

Vue.js 混入

混入 (mixins)定义了一部分可复用的方法或者计算属性。混入对象可以包含任意组件选项。当组件使用混入对象时，所有混入对象的选项将被混入该组件本身的选项。

来看一个简单的实例：

实例

```
var vm = new Vue({
  el: '#databinding',
  data: {
  },
  methods : {
  },
});
// 定义一个混入对象
var myMixin = {
  created: function () {
    this.startmixin()
  },
  methods: {
    startmixin: function () {
      document.write("欢迎来到混入实例");
    }
  }
};
var Component = Vue.extend({
  mixins: [myMixin]
})
var component = new Component();
```

[尝试一下 »](#)

选项合并

当组件和混入对象含有同名选项时，这些选项将以恰当的方式混合。

比如，数据对象在内部会进行浅合并（一层属性深度），在和组件的数据发生冲突时以组件数据优先。

以下实例中，Vue 实例与混入对象包含了相同的方法。从输出结果可以看出两个选项合并了。

实例

```
var mixin = {
  created: function () {
    document.write('混入调用' + '<br>')
  }
}

new Vue({
  mixins: [mixin],
```

```
created: function () {  
  document.write('组件调用' + '<br>')  
}  
});
```

输出结果为：

```
混入调用  
组件调用
```

尝试一下 »

如果 methods 选项中有相同的函数名，则 Vue 实例优先级会较高。如下实例，Vue 实例与混入对象的 methods 选项都包含了相同的函数：

实例

```
var mixin = {  
  methods: {  
    hellworld: function () {  
      document.write('HelloWorld 方法' + '<br>');  
    },  
    samemethod: function () {  
      document.write('Mixin: 相同方法名' + '<br>');  
    }  
  }  
};  
  
var vm = new Vue({  
  mixins: [mixin],  
  methods: {  
    start: function () {  
      document.write('start 方法' + '<br>');  
    },  
    samemethod: function () {  
      document.write('Main: 相同方法名' + '<br>');  
    }  
  }  
});  
vm.hellworld();  
vm.start();  
vm.samemethod();
```

输出结果为：

```
HelloWorld 方法  
start 方法  
Main: 相同方法名
```

[尝试一下 »](#)

以上实例，我们调用了以下三个方法：

```
vm.hellworld();  
vm.start();  
vm.samemethod();
```

从输出结果 methods 选项中如果碰到相同的函数名则 Vue 实例有更高的优先级会执行输出。

全局混入

也可以全局注册混入对象。注意使用！一旦使用全局混入对象，将会影响到所有之后创建的 Vue 实例。使用恰当时，可以为自定义对象注入处理逻辑。

实例

```
// 为自定义的选项 'myOption' 注入一个处理器。  
Vue.mixin({  
  created: function () {  
    var myOption = this.$options.myOption  
    if (myOption) {  
      console.log(myOption)  
    }  
  }  
})  
new Vue({  
  myOption: 'hello!'  
})  
// => "hello!"
```

[尝试一下 »](#)

谨慎使用全局混入对象，因为会影响到每个单独创建的 Vue 实例 (包括第三方模板)。

[← Vue.js 过渡 & 动画](#)[Vue.js Ajax\(vue-resource\) →](#)[✍ 点我分享笔记](#)