

Displaying Data and Responding to Events



John Papa

DEVELOPER ADVOCATE

@john_papa www.johnpapa.net



Data Bindings and Events



How do we display data?



How do we allow the user to edit data?



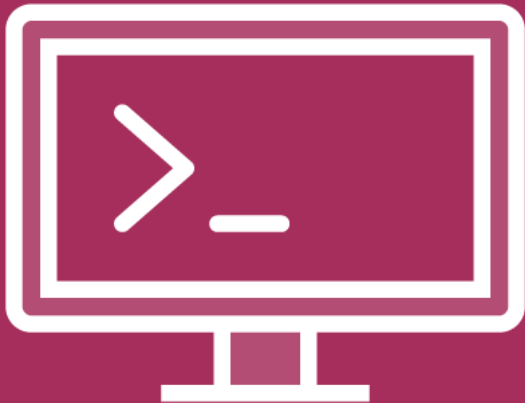
How do we make the UI change based on data?



How do we handle user interactions?



Demo



Binding to Properties



```
<a v-bind:href="github" target="_blank">
```

```
  <i class="fab fa-github fa-2x"></i>
```

Full v-bind syntax

```
</a>
```

```
<a :href="twitter" target="_blank">
```

```
  <i class="fab fa-twitter fa-2x"></i>
```

Shortcut syntax

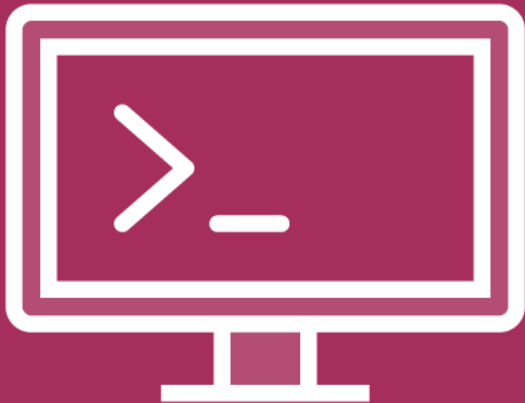
```
</a>
```

Property Binding

Bind to an HTML property



Demo



Displaying Text



```
<template>
  <div>Hello {{ name }}</div>
</template>

<script>
  export default {
    data() {
      return {
        name: "John"
      };
    }
  };
</script>
```

Displaying Text

Display models with `{{ model }}`

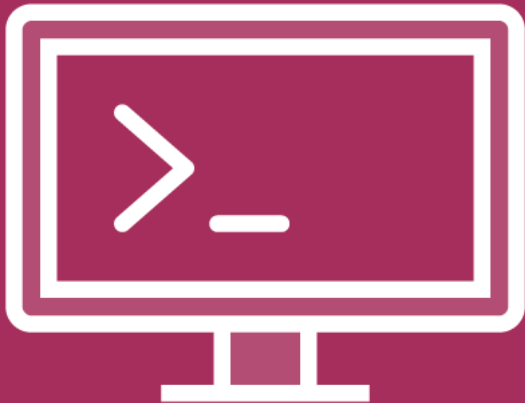
Can also use `v-text="model"`

Also known as interpolation

The data() function identifies the data models



Demo



Handling Events



Event Bindings

Execute when an event occurs

v-on:event="method-name"

@ is the shortcut syntax for v-on

heroes.vue

```
<template>

  <div>

    <input v-model="name" type="text">

    <button @click="ok">OK</button>

  </div>

</template>

<script>

  export default {

    data() { return { name: "John" }; },

    methods: {

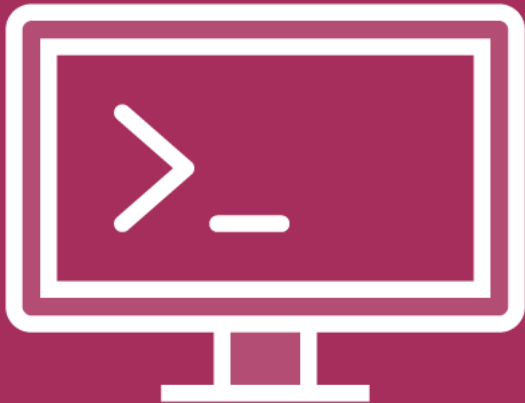
      ok() { /* do something */ }

    }

  };

</script>
```


Demo



Two-way Binding



Two-way Binding

The `hero.firstName` is shown in the input

The user types and the value of `hero.firstName` changes

heroes.vue

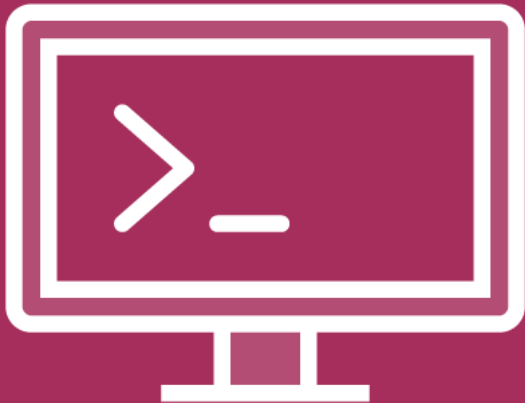
```
<div class="field">  
  <label class="label" for="firstName">first name</label>  
  <input class="input" id="firstName" v-model="hero.firstName" />  
</div>
```

Two-way binding



A blue line connects the `v-model="hero.firstName"` attribute in the code block above to the `Two-way binding` label in the block below.

Demo



Select boxes, key modifiers, and class bindings



```
<select id="power" v-model="hero.power" @keyup.esc="clearPower"  
  :class="{ invalid: !hero.power }">  
  <option disabled value>Select one</option>  
  <option>Speed</option>  
  <option>Invisibility</option>  
</select>
```

When the escape key is pressed

Bind the class name to a model

Key Modifiers and Class Bindings

Key modifiers bind to specific event scenarios

<https://vuejs.org/v2/guide/events.html#Key-Modifiers>

Class binding syntax `:class="{classname: expression}"`



Top Tips for Displaying Data

{{ }}

Interpolation

@event

Handle an event

:property

Bind to a property

v-model

Two-way binding



Summary



Display data

Let the user edit data

Change the UI based on data

Handle events from users

