

Посилання на елементи шаблонів. *ref*

<https://ua.vuejs.org/guide/essentials/template-refs.html>

`ref` використовується тоді, коли нам потрібен прямий доступ до базових елементів DOM

Задання посилання

Загальна форма	Приклад
<pre><тег ref="посилання"> ... </тег></pre>	<pre><input ref="age" type="number" v-model="age" /></pre>

Звертання за посиланням

Загальна форма	Приклад
<pre>this.\$refs.посилання</pre>	<pre>this.\$refs.age.focus()</pre>

Приклад. При відкритті сайту перевіряється вік користувача.

- 1) якщо не ввів, просимо ввести (поміщаємо фокус в елемент вводу)
- 2) якщо вік менше 18 років – доступ заборонено
- 3) якщо вік більше за 18 років – доступ дозволено

```
<div id="app">
  <div>
    <label>
      Введіть вік :
      <input ref="age" type="number" v-model="age" />
    </label>
  </div>

  <button @click="onStart">Go</button>

  <div v-if="message" class="error">{{message}}</div>
</div>
```

The screenshot shows a user interface element consisting of a label "Введіть вік :" followed by an input field and a button labeled "Go". The input field is highlighted with a red border, indicating it is the active or focused element. The entire form is enclosed in a light blue rectangular border.

Приклад. При відкритті сайту перевіряється вік користувача.

- 1) якщо не ввів, просимо ввести (поміщаємо фокус в елемент вводу)
- 2) якщо вік менше 18 років – доступ заборонено
- 3) якщо вік більше за 18 років – доступ дозволено

```
<div id="app">
  <div>
    <label>
      Введіть вік :
      <input ref="age" type="number" v-model="age" />
    </label>
  </div>

  <button @click="onStart">Go</button>

  <div v-if="message" class="error">{{message}}</div>
</div>
```

Введіть вік :

```
const app = createApp({
  data() {
    return {
      age: null,
      message: null,
    }
  },
  methods: {
    onStart() {
      if (!this.age) {
        this.$refs.age.focus()
        this.message = 'Введіть вік'
      } else {
        if (this.age < 18) this.message = 'Вхід заборонено'
        else this.message = 'Вхід дозволено'
      }
    },
  },
}).mount('#app')
```

Приклад. Вводиться логін і пароль. Поступово якщо якесь з полів не введено, то виводити повідомлення і поміщати фокус у потрібне поле

```
<div id="app">
  <div>
    <label>
      Login :
      <input ref="login" type="text" v-model="login" />
    </label>
  </div>
  <div>
    <label>
      Password :
      <input ref="password" type="password" v-model="password" />
    </label>
  </div>
  <button @click.prevent="onLogin">Go</button>
  <div v-if="message && isMessageVisible" class="error">{{message}}</div>
</div>
```

The screenshot shows a simple login interface. It consists of two text input fields, one labeled "Login" and another labeled "Password", both with red borders indicating they are currently selected or focused. Below these inputs is a single button labeled "Go". The entire form is enclosed in a light blue border.

Приклад. Вводиться логін і пароль. Поступово якщо якесь з полів не введено, то виводити повідомлення і поміщасти фокус у потрібне поле

```
<div id="app">
  <div>
    <label>
      Login : <input ref="login" type="text" v-model="login" />
    </label>
  </div>
  <div>
    <label>
      Password : <input ref="password" type="password" v-model="password" />
    </label>
  </div>
  <button @click.prevent="onLogin">Go</button>
  <div v-if="message && isMessageVisible" class="error">{{message}}</div>
</div>
```

A screenshot of a simple login form. It consists of two text input fields labeled "Login:" and "Password:", each with a red border around it. Below the inputs is a blue "Go" button.

```
data() {
  return {
    login: null,
    password: null,
    fieldName: null,
    message: null,
  }
},
computed: {
  isMessageVisible() {
    return this.fieldName && !this[this.fieldName];
  },
},
methods: {
  onLogin() {
    if (!this.login) {
      this.$refs.login.focus()
      this.fieldName = 'login'
      this.message = 'Задайте login'
    } else if (!this.password) {
      this.$refs.password.focus()
      this.fieldName = 'password'
      this.message = 'Задайте password'
    }
  },
},
```

Випадок, коли за спільним посиланням має декілька елементів (масив елементів)

Задання посилання

Загальна форма	Приклад
<p><u>Посилання на ітерований тег</u></p> <pre><тег v-for="..." ref="посилання"> ... </тег></pre>	<pre><input v-for="i in 3" :key="i" ref="mark" type="number" v-model="marks[i-1]" /></pre>
<p><u>Посилання на дочірній елемент ітерованого тегу</u></p> <pre><тег v-for="..." > ... <тег ref="посилання" > ... </тег> ... </тег></pre>	<pre><div v-for="i in 3" :key="i"> <label> Оцінка № {{i}} : <input ref="mark" type="number" v-model="marks[i-1]"/> </label> </div></pre>

Приклад. Вводиться 3 оцінки. Поступово якщо якась з оцінок не введена, то виводити повідомлення і поміщасти фокус у потрібне поле. Знайти мінімальну оцінку.

```
<div id="app">
  <div v-for="i in 3" :key="i">
    <label>
      Оцінка № {{i}} :
      <input ref="mark" type="number" v-model="marks[i-1]" />
    </label>
  </div>

  <button @click="getAverage">Знайти мінімальну оцінку</button>

  <div v-if="message" class="error">{{message}}</div>
</div>
```

Оцінка № 1 :	<input type="text"/>
Оцінка № 2 :	<input type="text"/>
Оцінка № 3 :	<input type="text"/>
<input type="button" value="Знайти сереню оцінку"/>	

Приклад. Вводиться 3 оцінки. Поступово якщо якась з оцінок не введена, то виводити повідомлення і поміщасти фокус у потрібне поле. Знайти мінімальну оцінку.

```
<div id="app">
  <div v-for="i in 3" :key="i">
    <label>
      Оцінка № {{i}} :
      <input ref="mark" type="number" v-model="marks[i-1]" />
    </label>
  </div>

  <button @click="getAverage">Знайти мінімальну оцінку</button>
  <div v-if="message" class="error">{{message}}</div>
</div>
```

Оцінка № 1 :

Оцінка № 2 :

Оцінка № 3 :

Знайти сереню оцінку

```
const app = createApp({
  data() {
    return {
      marks: [],
      message: null,
    }
  },
  methods: {
    getAverage() {
      let notValidInputIndex = -1
      for (let i = 0; i < 3; i++) {
        if (!this.marks[i]) {
          notValidInputIndex = i
          break
        }
      }
      if (notValidInputIndex !== -1) {
        this.$refs.mark[notValidInputIndex].focus()
        this.message = `Задайте оцінку № ${notValidInputIndex + 1}`
      } else {
        this.message = `Найменша оцінка: ${Math.min(...this.marks)}`
      }
    },
  },
}).mount('#app')
```

Приклад.