**쉽고 빠르게 배우는Vue.js 2 프로그래밍**

**지은이:알렉스 키리아키디스, 코스타스 매니아티스**

**출판사: 위키북스**

**소스코드: https://github.com/hootlex/the-majesty-of-vuejs-2/archive/master.zip**

1. **Vue.js 기초**
   1. **Vue.js 설치**
      1. **독립 실행 버전**

<http://vuejs.org/js/vue.js>

http://vuejs.org/js/vue.min.js

* + 1. **CDN 사용**

<https://cdn.jsdelivr.net/vue/2.1.10/vue.min.js>

https://cdnjs.cloudflare.com/ajax/libs/vue/2.1.10/vue.min.js

* + 1. **NPM 사용**

npm install vue

npm install vue@csp

npm install vuejs/vue#dev

* + 1. **Bower 사용**

bower install vue

* 1. **시작하기**
     1. **안녕하시나이까 전하!**

hello.html

<html>

<head>

<title>Hello Vue</title>

</head>

<body>

<div id=”app”>

<h1>{{message}}:: 바인딩 익스프레션</h1>

<input v-model=”message”>

</div>

</body>

<script src=”[https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js”></script](https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js)>

<script>

new Vue({

el: ‘#app’,

data: {

message: ‘Greetings your majesty!’

}

})

</script>

</html>

* + 1. **양방향 바인딩**
    2. **jQuery와 비교**
  1. **디렉티브**
     1. **v-show**

<html>

<head>

<title>Hello Vue</title>

</head>

<body>

<div id="app">

<h1 v-show="!message">You must send a message for help!</h1>

<textarea v-model="message"></textarea>

<button v-show="message">

Send word to allies for help!

</button>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '#app',

data: {

message: 'Greetings your majesty!'

}

})

</script>

</html>

* + 1. **v-if**

객체 자제가 사라짐

<html>

<head>

<title>Hello Vue</title>

</head>

<body>

<div id="app">

<h1 v-if="!message">You must send a message for help!</h1>

<textarea v-model="message"></textarea>

<button v-if="message">

Send word to allies for help!

</button>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '#app',

data: {

message: 'Greetings your majesty!'

}

})

</script>

</html>

**v-if template 사용**

template는 렌더링 되지 않음

<html>

<head>

<title>Hello Vue</title>

</head>

<body>

<div id="app">

<template v-if="!message">

<h1>You must send a message for help!</h1>

<p>Dispatch a messenger immediately</p>

<p>To nearby kingdom of Hearts!</p>

</template>

<textarea v-model="message"></textarea>

<button v-show="message">

Send word to allies for help!

</button>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '#app',

data: {

message: ''

}

})

</script>

</html>

* + 1. **v-else**

<html>

<head>

<title>Hello Vue</title>

</head>

<body>

<div id="app">

<h1 v-if="!message">You must send a message for help!</h1>

<h2 v-else>You have sent a message</h2>

<textarea v-model="message"></textarea>

<button v-show="message">

Send word to allies for help!

</button>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '#app',

data: {

message: ''

}

})

</script>

</html>

* + 1. **v-if vs. v-show**
  1. **리스트 렌더링**
     1. **부트스트랩 설치와 사용**

el: css 선택자나 실제 HTML 엘리먼트

css가 여러 개일 경우 첫번째

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Bootstrap</title>

</head>

<body>

<div class="container">

<h1>Hello Bootstrap, sit next to Vue.</h1>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container'

})

</script>

</html>

* + 1. **v-for**
       1. **범위를 가진 v-for**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Bootstrap</title>

</head>

<body>

<div class="container">

<h1>The multiplication of 4.</h1>

<ul class="list-group">

<li v-for="i in 11" class="list-group-item">

{{ i-1 }} times 4 equals {{ (i-1) \* 4 }}.

</li>

</ul>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container'

})

</script>

</html>

* + 1. **배열 렌더링**
       1. **배열 순회**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Bootstrap</title>

</head>

<body>

<div class="container">

<h1>array</h1>

<div>

<ul class="list-group">

<li v-for="story in stories" class="list-group-item">

Someone said "{{ story }}"

</li>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

stories:[

"I crashed my car today!",

"Yesterday, someone stole my bag!",

"Someone ate my chocolate...",

]

}

})

</script>

</html>

* + - 1. **객체 배열 순회**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Stories</title>

</head>

<body>

<div class="container">

<h1>object array</h1>

<div>

<ul class="list-group">

<li v-for="(story, index) in stories" class="list-group-item">

{{ index }} {{ story.writer }} said "{{ story.plot }}"

</li>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

stories:[

{

plot: "I crashed my car today!",

writer: "Alex"

},

{

plot: "Yesterday, someone stole my bag!",

writer: "John"

},

{

plot: "Someone ate my chocolate...",

writer: "Alex"

},

]

}

})

</script>

</html>

* + 1. **v-for를 이용한 객체 프로퍼티 순회**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Stories</title>

</head>

<body>

<div class="container">

<h1>object array</h1>

<div>

<ul class="list-group">

<li v-for="(value, key, index) in story" class="list-group-item">

{{ index }} : {{ key }} : {{ value }}

</li>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

story:

{

plot: "I crashed my car today!",

writer: "Alex",

upvotes: 47

}

}

})

</script>

</html>

* 1. **상호 작용**
     1. **이벤트 처리**
        1. **인라인 이벤트 핸들링**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Upvote</title>

</head>

<body>

<div class="container">

<button v-on:click="upvotes++">

Upvote! {{upvotes}}

</button>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

upvotes: 0

}

})

</script>

</html>

* + - 1. **메서드를 이용한 이벤트 처리**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Upvote</title>

</head>

<body>

<div class="container">

<button v-on:click="upvote">

Upvote! {{upvotes}}

</button>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

upvotes: 0

},

methods: {

upvote: function(){

this.upvotes++;

}

}

})

</script>

</html>

* + - 1. **v-on의 축약형**

<button @click="upvote">

Upvote! {{upvotes}}

</button>

* + 1. **이벤트 한정자**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Calculator</title>

</head>

<body>

<div class="container">

<h1>Type 2 numbers and choose operation</h1>

<form class="form-inline">

<input v-model.number="a" class="form-control">

<select v-model="operator" class="form-control">

<option>+</option>

<option>-</option>

<option>\*</option>

<option>/</option>

</select>

<input v-model.number="b" class="form-control">

<button type="submit" @click.prevent="calculate" class="btn btn-primary">

Calculate

</button>

</form>

<h2>Result: {{a}} {{operator}} {{b}} = {{c}}</h2>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

a: 1,

b: 2,

c: null,

operator: "+",

},

methods: {

calculate: function(event){

switch(this.operator){

case "+":

this.c = this.a + this.b

break;

case "-":

this.c = this.a - this.b

break;

case "\*":

this.c = this.a \* this.b

break;

case "/":

this.c = this.a / this.b

break;

}

}

}

})

</script>

</html>

.prevent

.stop

.capture

.self

* + 1. **키 한정자**

@keyup.13=”calculate”

enter

tab

delete

esc

space

up

down

left

right

<input v-model=”a” @keyup.enter=”calculate”>

* + 1. **계산된 프로퍼티**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div class="container">

a={{ a }}, b={{ b }}

<input v-model.number="a">

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

a: 1,

},

computed: {

b: function(){

return this.a + 1

}

}

})

</script>

</html>

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Calculator</title>

</head>

<body>

<div class="container">

<h1>Type 2 numbers and choose operation</h1>

<form class="form-inline">

<input v-model.number="a" class="form-control">

<select v-model="operator" class="form-control">

<option>+</option>

<option>-</option>

<option>\*</option>

<option>/</option>

</select>

<input v-model.number="b" class="form-control">

</form>

<h2>Result: {{a}} {{operator}} {{b}} = {{c}}</h2>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

a: 1,

b: 2,

operator: "+",

},

computed: {

c: function(){

switch(this.operator){

case "+":

return this.a + this.b

break;

case "-":

return this.a - this.b

break;

case "\*":

return this.a \* this.b

break;

case "/":

return this.a / this.b

break;

}

}

}

})

</script>

</html>

* 1. **필터**
     1. **필터링된 결과**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>User Stories</title>

</head>

<body>

<div class="container">

<h1>Let's hear some stories!</h1>

<div>

<h3>Alex's stories</h3>

<ul class="list-group">

<li v-for="(story, index) in storiesBy('Alex')" class="list-group-item">

{{ story.writer }} said "{{ story.plot }}"

</li>

</ul>

<h3>John's stories</h3>

<ul class="list-group">

<li v-for="(story, index) in storiesBy('John')" class="list-group-item">

{{ story.writer }} said "{{ story.plot }}"

</li>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

stories:[

{

plot: "I crashed my car today!",

writer: "Alex"

},

{

plot: "Yesterday, someone stole my bag!",

writer: "John"

},

{

plot: "Someone ate my chocolate...",

writer: "John"

},

{

plot: "Someone ate my chocolate...",

writer: "Alex"

},

]

},

methods:{

storiesBy: function(writer){

return this.stories.filter(function(story){

return story.writer === writer

})

},

}

})

</script>

</html>

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>User Stories</title>

</head>

<body>

<div class="container">

<h1>Let's hear some stories! ({{famous.length}})</h1>

<div>

<ul class="list-group">

<li v-for="(story, index) in famous" class="list-group-item">

{{ story.writer }} said "{{ story.plot }}"

and upvoted {{ story.upvotes }} times.

</li>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

stories:[

{

plot: "I crashed my car today!",

writer: "Alex",

upvotes: 28

},

{

plot: "Yesterday, someone stole my bag!",

writer: "John",

upvotes: 8

},

{

plot: "Someone ate my chocolate...",

writer: "John",

upvotes: 51

},

{

plot: "Someone ate my chocolate...",

writer: "Alex",

upvotes: 74

},

]

},

computed:{

famous: function(){

return this.stories.filter(function(item){

return item.upvotes > 25;

})

},

}

})

</script>

</html>

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>User Stories</title>

</head>

<body>

<div class="container">

<h1>Let's hear some stories!</h1>

<div>

<h3>Alex's stories</h3>

<ul class="list-group">

<li v-for="(story, index) in storiesBy('Alex')" class="list-group-item">

{{ story.writer }} said "{{ story.plot }}"

</li>

</ul>

<h3>John's stories</h3>

<ul class="list-group">

<li v-for="(story, index) in storiesBy('John')" class="list-group-item">

{{ story.writer }} said "{{ story.plot }}"

</li>

</ul>

<div class="form-group">

<label for="query">

What are you looking for?

</label>

<input v-model="query" class="form-control">

</div>

<h3>Search results:</h3>

<ul class="list-group">

<li v-for="story in search" class="list-group-item">

{{ story.writer }} said "{{ story.plot }}"

</li>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

stories:[

{

plot: "I crashed my car today!",

writer: "Alex"

},

{

plot: "Yesterday, someone stole my bag!",

writer: "John"

},

{

plot: "Someone ate my chocolate...",

writer: "John"

},

{

plot: "Someone ate my chocolate...",

writer: "Alex"

},

],

query: ''

},

methods:{

storiesBy: function(writer){

return this.stories.filter(function(story){

return story.writer === writer

})

},

},

computed:{

search: function(){

var query = this.query

return this.stories.filter(function(story){

return story.plot.includes(query)

})

}

}

})

</script>

</html>

* + 1. **결과 정렬**

compareFunction(a,b)가 0보다 작으면 b가 먼저

0이면 그대로 0보다 크면 a가 먼저

함수안에서는 a-b가 이루어짐

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>User Stories</title>

</head>

<body>

<div class="container">

<h1>Let's hear some stories!</h1>

<div>

<ul class="list-group">

<li v-for="(story, index) in orderedStories" class="list-group-item">

{{ story.writer }} said "{{ story.plot }}"

and upvoted {{ story.upvotes }} times.

</li>

</ul>

<button @click="order = order \* -1">Reverse Order</button>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

stories:[

{

plot: "I crashed my car today!",

writer: "Alex",

upvotes: 28

},

{

plot: "Yesterday, someone stole my bag!",

writer: "John",

upvotes: 8

},

{

plot: "Someone ate my chocolate...",

writer: "John",

upvotes: 51

},

{

plot: "Someone ate my chocolate...",

writer: "Alex",

upvotes: 74

},

],

order: -1

},

computed:{

orderedStories: function(){

var order = this.order;

return this.stories.sort(function(a,b){

return (a.upvotes - b.upvotes) \* order;

})

},

}

})

</script>

</html>

* + 1. **사용자 정의 필터**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>The Gotham Gazzete</title>

</head>

<body>

<div class="container">

<h1>Real identities of Super Heroes!</h1>

<div>

<ul class="list-group">

<li v-for="(hero, index) in heroes" class="list-group-item">

{{ hero | snitch }}

</li>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

Vue.filter('snitch', function(hero){

return hero.secretId + ' is '

+ hero.firstname + ' '

+ hero.lastname + ' in real life!'

})

new Vue({

el: '.container',

data: {

heroes:[

{ firstname: "Bruce", lastname: 'Wayne', secretId: 'Batman' },

{ firstname: "Clark", lastname: 'Kent', secretId: 'Superman' },

{ firstname: "Jay", lastname: 'Garrick', secretId: 'Flash' },

{ firstname: "Peter", lastname: 'Parker', secretId: 'Spider-Man' }

]

},

})

</script>

</html>

* + 1. **유틸리티 라이브러리**

<https://lodash.com>

<http://underscorejs.org>

<https://sugarjs.com/>

var kids = [

{ name: ‘Stan’, strength: 70, intelligence: 70},

{ name: ‘Kyle’, strength: 40, intelligence: 80},

{ name: ‘Eric’, strength: 45, intelligence: 80},

{ name: ‘Kenny’, strength: 100, intelligence: 70},

]

\_.orderBy(kids, [‘intelligence’, ‘strength’], [‘desc’, ‘asc’])

* 1. **컴포넌트**
     1. **컴포넌트란?**
     2. **컴포넌트 사용하기**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div class="container">

<story></story>

<story></story>

<story></story>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script>

Vue.component('story', {

template: '<h1>My horse is amazing!</h1>'

});

new Vue({

el: '.container',

})

</script>

</html>

* + 1. **템플릿**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div class="container">

<story></story>

<story></story>

<story></story>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script type="text/template" id="story1-template">

<h1>My horse is amazing!</h1>

</script>

<template id="story-template">

<h1>My horse is amazing!</h1>

</template>

Vue.component('story', {

template: '#story-template'

});

<script>

new Vue({

el: '.container',

})

</script>

</html>

* + 1. **프로퍼티**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div class="container">

<story plot="My horse is amazing."is-user='customer'></story>

<story plot="Narwhals invented Shish Kebab."></story>

<story plot="The dark side of the Force is stronger"></story>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script type="text/template" id="story1-template">

<h1>My horse is amazing!</h1>

</script>

<template id="story-template">

<h1>{{ plot }}</h1>

</template>

<script>

Vue.component('story', {

props: ['plot', 'isUser'],

template: '#story-template'

});

new Vue({

el: '.container',

})

</script>

</html>

property는 js는 카멜 케이스 html은 kebab-case(-)

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Awesome Stories</title>

</head>

<body>

<div class="container">

<story v-bind:story="{plot: 'My horse is amazing.', writer: 'Mr. Weebl'}"></story>

<story v-bind:story="{plot: 'Narwhals invented Shish Kebab.', writer: 'Mr. Weebl'}"></story>

<story v-bind:story="{plot: 'The dark side of the Force is stronger', writer: 'Darth Vader'}"></story>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script type="text/template" id="story1-template">

<h1>My horse is amazing!</h1>

</script>

<template id="story-template">

<h1>{{ story.writer }} said "{{ story.plot }}"</h1>

</template>

<script>

Vue.component('story', {

props: ['story'],

template: '#story-template'

});

new Vue({

el: '.container',

})

</script>

</html>

* + 1. **재사용성**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>User Stories</title>

</head>

<body>

<div class="container">

<h1>Let's hear some stories!</h1>

<div>

<h3>Alex's stories</h3>

<ul class="list-group">

<story v-for="story in storiesBy('Alex')" :story="story"></story>

</ul>

<h3>John's stories</h3>

<ul class="list-group">

<story v-for="story in storiesBy('John')" :story="story"></story>

</ul>

<div class="form-group">

<label for="query">

What are you looking for?

</label>

<input v-model="query" class="form-control">

</div>

<h3>Search results:</h3>

<ul class="list-group">

<story v-for="story in search" :story="story"></story>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

</body>

<template id="story-template">

<li class="list-group-item">

{{ story.writer }} said "{{ story.body }}"

</li>

</template>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

Vue.component('story', {

props: ['story'],

template: '#story-template'

});

new Vue({

el: '.container',

data: {

stories:[

{

body: "I crashed my car today!",

writer: "Alex"

},

{

body: "Yesterday, someone stole my bag!",

writer: "John"

},

{

body: "Someone ate my chocolate...",

writer: "John"

},

{

body: "Someone ate my chocolate...",

writer: "Alex"

},

],

query: ''

},

methods:{

storiesBy: function(writer){

return this.stories.filter(function(story){

return story.writer === writer

})

},

},

computed:{

search: function(){

var query = this.query

return this.stories.filter(function(story){

return story.body.includes(query)

})

}

}

})

</script>

</html>

* + 1. **종합 예제**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div id="app">

<div class="container">

<h1>Let's hear some stories!</h1>

<ul class="list-group">

<story v-for="story in stories" :story="story" :favorite="favorite"></story>

</ul>

</div>

<pre>

{{ $data }}

</pre>

</div>

<template id="story-template">

<li class="list-group-item">

{{ story.writer }} said "{{ story.body }}".

Story upvotes {{ story.upvotes }}.

<button v-show="!story.voted" @click="upvote" class="btn btn-default">Upvote</button>

<button v-show="!isFavorite" @click="setFavorite" class="btn btn-primary">Favorite</button>

<span v-show="isFavorite" class="glyphicon glyphicon-star pull-right" aria-hidden="true"></span>

</li>

</template>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script>

Vue.component('story', {

props: ['story', 'favorite'],

methods: {

upvote: function(){

this.story.upvotes += 1;

this.story.voted = true;

},

setFavorite: function(){

this.favorite = this.story;

}

},

template: '#story-template',

computed: {

isFavorite: function(){

return this.story == this.favorite;

}

}

});

new Vue({

el: '#app',

data: {

stories:[

{

body: "I crashed my car today!",

writer: "Alex",

upvotes: 28,

voted: false,

},

{

body: "Yesterday, someone stole my bag!",

writer: "John",

upvotes: 8,

voted: false,

},

{

body: "Someone ate my chocolate...",

writer: "John",

upvotes: 49,

voted: false,

},

{

body: "Someone ate my chocolate...",

writer: "Alex",

upvotes: 74,

voted: false,

},

],

favorite: {}

},

})

</script>

</html>

* 1. **사용자 정의 이벤트**

$on(event)

$emit(event)

$off()

$once(event)

* + 1. **발생과 청취**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Emit and Listen</title>

</head>

<body>

<div class="container text-center">

<p style="font-size: 140px;">

{{ votes }}

</p>

<button class="btn btn-primary" @click="vote">Vote</button>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script>

new Vue({

el: '.container',

data: {

votes: 0

},

methods: {

vote: function(writer){

this.$emit('voted')

}

},

created(){

this.$on('voted', function(button){

this.votes++

});

}

})

</script>

</html>

http://vuejs.org/api/#Instance-Methods-Events

**생명주기 훅**

|  |  |
| --- | --- |
| 훅 | 호출 시점 |
| beforeCreate | 인스턴스가 초기화된 후 데이터 감시 및 이벤트/감시자를 설정하기 전 |
| created | 인스턴스가 생성된 후 |
| beforeMount | 마운트가 시작되기 직전 |
| mounted | 인스턴스가 이제 막 DOM에 마운트 된 후 |
| beforeUpdate | 데이터 변경 시 가상 DOM이 다시 렌더링되고 패치되기 전 |
| updated | 데이터 변경 시 가상 DOM이 다시 렌더링되고 패치된 후 |
| activated | keep-alive 상태의 컴포넌트가 활성화될 때 |
| deactivated | keep-alive 상태의 컴포넌트가 비활성화될 때 |
| beforeDestroy | Vue 인스턴스가 파괴되기 직전 |
| destroyed | Vue 인스턴스가 파괴된 후 |

http://vuejs.org/api/#Options-Lifecycle-Hooks

* + 1. **부모-자식 간 통신**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Emit and Listen and Parent and Child</title>

</head>

<body>

<div class="container text-center">

<p style="font-size: 140px;">

{{ votes }}

</p>

<food @voted="countVote" name="Cheeseburger"></food>

</div>

<template id="food">

<button class="btn btn-default" @click="vote">{{ name }}</button>

</template>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script>

Vue.component('food', {

template: '#food',

props: ['name'],

methods: {

vote: function(){

this.$emit('voted');

}

}

});

new Vue({

el: '.container',

data: {

votes: 0

},

methods: {

countVote: function(){

this.votes++

}

},

})

</script>

</html>

* + 1. **인자 전달**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Parent and Child and param</title>

</head>

<body>

<div class="container text-center">

<p style="font-size: 140px;">

{{ votes }}

</p>

<div class="row">

<food @voted="countVote" name="Cheeseburger"></food>

<food @voted="countVote" name="Double Bacon Burger"></food>

<food @voted="countVote" name="Rodeo Burger"></food>

</div>

</div>

<template id="food">

<div class="text-center col-lg-4">

<p style="font-size: 40px;">

{{ votes }}

</p>

<button class="btn btn-default" @click="vote">{{ name }}</button>

</div>

</template>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script>

var bus = new Vue()

Vue.component('food', {

template: '#food',

props: ['name'],

data: function(){

return {

votes: 0

}

},

methods: {

vote: function(){

this.votes++

this.$emit('voted');

}

}

});

new Vue({

el: '.container',

data: {

votes: 0

},

methods: {

countVote: function(){

this.votes++

}

},

})

</script>

</html>

* + 1. **비부모 자식 간 통신**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Parent and Child and param</title>

</head>

<body>

<div class="container text-center">

<p style="font-size: 140px;">

{{ votes.count }}

</p>

<button class="btn btn-danger" @click="reset">Reset votes</button>

<hr>

<div class="row">

<food name="Cheeseburger"></food>

<food name="Double Bacon Burger"></food>

<food name="Rodeo Burger"></food>

</div>

<hr>

<h1>Log:</h1>

<ul class="list-group">

<li class="list-group-item" v-for="vote in votes.log">{{ vote }}</li>

</ul>

</div>

<template id="food">

<div class="text-center col-lg-4">

<p style="font-size: 40px;">

{{ votes }}

</p>

<button class="btn btn-default" @click="vote">{{ name }}</button>

</div>

</template>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script>

var bus = new Vue()

Vue.component('food', {

template: '#food',

props: ['name'],

data: function(){

return {

votes: 0

}

},

methods: {

vote: function(event){

//this.name을 사용하는 대신

//이벤트의 엘리먼트의 텍스트에 접근 가능

var food = event.srcElement.textContent;

this.votes++

bus.$emit('voted', food);

},

reset: function(){

this.votes = 0;

}

},

created(){

bus.$on('reset', this.reset);

}

});

new Vue({

el: '.container',

data: {

votes: {

count: 0,

log: []

}

},

methods: {

countVote: function(food){

this.votes.count++

this.votes.log.push(food + ' received a vote.')

}

,

reset: function(){

this.votes = {

count: 0,

log:[]

};

bus.$emit('reset');

}

},

created(){

bus.$on('voted', this.countVote);

}

})

</script>

</html>

* + 1. **이벤트 리스너 제거**

$off(): 모두 제거

$off([event]): 지정된 이벤트에 대한 모든 이벤트 리스너 제거

$off([event, callback)):지정된 콜백에 대한 이벤트 리스너 제거

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Parent and Child and param</title>

</head>

<body>

<div class="container text-center">

<p style="font-size: 140px;">

{{ votes.count }}

</p>

<button class="btn btn-danger" @click="reset">Reset votes</button>

<button class="btn btn-danger" @click="stop">No add any more</button>

<hr>

<div class="row">

<food name="Cheeseburger"></food>

<food name="Double Bacon Burger"></food>

<food name="Rodeo Burger"></food>

</div>

<hr>

<h1>Log:</h1>

<ul class="list-group">

<li class="list-group-item" v-for="vote in votes.log">{{ vote }}</li>

</ul>

</div>

<template id="food">

<div class="text-center col-lg-4">

<p style="font-size: 40px;">

{{ votes }}

</p>

<button class="btn btn-default" @click="vote">{{ name }}</button>

</div>

</template>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.16.4/lodash.min.js"></script>

<script>

var bus = new Vue()

Vue.component('food', {

template: '#food',

props: ['name'],

data: function(){

return {

votes: 0

}

},

methods: {

vote: function(event){

//this.name을 사용하는 대신

//이벤트의 엘리먼트의 텍스트에 접근 가능

var food = event.srcElement.textContent;

this.votes++

bus.$emit('voted', food);

},

reset: function(){

this.votes = 0;

}

},

created(){

bus.$on('reset', this.reset);

}

});

new Vue({

el: '.container',

data: {

votes: {

count: 0,

log: []

}

},

methods: {

countVote: function(food){

this.votes.count++

this.votes.log.push(food + ' received a vote.')

}

,

reset: function(){

this.votes = {

count: 0,

log:[]

};

bus.$emit('reset');

},

stop: function(){

bus.$off(['voted'])

}

},

created(){

bus.$on('voted', this.countVote);

}

})

</script>

</html>

* + 1. **이야기로 돌아가서**
  1. **클래스와 스타일 바인딩**
     1. **클래스 바인딩**
        1. **객체 문법**

<div v-bind:class="elClasses"></div>

data: {

elClasses: {

'classA': true,

'classB': false,

'classC': true

}

}

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div class="container text-center">

<div class="box" v-bind:class="{ 'red' : color, 'blue' : !color }"></div>

<div class="box" v-bind:class="{ 'purple' : color, 'green' : !color }"></div>

<div class="box" v-bind:class="{ 'red' : color, 'blue' : !color }"></div>

<div class="box" v-bind:class="{ 'purple' : color, 'green' : !color }"></div>

<button v-on:click="flipColor" class="btn btn-block btn-success">

Flip color!

</button>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.js"></script>

<script type="text/javascript">

new Vue({

el: '.container',

data: {

color: true

},

methods: {

flipColor: function(){

this.color = !this.color;

}

}

});

</script>

<style type="text/css">

.red{

background: #ff0000;

}

.blue{

background: #00ff00;

}

.purple{

background: #7B1FA2;

}

.green{

background: #4CAF50;

}

.box{

float: left;

width: 200px;

height: 200px;

margin: 40px;

border: 1px solid rgba(0, 0, 0, .2);

}

</style>

</html>

* + - 1. **배열 문법**

v-bind:class=[‘classA’, ‘classB’]

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div class="container text-center">

<div class="box" v-bind:class="[ color ? 'red' : 'blue' ]"></div>

<div class="box" v-bind:class="[ color ? 'purple' : 'green' ]"></div>

<div class="box" v-bind:class="[ color ? 'red' : 'blue' ]"></div>

<div class="box" v-bind:class="[ color ? 'purple' : 'green' ]"></div>

<button v-on:click="flipColor" class="btn btn-block btn-success">

Flip color!

</button>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.js"></script>

<script type="text/javascript">

new Vue({

el: '.container',

data: {

color: true

},

methods: {

flipColor: function(){

this.color = !this.color;

}

}

});

</script>

<style type="text/css">

.red{

background: #ff0000;

}

.blue{

background: #00ff00;

}

.purple{

background: #7B1FA2;

}

.green{

background: #4CAF50;

}

.box{

float: left;

width: 200px;

height: 200px;

margin: 40px;

border: 1px solid rgba(0, 0, 0, .2);

}

</style>

</html>

* + 1. **스타일 바인딩**
       1. **객체 문법**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div class="container text-center">

<div class="box" :style="niceStyle">3434</div>

<div class="box" :style="{'color':'blue', 'fontSize':'20px'}">3434</div>

<div class="box" :style="{'color': niceStyle.color, fontSize: niceStyle.fontSize}">4334</div>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.js"></script>

<script type="text/javascript">

new Vue({

el: '.container',

data: {

niceStyle: {

color: 'blue',

fontSize: '20px'

}

}

});

</script>

<style type="text/css">

.red{

background: #ff0000;

}

.blue{

background: #00ff00;

}

.purple{

background: #7B1FA2;

}

.green{

background: #4CAF50;

}

.box{

float: left;

width: 200px;

height: 200px;

margin: 40px;

border: 1px solid rgba(0, 0, 0, .2);

}

</style>

</html>

* + - 1. **배열 문법**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div class="container text-center">

<div class="box" :style="[niceStyle, badStyle]">1234</div>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.js"></script>

<script type="text/javascript">

new Vue({

el: '.container',

data: {

niceStyle: {

color: 'blue',

fontSize: '20px'

},

badStyle: {

fontStyle: 'italic'

}

}

});

</script>

<style type="text/css">

.red{

background: #ff0000;

}

.blue{

background: #00ff00;

}

.purple{

background: #7B1FA2;

}

.green{

background: #4CAF50;

}

.box{

float: left;

width: 200px;

height: 200px;

margin: 40px;

border: 1px solid rgba(0, 0, 0, .2);

}

</style>

</html>

* + 1. **바인딩 예제**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body class="container-fluid">

<div id="app">

<ul>

<li :class="{'completed' : task.done}"

:style="styleObject"

v-for="task in tasks">

{{task.body}}

<button @click="completeTask(task)" class="btn">

Just do it!

</button>

</li>

</ul>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.js"></script>

<script type="text/javascript">

new Vue({

el: '#app',

data: {

tasks: [

{body: "Feed the horses", done: true},

{body: "Wash armor", done: true},

{body: "Sharp sword", done: false},

],

styleObject: {

fontSize: '25px'

}

},

methods: {

completeTask: function(task){

task.done = !task.done;

}

}

});

</script>

<style type="text/css">

.completed {

text-decoration: line-through;

}

</style>

</html>

1. **API 사용하기**
   1. **API 사용하기**
      1. **CRUD**

Laravel: https://laravel.com/

Symfony2

Nette

CodeIgniter

Yii2

* + - 1. **예제 코드 내려받기**

1. 터미널을 열고 디렉토리 생성

mkdir ~/themajestyofvuejs2

2. 깃허브에서 소스코드 내려 받음

cd ~/themajestyofvue2

git clone <https://github.com/hootlex/the-majesty-of-vuejs-2>

cd ~/themajestyofvuejs2/apis/stories

sh setup.sh

Vagrant 사용시

Vagrant IP가 192.168.10.10 이면

$php artisan serve –port=3000 –host 0.0.0.0;

브라우저를 열고 192.168.10.10:3000 으로 접속

|  |  |
| --- | --- |
| 칼럼명 | 타입 |
| id | Integer, auto increment |
| plot | String |
| writer | String |
| upvotes | Integer, unsigned |

* + - 1. **API 엔드포인트**

|  |  |  |
| --- | --- | --- |
| HTTP 메서드 | URI | 동작 |
| GET/HEAD | api/stories | 모든 이야기를 가져옵니다 |
| GET/HEAD | api/stories/{id} | 특정 이야기를 조회 |
| POST | api/stories | 새 이야기 생성 |
| PUT/PATCH | api/stories/{id} | 기존 이야기 수정 |
| DELETE | api/stories/{id} | 특정 이야기 삭제 |

<http://localhost:3000/api/stories>

구글 JSONFormatter

https://chrome.google.com/webstore/detail/json-formatter/bcjindcccaagfpapjjmafapmmgkkhgoa

* + 1. **API**
  1. **실제 데이터 활용하기**
     1. **비동기로 데이터 가져오기**

<https://github.com/vuejs/vue-resource>

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div id="app">

<div class="container">

<h1>Let's hear some stories!</h1>

<ul class="list-group">

<story v-for="story in stories" :story="story">

</story>

</ul>

<pre>{{ $data }}</pre>

</div>

</div>

<template id="template-story-raw">

<li class="list-group-item">

{{ story.writer }} said "{{ story.plot }}"

<span>{{ story.upvotes }}</span>

</li>

</template>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.js"></script>

<script src="https://code.jquery.com/jquery-2.1.4.min.js"></script>

<script type="text/javascript">

Vue.component('story', {

template: '#template-story-raw',

props: ['story'],

});

var vm = new Vue({

el: '#app',

data: {

stories: []

},

mounted: function(){

$.get('/api/stories', function(data){

vm.stories = data;

})

}

});

</script>

</html>

* + 1. **리팩터링**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div id="app">

<table class="table table-striped">

<tr>

<th>#</th>

<th>Plot</th>

<th>Writer</th>

<th>Upvotes</th>

<th>Actions</th>

</tr>

<tr v-for="story in stories" is="story" :story="story"></tr>

</table>

<template id="template-story-raw">

<tr>

<td>

{{story.id}}

</td>

<td>

<span>

{{story.plot}}

</span>

</td>

<td>

<span>

{{story.writer}}

</span>

</td>

<td>

{{story.upvotes}}

</td>

</tr>

</template>

<p class="lead">Here's a list of all your stories.

</p>

<pre>{{ $data }}</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.js"></script>

<script src="https://code.jquery.com/jquery-2.1.4.min.js"></script>

<script type="text/javascript">

Vue.component('story', {

template: '#template-story-raw',

props: ['story'],

});

var vm = new Vue({

el: '#app',

data: {

stories: []

},

mounted: function(){

$.get('/api/stories', function(data){

vm.stories = data;

})

}

});

</script>

</html>

* + 1. **데이터 갱신**
    2. **데이터 제거**

<html>

<head>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" rel="stylesheet">

<title>Hello Vue</title>

</head>

<body>

<div id="app">

<table class="table table-striped">

<tr>

<th>#</th>

<th>Plot</th>

<th>Writer</th>

<th>Upvotes</th>

<th>Actions</th>

</tr>

<tr v-for="story in stories" is="story" :story="story"></tr>

</table>

<template id="template-story-raw">

<tr>

<td>

{{story.id}}

</td>

<td>

<span>

{{story.plot}}

</span>

</td>

<td>

<span>

{{story.writer}}

</span>

</td>

<td>

{{story.upvotes}}

</td>

<td>

<div class="btn-group">

<button @click="upvoteStory(story)" class="btn btn-primary">

Upvote

</button>

<button @click="deleteStory(story)" class="btn btn-danger">

Delete

</button>

</div>

</td>

</tr>

</template>

<p class="lead">Here's a list of all your stories.

</p>

<pre>{{ $data }}</pre>

</div>

</body>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.3.4/vue.js"></script>

<script src="https://code.jquery.com/jquery-2.1.4.min.js"></script>

<script type="text/javascript">

Vue.component('story', {

template: '#template-story-raw',

props: ['story'],

methods: {

upvoteStory: function(story){

story.upvotes++;

$.ajax({

url: '/api/stories/'+story.id,

type: 'PATCH',

data: story,

});

},

deleteStory: function(story){

var index = vm.stories.indexOf(story);

vm.stories.splice(index, 1);

$.ajax({

url: '/api/stories/'+story.id,

type: 'DELETE'

});

}

}

});

var vm = new Vue({

el: '#app',

data: {

stories: []

},

mounted: function(){

$.get('/api/stories', function(data){

vm.stories = data;

})

}

});

</script>

</html>

* 1. **HTTP 클라이언트 활용**
     1. **개요**Retiring vue-resource : <https://medium.com/the-vue-point/retiring-vue-resource-871a82880af4#.lew43e17f>

axios: <https://github.com/mzabriskie/axios>

* + 1. **vue-resource**

github: <https://github.com/vuejs/vue-resource>

cdnjs: <https://cdnjs.com/libraries/vue-resource>

this.$http({url: ‘/someUrl’, method: ‘GET’})

.then(function (response){

// 성공 콜백

}, function (response) {

// 오류 콜백

});

**축약형 요청 메서드**

this.$http.get(url, [data], [options]).then(successCallback, errorCallback);

this.$http.post(url, [data], [options]).then(successCallback, errorCallback);

this.$http.put(url, [data], [options]).then(successCallback, errorCallback);

this.$http.patch(url, [data], [options]).then(successCallback, errorCallback);

this.$http.delete(url, [data], [options]).then(successCallback, errorCallback);

* + 1. **axios**

axios는 Promise 기반 Http 클라이언트, vue-resource가 제공하는 거의 모든 API 제공, 취소 또한 지원. 타입스크립트도 지원

axios({

method: ‘get’,

url: ‘/user/kostas’

})

.then(function(response){

console.log(response);

})

.catch(function(error){

console.log(error);

});

axios.request(config)

axios.get(url[, config])

axios.delete(url[, config])

axios.head(url[, config])

axios.post(url[, data[, config]])

axios.put(url[, data[, config]])

axios.patch(url[, data[, config]])

vue-resource와 같이 this.$http를 사용하려면 Vue.prototype.$http = axios 처럼 설정

* + 1. **axios 사용하기**

stories.html

<script src=”<https://unpkg.com/axios/dist/axios.min.js>”></script>

app.js

mounted: function() {

axios.get(‘/api/stories’)

.then(function (response){

Vue.set(vm ,‘stories’, reponse.data)

// vm.stories = response.data

})

},

upvoteStory: function(story){

story.upvotes++;

axios.patch(‘/api/stories/’ + story.id, story)

}

deleteStory: function(story){

var index = this.$parent.stories.indexOf(story);

this.$parent.stories.splice(index, 1);

axios.delete(‘/api/stories/’ + story.id);

}

* + 1. **기능 개선**
       1. **이야기 수정**

<td>

<input v-if=”story.editing” v-model=”story.plot” class=”form-control”>

<span v-else>

{{story.plot}}

</span>

</td>

<td>

<input v-if=”story.editing” v-model=”story.writer” class=”form-control”>

<span v-else>

{{story.writer}}

</span>

</td>

<td>

{{story.upvotes}}

</td>

<td>

<div v-if=”!story.editing” class=”btn-group”>

<button @click=”upvoteStory(story)” class=”btn btn-primary”>

Upvote

</button>

<button @click=”editStory(story)” class=”btn btn-primary”>

Edit

</button>

<button @click=”deleteStory(story)” class=”btn btn-primary”>

Delete

</button>

</div>

</td>

Vue.component(‘story’, {

…

methods: {

…

editStory: function(story){

story.editing = true;

}

}

런타임시 프로퍼티 추가, 삭제

Vue.set , Vue.delete

mounted: function(){

var vm = this;

axios.get(‘/api/stories’)

.then(function (response){

var storiesReady = response.data.map(function (story){

story.editing = false;

return story;

})

vm.stories = storiesReady

//Vue.set(vm , ‘stories’, storiesReady)

});

}

<div class=”btn-group” v-else>

<button @click=”updateStory(story)” class=”btn btn-primary”>

Update Story

</button>

<button @click=”story.editing=false” class=”btn btn-default”>

Cancel

</button>

</div>

Vue.component(‘story’, {

…

methods: {

…

updateStory: function(story){

axios.patch(‘/api/stories/’ + story.id, story)

story.editing = false;

},

}

});

* + - 1. **새 이야기 만들기**
      2. **저장 및 수정 단위**
    1. **자바스크립트 파일**
    2. **소스코드**

<https://github.com/hootlex/the-majesty-of-vuejs-2/blob/master/apis/stories/public/stories.html>

<!doctype html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Stories</title>

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.2/css/bootstrap.min.css">

</head>

<body>

<main>

<div class="container">

<h1>Stories</h1>

<div id="v-app">

<table class="table table-striped">

<tr>

<th>#</th>

<th>Plot</th>

<th>Writer</th>

<th>Upvotes</th>

<th>Actions</th>

</tr>

<tr v-for="story in stories" :key="story.id" is="story" :story="story"></tr>

</table>

<p class="lead">Here's a list of all your stories.

<button @click="createStory()" class="btn btn-primary">Add a new one?</button>

</p>

<pre>{{ $data }}</pre>

</div>

</div>

</main>

<template id="template-story-raw">

<tr>

<td>

{{story.id}}

</td>

<td class="col-md-6">

<input v-if="story.editing" v-model="story.plot" class="form-control">

<!--in other occasions show the story plot-->

<span v-else>

{{story.plot}}

</span>

</td>

<td>

<input v-if="story.editing" v-model="story.writer" class="form-control">

<!--in other occasions show the story writer-->

<span v-else>

{{story.writer}}

</span>

</td>

<td>

{{story.upvotes}}

</td>

<td>

<div class="btn-group" v-if="!story.editing">

<button @click="upvoteStory(story)" class="btn btn-primary">Upvote</button>

<button @click="editStory(story)" class="btn btn-default">Edit</button>

<button @click="deleteStory(story)" class="btn btn-danger">Delete</button>

</div>

<div class="btn-group" v-else>

<!--If the story is taken from the db then it will have an id-->

<button v-if="story.id" class="btn btn-primary" @click="updateStory(story)">Update Story</button>

<!--If the story is new we want to store it-->

<button v-else class="btn btn-success" @click="storeStory(story)">Save New Story</button>

<!--Always show cancel-->

<button @click="story.editing=false" class="btn btn-default">Cancel</button>

</div>

</td>

</tr>

</template>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.5.17/vue.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/vue-resource/0.7.0/vue-resource.js"></script>

<script src='/js/app.js' type="text/javascript"></script>

</body>

</html>

<https://github.com/hootlex/the-majesty-of-vuejs-2/blob/master/apis/stories/public/js/app.js>

Vue.component('story', {

template: '#template-story-raw',

props: ['story'],

methods: {

deleteStory: function (story) {

var index = this.$parent.stories.indexOf(story);

this.$parent.stories.splice(index, 1)

this.$http.delete('/api/stories/' + story.id)

},

upvoteStory: function (story) {

story.upvotes++;

this.$http.patch('/api/stories/' + story.id, story)

},

editStory: function (story) {

story.editing = true;

},

updateStory: function (story) {

this.$http.patch('/api/stories/' + story.id, story)

//Set editing to false to show actions again and hide the inputs

story.editing = false;

},

storeStory: function (story) {

this.$http.post('/api/stories/', story).then(function (response) {

/\*

After the the new story is stored in the database fetch again all stories with

vm.fetchStories();

Or Better, update the id of the created story

\*/

Vue.set(story, 'id', response.data.id);

//Set editing to false to show actions again and hide the inputs

story.editing = false;

});

},

}

})

new Vue({

el: '#v-app',

data: {

stories: [],

story: {}

},

mounted: function () {

this.fetchStories()

},

methods: {

createStory: function () {

var newStory = {

plot: "",

upvotes: 0,

editing: true

};

this.stories.push(newStory);

},

fetchStories: function () {

var vm = this;

this.$http.get('/api/stories')

.then(function (response) {

// set data on vm

var storiesReady = response.data.map(function (story) {

story.editing = false;

return story

})

Vue.set(vm, 'stories', storiesReady)

});

},

}

});

* 1. **페이지 처리**