Vue.js

Vue.js

-
- IDEs
-
-

-
-
-
  -
- Vue
  -
  -
-
  -
  -
- props
  -
  -
- this component
  -
  -
-
  -
  -
-
  -
  -
- this.$parent
  -
  -
- this.$refs
  -
  -
-
  -
  -
- API
-
  -

Range slider

-
-
- demo

    -
-
    -
    -
        - ESLint
        - JSHint
-
    -
    - ?
    -
- mixins
    - ?
    - ?

app

Vue.js

Vue

100 demo

Vue

- :
- : 2 3
- :

- :
- **app-** :

-

```
<!--  -->
<app-header></app-header>
<user-list></user-list>
<range-slider></range-slider>

<!--  -->
<btn-group></btn-group> <!-- .  `button-group`  -->
<ui-slider></ui-slider> <!-- ui  -->
<slider></slider> <!--  -->
```

Vue.js 100% Javascript

-
-
- IDE IDE

method computed

~~~
<!--  -->
<template>
  <div id="example">
    <p>Original message: "{{ message }}"</p>
    <p>Computed reversed message: "{{ reversedMessage }}"</p>
  </div>
</template>
<script type="text/javascript">
  export default {
    data() {
        return() {
            message: 'Hello'
       }
   },
    computed: {
    reversedMessage: function() {
        return this.message.split('').reverse().join('')
     }
   }
 };
</script>


[ ](#)

###  props

 Vue.js  JavaScript  props  [JavaScript ](https://developer.mozilla.org/en-US/docs/Glossary/Primitive)

###

*  API
*  props  API  HTML(5)
*  prop
*

###

 props

```html
<!--  -->
<range-slider
  :values="[10, 20]"
  :min="0"
  :max="100"
  :step="5"
  @on-slide="updateInputs"
  @on-end="updateResults">
</range-slider>

<!--  -->
<range-slider :config="complexConfigObject"></range-slider>
~~~

props

Vue.js props API API

props Vue.js (`:attr="value" v-bind:attr="value"`) props

props

-
- `type`
- props prop

```
<template>
  <input type="range" v-model="value" :max="max" :min="min">
</template>
<script type="text/javascript">
  export default {
    props: {
      max: {
        type: Number, //
        default() { return 10; },
     },
      min: {
        type: Number,
        default() { return 0; },
     },
      value: {
        type: Number,
        default() { return 4; },
     },
   },
 };
</script>
```

this component

Vue.js this this component

ES6 var self = this; Vue

- **ES6** this
- this this this
- **ES6** “this”

```
<script type="text/javascript">
export default {
  methods: {
    hello() {
      return 'hello';
   },
    printHello() {
      console.log(this.hello());
   },
 },
};
</script>

<!--  -->
<script type="text/javascript">
export default {
  methods: {
    hello() {
      return 'hello';
   },
    printHello() {
      const self = this; //
      console.log(self.hello());
   },
 },
};
</script>
```

-
- propertiesdatacomputedwatches methods
- name; extends; props, data computed; components; watch methods; lifecycle methods
- `name` vue devtools
- CSS BEM rscss -
- .vue

```
<template lang="html">
  <div class="Ranger__Wrapper">
    <!-- ... -->
  </div>
</template>

<script type="text/javascript">
  export default {
    //  name
    name: 'RangeSlider',
    //  mixins
    mixins: [],
    //
    extends: {},
    //
    props: {
      bar: {}, //
      foo: {},
      fooBar: {},
   },
    //
    data() {},
    computed: {},
    //
    components: {},
    //
    watch: {},
    methods: {},
    //
    beforeCreate() {},
    mounted() {},
 };
</script>

<style scoped>
  .Ranger__Wrapper { /* ... */ }
</style>
```

Vue.js ViewModel

-
- DOM

-
- upload-successupload-error dropzone-upload-successdropzone-upload-error
- client-api-load drive-upload-success

this.$parent

Vue.js **`this.$parent`**

- Vue
-

- props
- props
-

this.$refs

Vue.js `ref` HTML `this.$refs` HTML `this.$refs` API `this.$refs`

- API
-

- API
-
- API
- props issue
- props
- **Props** API
- props events `this.$refs`
- DOM `this.$ref` `JQuerydocument.getElement*document.queryElement`

```
<!--  this.$refs -->
<range :max="max"
  :min="min"
  @current-value="currentValue"
  :step="1"></range>

<!--  this.$refs -->
<modal ref="basicModal">
  <h4>Basic Modal</h4>
  <button class="primary" @click="$refs.basicModal.hide()">Close</button>
</modal>
<button @click="$refs.basicModal.open()">Open modal</button>

<!-- Modal component -->
<template>
  <div v-show="active">
    <!-- ... -->
  </div>
</template>

<script>
  export default {
    // ...
    data() {
      return {
        active: false,
     };
   },
    methods: {
      open() {
        this.active = true;
     },
      hide() {
        this.active = false;
     },
   },
    // ...
 };
</script>
```

```
<!--  emited  this.$refs  -->
<template>
  <range :max="max"
    :min="min"
    ref="range"
    :step="1"></range>
</template>

<script>
  export default {
    // ...
    methods: {
      getRangeCurrentValue() {
        return this.$refs.range.currentValue;
     },
   },
    // ...
 };
</script>
```

Vue.js CSS

-
-

BEM OOCSS style scoped scoped class

```
<style scoped>
  /*  */
  .MyExample { }
  .MyExample li { }
  .MyExample__item { }

  /*  */
  .My-Example { } /* ,  BEM  */
</style>
```

API

Vue.js Vue API `README.md`

-
- API API
- API
- `README.md` GitHubBitbucketGitlab

`README.md`

```
range-slider/
 range-slider.vue
 range-slider.less
 README.md
```

README vue API

Range slider

range slider

noUiSlider touch

`<range-slider>`

| attribute | type | description | --- | --- | --- | `min` | Number | . | `max` | Number | . | `values` | Number[] *optional* | . . `values="[10, 20]"`. Defaults to `[opts.min, opts.max]`. | `step` | Number *optional* | 1. | `on-slide` | Function *optional* | `(values, HANDLE)` `on-slide={ updateInputs }`, `component.updateInputs = (values, HANDLE) => { const value = values[HANDLE]; }`. | `on-end` | Function *optional* | `(values, HANDLE)`

slider noUiSlider

demo

`index.html` demo

- demo
- demo
- demo

.vue `eslint-plugin-html` `vue-cli` `vue-cli`

-
-

`*.vue` `<script>` `vue props`

ESLint

ESLint ESLint HTML

`.eslintrc` ESlint IDE

```
{
  "extends": "eslint:recommended",
  "plugins": ["html"],
  "env": {
    "browser": true
 },
  "globals": {
    "opts": true,
    "vue": true
 }
}
```

ESLint

```
eslint src/**/*.vue
```

JSHint

JSHint HTML `--extra-ext` `--extract=auto`

`.jshintrc` ESlint IDE

```
{
  "browser": true,
  "predef": ["opts", "vue"]
}
```

JSHint

```
jshint --config modules/.jshintrc --extra-ext=html --extract=auto modules/
```

JSHint `vue` `html`

Vue.js

- , ;
- ;

?

-
- props & events

-
-
- “”

mixins

?

Mixins mixin mixin

?

mixin

```
const MenuMixin = {
  data () {
    return {
      language: 'EN'
   }
 },

  methods: {
    changeLanguage () {
      if (this.language === 'DE') this.$set(this, 'language', 'EN')
      if (this.language === 'EN') this.$set(this, 'language', 'DE')
   }
 }
}

export default MenuMixin
```

mixin

```
<template>
  <ul class="mobile">
    <li @click="changeLanguage">Change language</li>
  </ul>
</template>

<script>
  import MenuMixin from './MenuMixin'

  export default {
    mixins: [MenuMixin]
 }
</script>
```