Vue-Awesome-Swiper

基于 Swiper4、适用于 Vue 的轮播组件，支持服务端渲染和单页应用。
如果需要回退到 Swiper3，请使用 v2.6.7 版本。

Example

Demo Page
CDN Example
mobile fullpage example code
nuxt.js/ssr example code

Install

CDN

```
<link rel="stylesheet" href="path/to/swiper/dist/css/swiper.css"/>
<script type="text/javascript" src="path/to/swiper.js"></script>
<script type="text/javascript" src="path/to/vue.min.js"></script>
<script type="text/javascript" src="path/to/dist/vue-awesome-swiper.js"></script>
<script type="text/javascript">
  Vue.use(window.VueAwesomeSwiper)
</script>
```

NPM

```
npm install vue-awesome-swiper --save
```

Mount

mount with global

```
import Vue from 'vue'
import VueAwesomeSwiper from 'vue-awesome-swiper'

// require styles
import 'swiper/dist/css/swiper.css'

Vue.use(VueAwesomeSwiper, /* { default global options } */)
```

mount with component

```
// require styles
import 'swiper/dist/css/swiper.css'

import { swiper, swiperSlide } from 'vue-awesome-swiper'

export default {
  components: {
    swiper,
    swiperSlide
  }
}
```

mount with ssr

```
// If used in nuxt.js/ssr, you should keep it only in browser build environment
if (process.browser) {
  const VueAwesomeSwiper = require('vue-awesome-swiper/dist/ssr')
  Vue.use(VueAwesomeSwiper)
}
```

custom swiper plugin

```
import Swiper from 'swiper'
Swiper.use({
  name: 'pluginName',
  params: {
    pluginSwitch: false,
  },
  on: {
    init() {
      if (!this.params.pluginSwitch) return
      console.log('init')
    },
    // swiper callback...
  }
})
```

Difference（使用方法的异同）

SSR and the only difference in the use of the SPA:

- SPA worked by the `component`, find swiper instance by `ref attribute`.
- SSR worked by the `directive`, find swiper instance by `directive arg`.
- Other configurations, events are the same.

SPA

```
<!-- The ref attr used to find the swiper instance -->
<template>
  <swiper :options="swiperOption" ref="mySwiper" @someSwiperEvent="callback">
    <!-- slides -->
    <swiper-slide>I'm Slide 1</swiper-slide>
    <swiper-slide>I'm Slide 2</swiper-slide>
    <swiper-slide>I'm Slide 3</swiper-slide>
    <swiper-slide>I'm Slide 4</swiper-slide>
    <swiper-slide>I'm Slide 5</swiper-slide>
    <swiper-slide>I'm Slide 6</swiper-slide>
    <swiper-slide>I'm Slide 7</swiper-slide>
    <!-- Optional controls -->
    <div class="swiper-pagination"  slot="pagination"></div>
    <div class="swiper-button-prev" slot="button-prev"></div>
    <div class="swiper-button-next" slot="button-next"></div>
    <div class="swiper-scrollbar"   slot="scrollbar"></div>
  </swiper>
</template>

<script>
  export default {
    name: 'carrousel',
    data() {
      return {
        swiperOption: {
          // some swiper options/callbacks
          // 所有的参数同 swiper 官方 api 参数
          // ...
        }
      }
    },
    computed: {
      swiper() {
        return this.$refs.mySwiper.swiper
      }
    },
    mounted() {
      // current swiper instance
      // 然后你就可以使用当前上下文内的swiper对象去做你想做的事了
      console.log('this is current swiper instance object', this.swiper)
      this.swiper.slideTo(3, 1000, false)
    }
  }
</script>
```

Async data example

```
<template>
  <swiper :options="swiperOption">
    <swiper-slide v-for="slide in swiperSlides">I'm Slide {{ slide }}</swiper-slide>
    <div class="swiper-pagination" slot="pagination"></div>
  </swiper>
</template>

<script>
  export default {
    name: 'carrousel',
    data() {
      return {
        swiperOption: {
          pagination: {
            el: '.swiper-pagination'
          }
        },
        swiperSlides: [1, 2, 3, 4, 5]
      }
    },
    mounted() {
      setInterval(() => {
        console.log('simulate async data')
        if (this.swiperSlides.length < 10) {
          this.swiperSlides.push(this.swiperSlides.length + 1)
        }
      }, 3000)
    }
  }
</script>
```

SSR

```
<!-- You can custom the "mySwiper" name used to find the swiper instance in current component -->
<template>
  <div v-swiper:mySwiper="swiperOption" @someSwiperEvent="callback">
    <div class="swiper-wrapper">
      <div class="swiper-slide" v-for="banner in banners">
        <img :src="banner">
      </div>
    </div>
    <div class="swiper-pagination"></div>
  </div>
</template>

<script>
  export default {
    data () {
      return {
        banners: [ '/1.jpg', '/2.jpg', '/3.jpg' ],
        swiperOption: {
          pagination: {
            el: '.swiper-pagination'
          },
          // some swiper options...
        }
      }
    },
    mounted() {
      setTimeout(() => {
        this.banners.push('/4.jpg')
        console.log('banners update')
      }, 3000)
      console.log(
        'This is current swiper instance object', this.mySwiper,
        'It will slideTo banners 3')
      this.mySwiper.slideTo(3, 1000, false)
    }
  }
</script>
```