

Paris - 2023



Vue.js

Animated by : Mohammed BAHRI
(Developer & Web Trainer)

VUE.JS

Learn Vue.js

PROGRAM

- ❑ Introduction
- ❑ The essentials of view.js
- ❑ Advanced aspects
- ❑ More: Exploiting an API (REST)
- ❑ Conclusion

WELCOME : Roundtable

Who are you, your job?

What are your professional goals?

What are your knowledge on the subject?



INTRODUCTION



Vue.js : Introduction

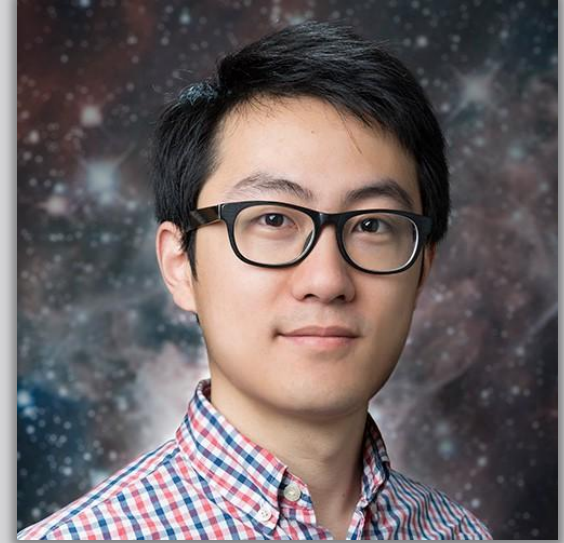
Vue.js (also called simply View), is an open-source JavaScript framework used to build user interfaces.

The library is used by Adobe, Alibaba and Gitlab.

Vue was created by Evan You after working for Google using AngularJS in several projects.

Vue.js : Introduction

- Creator : Evan You (尤雨溪)
- First version : February 11, 2014
- Latest Version : 2.6.10 (March 20, 2019)
- Deposit : <https://github.com/vuejs/view>
- Written in Javascript
- Type : JavaScript Framework
- License : MIT License
- Website : <https://vuejs.org>



Vue.js : Installation

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
</body>
</html>
```

Vue.js : Instance

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  {{ message }}
</div>

<script>
const app = Vue.createApp({
  data() {
    return {
      message: 'Hello World!'
    }
  }
}).mount('#app')
</script>
</body>
</html>
```


Vue.js : Data (attribute) : Array

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  {{ cars[0] }}
  {{ cars[1] }}
  {{ cars[2] }}
</div>
<script>
const app = Vue.createApp({
  data() {
    return {
      cars : [ 'BMW', 'RENAULT', 'PEUGEOT' ]
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Data (attribute) : Object

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  {{ cars.v1 }}
  {{ cars.v2 }}
  {{ cars.v3 }}
</div>
<script>
const app = Vue.createApp({
  data() {
    return {
      cars: { v1:'BMW', v2: 'RENAULT', v3: 'PEUGEOT' }
    }
  }
}).mount('#app')
</script>

</body>
</html>
```

Vue.js : Exercise 1

Create an instance containing the name, age and occupation of a person.

Display information using expressions (`{{ }}`).

Example:

Hello Bahri, you are 32 years old and you are web trainer.

Vue.js : Corrected 1

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p>Hello {{ name }}, you are {{ age }} years old and you are {{ profession }}.</p>
</div>
<script>
const app = Vue.createApp({
  data() {
    return {
      name: 'Bahri',
      age: 32,
      profession: 'web trainer'
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 2

Transform the table below into Vuejs data and then display it using bootstrap4.

Email informations

SNI	NAME	SPECIALITY	AVG	MENTION
S123	ALLON	TIN	15	Good
S321	BACARD	CDI	14	Good
S323	ALAIN	IPS	16	Very good

Vue.js : Corrected 2 (1)

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
<script src="https://unpkg.com/vue@next"></script>
<body>
  <div id="app" class="container">
    <h1>{{ parentMessage }}</h1>
    <table class="table table-bordered text-center">
      <tr>
        <th>SNI</th>
        <th>NAME</th>
        <th>SPECIALITY</th>
        <th>AVG</th>
        <th>MENTION</th>
      </tr>
      <tr>
        <td> {{ items[0].nsi }} </td>
        <td> {{ items[0].name }} </td>
        <td> {{ items[0].speciality }} </td>
        <td> {{ items[0].avg }} </td>
        <td> {{ items[0].mention }} </td>
      </tr>
    </table>
```

Vue.js : Corrected 2 (2)

```
<script>
const app = Vue.createApp({
  data(){
    return{
      parentMessage: 'Email informations',
      items: [
        { nsi : 'S123', name : 'ALLON', speciality : 'TIN', avg : 15, mention : 'Good'},
        { nsi : 'S321', name : 'DAVID', speciality : 'IPS', avg : 16, mention : 'Good'},
        // ...
      ]
    }
  }
}).mount('#app')
</script>
</div>
</body>
</html>
```

Vue.js : Function (Method)

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p>{{ message }}</p>
  <button v-on:click="reverseMessage">Reverse Message</button>
</div>
<script>
const app = Vue.createApp({
  data() {
    return {
      message: 'Hello World!'
    }
  },
  methods: {
    reverseMessage: function () {
      this.message = this.message.split('').reverse().join('')
    }
  }
}).mount('#app')
</script>
</body>
</html>
```


Vue.js : Function (Method) with {{ ... }}

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p>{{ hello('Daniel') }}</p>
</div>
<script>
const app = Vue.createApp({
  methods: {
    hello: function (name) {
      return 'Hello ' + name
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 3

Create a method taking the name, age and occupation of a person as parameter and then returns them.

Call the method using expressions (`{{ }}`).

Example:

Hello Bahri, you are 32 years old and you are web trainer.

Vue.js : Corrected 3

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p>{{ getMessage('Bahri', 32, 'web trainer') }}</p>
</div>
<script>
const app = Vue.createApp({
  methods: {
    getMessage: function (name, age, profession) {
      return 'Hello ' + name + ', you are ' + age + ' years old and you are ' + profession
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 4

Create an instance containing the name, age and occupation of a person.

Create a method (named constructor ()) to initialize the attributes of the instance.

Create a method (named display ()) to display the instance data.

Test the methods using expressions ({{ }}).

Example:

Hello Bahri, you are 32 years old and you are web trainer.

Vue.js : Corrected 4

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  {{ constructor('Bahri',32,'web trainer') }}
  <p>{{ show() }}</p>
</div>
<script>
const app = Vue.createApp({
  methods: {
    constructor: function(name, age, profession) {
      this.name = name;
      this.age = age;
      this.profession = profession;
    },
    show: function() {
      return 'Hello ' + this.name + ', you are ' + this.age + ' years old.'
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 5

Create an instance containing methods to do:

The addition, subtraction, multiplication and division of two parameters.

Test the methods using expressions (`{{ }}`).

Example:

$$12 + 3 = 15$$

$$12 - 3 = 9$$

$$12 * 3 = 36$$

$$12 / 3 = 4$$

$$12 ^ 3 = 1728$$

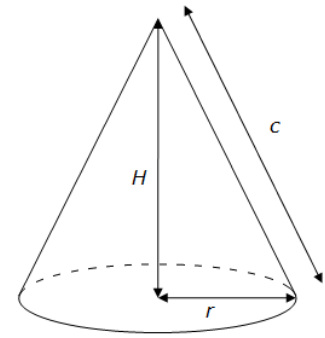
Vue.js : Corrected 5

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p>{{ addition(12,3) }}</p>
</div>
<script>
const app = Vue.createApp({
  methods: {
    addition: function(x,y){ return x + ' + ' + y + ' = ' + (x + y); },
    subtraction: function(x,y){ return x + ' - ' + y + ' = ' + (x - y); },
    multiplication: function(x,y){ return x + ' * ' + y + ' = ' + (x * y); },
    division: function(x,y){ return x + ' / ' + y + ' = ' + (x / y); },
    exponentiation: function(x,y){ return x + ' ^ ' + y + ' = ' + Math.pow(x,y); }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 6

Write a parameterized function that calculates the volume of a cone.

Rule: $\text{Volume} = (\text{Radius}^2 * \text{Height} * \pi) / 3$



Vue.js : Corrected 6

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  {{ volumeConeCalculate(12,3) }}
  <p>{{ vc }}</p>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      vc:0,
    }
  },
  methods: {
    volumeConeCalculate: function(radius, height){
      this.vc = (height * Math.pow(radius,2) * Math.PI) / 3;
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 7

Create an instance containing methods for formatting a string in :
Upper, Lower and capitalize case.

Test the methods using expressions (`{{ }}`).

Example:

I AM A STRING

i am a string

I am a string

Vue.js : Corrected 7

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p>{{ toUpper('i am a string') }}</p>
  <p>{{ toLower('I am a STRING') }}</p>
  <p>{{ toCapitalize('i AM a string') }}</p>
</div>
<script>
const app = Vue.createApp({
  methods: {
    toUpper: function (str) { return str.toUpperCase() },
    toLower: function (str) { return str.toLowerCase() },
    toCapitalize: function (str) { return str.charAt(0).toUpperCase() + str.slice(1).toLowerCase() }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : V-Bind

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p v-bind:title="message">
    Hover your mouse over me for a few seconds to see my dynamically bound title!
  </p>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      message: 'You loaded this page on ' + new Date().toLocaleString()
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Bind (Shortcuts)

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p :title="message">
    Hover your mouse over me for a few seconds to see my dynamically bound title!
  </p>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      message: 'You loaded this page on ' + new Date().toLocaleString()
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Loop

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <span v-for="n in 10"> {{ n }} </span>
</div>
<script>
const app = Vue.createApp({
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Loop

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<ul id="app">
  <li v-for="item in items">
    {{ item }}
  </li>
</ul>
<script>
const app = Vue.createApp({
  data(){
    return {
      items: ['BMW', 'Renault']
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Loop

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<ul id="app">
  <li v-for="(item, index) in items">
    {{ index }} - {{ item }}
  </li>
</ul>
<script>
const app = Vue.createApp({
  data(){
    return {
      items: ['BMW', 'Renault']
    }
  }
}).mount('#app')
</script>
</body>
</html>
```


Vue.js : Exercise 8

```
items:
[
  { Sender:'Nicolas Elbaz', Recipient:'Christel Lavaut' },
  { Sender: 'Jérôme Claustre', Recipient: 'Mohammed Bahri' }
]
```



Email informations

Email : 1

- Sender - Nicolas Elbaz
- Recipient - Christel Lavaut

Email : 2

- Sender - Jérôme Claustre
- Recipient - Mohammed Bahri

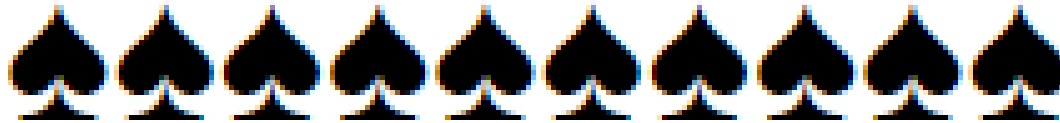
Vue.js : Corrected 8

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <span>{{ parentMessage }}</span>
  <ul v-for="(item, index) in items">
    <u> Email : {{ index + 1 }} </u>
    <li v-for="(it, ix) in item">
      {{ ix }} - {{ it }}
    </li>
  </ul>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      parentMessage: 'Email informations',
      items: [{ Sender: 'Nicolas Elbaz', Recipient: 'Christel Lavaut' },
               { Sender: 'Jérôme Claustre', Recipient: 'Mohammed Bahri' }]
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 9

Create a loop producing a horizontal line of 10 hearts.

Example:



Vue.js : Corrected 9

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <span v-for="n in 10">&spades;</span>
</div>

<script>
const app = Vue.createApp({
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 10

Use the loops to produce the rendering below :



Vue.js : Corrected 10

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <ol type="a">
    <span v-for="n in 5">
      <li>
        <span v-for="n in 10">
          
        </span>
      </li>
    </span>
  </ol>
</div>
<script>
const app = Vue.createApp({
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Condition

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <span v-if="seen">Now you see me</span>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      seen: true
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Condition

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <div v-if="type == 'A'">
    Yes
  </div>
  <div v-else-if="type == 'B'">
    No
  </div>
  <div v-else>
    I don't know!
  </div>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      type: 'A'
    }
  }
}).mount('#app')
</script>
</body>
</html>
```


Vue.js : Using ternary

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <p>{{ (age >= 18) ? 'YES' : 'NO' }}</p>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      age: 17
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 11

Write a conditional expression to display a message if a person is both a major and a developer.

Example : "Hello Sarah! You are welcome!".

If not, the expression displays "Sorry! You are not authorized".

Vue.js : v-if vs v-show

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <div v-if="access">Hello! You are welcome!</div>
  <!-- -->
  <div v-show="access">Hello! You are welcome! </div>
  <!-- div style="display: none;">Hello! You are welcome! </div -->
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      access:false
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Events

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <button v-on:click="counter += 1">Add 1</button>
  <!-- button @click="counter += 1">Add 1</button -->
  <p>Le bouton ci-dessus a été cliqué {{ counter }} fois.</p>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      counter: 0
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 12

Random of a letter between A and K

Start Random

The lettre is : D

Vue.js : Exercise 13

mouseover



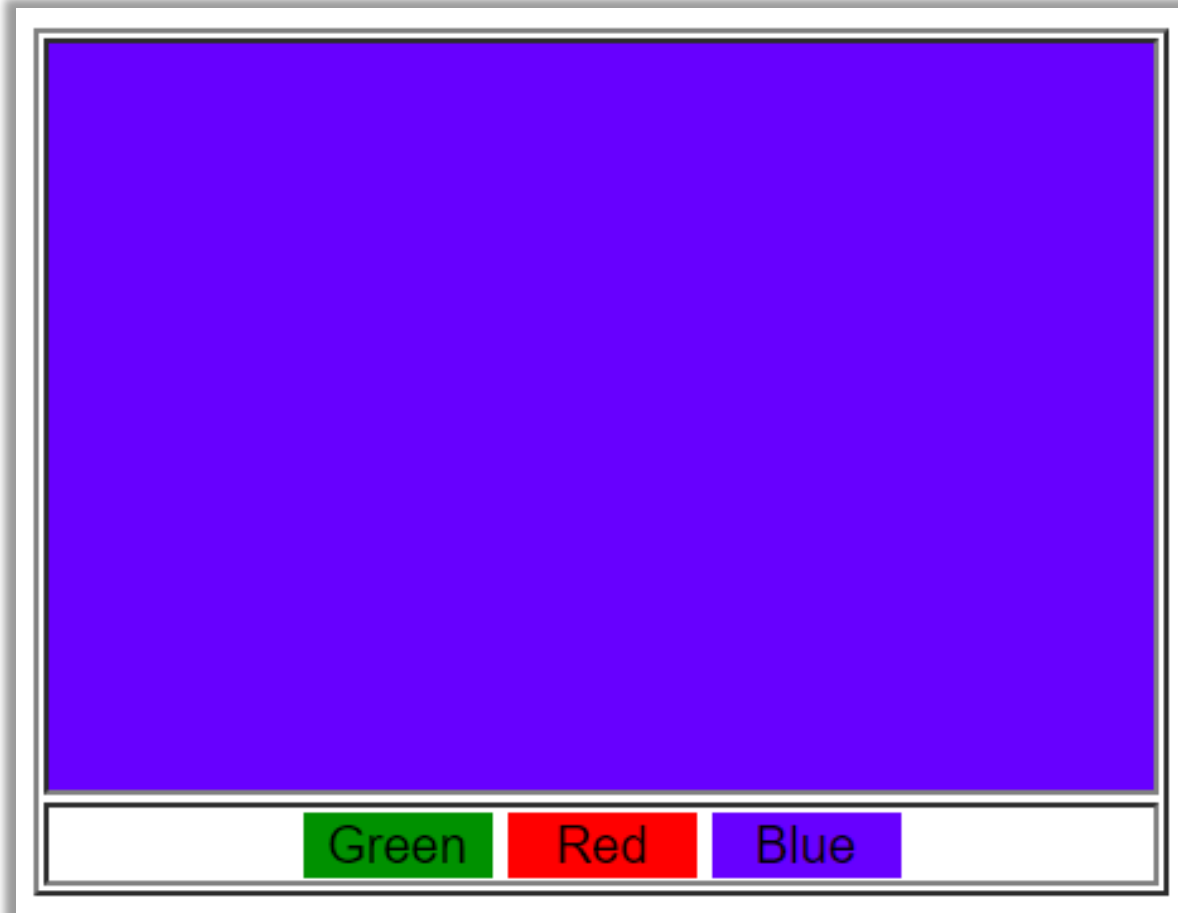
mouseout



Vue.js : HTML

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <span v-html="msg"></span>
  <span>{{ msg }}</span>
</div>
<script>
const app = Vue.createApp({
  data(){
    return {
      msg: '<p><strong>Hello</strong></p>'
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 14



Vue.js : Exercise 15



Vue.js : CSS

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<style>
  .active { color:green }
</style>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <div v-bind:class="{ active: isActive }">Some text...</div>
</div>
<script>
const app = Vue.createApp({
  data() {
    return {
      isActive: true,
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : CSS

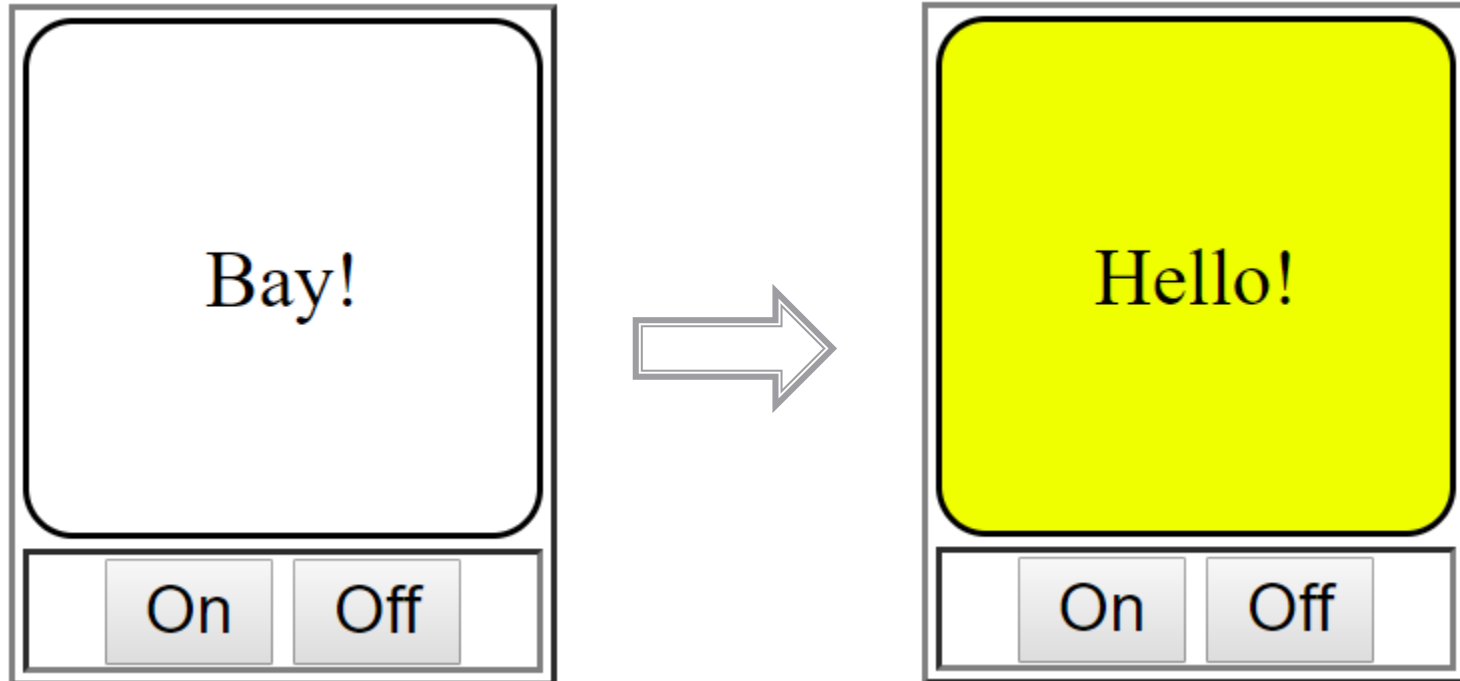
```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<style>
  .active { color:green }
</style>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <div v-bind:style="{ color: activeColor, fontSize: fontSize + 'px' }">
    Some text...
  </div>
</div>
<script>
const app = Vue.createApp({
  data() {
    return {
      activeColor: 'red',
      fontSize: 30
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : CSS

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<style>
  .active { color: green }
</style>
<script src="https://unpkg.com/vue@next"></script>
<body>
<div id="app">
  <div v-bind:style="styleObject">Some text...</div>
</div>
<script>
const app = Vue.createApp({
  data() {
    return {
      styleObject: {
        color: 'red',
        fontSize: '13px'
      }
    }
  }
}).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 16

CSS



Vue.js : V-Model

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
  <div id="app">
    <input v-model="message">
    <p>{{ message }}</p>
  </div>
<script>
  const app = Vue.createApp({
    data() {
      return {
        message: 'Hello Vue!'
      }
    }
  }).mount('#app')
</script>
</body>
</html>
```

Vue.js : Input validation

```
<!DOCTYPE html>
<html>
<meta charset="utf-8"/>
<script src="https://unpkg.com/vue@next"></script>
<body>
  <div id="app">
    <label>Enter a name </label>
    <input type="text" v-model="name" name="name">
    <input type="submit" value="Submit" v-on:click="checkForm">
    <p> {{ message }} </p>
  </div>
<script>
  const app = Vue.createApp({
    data() {
      return {
        name: null, message: null
      }
    },
    methods: {
      checkForm: function () {
        if (this.name) { this.message = 'Hello ' + this.name; }
        else { this.message = 'Required field!'; }
      }
    }
  }).mount('#app')
</script>
</body>
</html>
```

Vue.js : Exercise 17

Please correct the following error(s):

- Name required.
- Age required.

Name

Age

Name

Age

Hello Marck

Vue.js : Exercise 18

Calculator

First number required.

▼ Operation required.

Second number required.

Submit



Calculator

3

+

10

Submit

$3 + 10 = 13$

Vue.js : Exercise 19

Currency converter

Enter a sum

- ☐ Convert to Dollar (\$)
☐ Convert to Euro (€)

Submit

Please complete all fields!

Currency converter

Enter a sum




- ☒ Convert to Dollar (\$)
☐ Convert to Euro (€)

Submit

10 euro(s) = 11 dollar(s).

Vue.js : Exercise 20

Quiz App

	<p><input type="radio"/> Angular</p> <p><input checked="" type="radio"/> Vue</p> <p><input type="radio"/> React</p> <p><input type="radio"/> jQuery</p>
	<p><input checked="" type="radio"/> Angular</p> <p><input type="radio"/> Vue</p> <p><input type="radio"/> lievre</p> <p><input type="radio"/> jQuery</p>
	<p><input type="radio"/> Angular</p> <p><input type="radio"/> Vue</p> <p><input type="radio"/> React</p> <p><input checked="" type="radio"/> jQuery</p>
<div>Send</div>	

You have 2/3 good answer(s).