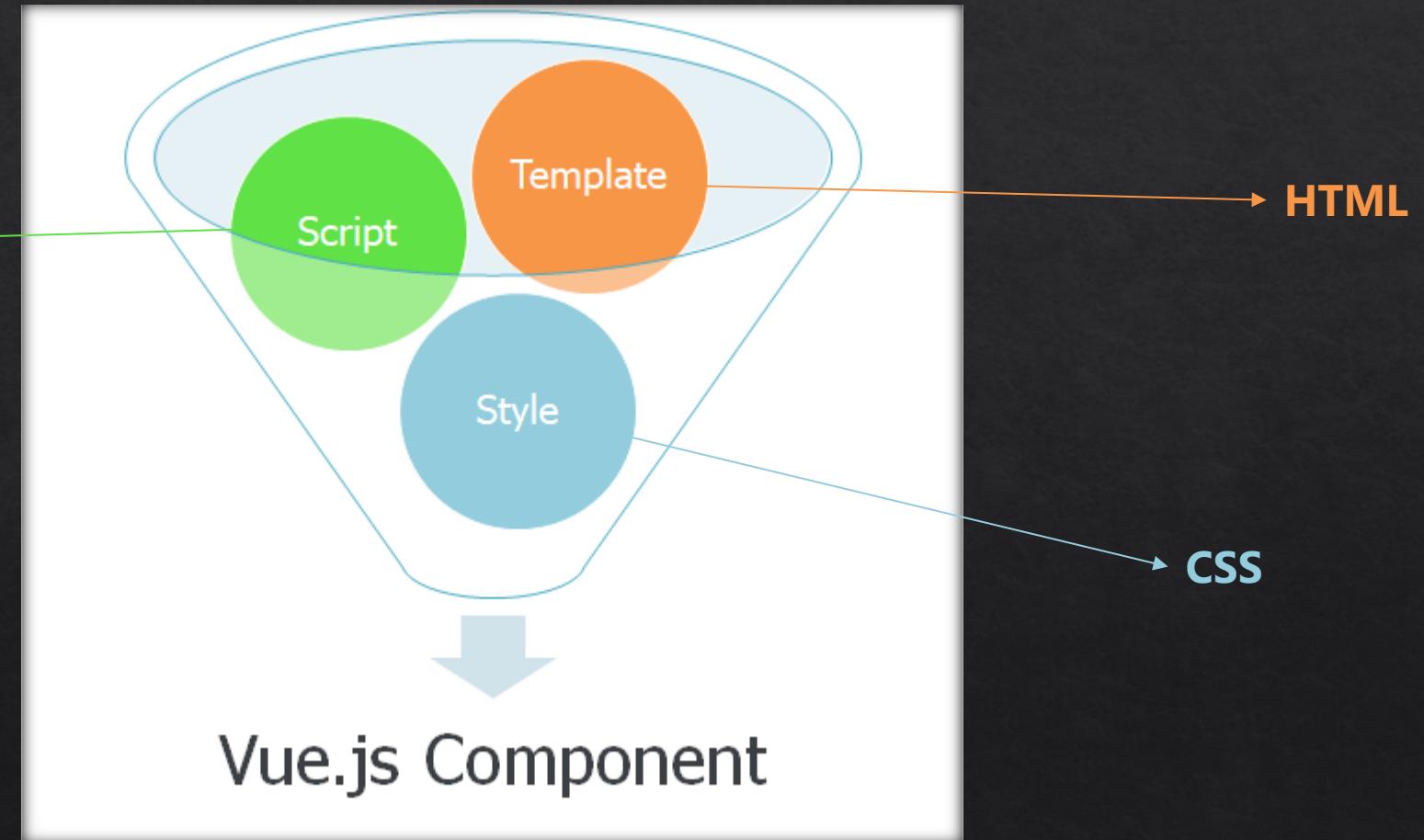




Review



PARTS OF A COMPONENT





TEMPLATE

Review



```
1 <template>
2   <main>
3     <section id="opening">
4       <h1>Aw Nuts!</h1>
5       <div>
6         <p>
7           The page you were looking for couldn't be found.
8           A dog probably ate it.
9         </p>
10        </div>
11      </section>
12    </main>
13 </template>
```

HTML



SCRIPT

Review



```
1 <script>
2   export default {
3     name: "MyComponentName",
4     data: () => {
5       return {
6         searchText: 'Hello Vue.js'
7       };
8     }
9   };
10 </script>
```





Review



STYLE

CSS

```
1 <style>
2   table {
3     font-size: medium;
4     color: #ffffff;
5     text-shadow: #3b3e42 1px 1px;
6     border: #8cc63f solid 0.15rem;
7     border-collapse: collapse;
8   }
9
10  thead th {
11    text-align: left;
12    text-transform: capitalize;
13    background-color: #00adee;
14  }
15
16  tbody th {
17    text-align: left;
18    padding-right: 0.5rem;
19  }
20
21  tbody tr:nth-child(odd) {
22    background-color: #999999;
23  }
24  tbody tr:nth-child(even) {
25    background-color: #3b3e42;
26  }
27 </style>
```



COMPUTED PROPERTIES

Review



```
1 computed: {
2   filteredQuestions() {
3     return this.questionList.filter(q =>
4       this.doesQuestionMatchSearch(q, this.searchText.toLowerCase())
5     );
6   },
7   isBusy() {
8     return this.workRemaining > 0;
9   }
10 },
```





V-BIND

Review



```
1 <template>
2   <main>
3     <section id="opening">
4       <h1>Search Results</h1>
5       <div>
6         <p>You searched for {{searchText}}</p>
7         
9       </div>
10      </section>
11    </main>
12 </template>
```



V-BIND:CLASS

Review



```
1 <article v-bind:class="{ blur: !question.isAnswerVisible,  
2                           correct: question.isCorrect == true,  
3                           incorrect: question.isCorrect == false}">  
4     <!-- Omitted -->  
5 </article>
```





V-FOR

Review



```
1 <div class="form-check" v-for="cat of categories" v-bind:key="cat.id">
2   <input type="checkbox" v-bind:id="cat.id" v-bind:name="cat.tag" />
3   <label>{{cat.tag}}</label>
4 </div>
```





Review



```
1 <div class="form-group">
2   <label for="question">Question:</label>
3   <input id="question"
4     type="text"
5     placeholder="Enter a question name"
6     v-model="question"
7     required />
8 </div>
```

Two-way data binding

v-model on input fields

You bind a data property to an input element using an attribute on the HTML elements called **v-model**. This style of data binding is two-way data binding. You can do that with text input elements with the v-model syntax:





V-MODEL MODIFIERS

Review



- **v-model.number** - Converts the input to a number and stores the number
- **v-model.trim** - Trims any whitespace at the left / right of the string
- **v-model.lazy** - Waits for the input element to lose focus





Review



Just a note when working with checkboxes.



CHECKBOXES AND V-MODEL

Multiple checkboxes, bound to the same Array:

HTML

```
<input type="checkbox" id="jack" value="Jack" v-model="checkedNames">
<label for="jack">Jack</label>
<input type="checkbox" id="john" value="John" v-model="checkedNames">
<label for="john">John</label>
<input type="checkbox" id="mike" value="Mike" v-model="checkedNames">
<label for="mike">Mike</label>
<br>
<span>Checked names: {{ checkedNames }}</span>
```

JS

```
new Vue({
  el: '...',
  data: {
    checkedNames: []
  }
})
```

Jack John Mike
 Checked names: ["Jack", "John"]

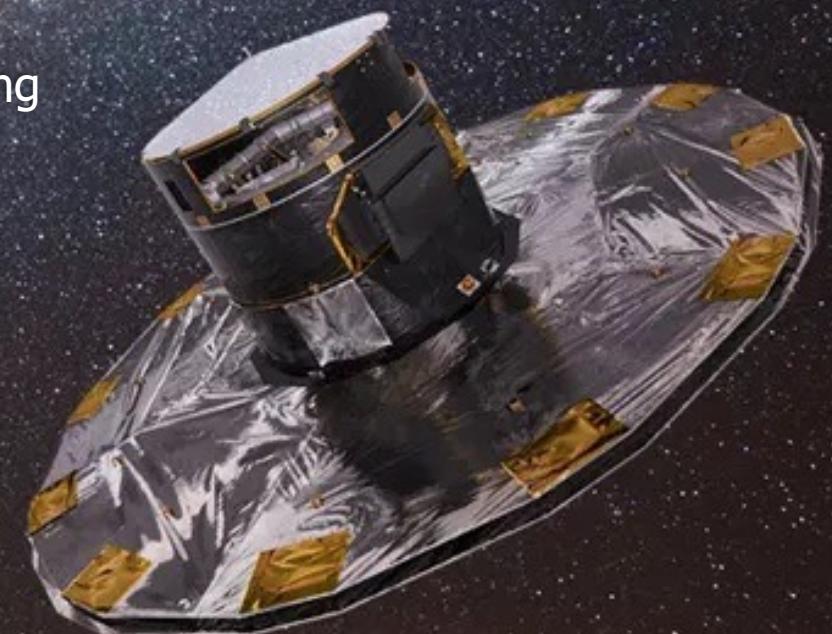


Lecture



AGENDA

- Forms
- Showing / Hiding
- Event Handling
- Methods
- Modifiers





Lecture



VUE - HTML FORMS

ADD NEW REVIEW FORM

LET'S CODE!



Lecture Code



Product Reviews for Cigar Parties for Dummies
Host and plan the perfect cigar party for all of your squirrelly friends.

2.75	1	0	2	1	0
Average Rating	1 Star Review	2 Star Reviews	3 Star Reviews	4 Star Review	5 Star Reviews

Add Review

Name:

Title:

Rating:

Review:

Save Cancel

Malcolm Gladwell
 What a book!

If certainly is a book. I mean, I can see that. Pages kept together with glue and there's writing on it, in some language.

Favorite?

Tim Ferriss
 Had a cigar party started in less than 4 hours.

It should have been called the four hour cigar party. That's amazing. I have a new idea for muse because of this.



Lecture



HOW DO WE ADD MORE COMPLEX BEHAVIOR TO OUR COMPONENTS?



METHODS



Lecture



```
1 <script>
2 export default {
3   // Methods contain functions which can be invoked from event handlers or other code
4   methods: {
5     /**
6      * Shows the add new question region
7      */
8     showQuestionForm() {
9       this.showAddQuestion = true;
10    },
11    /**
12     * Hides the add new question region
13     */
14    hideQuestionForm() {
15      this.showAddQuestion = false;
16    }
17  },
18  data() {
19    return {
20      showAddQuestion: false,
21    };
22  }
23};
24 </script>
```



TOGGLING FORM VISIBILITY



Lecture



Product Reviews for Cigar Parties for Dummies
Host and plan the perfect cigar party for all of your squirrelly friends.

2.75	1	0	2	1	0
Average Rating	1 Star Review	2 Star Reviews	3 Star Reviews	4 Star Review	5 Star Reviews

Add Review

Name:

Title:

Rating: 1 Star

Review:

Malcolm Gladwell

★★★ What a book!

It certainly is a book. I mean, I can see that. Pages kept together with glue and there's writing on it.
in some language.

Favorite?

Tim Ferriss

★★★★★ Had a cigar party started in less than 4 hours.

It should have been called the four hour cigar party. That's amazing. I have a new idea for muse
because of this.



Lecture



CONDITIONAL DISPLAY



Lecture



V-IF

```
2
3 <BusySpinner v-if="isBusy" />
4
```



V-ELSE



Lecture



```
1 <section id="questionList">
2
3   <BusySpinner v-if="isBusy" />
4
5   <div class="questionContainer" v-else>
6     <QuestionCard v-for="q in filteredQuestions" v-bind:key="q.question" v-bind:question="q"></QuestionCard>
7   </div>
8
9 </section>
```





Lecture



V-SHOW

```
1 <span class="showAnswer"  
2     v-show="!question.isAnswerVisible">  
3     {{showAnswerText}}  
4 </span>
```



Lecture



EVENT HANDLING



Lecture



EVENT HANDLING

Inline event handlers

Inline event handlers work well for basic operations.

```
<template>
  <div id="bridgeCommand">
    <a href="#" id="warpSpeedAccelerator" class="btn" v-on:click="warpSpeed += 1">Warp Drive</a>
    <p>The warp drive accelerator button was clicked, proceed to warp factor {{ warpSpeed }} </p>
  </div>
</template>
```



EVENT HANDLING

Method event handlers

Effectively functions the same way as the inline version but allows you to extract the logic and put it into a method that contains more complex operations.

```
<template>
  <div id="bridgeCommand">
    <a href="#" id="warpSpeedAccelerator" class="btn" v-on:click="setWarpSpeed(1)">Warp Drive</a>
    <p>The warp drive accelerator was clicked, proceed to warp factor {{ warpSpeed }} </p>
  </div>
</template>

<script>
  export default {
    name: "star-trek-command",
    data() {
      return {
        warpSpeed: 0
      };
    },
    methods: {
      setWarpSpeed(factor) {
        this.warpSpeed += factor;
      }
    }
  };
</script>
```

V-ON

Lecture





Lecture



V-ON

Event to respond to

```
1 <input type="button" value="Cancel" v-on:click="hideQuestionForm">
```

JS Code or Method to Invoke



Lecture



V-ON EVENTS

- v-on:**click**="someMethod"
- v-on:**change**="someMethod"
- v-on:**submit**="someMethod"
- v-on:**keyup**="someMethod"
- v-on:**blur**="someMethod"
- ...
- Basically anything we had before, just in Vue.



Lecture



V-ON KEYBOARD EVENTS

- v-on:keyup.**enter**="someMethod"
- v-on:keyup.**space**="someMethod"
- v-on:keyup.**page-down**="someMethod"
- v-on:keyup.**up**="someMethod"
- v-on:keyup.**down**="someMethod"
- v-on:keyup.**left**="someMethod"
- v-on:keyup.**right**="someMethod"



Lecture



V-ON AND METHODS

```
1 <input type="checkbox"
2     id="selectAll"
3     v-on:change="selectAllChanged($event)"
4     v-bind:checked="areAllSelected" />
```

```
1 methods: {
2   selectAllChanged(e) {
3     if (e.target.checked) {
4       // Do something if checked
5     } else {
6       // Do something if unchecked
7     }
8   },
9 },
10 // other code omitted...
```



V-ON AND METHODS

Event argument - \$event

Lecture



```
1 <input type="checkbox"
2   id="selectAll"
3   v-on:change="selectAllChanged($event)"
4   v-bind:checked="areAllSelected" />
```

```
1 methods: {
2   selectAllChanged(e) {
3     if (e.target.checked) {
4       // Do something if checked
5     } else {
6       // Do something if unchecked
7     }
8   },
9 },
10 // other code omitted...
```

In Vanilla JavaScript, you have access to the original DOM event. From there, you can determine where the event originated from. In Vue, the original DOM event is passed to the method by telling the listener to pass it using **\$event**.





V-ON MODIFIERS



- **v-on:click.stop** - Identical to **event.stopPropagation()**
- **v-on:click.prevent** - Identical to **event.preventDefault()**

Lecture



These also exist, but you'll likely never use them:

- **v-on:click.self** - Ignores bubbled up events from children
- **v-on:click.capture** - Uses capturing instead of bubbling
- **v-on:click.once** - Only care about the first occurrence





Lecture Code



TOGGLING FORM VISIBILITY

LET'S CODE!

Product Reviews for Cigar Parties for Dummies
Host and plan the perfect cigar party for all of your squirrelly friends.

2.75	1	0	2	1	0
Average Rating	1 Star Review	2 Star Reviews	3 Star Reviews	4 Star Review	5 Star Reviews

Add Review

Name:

Title:

Rating: 1 Star

Review:

Malcolm Gladwell
 What a book!

If certainly is a book. I mean, I can see that. Pages kept together with glue and there's writing on it. in some language.

Favorite?

Tim Ferriss
 Had a cigar party started in less than 4 hours.

It should have been called the four hour cigar party. That's amazing. I have a new idea for muse because of this.

ADDING REVIEWS LET'S CODE!



Lecture Code



Product Reviews for Cigar Parties for Dummies
Host and plan the perfect cigar party for all of your squirrelly friends.

2.75	1	0	2	1	0
Average Rating	1 Star Review	2 Star Reviews	3 Star Reviews	4 Star Review	5 Star Reviews

Add Review

Name:

Title:

Rating:

Review:

Malcolm Gladwell

★★★ What a book!

It certainly is a book. I mean, I can see that. Pages kept together with glue and there's writing on it.
in some language.

Favorite?

Tim Ferriss

★★★★★ Had a cigar party started in less than 4 hours.

It should have been called the four hour cigar party. That's amazing. I have a new idea for muse
because of this.



Lecture Code



FILTERING REVIEWS LET'S CODE!

Product Reviews for Cigar Parties for Dummies

Host and plan the perfect cigar party for all of your squirmly friends.

2.75	1	0	2	1	0
Average Rating	1 Star Review	2 Star Reviews	3 Star Reviews	4 Star Review	5 Star Reviews

[Add Review](#)

Name:

Title:

Rating: 1 Star

Review:

[Save](#) [Cancel](#)

Malcolm Gladwell

★★★ What a book!

If certainly is a book. I mean, I can see that. Pages kept together with glue and there's writing on it, in some language.

Favorite?

Tim Ferriss

★★★★ Had a cigar party started in less than 4 hours.

It should have been called the four hour cigar party. That's amazing. I have a new idea for muse because of this.



ADDITIONAL RESOURCES

- [Bootcamp OS](#)
- [Vue.js Documentation - \(https://vuejs.org/v2/guide/\)](#)
- [Vue CLI – \(https://cli.vuejs.org/\)](#)



Lecture Code





Exercise HW

Module 3

Vue Event Handling

New Tools! Let's Code!

Vue.js Event Handling



Developer's Toolbox



STUDENT EXERCISE

DUE NEXT BUSINESS DAY 11:59PM



Exercise HW



FIRST NAME	LAST NAME	USERNAME	EMAIL ADDRESS	STATUS
John	Smith	jsmith	jsmith@gmail.com	Active
Anna	Bell	abell	abell@yahoo.com	Active
George	Best	gbest	gbest@gmail.com	Disabled
Ben	Carter	bcarter	bcarter@gmail.com	Active
Katie	Jackson	kjackson	kjackson@yahoo.com	Active
Mark	Smith	msmith	msmith@foo.com	Disabled